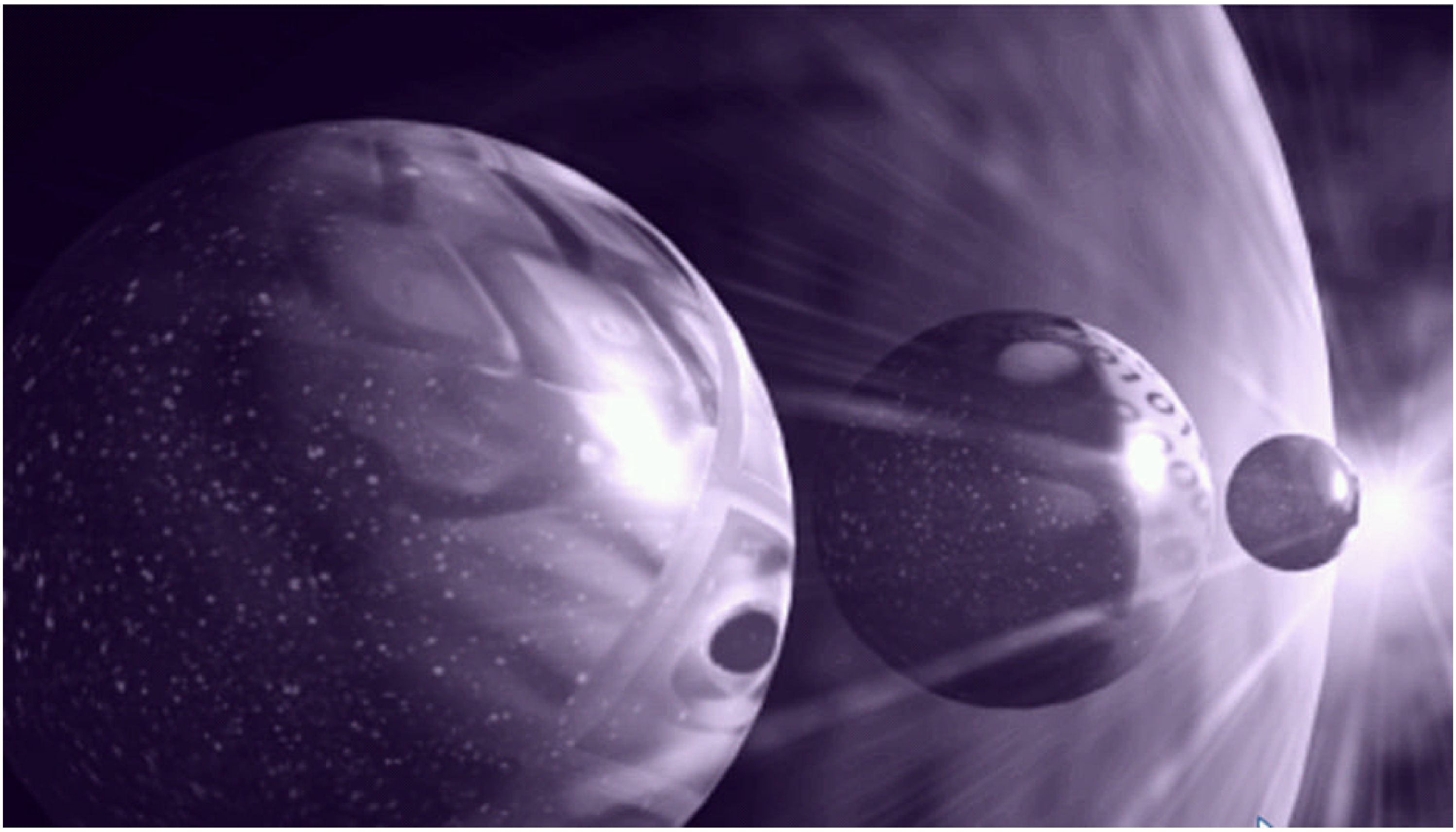
WebSphere software

IBM

Guide to WebSphere Portal

April 2002



e business software

Contents

Figures

Introduction

The Portal Vision

Portals serve as a simple, unified access point to web applications. Portals also do much more: they provide additional valuable functions like security, search, collaboration, and workflow. A portal delivers integrated content and applications, plus a unified, collaborative workplace. Indeed, portals are the next-generation desktop, delivering e-business applications over the web to all kinds of client devices.

A complete portal solution should provide users with convenient access to everything they need to get their tasks done, anytime, anywhere, in a secure manner. IBM's vision is that portals are the key to reach and user experience. That is, portals provide the tools and user interface to access information and applications, and to manage the selection and personalization of content.

THE WEBSPHERE PLATFORM

IBM WebSphere Portal is part of the WebSphere software platform. The platform fundamentally does three things:

1. Provides access to information across a spectrum of users, devices, and customization options

2. Integrates and automates business processes

3. Builds, connects, and manages applications.

These three areas of functionality organize the broad range of WebSphere software into three classes:

> **Foundation and tools** for building, running, and deploying applications. WebSphere Application Server, MQ messaging, and state-of-the-art development tools form a solid base for the platform. The foundation and tools provide the Internet expertise you need, enable you to build and use Web Services, and link you to a greater technical community of developers and other WebSphere users.

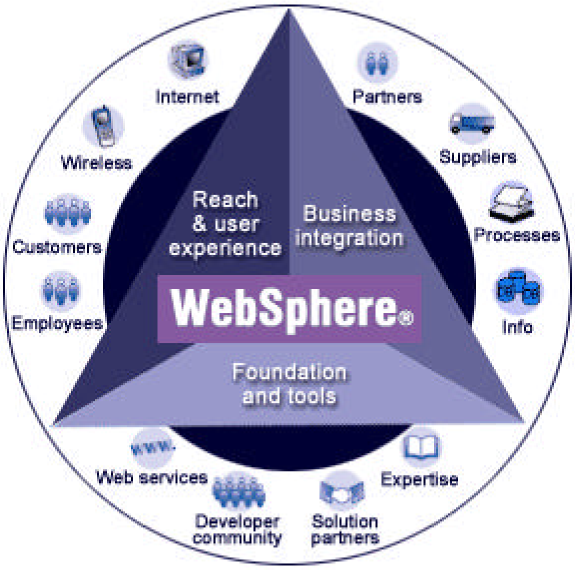


Figure 0.1: The WebSphere platform

Business integration for integrating internal business processes, including processes that involve business partners. WebSphere offerings such as WebSphere Business Integrator make it easy for your company to implement applications and business processes, including supply chain management and the integration of existing processes with the Web.

Reach and user experience for personalizing Web-based content and making it accessible to any device. These WebSphere products fine-tune your users' experience and provide broad access for your customers, employees, business partners and remote branch offices.

WebSphere Portal leads the Reach and User Experience part of the WebSphere Platform. It provides an extensible framework for interacting with enterprise applications, content, people, and processes. Self-service features allow end users to personalize and organize their own view of the portal, to manage their own profiles and to publish and share documents with their colleagues. WebSphere Portal provides additional services such as single sign-on, security, web content publishing, search and, personalization, collaboration services, enterprise application integration, support for mobile devices, and site analytics.

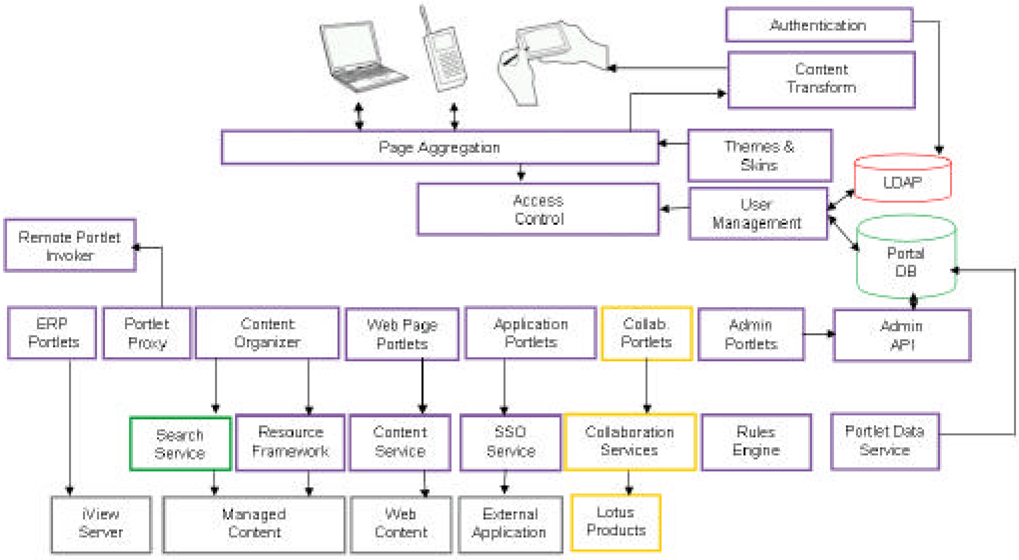


Figure 0.2: WebSphere Portal Architecture

WebSphere Portal is the industry's most comprehensive portal solution, and represents the de-facto standard e-business architecture. WebSphere Portal integrates both IBM and business partner technologies to realize this architecture. IBM is also extending its portal offering to deliver highly personalized and context-sensitive applications, accessible from any device, anytime.

As portals enter their third generation of maturity, WebSphere Portal leads the way with its concepts of delegated administration, cascading page layouts, portal federation through web services, advanced portlet application concepts, business process integration, knowledge management, and advanced personalization. In complementary offerings, additional pervasive computing functions are enabled, such as intelligent notification, offline browsing, and data synchronization.

SUPPORTING PRODUCTS

The following products are also included with the portal installation discs:

- DB2 Universal Database
- SecureWay Directory
- IBM HTTP Server
- WebSphere Application Server Advanced Edition
- Lotus Domino Application Server
- Lotus Sametime
- Lotus QuickPlace
- Lotus Workflow

See [1] for supported platforms, system requirements, and version numbers.

WebSphere Portal provides three offerings, each designed to provide the infrastructure you need to build and deploy highly scalable portals. All three offerings share a common framework (the portal server) plus additional products and services. The portal server provides common services such as application connectivity, integration, administration, and presentation that are required across portal environments.

The following table summarizes the components of the WebSphere Portal offerings.

WEBSPHERE PORTAL ENABLE	
Portal Server	Provides presentation, user management, security, and other services for constructing the portal site.
WebSphere Personalization	Provides personalization technologies for targeting Web content to meet user needs and preferences: ▪ Rules-based personalization, where the business manager defines a set of business rules that determine which Web content is displayed for a particular user. ▪ Recommendations, using advanced statistical models and other matching techniques to extract trends from the behavior of Web site visitors. This approach adapts to changing trends in visitor interests without creating new business rules. ▪ Campaign management, for e-mail and web-based promotions, such as enrollment offers or product introductions.
Web Content Publisher	Provides tools for contributing content and documents to the portal.
WebSphere Studio Application Developer	Professional developer tools for creating, testing, debugging, and deploying portlets, servlets, and other assets related to portals and web applications.
WEBSPHERE PORTAL EXTEND	
All products in Enable	Same portal server and personalization functions.
Lotus Collaboration Components	A series of collaborative service components for the portal, including instant messaging and access to team rooms.
Lotus Collaborative Places	Provides portal community services and portlets that help teams share portal pages, applications, documents, messaging, and other collaboration tools.
Lotus Domino Extended Search	Provides parallel, distributed, heterogeneous searching capability. Searches Lotus Notes databases, legacy data stores, Web search sites, Microsoft Index Server, Site Server and Exchange 2000, as well as multiple Lotus Notes domains.
WebSphere Site Analyzer	Analyzes Web content integrity and site performance, provides usage statistics, and analyzes portal server logs. Reports from site analyzer reveal information that can be used to improve the portal for better user experience.
WEBSPHERE PORTAL EXPERIENCE	
All products in Extend	Same portal server, personalization, and collaboration functions.
IBM Content Manager	Data store for a broad spectrum of digital business information -- from scanned images, facsimiles, and PC files to XML and rich multimedia and web content. Content Manager also provides folder management and document workflow. It provides the content infrastructure for applications from call centers, high-volume claims processing, and accounts payable, to e-commerce catalogues and e-learning.

IBM Tivoli Access Manager	Security policy management tools for e-business and distributed applications.

Table 0.3: WebSphere Portal offerings

WEBSPHERE PORTAL

Portlets
Content and Search
Security
Personalization
Administration
Collaboration
Application Integration
Mobile Portals

Customers choosing one of the WebSphere Portal offerings can realize tangible business and technical benefits:

- Revenue benefits, as a result of tighter relationships with customers or partners, work force productivity, innovation and reduced cycle times
- Operational cost reduction, as a result of operational efficiency, better information flow and knowledge, and consistent infrastructure
- Increased employee productivity and improved decision making because of access to more relevant information, and a single access point to applications and collaboration tools
- Better security and single sign-on, resulting in fewer passwords to administer and better user experience
- Reduced training costs, resulting from common presentation and a consistent user interface
- Unification of applications, giving them a longer useful life, with new ways of accessing them through desktop and pervasive devices.

The WebSphere Portal offerings allow an e-business to quickly capitalize on their digital and human assets while presenting a first-class Web experience to their employees, partners, and customers.

This document is intended to help independent software vendors and application architects plan their use of WebSphere Portal. It explains portal applications, content, security, (authentication, authorization, and single sign-on), user management, administration, personalization, collaboration, and more.

Portlets

Content and Applications for the Portal

Portlets are the heart of a portal. The term “portlet” refers to a small portal application, usually depicted as a small box in the web page. Portlets are reusable components that provide access to applications, web-based content, and other resources. Web pages, web services, applications, and syndicated content feeds can be accessed through portlets. Companies can create their own portlets or select from a catalog of portlets created by IBM and by IBM business partners.

Any particular portlet is developed, deployed, managed, and displayed independent of other portlets. Administrators and end users create personalized portal pages by choosing and arranging portlets, resulting in web pages such as the one shown in Figure 1.1.

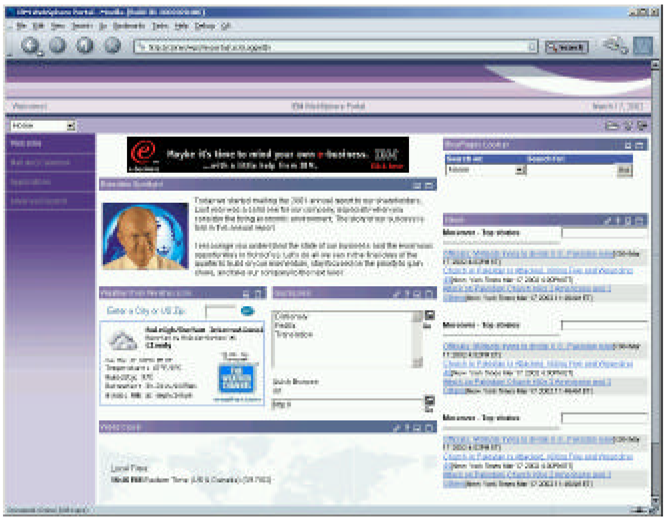

Maybe it's time to mind your own e-business. IBM
...with a little help from IBM.
Enter a City or US Zip
Local Time

Figure 1.1: A typical portal page

The portal server already includes a rich set of standard portlets for displaying syndicated content, performing XML transformation, accessing existing web pages, Lotus Notes and Microsoft Exchange productivity applications, Sametime instant messaging and Lotus QuickPlace team rooms. The following table lists a few of the portlets that are available from IBM.

IBM PORTLETS	
Banner Ad	Displays a predefined banner ad.
Bookmarks	Displays bookmarks for visiting commonly visited websites.
Clipping	Clipping allows the administrator to clip content, images, tables, forms, table cells, and more from websites.
Comma Separated File	Displays data from a file in a table.
EIP Search	Presents an interface to search capabilities provided by a local or remote Enterprise Information Portal server.
File Viewer	Browse an Excel, PowerPoint, Word, PDF or RTF file.
File Download	Allows a user to connect to an FTP or HTTP file server, and download files.
Host On Demand	Establish Host On Demand sessions in a portlet.
JSP	Displays a Java Server Pages fragment.
Multimedia Player	Includes QuickTime, Crescendo Player, Flash Player, Real Player and Media Player portlets.
SQL Portlet	Displays the results of SQL queries and displays a list of bookmarks for saved queries.
Image Viewer	Displays an image from a URL. It allows the user to select a URL and modify the zoom level.
LDAP Search	Allows users to search any LDAP directory that uses the inetOrgPerson schema.
Lotus iNotes	Opens a user's calendar, address book, e-mail, or to-do list.
Lotus Notes Mail (IIOP)	Provides access to a user's Lotus Notes mail database for devices that support HTML or WML browsing.
Lotus QuickPlace	Launches Lotus QuickPlace.
Lotus Sametime	Launches Lotus Sametime in a browser.
Microsoft Exchange 2000	Provides access to a user’s Exchange e-mail.
Microsoft Exchange 5.5	Connects to an Exchange 5.5 server by interfacing with ASP files installed on the Exchange server.
Moreover News	Allows users to view many news headlines compiled by Moreover, such as consumer health news, travel news, etc.
POP3 & IMAP	Receive and Send web-based e-mail.
Reminders	Lets users save and display short text messages.
Web Search	Lets the user search the web using popular search engines.
World Clock	Allows users to view current local times around the world.
XSL Portlet	Displays the result of an XML/XSL transformation

Table 1.2: Sampling of IBM portlets

An extensive third party partner program makes many more portlets available from the portlet catalog download site [1]. The following table includes a sampling of portlets available from IBM's business partners:

PARTNER PORTLETS	
Atomica	Answers on demand, accesses Atomica's comprehensive set of general and industry-specific references.
Crystal Decisions	Crystal portfolio of managed reports.
divine	Portlets for content contribution, approval delivery, and management.
Factiva	Search and Track Modules with a free 30-day preview.
FatWire	Content management portlets.
Hoovers	Business news, industry news, SEC filings, travel guides, and search portlets for Hoover's online information.
Inktomi	Portlet for Inktomi's search engine, with a free 30 day trial available.
Kivera	An evaluation portlet that demonstrates location-based services, including mapping, driving directions and business listing searches.
OnePage	Uses the Content Connect Studio Application to build portlets from information in html pages, databases, applications, EJB's, XML.
Presence Online	The Aptrix portlet client displays the Aptrix web page within a portlet window.
Screaming Media	Breaking news headlines.
Stellent	Portlets to submit, manage, search and approve business content.
YellowBrix	Business intelligence, stock prices, news, and company profiles, plus horoscopes and weather forecasts.

Table 1.3: Sampling of business partner portlets

Portlet Applications

Portlets are more than simple views of existing web content. A portlet is a complete application, following a standard model-view-controller design. Portlets have multiple states and view modes, plus event and messaging capabilities.

Portlets run inside the portlet container of a portal server, similar to a servlet running on an application server. The portlet container provides a runtime environment in which portlets are instantiated, used, and finally destroyed. Portlets rely on the portal infrastructure to access user profile information, participate in window and action events, communicate with other portlets, access remote content, lookup credentials, and to store persistent data.

Generally, portlets are administered more dynamically than servlets. For example, portlet applications consisting of several portlets can be installed or removed while the server is running. The settings and access rights of a portlet can be changed by an administrator while the portal is running, even in a production environment.

PORTLET MODES

Portlet modes allow a portlet to display a different user interface, depending on the task required of the portlet. A portlet has several modes of display, which can be invoked by icons on the portlet title bar: **view**, **help**, and **edit**.

A portlet is initially displayed in its view mode. As the user interacts with the portlet, it may display a sequence of view states, such as forms and responses, error messages, and other application-specific states.

Help mode is used to provide user assistance about the portlet. Edit mode provides a page for users to change the portlet settings. For example, a weather portlet might provide an edit page for users to specify their location. Users must be logged into the portal to access edit mode.

Each portlet mode can be displayed in normal, maximized or minimize states. When a portlet is maximized, it is displayed in the entire body of the portal page, replacing the view of other portlets. When a portlet is minimized, only the portlet title bar is displayed on the portal page.

Portlet API

Portlets are a special subclass of `HttpServlet`, with properties that allow them to easily plug into and run in the portal server. Portlets are assembled into a larger portal page, with multiple instances of the same portlet displaying different data for each user. Portlets rely on the portal infrastructure to access user profile information, participate in window and action events, communicate with other portlets, access remote content, lookup credentials, and to store persistent data. The portlet API provides standard interfaces for these functions.

The portlet API defines a common base class and interfaces for portlets, in order to cleanly separate the portlet from the portal infrastructure. In most respects, the portlet API is an extension of the servlet API, except that it restricts certain functions to a subset that makes sense for portlets running in the context of a portal. For example, unlike servlets, portlets may not send errors or redirects as a response. This is only be done by the portal itself, which controls the overall response page.

The markup fragments that portlets produce may contain links, actions and other content. The Portlet API defines URL rewriting methods that allow portlets to transparently create links, without needing to know how URLs are structured in the particular portal.

PORTLET PERFORMANCE

Since portlets are servlets, similar reentrancy and performance considerations apply to both. A single portlet instance is shared among all requesters. There are a limited number of threads that process portlets and servlets, so it is important for each portlet to do its job as quickly as possible so that response time for the whole page is optimized.

Just as with servlet programming, you should consider optimizations such as limiting the use of the synchronized methods, limiting the use of expensive string operations, avoiding long running loops, and minimizing the number of objects that are created. Another optimization is to use Java Server Pages for rendering the portlet's views; in general, views created with Java Server Pages are faster than views created with XSL.

Usually, many portlets are invoked in the course of handling a single request, each one appending its content into the overall page. Some portlets can be rendered in parallel, so that the portal server assembles all the markup fragments when all the portlets finish or time out. This improves the performance of portlets that access remote data by HTTP or SOAP requests. However, not all portlets are thread-safe; for example, portlets that access protected resources or which use Enterprise Java Beans may not be run in parallel. The portlet deployment descriptor indicates whether the portlet is considered thread-safe; portlets that are not thread-safe will be rendered sequentially.

Portlet output can also be cached. The caching policies are configured in the portlet deployment descriptor, including an expiration time and whether or not the portlet markup can be shared among users or is user-specific.

STANDARDS

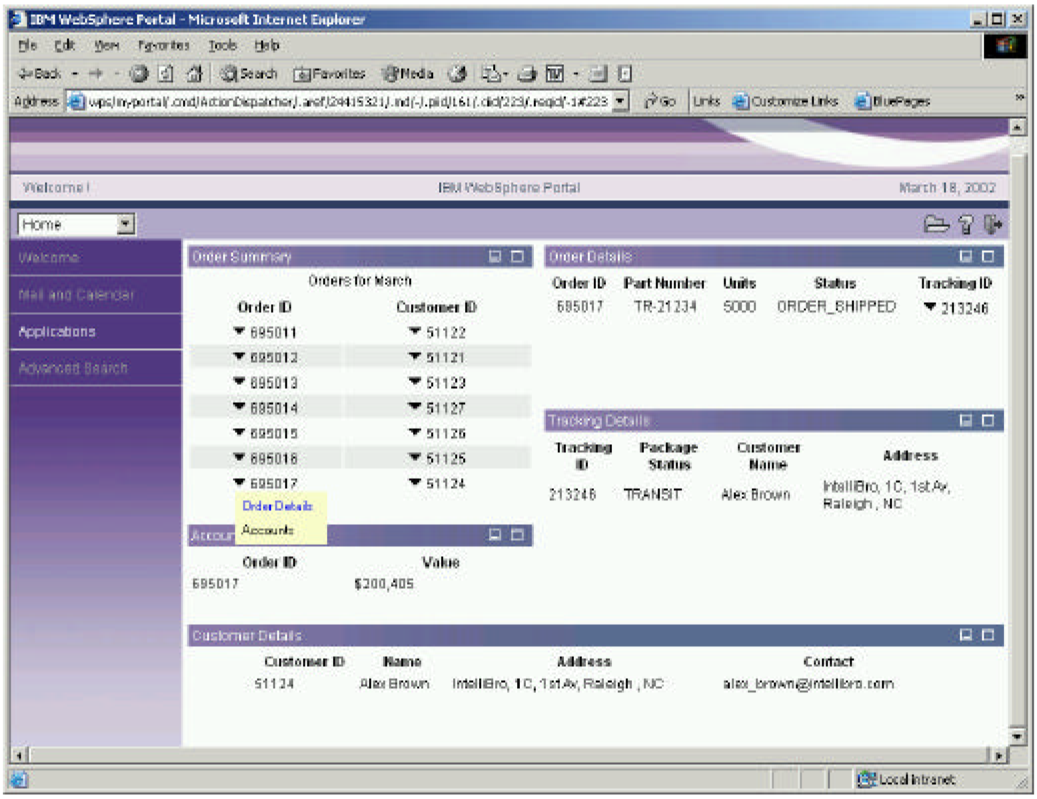
As portals continue to evolve as the new desktop and integration standard, IBM is leading efforts to standardize the application programming interfaces between portals and other applications. In particular, the Java Community Process (JCP) and the Organization for the Advancement of Structured Information Standards (OASIS) are working cooperatively to standardize the Java and XML technology needed to link portals to disparate applications.

JSR 168 [18] will be co-led by IBM and Sun, and seeks to achieve interoperability between local portlets and portal servers. IBM initiated the Portlet API, and will be responsible for providing an open source reference implementation at the Apache Jakarta project. Sun will edit the specification and will provide the freely accessible Compliance Test Kit.

OASIS recently announced the formation of the Web Services for Remote Portals (WSRP) Technical Committee. Chaired by IBM, the WSRP [19] committee has the charter to create an XML and Web services standard that will allow the interoperability of visual, user-facing services with portals or other web applications.

Portlet Communications

The portal server provides a mechanism for portlets to communicate with each other, exchanging data or other messages. In a production portal, portlet communication could be used to copy common data between portlets. This saves redundant typing by the user and makes the portal easier to use. For example, one portlet might display information about accounts while a second portlet displays information about transactions that have occurred for one of the accounts over the last 30 days. To do this, the transactions portlet needs to obtain the corresponding account information when it displays the transaction details.


IBM WebSphere Portal - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address wps/myportal/.cmd/ActionDispatcher/.aref/24415321/.md/-/.pid/161/.cid/223/.reqid/-1#223
Go Links Customize Links BluePages
Welcome!
IBM WebSphere Portal
March 18, 2002
Home
Welcome
Mail and Calendar
Applications
Advanced Search
Order Details
Accounts
Local intranet

Order Summary

Orders for March

Order ID	Customer ID
695011	51122
695012	51121
695013	51123
695014	51127
695015	51126
695016	51125
695017	51124

Order Details

Order ID	Part Number	Units	Status	Tracking ID
695017	TR-21234	5000	ORDER_SHIPPED	213246

Tracking Details

Tracking ID	Package Status	Customer Name	Address
213246	TRANSIT	Alex Brown	IntelliBro, 1C, 1st Av, Raleigh , NC

Accounts

Order ID	Value
695017	$200,405

Customer Details

Customer ID	Name	Address	Contact
51124	Alex Brown	IntelliBro, 1C, 1st Av, Raleigh , NC	alex_brown@intellibro.com

Figure 1.4: A portlet communication scenario

This is accomplished by communication between the two portlets, using portlet **actions** and portlet **messages**. In this example, the account portlet creates a portlet action and encodes it into the URL that is rendered for displaying transactions. When the link is clicked, the **action listener** is called, which then sends a portlet message to send the necessary data.

CLICK-TO-ACTION

Using the event and message feature helps unify portlet applications that access different backend applications. IBM provides portlets that enable automatic data exchange based on user interaction. These are called **click-to-action** portlets.

The objective of the click-to-action portlets is to increase the productivity of portal users working with multiple portlets by easily enabling them to send information from one portlet to another. For example, users can click on information that is displayed in one portlet and transfer that information to another portlet. The portlet receiving the information processes it and updates its display.

Click-to-Action automatically matches the portlet information sources and possible actions based on their data type compatibility. Click-to-action does not rely on drag and drop or other non-standard browser features. A unique advantage of click-to-action is that is it designed to work in different browsers, making it more accessible to users.

Discoverable Services

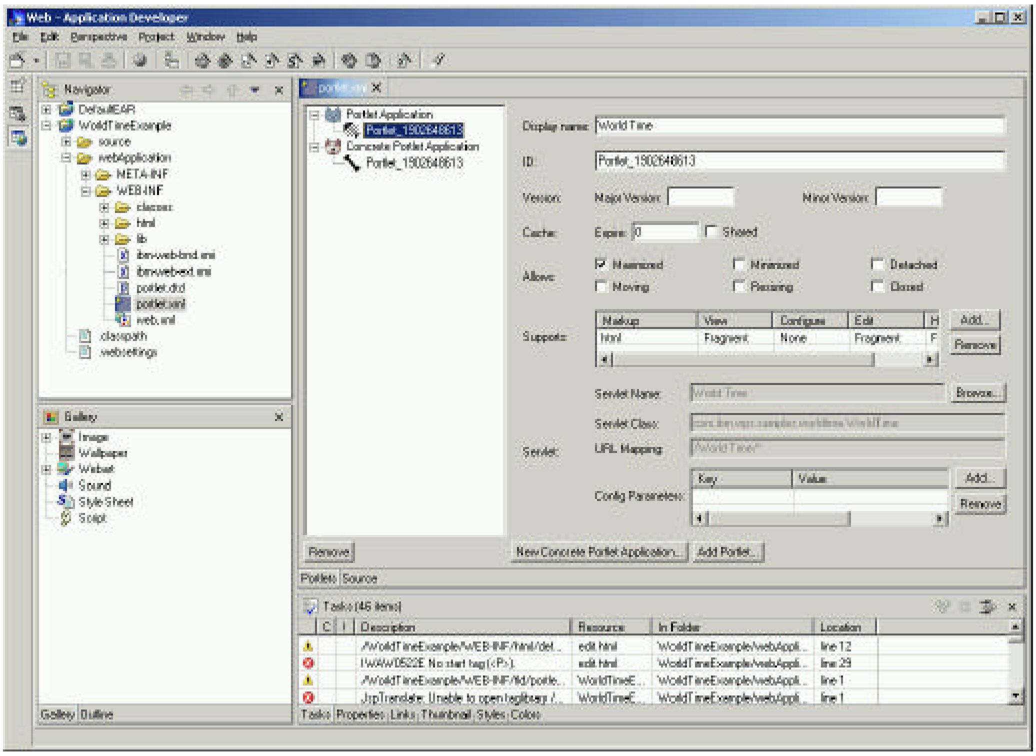
The portlet API provides an interface to enable dynamic discovery of available services. Each service is registered in a portal configuration file, and is accessed from the `PortletContext.getService()` method, which looks up the factory for the service, creates the service, and returns it to the portlet. This makes services available to all portlets without having to package the service code with the portlet. The implementation of such a service can be exchanged or enhanced transparently, without affecting the portlet.

The portal server provides discoverable services for its credential vault, for managing persistent TCP/IP connections, and for managing the portal's content repository. New services may be implemented by portal developers, such as search, location, notification, content access, or mail services.

Tools

Portlets can be grouped together in a **portlet application**. Portlet applications are distributed and deployed using web archive files (WAR). There are portlet-specific extensions to the standard web application deployment descriptor.

WebSphere Studio Application Developer provides an excellent development, test, and debug environment for portlet applications. You can implement the Java classes for portlet classes, and also test and debug the Java code. Studio also provides tools for creating Java Server Pages, HTML pages, images, and other related portal resources. The portal toolkit (downloaded from the web) provides plugins to help build the portlet deployment descriptor and package it into a WAR file, and then easily deploy it to the runtime portal server.



Display name: World Time

ID: Portlet_1902648613

Version: Major Version: Minor Version:

Cache: Expire: 0 □ Shared

Allows: ☑ Maximized □ Minimized □ Detached □ Moving □ Resizing □ Closed

Supports:

Markup	View	Configure	Edit	H
html	Fragment	None	Fragment	F

Servlet: Servlet Name: World Time

Servlet Class: com.ibm.wps.samples.worldtime.WorldTime

URL Mapping: /WorldTime/*

Config Parameters:

Key	Value

Tasks (46 items)

	C	!	Description	Resource	In Folder	Location
⚠			/WorldTimeExample/WEB-INF/html/del...	edit.html	WorldTimeExample/webAppli...	line 12
⊗			IWAW0522E No start tag (<P>).	edit.html	WorldTimeExample/webAppli...	line 29
⚠			/WorldTimeExample/WEB-INF/tld/portle...	WorldTimeE...	WorldTimeExample/webAppli...	line 1
⊗			JspTranslate: Unable to open taglibrary /...	WorldTimeE...	WorldTimeExample/webAppli...	line 1

Figure 1.5: WebSphere Studio Application Developer with portal plugins

2

Content and Search

View, Organize, and Share Portal Content

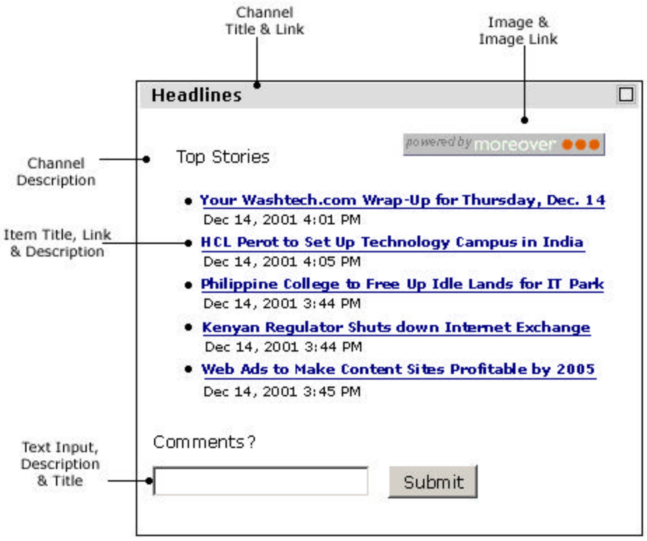
When companies deploy portals, they have a significant need for content aggregation, rapid and large-scale deployment, search, and delivery of personalized content. WebSphere Portal meets the portal’s content delivery needs by supporting syndicated content, by integration with web content management systems, and by providing built-in content organizing portlets.

Syndicated Content

A key concept related to portal technology is **syndication,** which is about delivering fresh, personalized, and filtered content and services from multiple sources to subscribers. Typically, the content is related to news, finance, and entertainment. Portal partners include popular content providers, such as Moreover, YellowBrix, Hoovers, Factiva, NewsEdge, MediaApps, DataMonitor, and Screaming Media and others.

Companies are embracing syndication concepts and standards to automate the publishing of electronic catalogs and other internal information, and to make this information available to workers through enterprise portals.

A popular and useful format for syndicated news and entertainment content is Rich Site Summary (RSS) [6]. Content can be published directly from the content management system into Rich Site Summary and Open Content Syndication (OCS) channels, where it can easily be displayed by the portal server’s built-in RSS portlet. This **self-syndication** concept defines a procedure for editing, managing, and publishing your own sources of content.



Headlines

Top Stories

- Your Washtech.com Wrap-Up for Thursday, Dec. 14
  Dec 14, 2001 4:01 PM
- HCL Perot to Set Up Technology Campus in India
  Dec 14, 2001 4:05 PM
- Philippine College to Free Up Idle Lands for IT Park
  Dec 14, 2001 3:44 PM
- Kenyan Regulator Shuts down Internet Exchange
  Dec 14, 2001 3:44 PM
- Web Ads to Make Content Sites Profitable by 2005
  Dec 14, 2001 3:45 PM

Comments?

Figure 2.1: A Rich-Site Summary example

Web Content Management

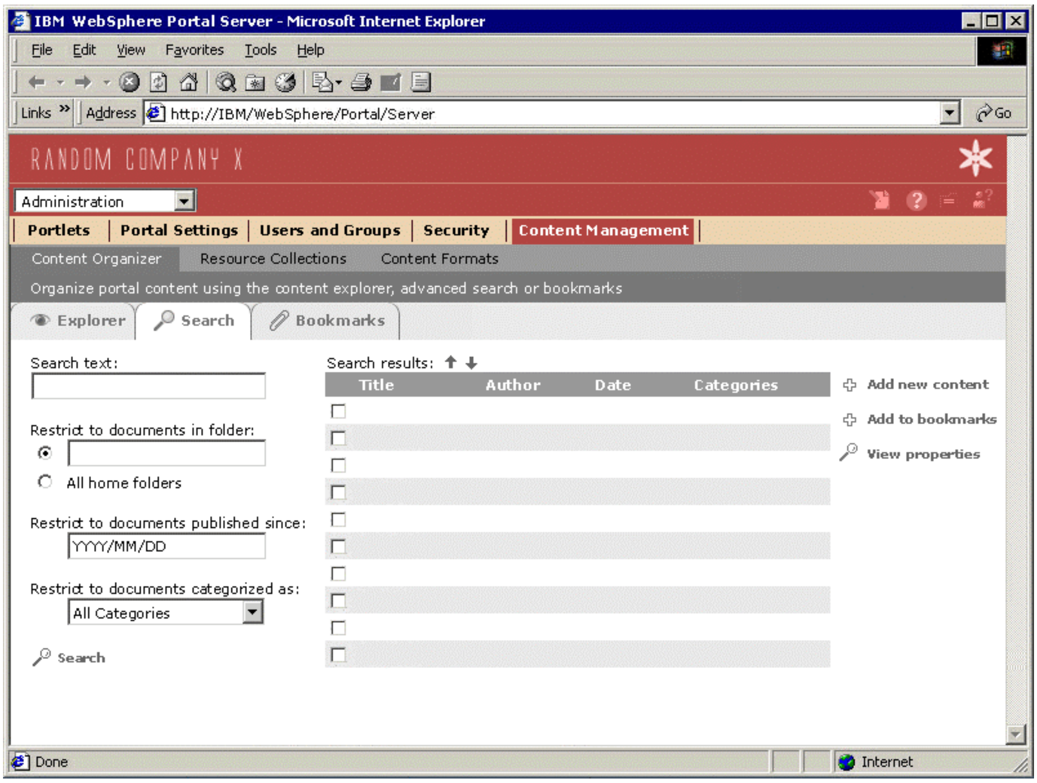
Web content management deals with creating, approving and publishing Web content from content creators to Web servers. The steps of this process include defining content types, roles, publication options, destination specifications and workflow processes. There are many content management vendors in the marketplace today, including Interwoven, Vignette, Documentum, FatWire, Stellant, Aptrix, and others.

Although each of these products works differently, generally, they are designed to create, maintain and publish collections of documents that can be made available to users via the portal. To illustrate how this done, WebSphere Portal includes integration kits that illustrate specific steps of how to publish RSS content from several of the web content management products.

Content contribution and approval operations of the web content management system can also be accessed through portlets that are provided by their respective companies. These portlets provide a user interface into various aspects of the content management process, such as content submission, workflow management, content approval, and even staging or publishing.

Content Organizer

The portal server includes a content organizer portlet shown in Figure 2.2, which enables portal users to contribute and share documents. The content organizer portlet provides a workspace for storing, navigation, viewing, and searching portal documents and other content. The organizer is pre-configured to work with plain text files, HTML files, syndicated content (RSS) and some XML formats. Additional content types, formats, and back-end systems can be integrated easily.


IBM WebSphere Portal Server - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Links Address
Go
RANDOM COMPANY X
Portlets | Portal Settings | Users and Groups | Security | Content Management
Content Organizer Resource Collections Content Formats
Organize portal content using the content explorer, advanced search or bookmarks
Explorer Search Bookmarks
Search results:
Title Author Date Categories
Add new content
Add to bookmarks
View properties
Search
Done
Internet

Address: http://IBM/WebSphere/Portal/Server

Administration

Search text:

Restrict to documents in folder:
●
○ All home folders

Restrict to documents published since:
YYYY/MM/DD

Restrict to documents categorized as:
All Categories

Figure 2.2: Portal Content Organizer.

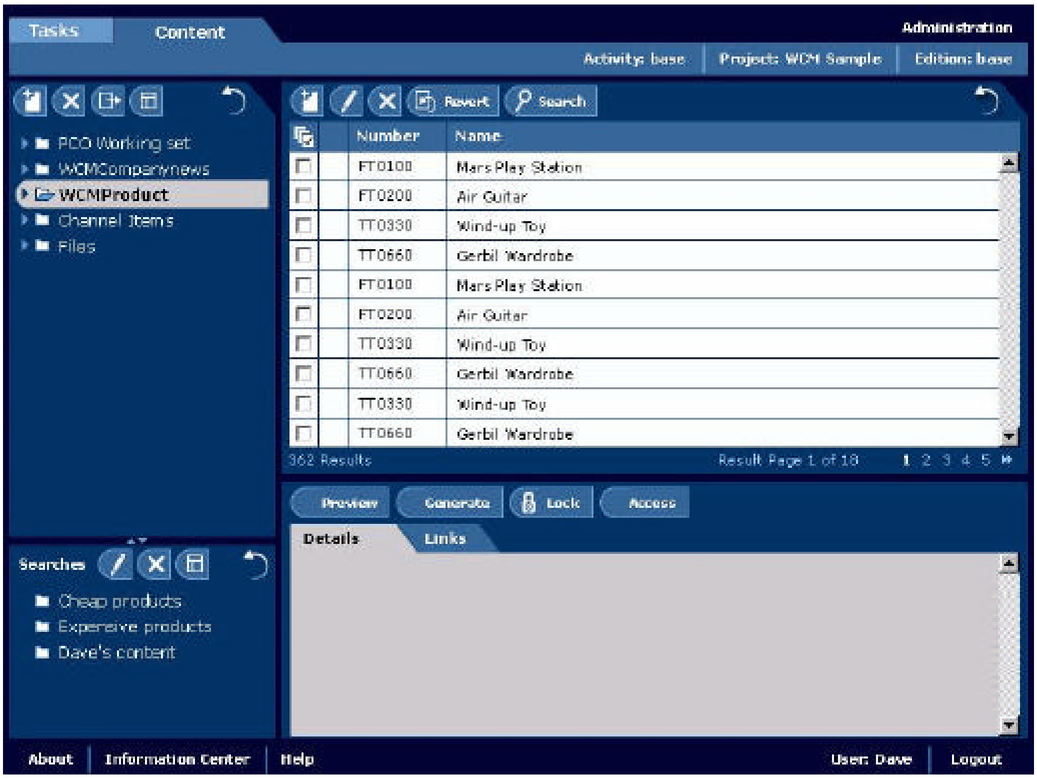
This portlet help users organize the content they have seen, content they want to read or content they want to share with colleagues. The content organizer maintains properties and attributes of content, assigns unique ids to documents, and handles conversion among several content formats. Content maintained by the content organizer can be searched by the portal's built-in search service.

The content model used by the organizer is a lightweight, open, extensible content model based upon the WebSphere Personalization resource engine. The Web Content Publisher tools can be used to publish documents and make them available for use in the portal. By implementing the resource engine's content model interfaces, third party web content management systems can also be integrated.

Web Content Publishing

The Web Content Publisher is a tool intended for users that need to contribute content to a web site on a regular or occasional basis. This includes business users creating template-driven content such as press releases or product information, and graphic artists creating and editing artwork using their favorite tools. Content Builder supports content contribution via templates (forms), as well the contribution of files such as images, HTML or JSPs which are created and edited using popular tools such as HomePage Builder, Dreamweaver, FrontPage, PhotoShop or Word.

Web Content Publisher provides a web browser interface that enables users to contribute content to a web site in an easy to use, quick, and controlled manner. This allows large teams to work together, jointly where necessary, but with enough isolation that they aren't constantly tripping over each other's changes.



Number	Name
FT0100	Mars Play Station
FT0200	Air Guitar
TT0330	Wind-up Toy
TT0660	Gerbil Wardrobe
FT0100	Mars Play Station
FT0200	Air Guitar
TT0330	Wind-up Toy
TT0660	Gerbil Wardrobe
TT0330	Wind-up Toy
TT0660	Gerbil Wardrobe

362 Results Result Page 1 of 18 1 2 3 4 5

Figure 2.3: Content publishing tools

You can use the content publishing tools to manage content that is served through the content organizer or other portlets within the portal server. It is also possible to categorize content and to schedule the import of syndicated content.

WORKFLOW

Web Content Publisher manages the task lists of each user, maintains access control over what content each user can see or change, and coordinates the approval and publishing process when the content is ready.

To set up workflow, use Lotus Workflow Architect to graphically design one or more processes. Each step in the process is assigned to a specific user role, such as an editor or an approver. When the process starts, a work item is put on the work queue of each person in the specified role. As users claim activities and complete the steps, more work is put on the next queue defined in the workflow.

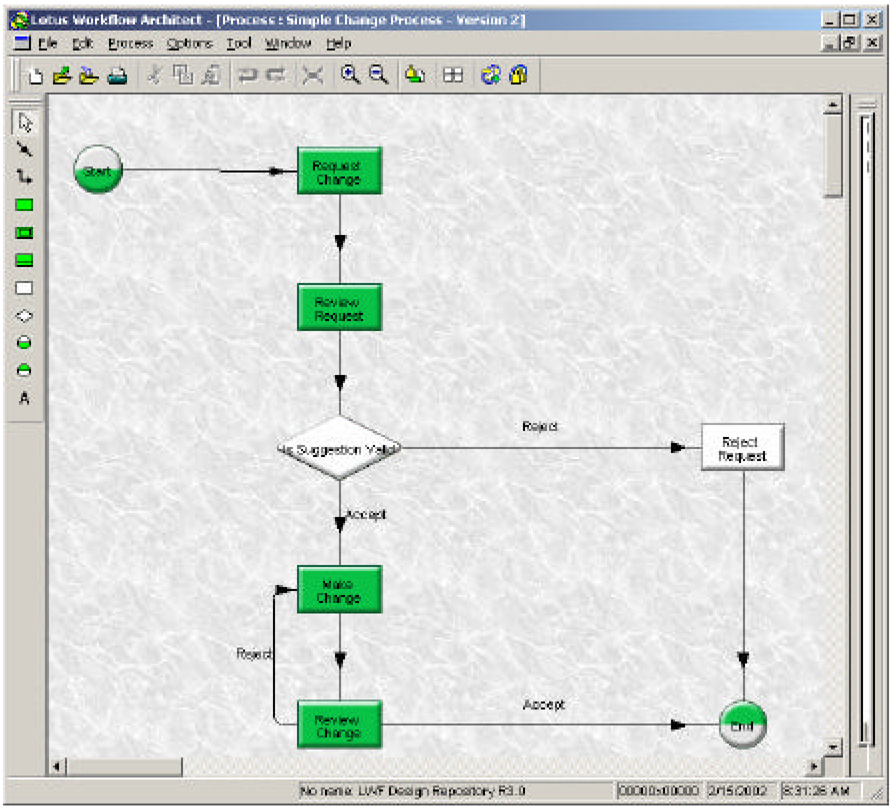

Lotus Workflow Architect - [Process : Simple Change Process - Version 2]
File Edit Process Options Tool Window Help
Start
Request Change
Review Request
Is Suggestion Valid
Reject
Reject Request
Accept
Make Change
Reject
Review Change
Accept
End
No name LWF Design Repository R3.0
00000:00000
2/15/2002
8:31:26 AM

Figure 2.4: Content publishing workflow

DIGITAL CONTENT LIBRARIES

For managing and searching large collections of digital content, the WebSphere Portal Experience offering includes IBM Content Manager [13]. Additional search portlets from leading search vendors (Verity, Inktomi, Autonomy) are also available and can be found on the portlet catalog.

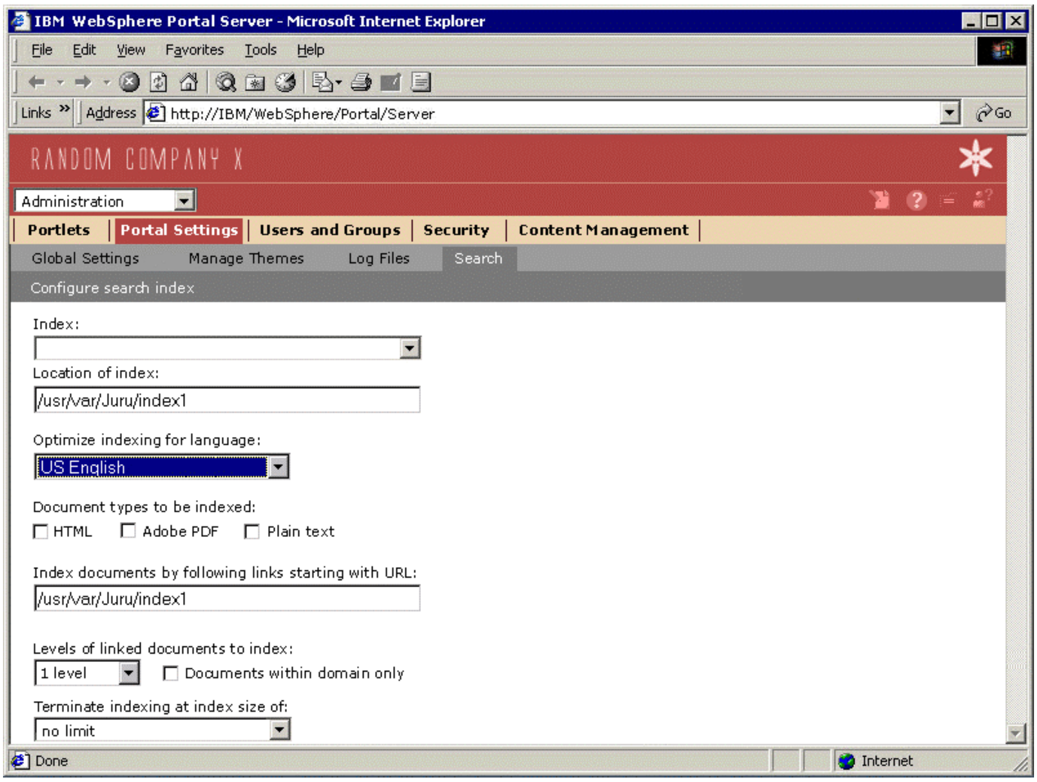
Search

WebSphere Portal provides integrated text search capabilities, including a search portlet, a crawler, and a document indexer. The search service can search the portal's document repository as well as Internet content.

The portal server's built-in search engine is optimized for full-text searching of small and medium-sized collections where precision is essential. It efficiently applies state-of-the-art search algorithms producing high quality search results.

The search engine supports free-text queries, with query assistance and query word completion. Search queries use advanced query operators (+ or -) to indicate keywords that must be in the document or keywords that must not be in the document. The search engine can search documents in any language, and also supports synonyms and stop word lists. Search results include document summarization and search results clustering.

To prepare for searching, the search engine builds a full text index in order to search documents that are stored in the local file system. The indexer supports multi-word indexing for disambiguation and high precision. The index can be compressed, and the size can be controlled for situations where the size of the index needs to be limited. Administration portlets are provided for creating, updating, and managing the index.


IBM WebSphere Portal Server - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Links Address http://IBM/WebSphere/Portal/Server Go
RANDOM COMPANY X
Administration
Portlets | Portal Settings | Users and Groups | Security | Content Management
Global Settings Manage Themes Log Files Search
Done Internet

Configure search index

Index:

Location of index: /usr/var/Juru/index1

Optimize indexing for language: US English

Document types to be indexed: □ HTML □ Adobe PDF □ Plain text

Index documents by following links starting with URL: /usr/var/Juru/index1

Levels of linked documents to index: 1 level □ Documents within domain only

Terminate indexing at index size of: no limit

Figure 2.5: Managing the search index

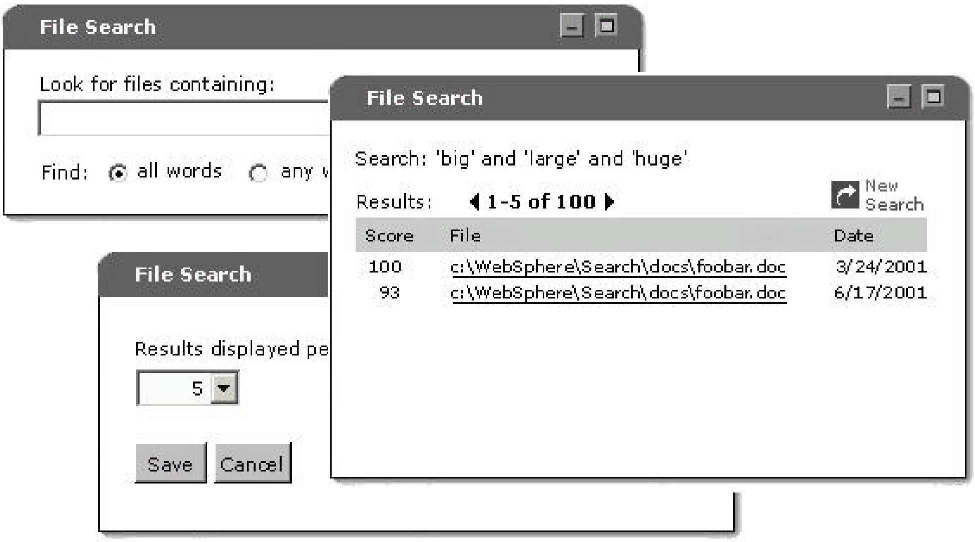
Federated Search

WebSphere Portal also integrates several other search technologies. Portlets using Lotus Domino Extended Search and IBM Enterprise Information Portal search can access and aggregate other search engines and indexes in a distributed fashion.

Customers seeking support for large document collections or support for searching a wide range of document types and data sources should consider using Extended Search (included with the WebSphere Portal Extend offering) or IBM's Enterprise Information Portal [12].

LOTUS DOMINO EXTENDED SEARCH

Lotus Domino Extended Search (ES) provides distributed, heterogeneous searching across Domino servers, databases, and the Internet, without the user having to know the details of these various systems. The result is single-point of access to a variety of data sources without requiring a new, central index. ES can search and retrieve documents from repositories that include Lotus Notes 4.X and 5.X, Domino.doc, and R5 Domain Index.



Search: 'big' and 'large' and 'huge'

Results: ◀ 1-5 of 100 ▶

Score	File	Date
100	c:\WebSphere\Search\docs\foobar.doc	3/24/2001
93	c:\WebSphere\Search\docs\foobar.doc	6/17/2001

Figure 2.6: Extended Search portlets

ES also searches external sources such as Microsoft Index Server and Site Server, LDAP-compliant directories, 18 popular Web search sites and News sites, commercial content providers, and relational databases such as IBM DB2, Oracle, Sybase, MS SQL-Server, and other ODBC compliant databases. Results can be ranked by relevancy over multiple data stores.

ADVANCE SEARCH AND DOCUMENT PROCESSING

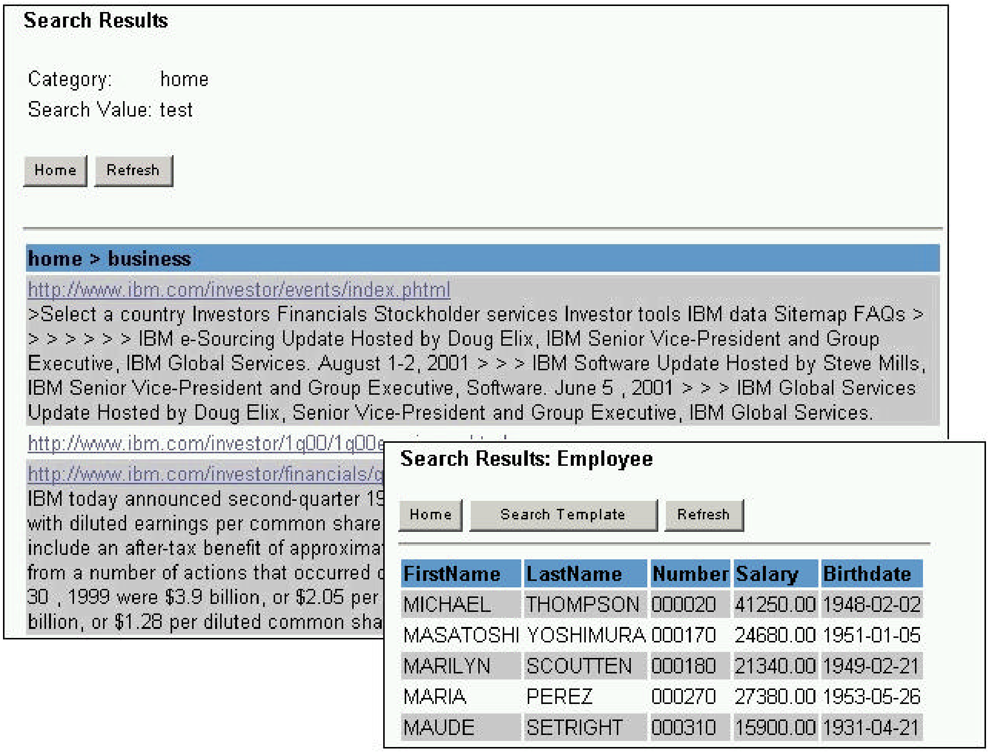
IBM Enterprise Information Portal (EIP) can manage data access across multiple sources such as content management repositories, email systems, relational databases, file systems, web sites (both intranet and Internet), and more. EIP integrates data sources across the enterprise, with a unified set of APIs to simplify programming and speed development and deployment, while providing an interface layer that isolates portal applications from changes to underlying data repositories.

Documents can be full-text indexed/searched using the EIP crawler and text search features. Formatted document types handled by INSO technology are supported, in addition to standard markup text such as HTML and XML. Documents can be categorized into taxonomies, enabling search by category. APIs are provided for capturing and storing other metadata about documents.

EIP provides connectors for a variety of repositories, provided by IBM, Lotus, and other vendors such as Documentum and Filenet. Federated searches can be applied across multiple such repositories, and can exercise searching based on metadata, full text, and other specialized search properties such as Query by Image Content (QBIC) - depending on the search services enabled for each repository.

The Text Analysis features of EIP support creating full-text indices, and subsequent searching across all the text portions of the content sources configured for use in the portal. Sources can be accessed for indexing via the web crawler, or via a metadata search.

Portlets for accessing EIP advanced and federated search functions are available from the portlet catalog.



Search Results

Category: home
Search Value: test

home > business

http://www.ibm.com/investor/events/index.phtml
>Select a country Investors Financials Stockholder services Investor tools IBM data Sitemap FAQs > > > > > > > IBM e-Sourcing Update Hosted by Doug Elix, IBM Senior Vice-President and Group Executive, IBM Global Services. August 1-2, 2001 > > > IBM Software Update Hosted by Steve Mills, IBM Senior Vice-President and Group Executive, Software. June 5 , 2001 > > > IBM Global Services Update Hosted by Doug Elix, Senior Vice-President and Group Executive, IBM Global Services.

http://www.ibm.com/investor/1q00/1q00e

http://www.ibm.com/investor/financials/g
IBM today announced second-quarter 19
with diluted earnings per common share
include an after-tax benefit of approxima
from a number of actions that occurred o
30 , 1999 were $3.9 billion, or $2.05 per
billion, or $1.28 per diluted common sha

Search Results: Employee

FirstName	LastName	Number	Salary	Birthdate
MICHAEL	THOMPSON	000020	41250.00	1948-02-02
MASATOSHI	YOSHIMURA	000170	24680.00	1951-01-05
MARILYN	SCOUTTEN	000180	21340.00	1949-02-21
MARIA	PEREZ	000270	27380.00	1953-05-26
MAUDE	SETRIGHT	000310	15900.00	1931-04-21

Figure 2.7: Advanced and federated search portlets

3

Security

User Management, Access Control, and Single Sign-on

With the explosive growth of business-to-business and business-to-consumer Web applications, e-businesses need to protect critical information assets from intruders and hackers. Service providers need similar protection when re-hosting e-business content and applications for their customers. Portal applications and resources are protected by

- Managing user profiles (member services)
- Verifying user identity (authentication)
- Managing access to backend applications (single sign-on)
- Enforcing access policies (authorization)

This chapter highlights some of the security features that applications and portal administrators can exploit, to better protect the portal's valuable information assets.

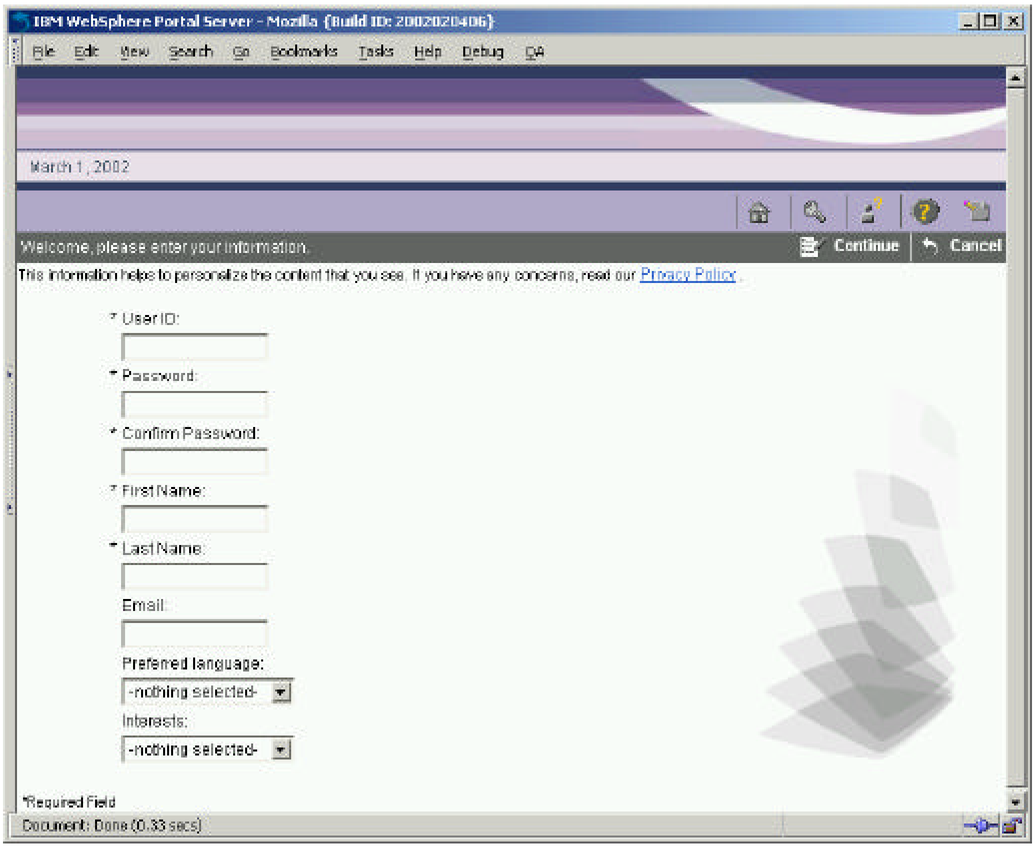
SUPPORTED LDAP DIRECTORIES

- Lotus Domino
- IBM SecureWay® Directory
- iPlanet™ Directory Server
- Microsoft® Active Directory

Member Services

Centralized administration of user identities, credentials and permissions is desirable in many environments. The portal server includes facilities for defining portal users and managing user access rights.

The user and group subsystem includes Web pages where users can register and manage their own account information, administration portlets for managing user accounts and group information, plus a repository that stores all the information about portal users. It provides services to create, read, update, and delete users or groups in the repository. User profile information includes personal information such as a user's name and user id, plus preference information such as news topics of interest, preferred language, etc. A user may be a member of one or more groups, and groups can contain other groups.


IBM WebSphere Portal Server - Mozilla {Build ID: 2002020406}
File Edit View Search Go Bookmarks Tasks Help Debug QA
Continue
Cancel

March 1, 2002

Welcome, please enter your information.

This information helps to personalize the content that you see. If you have any concerns, read our Privacy Policy.

* User ID:

* Password:

* Confirm Password:

* First Name:

* Last Name:

Email:

Preferred language: -nothing selected-

Interests: -nothing selected-

*Required Field

Document: Done (0.33 secs)

Figure 3.1: Self-enrollment page

SUPPORTED DATABASES

- DB2
- Oracle

The default set of user profile attributes is based on the inetOrgPerson schema, which is supported by most LDAP directories:

BusinessCategory
cn
departmentNumber
employeeNumber
employeeType
facsimileTelephoneNumber
givenName
homePhoneShipments
homePostalAddress
mail
manager
mobile
pager
postalAddress
postalCode
preferredLanguage
roomNumber
sn
st
street
telephoneNumber
title
uid
userPassword

The user repository may consist of multiple data sources. By default, the repository consists of two data sources: it is a combination of a database and a directory server. The database may be either a DB2 or Oracle database. Any one of several LDAP [5] directory products are supported, including the Netscape (iPlanet) Directory Server, Microsoft Active Directory, Domino Name and Address Book, and IBM's SecureWay Directory Server.

The mapping of user profile attributes to LDAP object classes is defined using in the file `wms.xml`. This file specifies the names of the various data repositories, their implementation classes, and the mapping between each attribute in the user object to an attribute in the actual repository.

This mapping file also includes metadata about each attribute such as its data type, whether it is required, whether it can have multiple values, etc.

```
<?xml version="1.0"?>
<!DOCTYPE wms SYSTEM "wms.dtd">
<wms profileDataStorage="LDAP+DB" accountNameContext="realmName">
      <repository repositoryName="wmsDB"
                  repositoryImplClass="com.ibm.websphere.wms.wmsDbAdapter"
                  wmsNodes="o=Default Org,o=Root Org"
                  deleteMember="yes" />
      <repository repositoryName="defaultLdapRepository"
                  repositoryImplClass="com.ibm.xxx.xxx.xxx"
                  wmsNodes="o=USdiv,o=Root Org; o=CDNdiv,o=Root Org"
                  deleteMember="yes" />
     ...
    <entry memberType="Person" >
    <wmsAttribute name="firstName"
                              valueType="String"
                              valueLength=128
                              encrypt="no"
                              required="yes"
                              multi-valued="no"
                              searchable="yes"
                              readOnly="false" >
     </wmsAttribute>
      ...
     </entry>
     ...
</wms>
```

The file `defaultLdapRepository.xml` specifies the details of how each attribute is mapped to the LDAP directory.

```
<?xml version="1.0"?>
<!DOCTYPE defaultLdapRepository SYSTEM "pluginRepository.dtd">
<defaultLdapRepository>
         <envProperty name="ldapHost" value="papaya.torolab.ibm.com" />
         <envProperty name="ldapPort"  value="389" />
         <envProperty name="ldapType" value="SecureWay" />
         <envProperty name="userObjectClass"
               value="top;person;organizationalPerson;inetOrgPerson" />
         …
         <entry memberType="Person">
           <map>
              <wmsAttribute   name="businessTitle" />
                 <pluginAttribute name="title" required="no"
                      multi-valued="yes"
                      searchable="yes"  readOnly="false"/>
           </map>
```

```
            </entry>
            ...
    </defaultLdapRepository>
```

ADMINISTRATION

Administration of users and groups can be performed by users themselves (“self care”) or by portal administrators. The portal server includes forms for registering new users (see *Figure 3.1: Self-enrollment page*), as well as administration portlets for updating user and group information.

The registration and self-care forms are easily modified to accommodate new attributes. You can simply add new data entry fields to the form, matching the field identifiers with the new attributes names. The enrollment servlet will automatically store the new data in the corresponding user attributes. The WebSphere Portal InfoCenter includes more information about customizing the implementation of the user repository, registration and self-care pages, and data validation classes.

Authentication

Authentication is the process of establishing a user’s identity. Usually, the portal server uses the authentication services provided by WebSphere Application Server [2]. Another option is to use a third-party authentication server (such as Tivoli Access Manager WebSeal or Netegrity SiteMinder) that has a trusted association with the application server.

Identifying the User

Portal server uses form-based authentication. Form-based authentication means that a user is prompted through an HTML form for the user ID and password for authentication when trying to access the portal. The portal server requests the application server to validate the authentication information against a Lightweight Directory Access Protocol (LDAP) user registry.

WebSphere Application Server uses Lightweight Third Party Authentication (LTPA) as the authentication mechanism. A Common Object Request Broker Architecture (CORBA) Credential is used to represent authenticated users and their group memberships. When a user tries to access a protected resource, the application server intercepts the request and redirects the request to the login form. This form posts the user ID and password to the portal which requests the application server to authenticate the user. If the user can be authenticated, a valid CORBA credential is created and an LTPA cookie is stored on the user's machine.

WEBSPHERE EVERYPLACE SERVER

If your system uses WebSphere Everyplace Server [14], the portal can be configured to use the authentication proxy of WebSphere Everyplace Authentication Server to protect URLs and authenticate users. This allows single sign-on to applications and services provided by Everyplace Server.

In a WebSphere Everyplace Server environment, the authentication server uses HTTP basic authentication. HTTP basic authentication means that a user is prompted by a browser window for the user ID and password for authentication. When a user tries to access the portal, the authentication proxy intercepts the request and sends an authentication request back to the user. When the user responds, the authentication proxy validates the authentication information against the user registry of Tivoli Personalized Services Manager (TPSM).

THIRD-PARTY AUTHENTICATION SERVERS

If your system uses another third-party authentication server, trust needs to be established between that proxy and WebSphere Application Server. This is done using a **Trust Association Interceptor** (TAI) module, which converts security information specific to the authentication proxy into a format that can be handled by the application server. The supported authentication mechanism depends on the capabilities of the third-party product.

When a user tries to access the portal, the third-party authentication proxy intercepts the request and challenges the user to authenticate. After a successful login, the original user request, along with additional security information in the request header, is passed to the application server. The format and content of this information is vendor specific. WebSphere Application Server uses the TAI module (that is specific to the third-party product) to extract the necessary security information from the request header.

TAI modules for IBM Tivoli Access Manager and Netegrity SiteMinder are packaged with the portal server (all editions). The WebSphere Application Server InfoCenter includes information about creating custom TAI modules for other third-party reverse proxy servers.

Single Sign-On

The portal server provides comprehensive single sign-on (SSO) support. Users want to be able to log on only once, and be known to the different parts of the portal server with the same consistent user credentials. Users should not be asked to do multiple logons simply because they access different portal applications.

The portal server supports single sign-on realms using WebSphere Application Server as well as authentication proxies such as IBM Tivoli Access Manager's WebSeal or Netegrity SiteMinder. This means that the user needs to log on only once to gain access to all enterprise applications that are installed within the single sign-on realm.

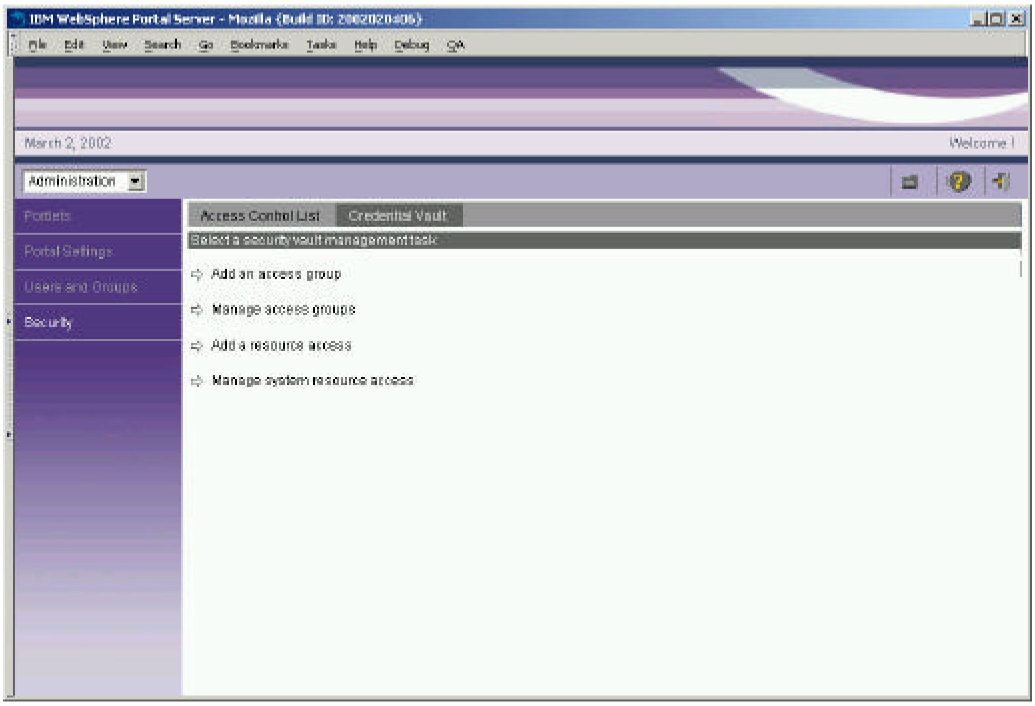
The WebSphere Application Server uses Lightweight Third Party Authentication tokens to provide single sign-on. When a user is authenticated, the portal server creates an LTPA single sign-on cookie containing the authenticated user credential. This encrypted cookie conforms to the format used by WebSphere Application Server and can be decrypted by all application servers in the shared domain, provided they all have the same cipher key. This cookie enables all servers in the cluster to access the user's credentials without additional prompting, resulting in a seamless single sign-on experience for the user. To benefit from the LTPA method of single sign-on, the user's browser must support cookies and have its support for **session cookies** enabled.

Note: *Session* cookies are the temporary kind that applications use to remember and convey transient session information between a set of related HTTP messages. They are unlike the *permanent* type of cookies that are written to the user's file system, which some users find objectionable.

CREDENTIAL VAULT

Passing the User Credential to the Application

Many portlets need to access remote applications that require some form of user authentication. For accessing applications outside the portal's realm, portal server provides a credential vault service that portlets can use to store user ID and password (or other credentials) for a user login to an application. Portlets can use these on behalf of the user to access remote systems. The credential vault supports either local database storage or IBM Tivoli's Access Manager for secure storage and retrieval of credentials.



Select a security vault management task

Figure 3.2: Setting up the credential vault.

Portlets obtain credentials by obtaining a `CredentialVaultPortletService` object and calling its `getCredential` method. With the returned credential, there are two options:

1. Use passwords or keys from a **passive credential**, passing them in application-specific calls. Portlets that use passive credentials need to extract the secret out of the credential and do all the authentication communication with the backend application.

2. Call the authenticate method of an **active credential**. Active credential objects hide the credential's secret from the portlet, with no way to extract it out of the credential. Active credentials provide additional methods to perform the authentication.

The latter case allows portlets to trigger authentication to remote servers using basic authorization, SSL client authentication, digest authentication, or LTPA without knowing the credential values. Using active credentials means that the portal authenticates on behalf of the portlet, and the portlet can simply use the open connection. While this may not be possible for all cases, it is the preferred technique.

For secure transmission of data, portlets can request a secure session (HTTPS) for accessing web applications.

PERSISTENT CONNECTIONS

Portlets that depend on remote connections require some way of maintaining that connection as users navigate through the portal. The portal provides a **persistent backend connection** service, that maintains TCP/IP connections across page changes.

Some remote applications use forms-based logins and store cookies during the login form processing. The `HttpFormBasedCredential` can be used for handling these form-based logins and will store all the cookies that are returned as a result. For subsequent calls, the portlet can then ask the credential for an **authenticated connection**. This gives an HTTP connection with these cookies already set in the header. This way, portlets can maintain persistent, secure back-end connections.

JAVA SECURITY

The portal server implements the Java Authentication and Authorization Service (JAAS) architecture. JAAS provides a means for authenticating subjects and for providing fine-grained access control. JAAS is part of the standard Java security model; it gives applications independence from the underlying authentication and authorization mechanisms being used.

JAAS performs login and logout operations using a modular service provider interface. Credentials that are established through the portal server's JAAS login modules include CORBA credentials, user and group distinguished names, user id and password, and LTPA tokens. In a distributed J2EE environment, portlets can use the JAAS API to access JAAS-enabled backend applications.

Authorization

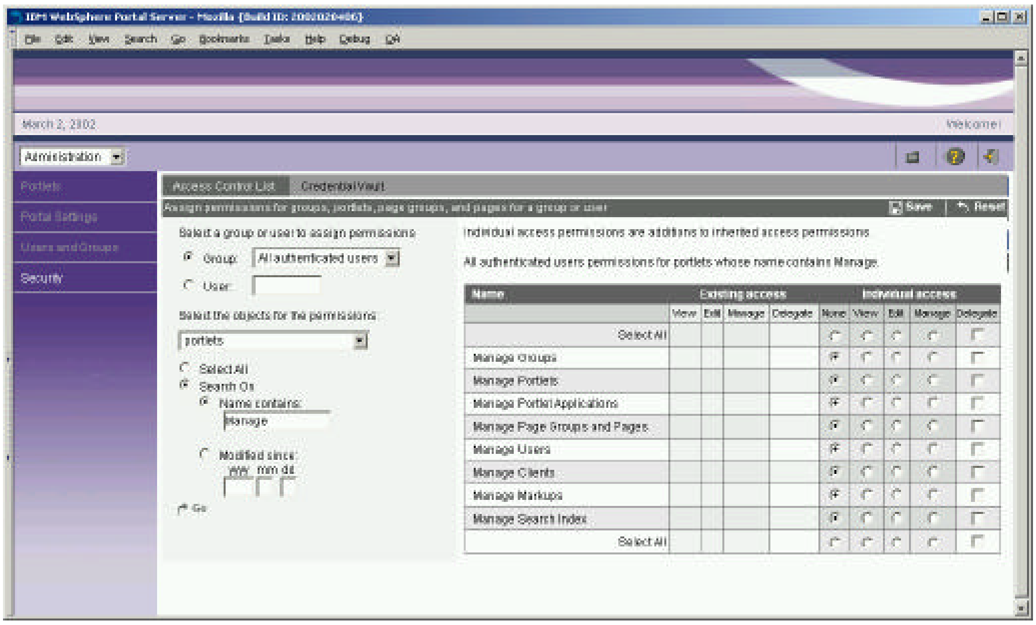
After determining the user's identity, the portal server consults locally cached access control lists to determine which pages and portlets a user has permission to access.

Authorizing the User's Access

The portal server enforces access control to portal assets, including pages, portlets, page groups, and user groups. The access control lists are stored in the portal's administration database. It is also possible to manage access control for specific resources in an external security manager, such IBM Tivoli Access Manager or Netegrity SiteMinder.

IBM Tivoli Access Manager is a robust and secure policy management tool for e-business and distributed applications. It uniquely addresses the challenges of e-business security - escalating costs, growing complexity, and the need for uniform security policies across platforms. 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. Access Manager is included with the WebSphere Portal Experience offering. For more information, see the IBM Tivoli Access Manager Web site [3].

Access permissions are maintained using the Access Control administration portlet. Use this portlet to grant **view**, **edit**, and **manage** permissions to individual users or to groups of users, so that they may access specific portlets, pages, or page groups.



Assign permissions for groups, portlets, page groups, and pages for a group or user

Select a group or user to assign permissions

● Group: All authenticated users

○ User:

Select the objects for the permissions:

portlets

○ Select All

● Search On

● Name contains: Manage

○ Modified since: yyyy mm dd

Individual access permissions are additions to inherited access permissions.

All authenticated users permissions for portlets whose name contains Manage.

Name	Existing access				Individual access				
	View	Edit	Manage	Delegate	None	View	Edit	Manage	Delegate
Select All					○	○	○	○	□
Manage Groups					●	○	○	○	□
Manage Portlets					●	○	○	○	□
Manage Portlet Applications					●	○	○	○	□
Manage Page Groups and Pages					●	○	○	○	□
Manage Users					●	○	○	○	□
Manage Clients					●	○	○	○	□
Manage Markups					●	○	○	○	□
Manage Search Index					●	○	○	○	□
Select All					○	○	○	○	□

Figure 3.3: Managing access rights in the portal

Use the form on the left to select the users or groups and the portal resources you want to work with. When you submit the form, the list on the right is refreshed to show the current permissions. Update the list to reflect the new permissions that you want to grant, and then save your changes.

Granting view access to a page or page groups means that other users will see those page groups and pages when they log in. Granting view access to a portlet means that users can add it to their pages when they customize their portal experience. Granting edit access means that a user can set the portlet settings or change the contents of a page. Granting manage access means that a user can perform view and edit operations, and can also delete the portlet or page.

DELEGATED ADMINISTRATION

Granting view access to administration portlets is an effective way of delegating certain administrative tasks to other portal users. Those users can simply add the administration portlets to their personal pages, and then can perform whatever task the portlet is designed to do. This way, the user doesn’t have to be given all administrative privileges or added to the portal administrator’s group. Their administrative abilities are limited to only those tasks covered by the authorized portlets.

Personalizing the Portal

Customizing the User Experience

Optimizing each user's experience in the portal is one of the key goals of WebSphere Portal. To this end, the portal server provides end-user and administrative interfaces for customizing the content of portal pages, as well as the look and layout of the pages. With these tools, users can customize their own pages by selecting portlets and customizing the settings of each one. Users can also change the page layout and the color scheme (if the administrator has decided to allow this).

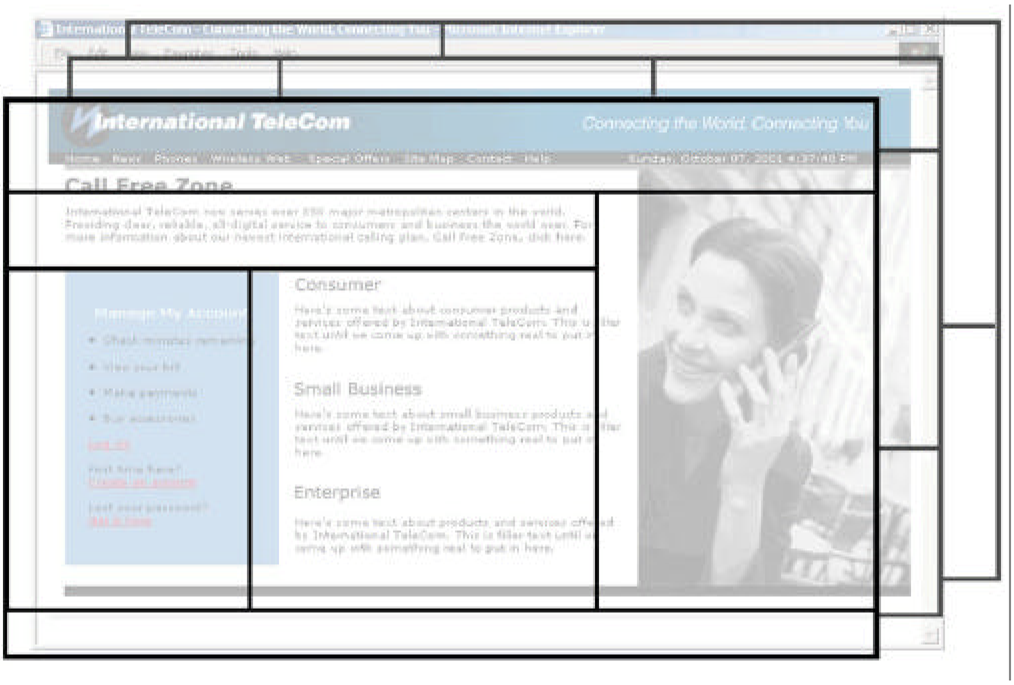


Figure 4.1: Portal page elements

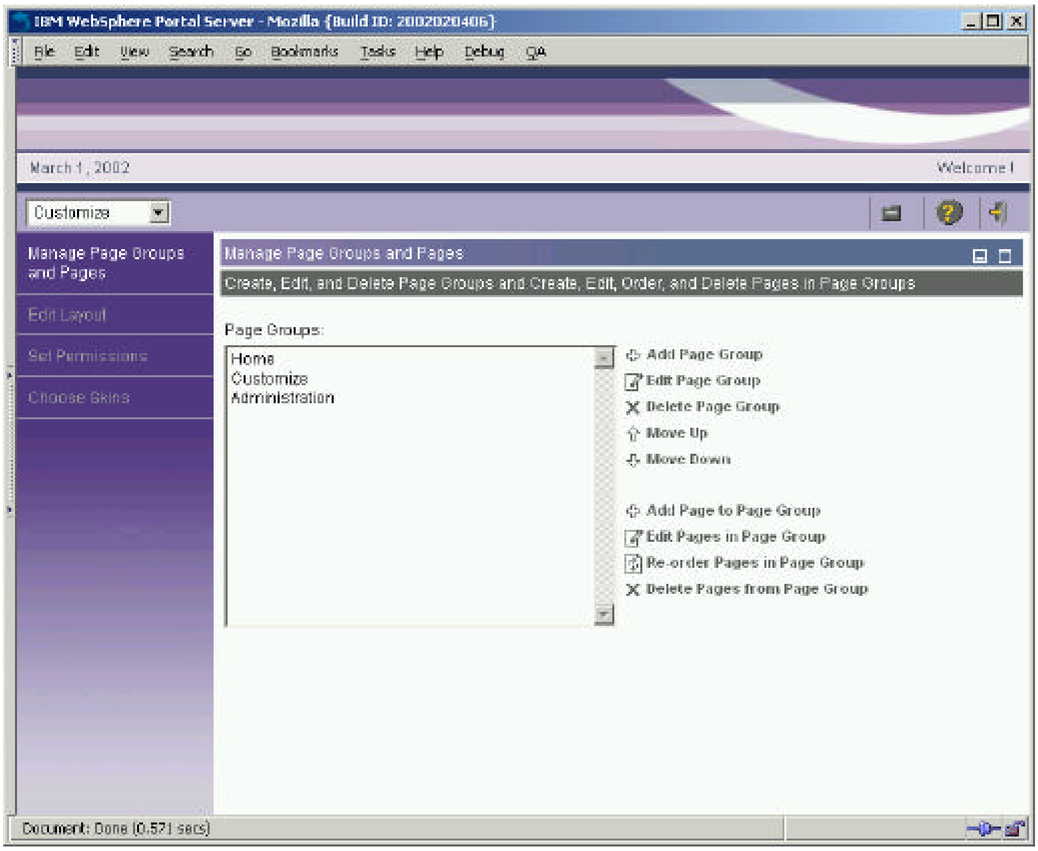
Customizing Pages

Users can have one or more personalized pages, navigation to each one from the home page. Pages are arranged into **page groups**. Each page group can have its own choice of color themes, skins and page layouts. **Themes** are used to define the fonts, colors, spacing, and other visual elements; themes consist of cascading style sheets, JSP files, and images. **Skins** are decorations and controls placed around portlets, such as title bars, borders, shadows, etc. Since the look and feel of each page group can be completely different, page groups can be used to create multiple virtual portals running on one portal server.

In a page group, each personalized page can have a different set of portlets. The portlets on page can be selected by end users or by administrators, depending on their access rights for the page. Administrators can specify that certain portlets are required, so that end users cannot remove them or rearrange them. Pages can also be re-arranged to get a different navigation order.

All of the functions related to customizing page layouts, page contents, color themes and skins are found in the pages of the *Customize* page group – we call this page group the **customizer** for short. Using the customizer, users can see the arrangement of the page and can move portlets around easily.

In this release of WebSphere Portal, pages can be arranged into **page groups**, for better organization and navigation of the portal experience. The user interface for managing page groups and pages is shown below.


IBM WebSphere Portal Server - Mozilla {Build ID: 2002020406}
File Edit View Search Go Bookmarks Tasks Help Debug QA
March 1, 2002
Welcome !
Customize
Manage Page Groups and Pages
Edit Layout
Set Permissions
Choose Skins
Manage Page Groups and Pages
Add Page Group
Edit Page Group
Delete Page Group
Move Up
Move Down
Add Page to Page Group
Edit Pages in Page Group
Re-order Pages in Page Group
Delete Pages from Page Group
Document: Done (0.571 secs)

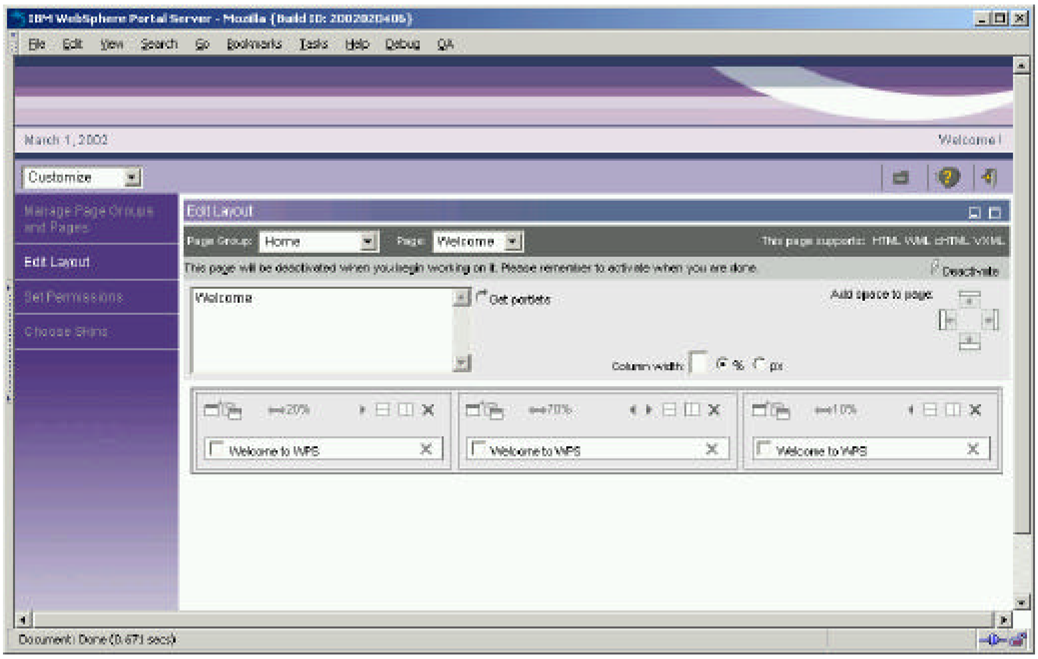
Create, Edit, and Delete Page Groups and Create, Edit, Order, and Delete Pages in Page Groups

Page Groups:

Home
Customize
Administration

Figure 4.2: Managing page groups and pages

On the **Edit Layout** page tab of the customizer (figure 4.3), the basic structure of a page is defined. Page layouts are fully dynamic, with the possibility of creating any arrangement of rows and columns. To create new rows or columns, use the split controls ▸ ⊟ ◫ ✕ to subdivide the page. Columns can have a fixed size, specified in pixels or in a percentage of the overall page size, using the width controls Column width: ◉ % ○ px. To place portlets on the page, choose the portlet you want from the list, then click the add button to place it in one of the column or row spaces.


IBM WebSphere Portal Server - Mozilla {Build ID: 2002020406}
File Edit View Search Go Bookmarks Tasks Help Debug QA
March 1, 2002
Welcome!
Customize
Manage Page Groups and Pages
Edit Layout
Set Permissions
Choose Skins
Edit Layout
This page supports: HTML WML cHTML VXML
Deactivate
Get portlets
Add space to page
20%
70%
10%
Welcome to WPS
Welcome to WPS
Welcome to WPS
Document: Done (0.671 secs)

Page Group: Home
Page: Welcome

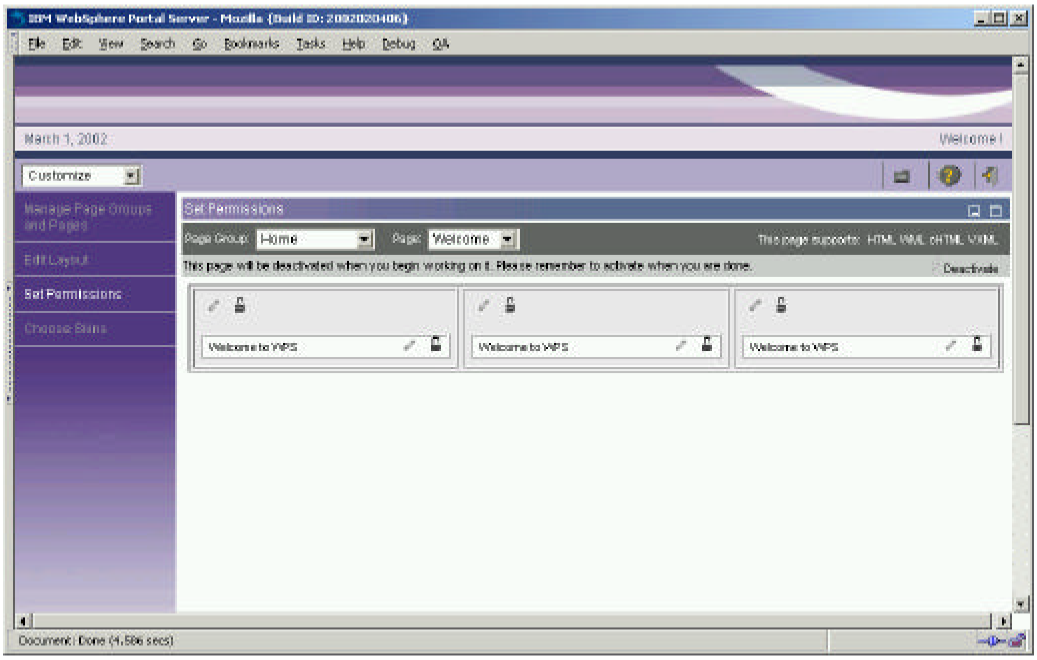
This page will be deactivated when you begin working on it. Please remember to activate when you are done.

Column width: ● % ○ px

Figure 4.3: Designing a page layout

CASCADING PORTALS

Administrators at different levels of the organization can lock the layout or content of any area of the page. For example, locking the placement of portlets means that users cannot move them or remove them from their pages. This way, a higher-level administrator can set up the basic structure of the page, and can fix certain portions or leave others open to modification by other administrators or by end users. Lower levels may not override any restrictions imposed by higher levels.


IBM WebSphere Portal Server - Mozilla {Build ID: 2002020406}
File Edit View Search Go Bookmarks Tasks Help Debug QA
March 1, 2002
Welcome !
Customize
Manage Page Groups and Pages
Edit Layout
Set Permissions
Choose Skins
Set Permissions
Deactivate
Welcome to WPS
Welcome to WPS
Welcome to WPS
Document: Done (4.586 secs)

Page Group: Home

Page: Welcome

This page supports: HTML WML cHTML VXML

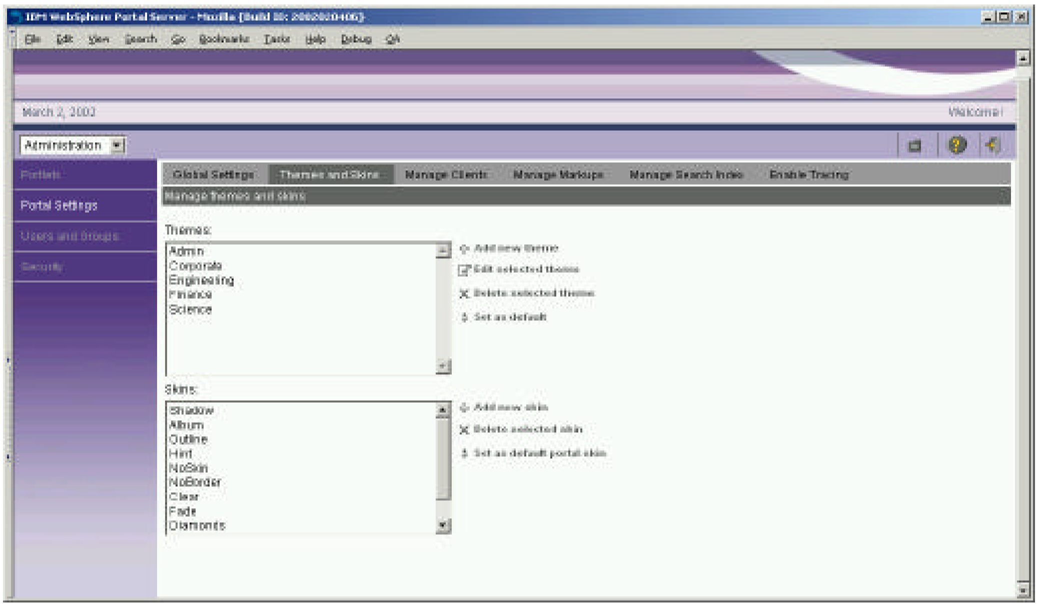
This page will be deactivated when you begin working on it. Please remember to activate when you are done.

Figure 4.4: Locking portlets and layout areas

SKINS AND THEMES

The portal server uses a system of Java Server Pages templates, cascading style sheets, and images to define the look of the portal pages. You can modify these to control any of the visual aspects of the portal, perhaps to add company-specific brand elements, or to achieve a different color scheme and visual style.

The system for defining color themes and portal skins has been enhanced to support multiple skins per theme, additional branding elements, navigation styles, and dynamic, browser-independent cascading style sheets. The file and folder structure for themes has also been changed for easier distribution and installation.



Themes:

Admin
Corporate
Engineering
Finance
Science

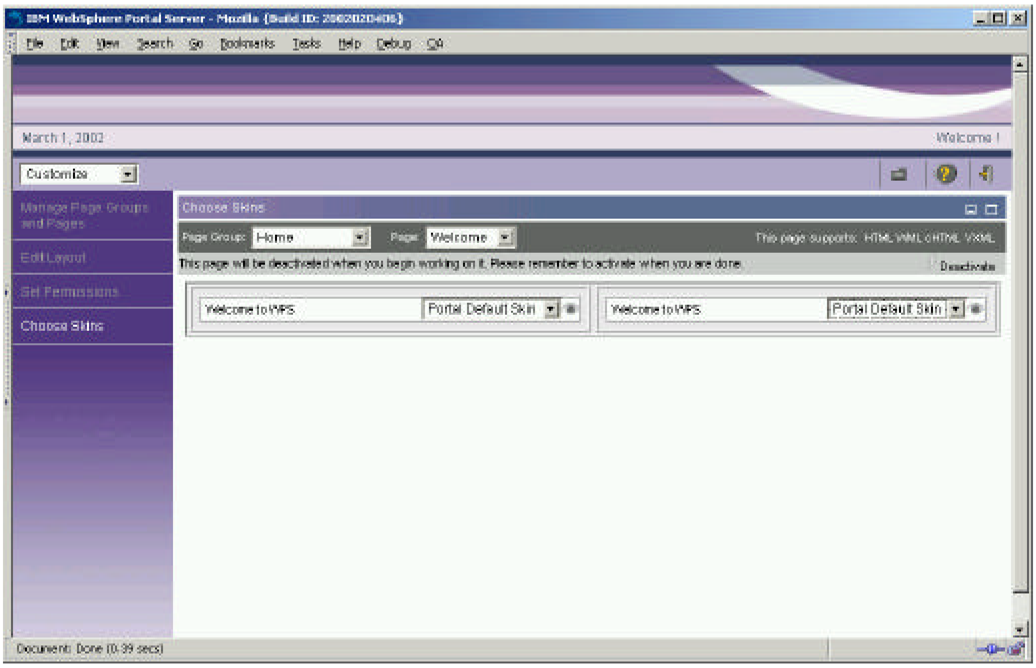
Skins:

Shadow
Album
Outline
Hint
NoSkin
NoBorder
Clear
Fade
Diamonds

Figure 4.5: Managing themes and skins

Stylesheets are dynamically created taking into account the `client`, `version` and `locale` parameters passed to it. The pre-determined settings will be located in properties files and include information such as: “If the client is the Netscape browser, increase the font size by 1.” or “If the client is using the Japanese locale, disable the bold attribute and increase the font size by 1”.

Skins and themes can now be applied to a page group, rather than being restricted to having only one for the overall portal. Different skins can be applied individually to portlets, so that the portal look can be fine-tuned to meet any needs.



IBM WebSphere Portal Server - Mozilla {Build ID: 2002020406}
File Edit View Search Go Bookmarks Tasks Help Debug QA
March 1, 2002
Welcome !
Customize
Manage Page Groups and Pages
Edit Layout
Set Permissions
Choose Skins
Choose Skins
Deactivate
Document: Done (0.39 secs)

Page Group: Home
Page: Welcome
This page supports: HTML WML cHTML VXML

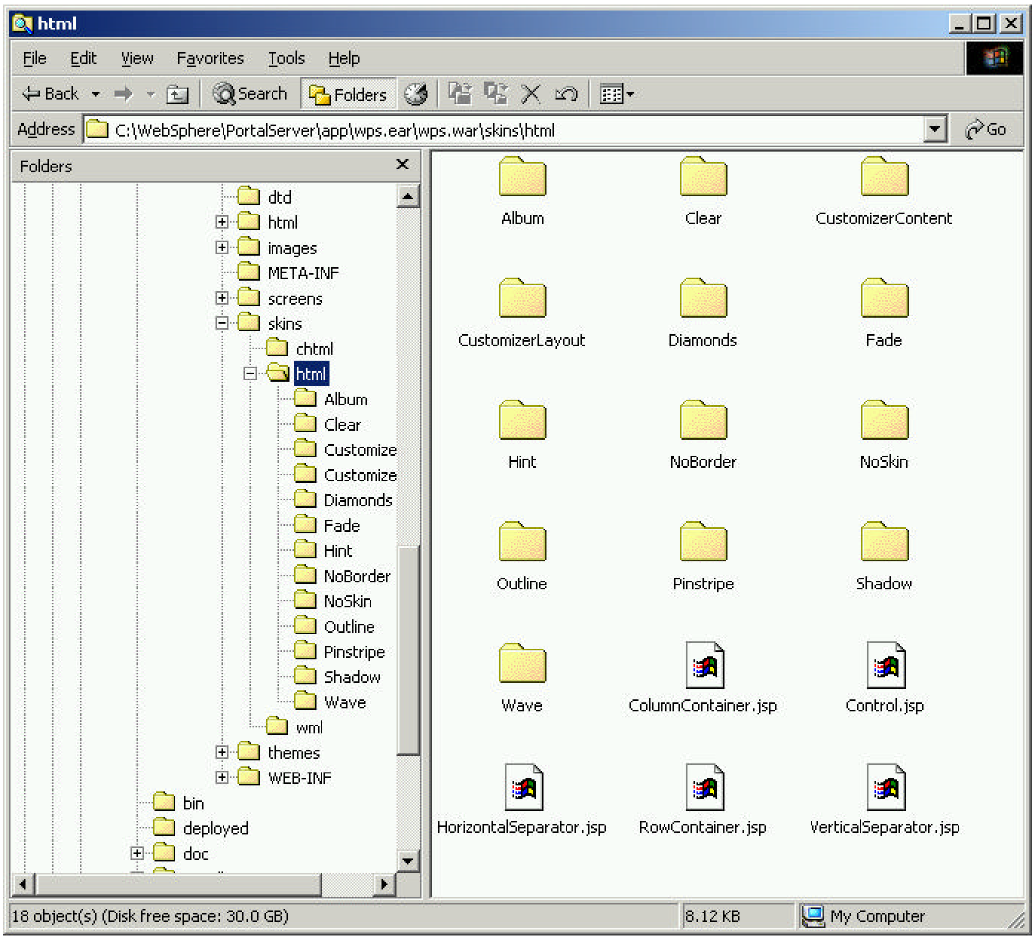
This page will be deactivated when you begin working on it. Please remember to activate when you are done.

Welcome to WPS: Portal Default Skin
Welcome to WPS: Portal Default Skin

Figure 4.6: Assigning skins to portlets

BRAND ELEMENTS

All of the visual elements of the portal, including the masthead, the navigation areas, graphics, portlet title areas, and stylesheets can be changed to give the portal a custom look. Standard file formats, such as JPEG, GIF, CSS, and JSP files are used for defining the look and the layout of the portal.

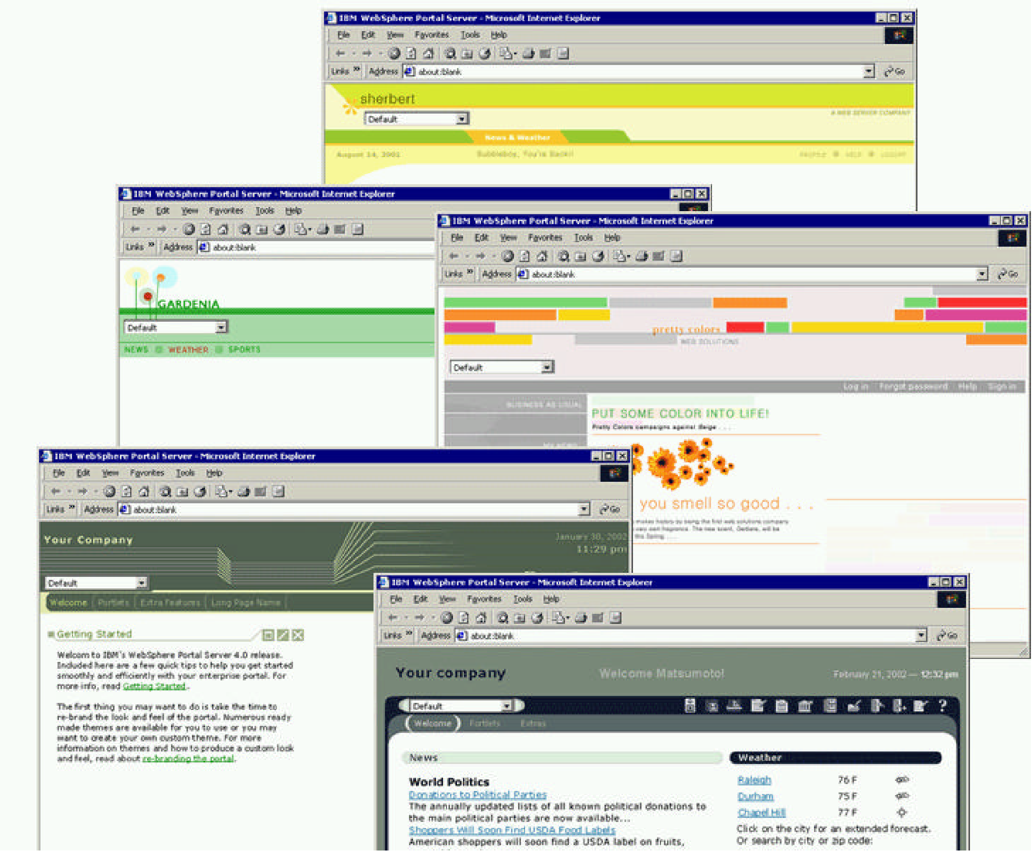


Address C:\WebSphere\PortalServer\app\wps.ear\wps.war\skins\html

18 object(s) (Disk free space: 30.0 GB) | 8.12 KB | My Computer

Figure 4.7 Skin and theme files and folders

If you examine the structure of the portal server's installation folder, you will notice folders named `skins` and `themes`, with folders `html`, `wml`, and `chtml` beneath them. These folders contain most of the files used for defining the basic structure of the portal's home page, its color schemes, and portlet decorations. You can make copies of these folders and modify the contents to achieve the visual image that your company needs. The theme administration portlet registers the new files.



Welcom to IBM's WebSphere Portal Server 4.0 release. Included here are a few quick tips to help you get started smoothly and efficiently with your enterprise portal. For more info, read Getting Started.

The first thing you may want to do is take the time to re-brand the look and feel of the portal. Numerous ready made themes are available for you to use or you may want to create your own custom theme. For more information on themes and how to produce a custom look and feel, read about re-branding the portal.

Donations to Political Parties
The annually updated lists of all known political donations to the main political parties are now available...
Shoppers Will Soon Find USDA Food Labels
American shoppers will soon find a USDA label on fruits,

Click on the city for an extended forecast.
Or search by city or zip code:

Figure 4.8 Example Themes

VIRTUAL PORTALS

Using different themes for each page group, a single installation of the portal server can give the appearance of supporting many "virtual" portals. For example, a company might want to have a different portal for each division, or it might want to have special branding of its business-to-business portal for each business partner. Each virtual portal can be designed using one or more page groups, each with its own themes, skins, page layouts, and access permissions. Administration of the associated portal resources can be delegated to a different administrator.

Universal Access

Earlier, we hinted at the portal's support for international use. The entire system of page templates, themes, skins, and portlet rendering is fully enabled for internationalization and for accessibility to people with disabilities [17]. The portal server generates markup that complies with the American Disability Act (ADA) as defined in Section 508 Web

Accessibility Standards and meets the guidelines of the W3C Web Accessibility Initiative.

For globally accessible portals, the portal server will search for and select the proper JSP pages based on the target browser, and target browser's settings for language and country. To enable a portlet for language selection, its JSP files are packaged using the following directory structure:

```
WAR / markup_type / language / country / variant /jspname.jsp
```

The portal server searches for the JSP for a portal from the most specific to the least specific, as in the following example.

1. /html/en/US/IE/portletJSP.jsp
2. /html/en/US/portletJSP.jsp
3. /html/en/portletJSP.jsp
4. /html/portletJSP.jsp
5. /en/US/IE/portletJSP.jsp
6. /en/US/portletJSP.jsp
7. /en/portletJSP.jsp
8. /portletJSP.jsp

Another good technique for handling translation issues is to use the portal server's JSP tag library. It includes a tag for identifying and separating translatable text, so that you don't have to create and maintain separate JSP files for each language. Using this technique, you could create one JSP view, tagged like this:

```
<H2><wps:text key="heading.hello" bundle="mystrings"/></H2>
```

When this page is processed, the strings are substituted using standard Java resource bundles.

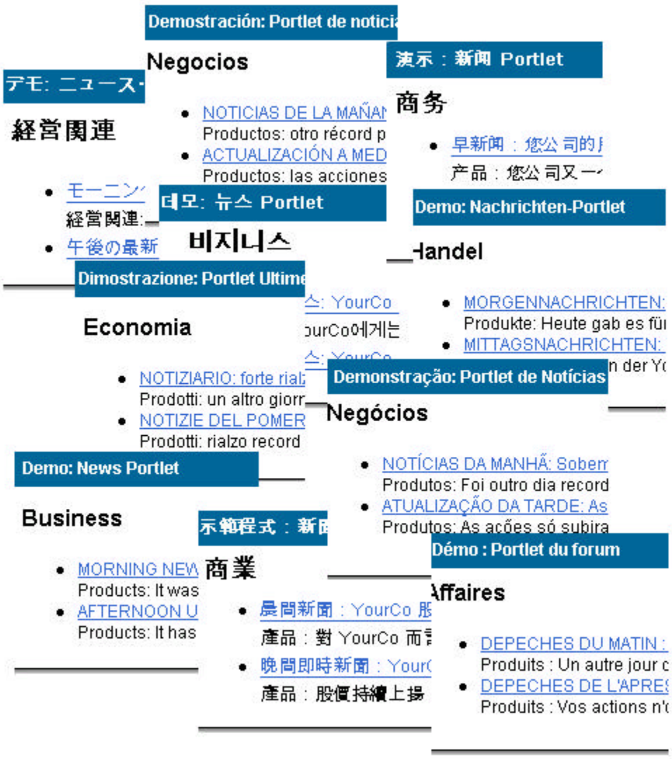


Figure 4.9:Portlets in many languages

For portals that need to support many languages or for portals that include rapidly changing portal content, WebSphere Translation Server [16] offers automatic translation technology. This technology is useful for either real-time or offline translation of human languages.

WebSphere Translation Server provides quick, inexpensive, convenient "gist" text translations whenever a professional translation is not feasible because of availability, time, or cost. The dictionaries can be tuned so that idiomatic expressions and specialized terminology are interpreted correctly. It supports bi-directional translation of content for English to and from French, Italian, German, Spanish, Japanese, and Chinese (simplified and traditional). Unidirectional translation of content from English to Korean and Brazilian Portuguese is also supported.

Personalization

PERSONALIZATION TECHNOLOGY

- Rules Engine
- Filtering Engine
- Campaign Manager
- Preview Manager

The WebSphere Portal offerings include the WebSphere Personalization server [4]. The purpose of the personalization server is to select content for users, based on information in their profiles and on business logic.

WebSphere Personalization provides facilities that allow subject matter experts to select content suited to the unique needs and interests of each site visitor. The web-based tools help companies quickly and easily leverage content created by line of business and subject matter experts.

A personalization solution involves three basic components:

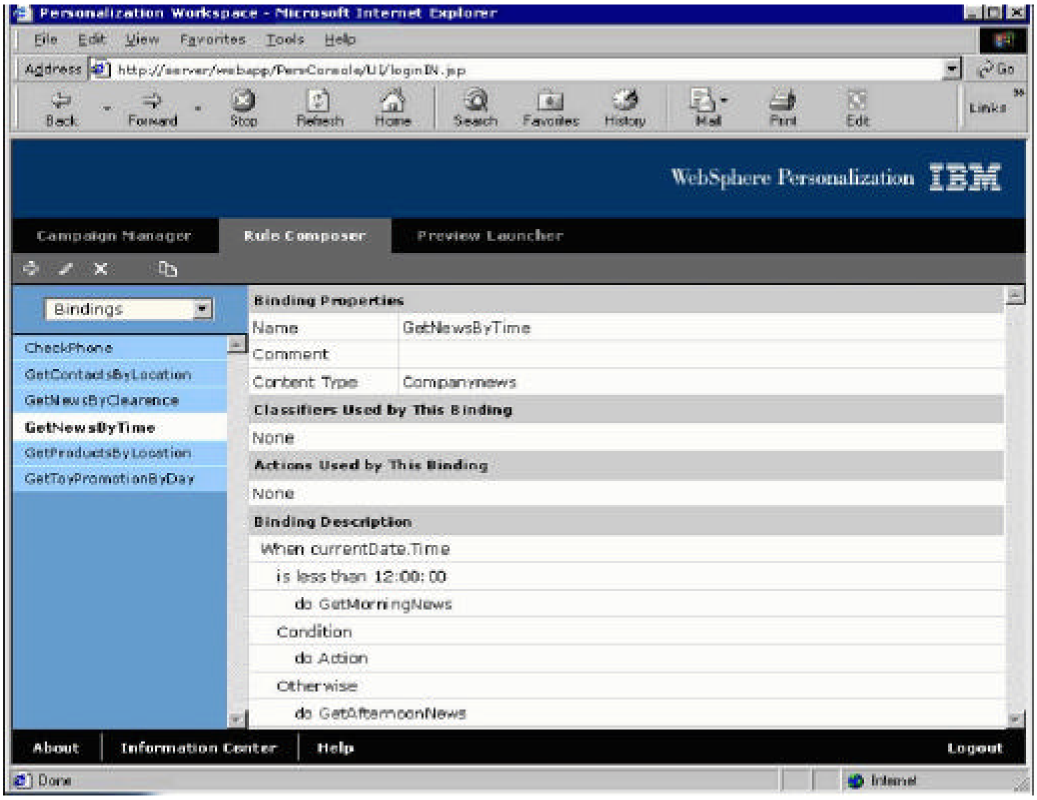
1. User Profile: information about users of the site, including attributes about the users

2. Content Model: defines attributes about the content, such as product descriptions, articles, and other information

3. Matching Technology: engines that match users to the right content; includes filtering, rules, recommendation engines or combinations of all three.

The WebSphere Personalization server and WebSphere Portal server share a common user profile and a common content model. The model is based on the WebSphere resource framework interfaces classes. This means that personalization rules can easily be added to portlets to select portal content and target it to the portal's registered users.

The basic steps involved in personalization involve classifying site visitors into segments, and then targeting relevant content to each segment. Business experts create the rules for classifying users and selecting content, using web-based tools.



Binding Properties

Name	GetNewsByTime
Comment	
Content Type	Companynews

Classifiers Used by This Binding

None

Actions Used by This Binding

None

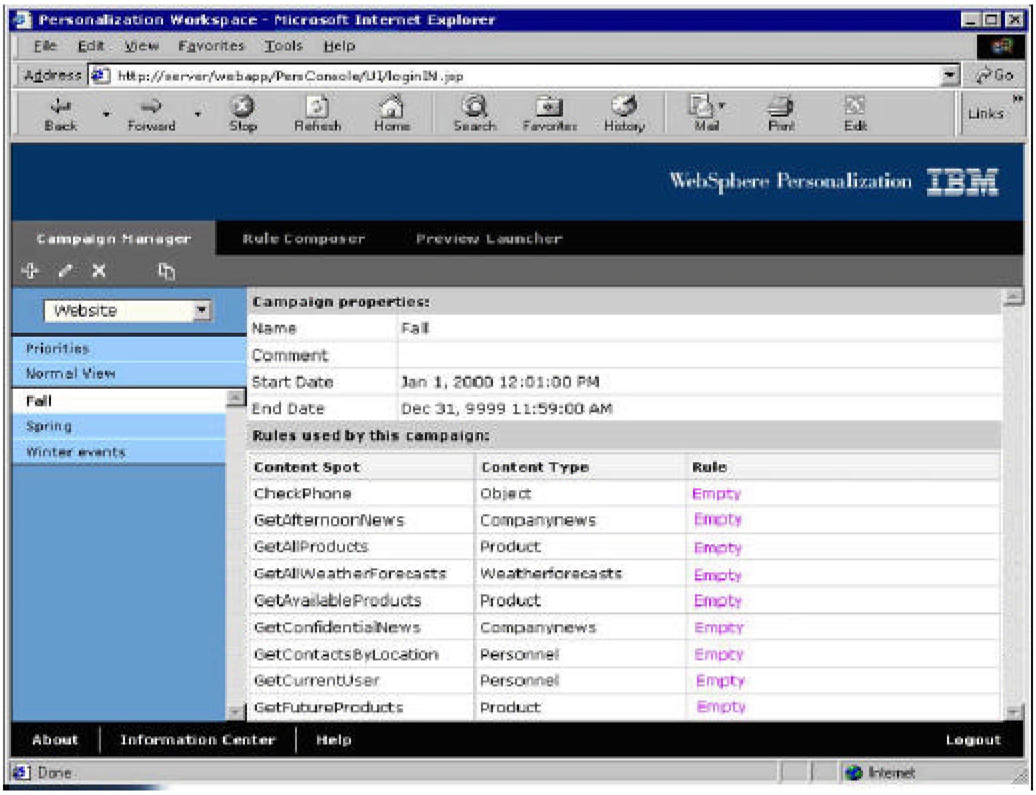
Binding Description

```
When currentDate.Time
  is less than 12:00: 00
    do GetMorningNews
  Condition
    do Action
  Otherwise
    do GetAfternoonNews
```

Figure 4.10: Creating business rules for personalization

WebSphere Personalization also includes a recommendation engine, which provides collaborative filtering capabilities. Collaborative filtering uses statistical techniques to identify groups of users with similar interests or behaviors. Inferences can be made about what a particular user might be interested in, based on the interests of the other members of the group.

Also included with WebSphere Personalization are new campaign management tools. Campaigns are sets of business rules that work together to accomplish a business objective. For example, an HR manager may want to run a campaign to encourage employees to enroll in a stock purchase plan. The HR manager would define a set of rules as shown to accomplish this business objective. Campaigns have start and stop dates and times and can be e-mail and web page based. Several campaigns can run simultaneously and they can be prioritized.


Personalization Workspace - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Address http://server/webapp/PersConsole/UI/loginIN.jsp
Go
Back Forward Stop Refresh Home Search Favorites History Mail Print Edit
Links
WebSphere Personalization IBM
Campaign Manager
Rule Composer
Preview Launcher
Website
Priorities
Normal View
Fall
Spring
Winter events
About
Information Center
Help
Logout
Done
Internet

Campaign properties:

Name	Fall
Comment	
Start Date	Jan 1, 2000 12:01:00 PM
End Date	Dec 31, 9999 11:59:00 AM

Rules used by this campaign:

Content Spot	Content Type	Rule
CheckPhone	Object	Empty
GetAfternoonNews	Companynews	Empty
GetAllProducts	Product	Empty
GetAllWeatherForecasts	Weatherforecasts	Empty
GetAvailableProducts	Product	Empty
GetConfidentialNews	Companynews	Empty
GetContactsByLocation	Personnel	Empty
GetCurrentUser	Personnel	Empty
GetFutureProducts	Product	Empty

Figure 4.11: Setting up campaign parameters

Implicit profiling services can collect real time information about site visitor actions and then construct personalization business rules using this data. Implicit profiling tracks the areas of a site where a user is most active in order to determine the user's interests. For example, if a user clicks on articles about football, it is possible to determine that he is interested in sports, even if the he did not explicitly indicate this interest when registering at the portal.

To analyze the effectiveness of the site and its personalization strategies, the server provides logs that can be analyzed by WebSphere Site Analyzer. Site Analyzer can then create reports for the portal's business owner. This helps the company measure the effectiveness of the business rules and campaigns in achieving their objectives.

5

Administration

Managing Portal Resources Effectively

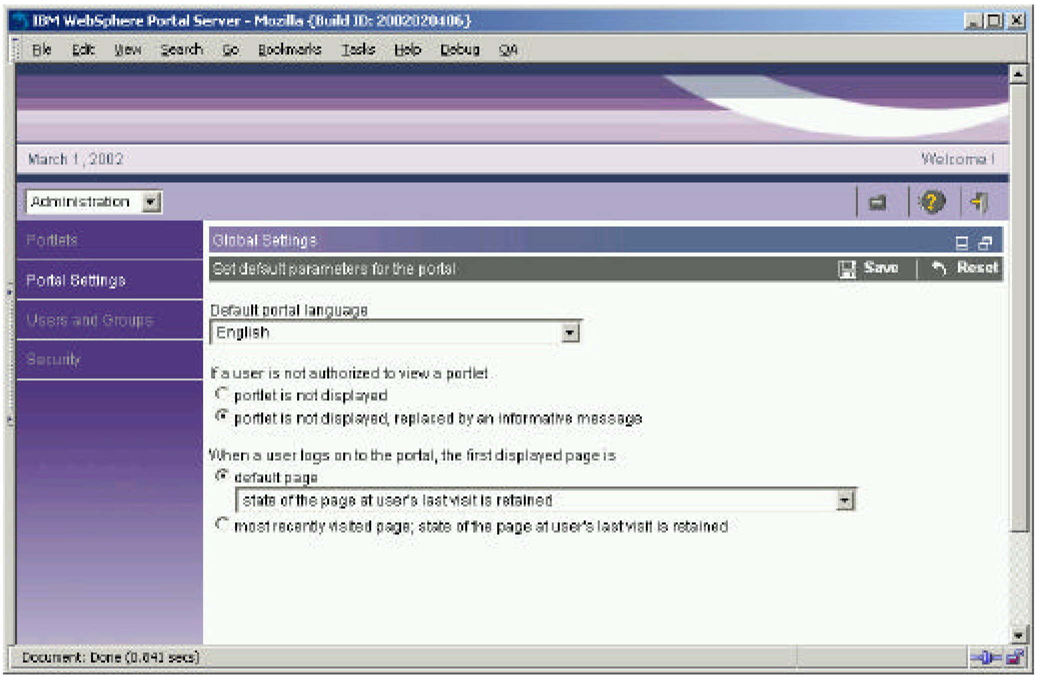
Administration of the portal is done through the portal itself, either in a centralized or delegated fashion. Administrators can deliver a new service to users simply by adding new portlets to the pages of the portal. Since these are portlets, just like bookmarks or reminders or news or any other portlets, administrators can control access to them, place them on portal pages, and perform any of the usual steps.

Administrative portlets are provided for adding portlets to the portal's registry, managing users, groups, and access control lists, clipping web pages, publishing web services, setting portal-wide settings, managing logs, and other common tasks. In this chapter, we'll describe some of the new administration portlets and describe what they do for you. You've already seen some of these in previous chapters, so we'll just focus on the ones that have not been covered already.

Portal Settings

In the global settings portlet, administrators can change portlet settings such as the default language, the cache timeout values, etc.

In this version of WebSphere Portal, there are some new settings that control how new user sessions are handled, and what to do when a user tries to access a portlet without authorization. Unauthorized access can be ignored (in other words, the portlet is not displayed), or the portlet can be replaced by an informative message so that the user can take the necessary actions to correct the situation. Returning users may wish to pick up where they previously left off, so there is a setting to retain the state of the last visit and return to that page next time.


IBM WebSphere Portal Server - Mozilla {Build ID: 2002020406}
File Edit View Search Go Bookmarks Tasks Help Debug QA
March 1, 2002
Welcome !
Administration
Portlets
Portal Settings
Users and Groups
Security
Global Settings
Save
Reset
Document: Done (0.841 secs)

Set default parameters for the portal

Default portal language
English

If a user is not authorized to view a portlet
○ portlet is not displayed
● portlet is not displayed, replaced by an informative message

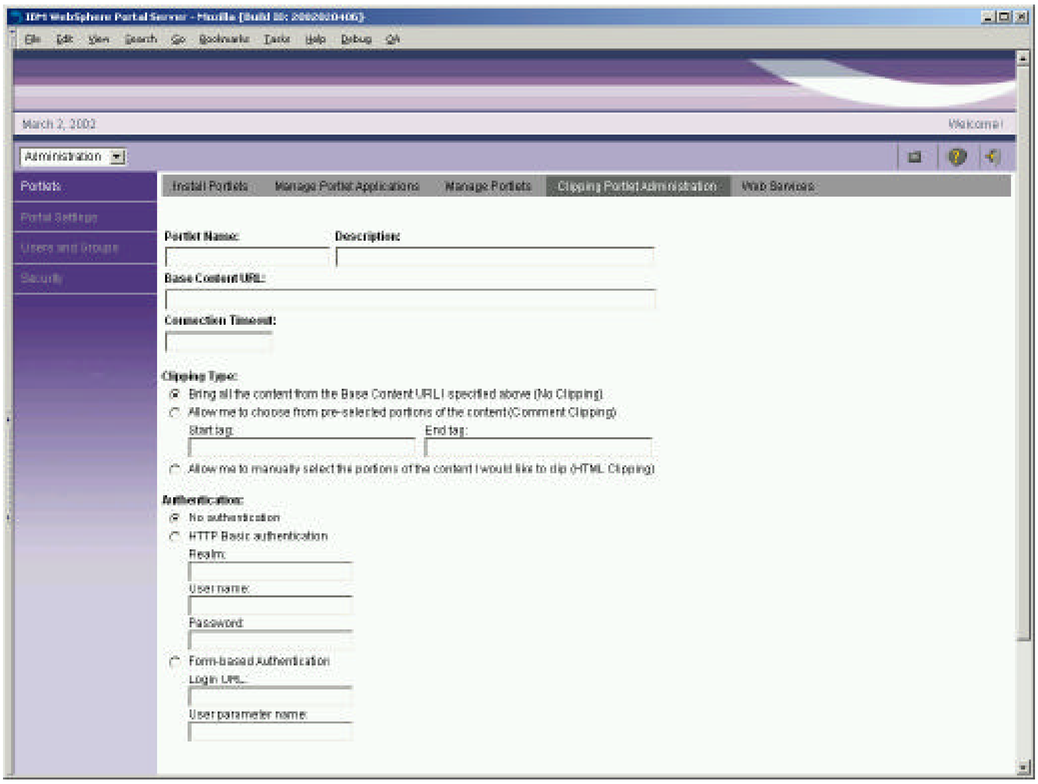
When a user logs on to the portal, the first displayed page is
● default page
state of the page at user's last visit is retained
○ most recently visited page; state of the page at user's last visit is retained

Figure 5.1: General portal settings

Web Clipping Portlet

Reusing Web Content

One of the most important new portlets is the Web clipping portlet. This portlet is used to display sections of existing web pages. You can visually select portions of the page or clip all the text between specific tags. This way, you can precisely control what markup is extracted. The portlet can optionally rewrite the links inside the clipped page, which is useful for displaying existing pages without leaving the portal's navigation structure. Each time you clip a web page, a new portlet is created in the portal's registry. Whenever the new portlet is displayed, it retrieves the current version of the web page and extracts the clipped portion to display.


IBM WebSphere Portal Server - Mozilla {Build ID: 2002020406}
File Edit View Search Go Bookmarks Tasks Help Debug QA
March 2, 2002
Welcome!
Administration
Portlets
Portal Settings
Users and Groups
Security
Install Portlets
Manage Portlet Applications
Manage Portlets
Clipping Portlet Administration
Web Services

Portlet Name:

Description:

Base Content URL:

Connection Timeout:

Clipping Type:

● Bring all the content from the Base Content URL I specified above (No Clipping)

○ Allow me to choose from pre-selected portions of the content(Comment Clipping)

Start tag:

End tag:

○ Allow me to manually select the portions of the content I would like to clip (HTML Clipping)

Authentication:

● No authentication

○ HTTP Basic authentication

Realm:

User name:

Password:

○ Form-based Authentication

Login URL:

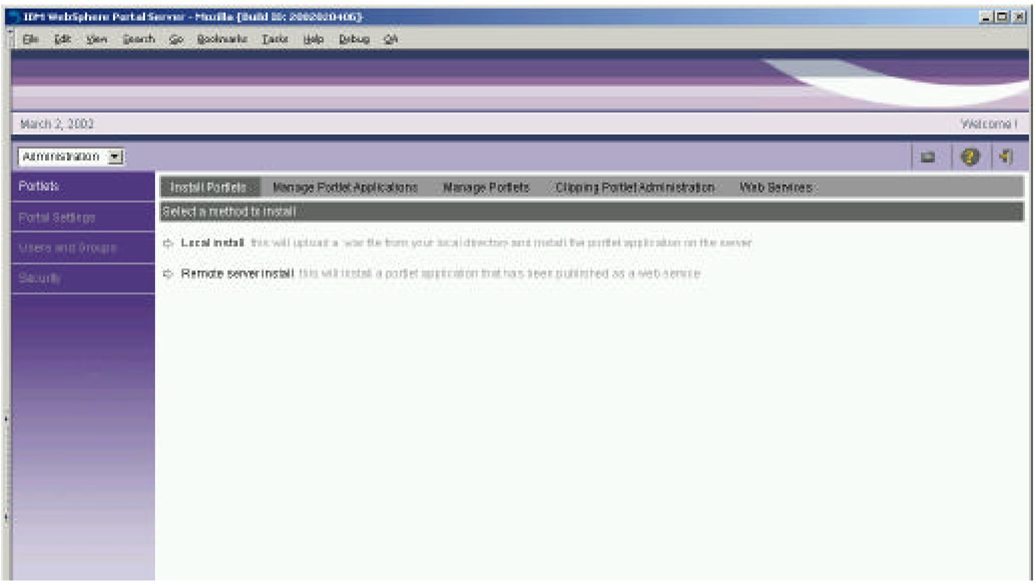
User parameter name:

Figure 5.2: Clipping web pages

Some sites that you clip might require authentication. The clipping portlet provides options for no security, basic authentication, or form-based authentication. The credentials can be provided by the user, or filled in by the administrator.

Managing Portlets

When installing new portlets, you can use web archive files (WAR) from your local file system, or you can install portlets that have been previously published in a UDDI directory (we'll explain more about this later).


IBM WebSphere Portal Server - Mozilla {Build ID: 2002020406}
File Edit View Search Go Bookmarks Tasks Help Debug QA
March 2, 2002
Welcome !
Administration
Portlets
Portal Settings
Users and Groups
Security
Install Portlets
Manage Portlet Applications
Manage Portlets
Clipping Portlet Administration
Web Services

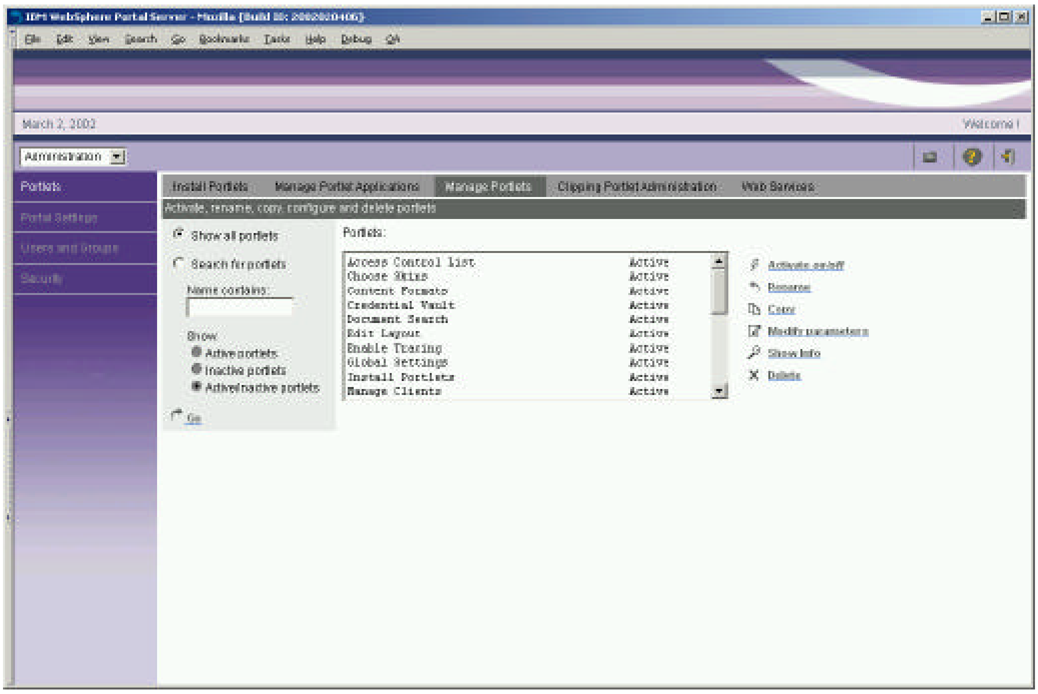
Select a method to install

Local install this will upload a .war file from your local directory and install the portlet application on the server

Remote server install this will install a portlet application that has been published as a web service

Figure 5.3: Installing new portlets

The new portlet is automatically activated, but with no special permissions. Use the Access Control portlet (see Figure 3.3: Managing access rights in the portal) to determine which users and groups can view, edit, or manage the new portlet.



Activate, rename, copy, configure and delete portlets

● Show all portlets

○ Search for portlets

Name contains:

Show:
- Active portlets
- Inactive portlets
- ● Active/inactive portlets

Portlets:

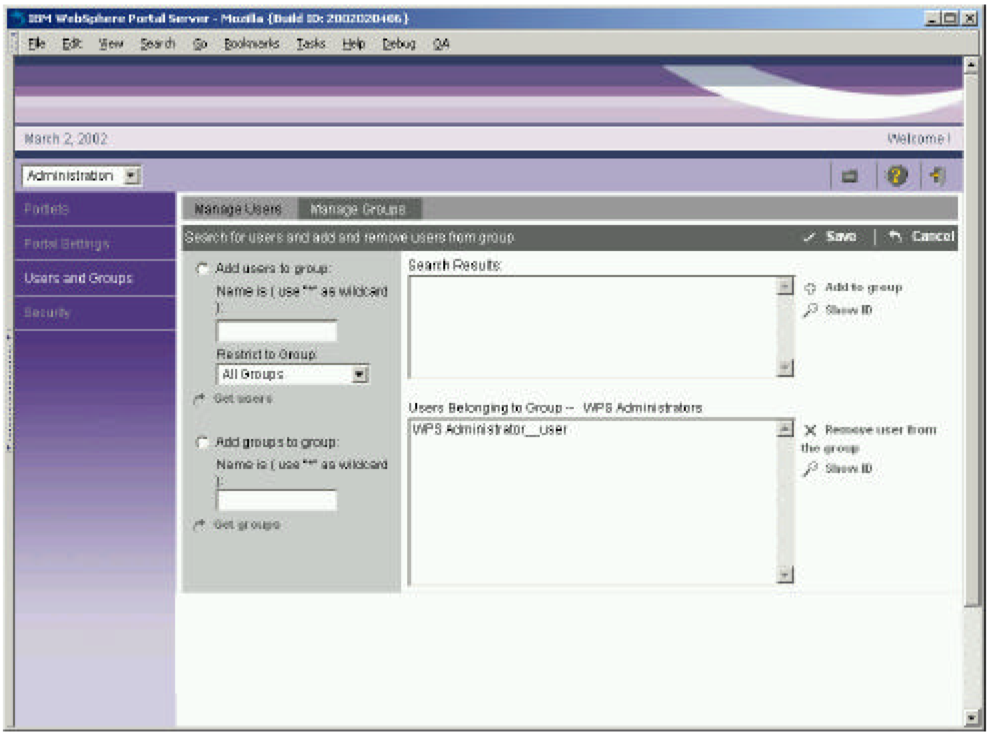
Access Control list	Active
Choose Skins	Active
Content Formats	Active
Credential Vault	Active
Document Search	Active
Edit Layout	Active
Enable Tracing	Active
Global Settings	Active
Install Portlets	Active
Manage Clients	Active

Figure 5.4: Managing the portlet catalog

Once the portlet is installed, you can copy it, set its configuration parameters, activate or deactivate it, or uninstall the portlet, as shown in *Figure 5.4: Managing the portlet catalog.*

Users and Groups

In previous versions of WebSphere Portal, it was necessary to use your LDAP directory's administration tools to manage user and group information. Now, portlets have been added so that you can manage user and group information without leaving the portal. You can also manage a user's group memberships.


IBM WebSphere Portal Server - Mozilla {Build ID: 2002020406}
File Edit View Search Go Bookmarks Tasks Help Debug QA
March 2, 2002
Welcome!
Administration
Portlets
Portal Settings
Users and Groups
Security
Manage Users
Manage Groups
Search for users and add and remove users from group
Save
Cancel
Add users to group:
Name is (use "*" as wildcard):
Get users
Add groups to group:
Name is (use "*" as wildcard):
Get groups
Search Results:
Add to group
Show ID
Remove user from the group
Show ID

Restrict to Group: All Groups

Users Belonging to Group -- WPS Administrators

WPS Administrator__user

Figure 5.5: Managing users and group memberships

The portal server uses group membership information to determine what page groups, pages, and portlets a user is authorized to view and edit. Users can be members of one or more groups, and groups may contain other groups. Users will be allowed access to portal resources when access is granted to any group that the user belongs to. Access rights can also be granted to specific individuals, but most companies find that it is easier to manage the access rights of groups instead.

A programmatic exit is available for obtaining group membership information from an external source. When a company wants to manage access rights outside the portal, we recommend managing group membership information externally, but continuing to use the portal's access control portlet to define which groups can perform what operations on portal resources. This is because most external security managers are not designed to handle large numbers of pages, page groups, portlets and their semantics.

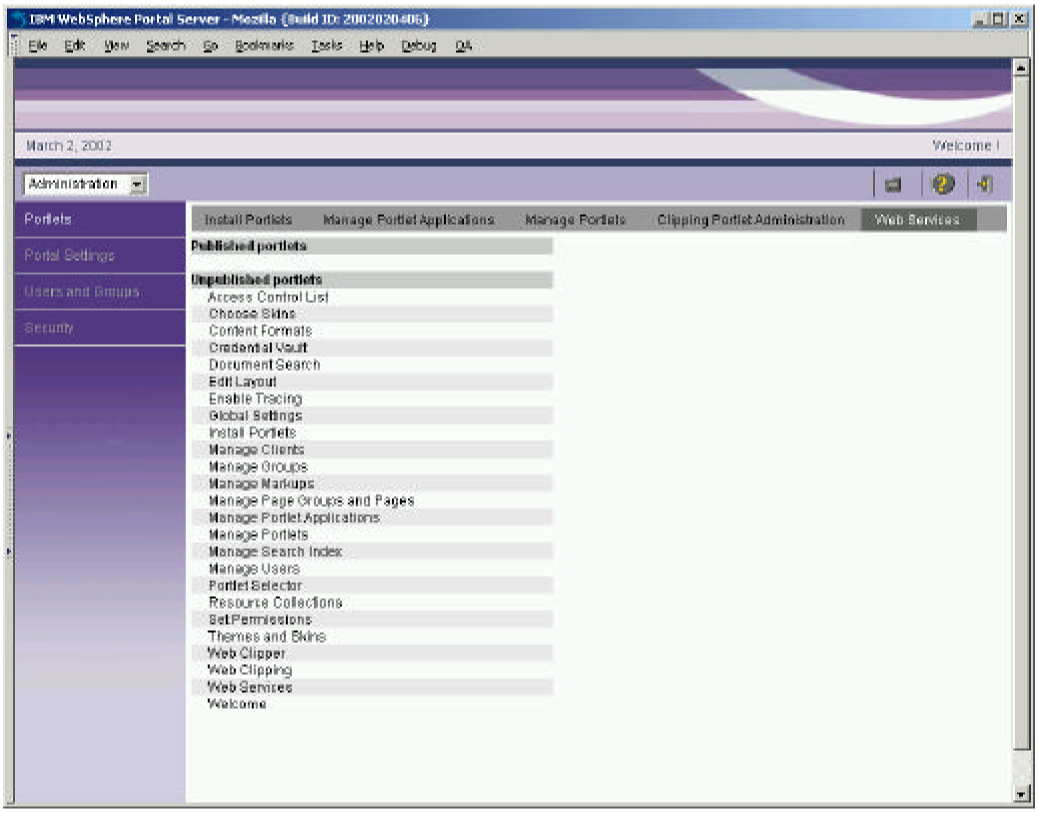
Web Services

A web service [15] is an interface that describes a collection of network accessible operations. The interface is described using a standard XML description language called Web Service Description Language (WSDL), so that the service can be invoked without prior knowledge of the platform, language, or implementation design of the web service. Web Services are located using the standard Universal Description and Discovery Interface (UDDI), which may be a private or public registry of web services.

WebSphere Portal provides extensive support for web services. Portal administrators can publish and bind remote portlets as web services, making the remote portlets available in the portal's registry dynamically. When a remote portlet is used, its services are invoked using Simple Object Access Protocol (SOAP).

FEDERATED PORTALS

To see how portlet web services might be useful, consider a large corporation that has several different portals, such as an employee portal, a supplier portal and a human resources portal. Each of these portals offers portlets to its users, and may choose to publish some of its portlets as web services, making them available for access through other portals.


IBM WebSphere Portal Server - Mozilla {Build ID: 2002020406}
File Edit View Search Go Bookmarks Tasks Help Debug QA
March 2, 2002
Welcome !
Administration
Portlets
Portal Settings
Users and Groups
Security
Install Portlets
Manage Portlet Applications
Manage Portlets
Clipping Portlet Administration
Web Services

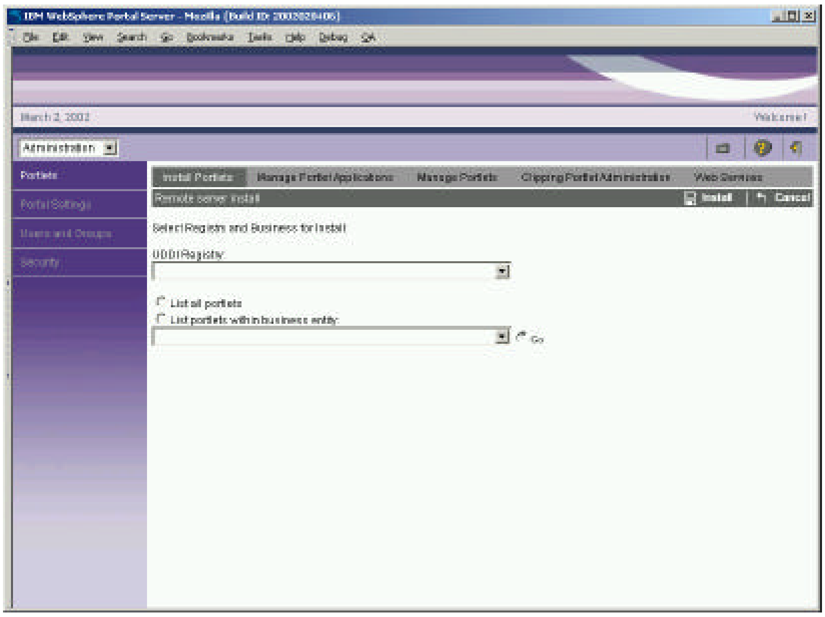
Published portlets

Unpublished portlets

Access Control List
Choose Skins
Content Formats
Credential Vault
Document Search
Edit Layout
Enable Tracing
Global Settings
Install Portlets
Manage Clients
Manage Groups
Manage Markups
Manage Page Groups and Pages
Manage Portlet Applications
Manage Portlets
Manage Search Index
Manage Users
Portlet Selector
Resource Collections
Set Permissions
Themes and Skins
Web Clipper
Web Clipping
Web Services
Welcome

Figure 5.6: Publishing a portlet as a web service

Completing the publishing step puts an entry for the portlet into a UDDI directory. An administrator at another portal can browse the UDDI directory to find all the portlets that have previously been published, and bind these into their local portal. This makes the portlet available as if it were locally installed. Of course, the portlet is actually running remotely, on the original portal server that published it.



Select Registry and Business for Install

UDDI Registry:

○ List all portlets

○ List portlets within business entity:

Figure 5.7: Binding portlet as a web services into a local portal

The effect is to have a **federated portal**, where portlets may be running at any location in the network of portals.

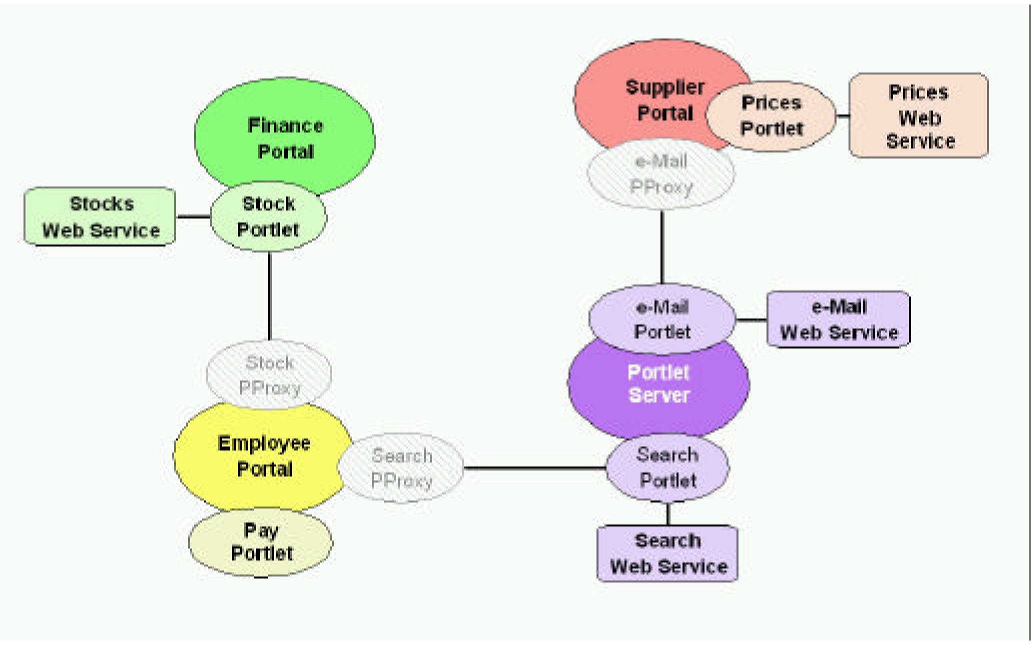


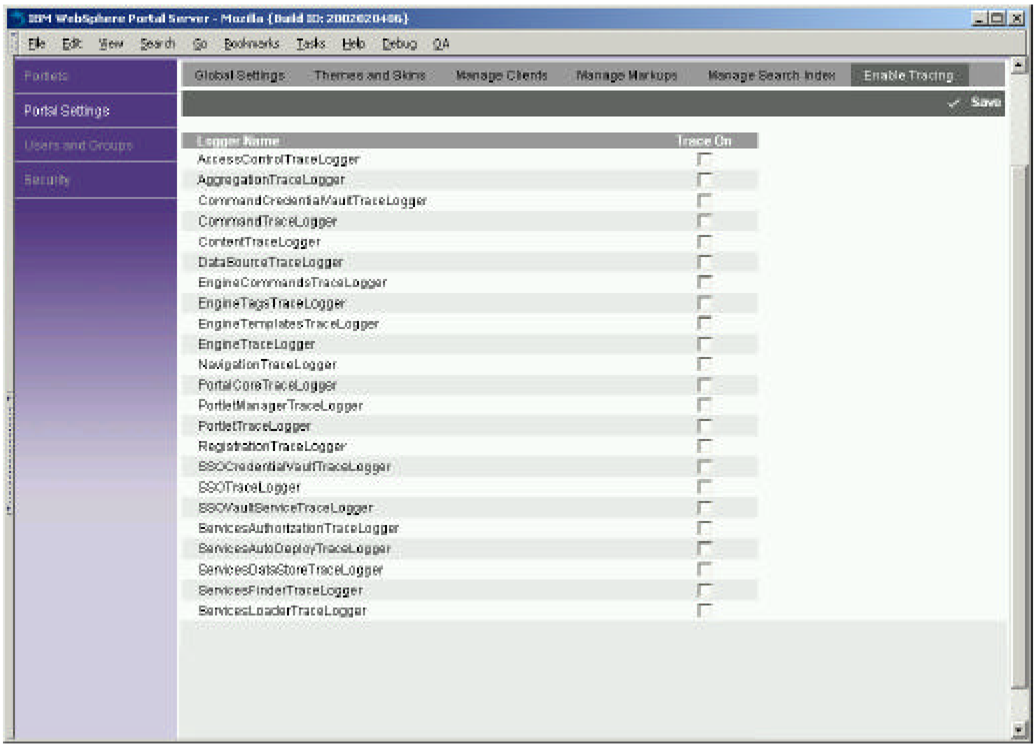
Figure 5.8: Federated Portals

Individual portlets can also use web services internally, to deliver their functionality. For example, a search portlet might query the user for a search string, then use a search web service to search the Internet. A calendar portlet might act as a front end, providing views for a calendar web service.

WebSphere Studio Application Developer [7] provides development tools for quickly developing web services and for generating proxy classes from WSDL descriptions.

Logging

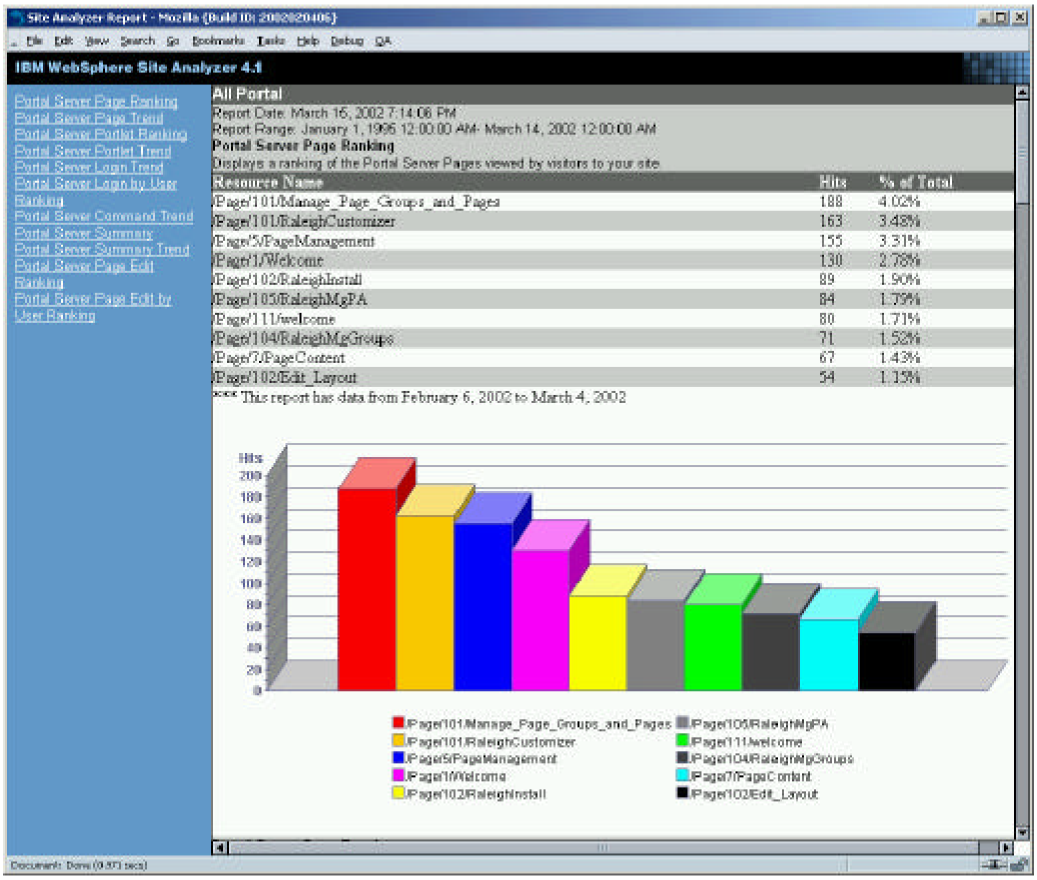
Administrators can control the tracing and logging activity through the Log Files portlet and also by modifying the configuration properties files of the logging subsystem.



Logger Name	Trace On
AccessControlTraceLogger	□
AggregationTraceLogger	□
CommandCredentialVaultTraceLogger	□
CommandTraceLogger	□
ContentTraceLogger	□
DataSourceTraceLogger	□
EngineCommandsTraceLogger	□
EngineTagsTraceLogger	□
EngineTemplatesTraceLogger	□
EngineTraceLogger	□
NavigationTraceLogger	□
PortalCoreTraceLogger	□
PortletManagerTraceLogger	□
PortletTraceLogger	□
RegistrationTraceLogger	□
SSOCredentialVaultTraceLogger	□
SSOTraceLogger	□
SSOVaultServiceTraceLogger	□
ServicesAuthorizationTraceLogger	□
ServicesAutoDeployTraceLogger	□
ServicesDataStoreTraceLogger	□
ServicesFinderTraceLogger	□
ServicesLoaderTraceLogger	□

Figure 5.9: Managing log files

The portal server also records user activity in logs that can be processed by WebSphere Site Analyzer [11]. Overall usage statistics such as logins and logouts, enrollments, and error conditions are tracked. Portlet and page usage statistics, including portlet actions, number of views, modifications, etc. are also tracked.



Site Analyzer Report - Mozilla {Build ID: 2002020406}
File Edit View Search Go Bookmarks Tasks Help Debug QA
IBM WebSphere Site Analyzer 4.1
Portal Server Page Ranking
Portal Server Page Trend
Portal Server Portlet Ranking
Portal Server Portlet Trend
Portal Server Login Trend
Portal Server Login by User Ranking
Portal Server Command Trend
Portal Server Summary
Portal Server Summary Trend
Portal Server Page Edit Ranking
Portal Server Page Edit by User Ranking

All Portal

Report Date: March 16, 2002 7:14:06 PM

Report Range: January 1, 1996 12:00:00 AM- March 14, 2002 12:00:00 AM

Portal Server Page Ranking

Displays a ranking of the Portal Server Pages viewed by visitors to your site

Resource Name	Hits	% of Total
/Page/101/Manage_Page_Groups_and_Pages	188	4.02%
/Page/101/RaleighCustomizer	163	3.48%
/Page/5/PageManagement	155	3.31%
/Page/1/Welcome	130	2.78%
/Page/102/RaleighInstall	89	1.90%
/Page/105/RaleighMgPA	84	1.79%
/Page/111/welcome	80	1.71%
/Page/104/RaleighMgGroups	71	1.52%
/Page/7/PageContent	67	1.43%
/Page/102/Edit_Layout	54	1.15%

*** This report has data from February 6, 2002 to March 4, 2002

Figure 5.10: A portal usage report from WebSphere Site Analyzer

The following table describes the available reports:

PORTAL USAGE REPORTS

Project Summary	Displays summary statistics about the analysis, such as the total number of hits, sessions, visitors, and page views.
User Agent Ranking	Displays a ranking of user agents used by visitors to your site.
Browser Ranking	Displays a ranking of browsers used by visitors to your site.
Platform Ranking	Displays a ranking of users to your site by frequency of visits.
User Ranking	Displays a ranking of users to your site by frequency of visits.
Page Ranking	Displays a ranking of the pages viewed or editing by visitors to your site.
Portlet Ranking	Displays a ranking of the portlets viewed by visitors to your site.
Commands Ranking	Displays a ranking of the commands executed by visitors to your site, such as editing pages, creating new pages, creating or deleting users and groups.

Logins	Displays the number of logins/logouts into the portal.
Resource Ranking	Displays a ranking of the resources viewed by visitors to your site.

Table 5.11: Portal usage analysis

Collaboration

Creating Portal Communities

Corporate portals connect people to the applications, content, and resources they need. Portals also connect people to each other, through community pages, shared bookmarks, contact lists, and personal productivity portlets.

In WebSphere Portal Extend, Lotus Collaborative Places are introduced. A Lotus Collaborative Place is a community area in the portal, which organizes portal pages and applications, helping team members work together on common projects. Collaborative Places help users speed up communication, make effective decisions, and reuse information.

The term “collaborative place” means a group of portal pages, together with portlets, pre-configured applications, authorization groups, and membership services. Collaborative Places can be personal or shared; with shared workplaces, access can be restricted or the place can be made public.

When users visit a place, the home page looks similar to the standard portal home page, except that some additional place-related tools are included, as shown in the following figure:

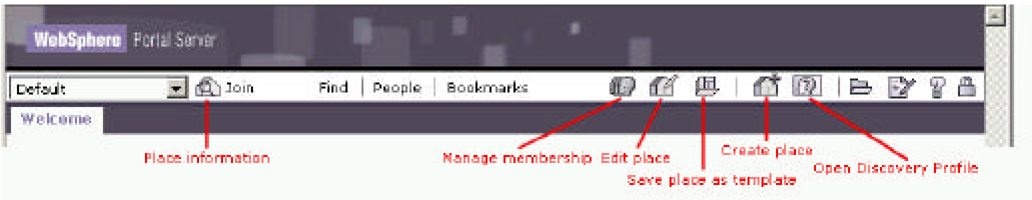


Figure 6.1: Place tools

PERSONAL AND SHARED PLACES

Places are predefined portal workspaces with pages and portlets for managing daily work activities, content, applications and processes.

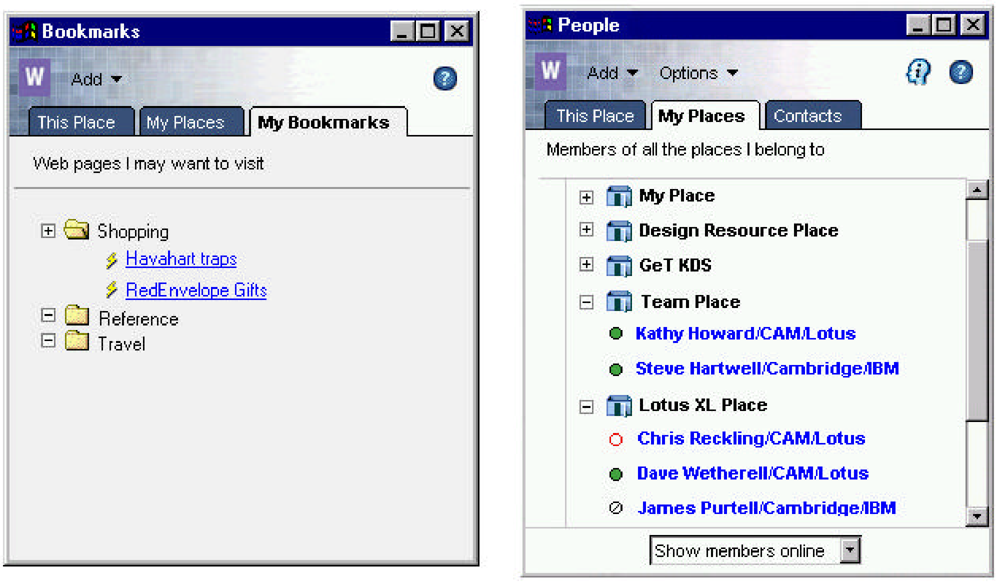
Built-in collaboration tools include online people awareness, instant messaging, discussion areas, shared document libraries with version control, group calendars, task tracking, shared bookmarks and more. Places can be customized by administrators or end users by organizing content into pages, adjusting page layouts, assigning membership access and decorating the place by selecting themes. Places can be open to all portal users or restricted to a specific team or community.

There are two main types of places:

- A **personal place** provides a workspace for individual users to manage daily work needs. This includes portlets for email, calendar, corporate resources and project databases specific to that user.
- A **shared place** can be used for teams working on a project (e.g. a team working on a Merger or Acquisition) or for enterprise-wide communities (e.g. a Sales Community) that need to access the same resources and need a place to share information, store documents, participate in discussions and so on. The content and applications surfaced in a shared place are specific to the context of the place and its members.

Access to a Collaborative Place is controlled through a membership list. Members can be **managers**, **designers**, or **participants** in the community. Managers are allowed to add new members and assign different roles to workplace members. Designers can change the page layouts, contents, and bookmarks. Participants can visit and use the workplace, but cannot edit membership, layout, or bookmarks.

Portal users can store personal bookmarks or bookmarks for the entire team. They are stored on the server instead of on the browser so that bookmarks can be seen by all members of a shared place. Bookmarks can be used to point to important news articles or important information relevant to the team.



Web pages I may want to visit

- Shopping
  - Havahart traps
  - RedEnvelope Gifts
- Reference
- Travel

Members of all the places I belong to

- My Place
- Design Resource Place
- GeT KDS
- Team Place
  - ● Kathy Howard/CAM/Lotus
  - ● Steve Hartwell/Cambridge/IBM
- Lotus XL Place
  - ○ Chris Reckling/CAM/Lotus
  - ● Dave Wetherell/CAM/Lotus
  - ⊘ James Purtell/Cambridge/IBM

Figure 6.2: Community bookmarks and membership lists

Portal members can easily see who else is online and then send an instant message, send an email or add a person to their contact list. Collaboration portlets have advanced built-in features that allow portal users to take actions on documents or user names that appear in a portlet. Directly from the portlet, a portal user can see if other users are online and can select from a menu of options to interact with another user.

PLACE TEMPLATES

Lotus Collaborative Places can be saved as templates so that it's easy to make new ones based on the template. Place templates include a collection of pages and portlets suited to a specific task. A place template also has a theme for styling, plus information such as the place name, description, contact name, and contact email address. A place can be set to inherit changes from its template (or not), so that future changes can be propagated to the new places.

For example, an administrator can configure a personal place to include portlets to email, commonly used corporate databases, and useful links. The place can be saved as a template and reused for new employees. A new employee logs into the portal and automatically has a preconfigured personal place that can be easily customized with project information or personal preferences.

Similarly, a shared place can be saved and reused by other portal users to really take advantage of knowledge management. For example, a sales team that is responding to a Request for Proposal (RFP) may use a shared place to collaborate on collecting the information, creating the document, creating a customer presentation etc. Once the team finishes collecting the information, the knowledge is captured and stored in the place including the names of people who worked on the RFP, the documents and presentations, discussions and so on. This place can be saved and reused by the same team on future RFP's or by other sales teams who have to respond to a similar proposal.

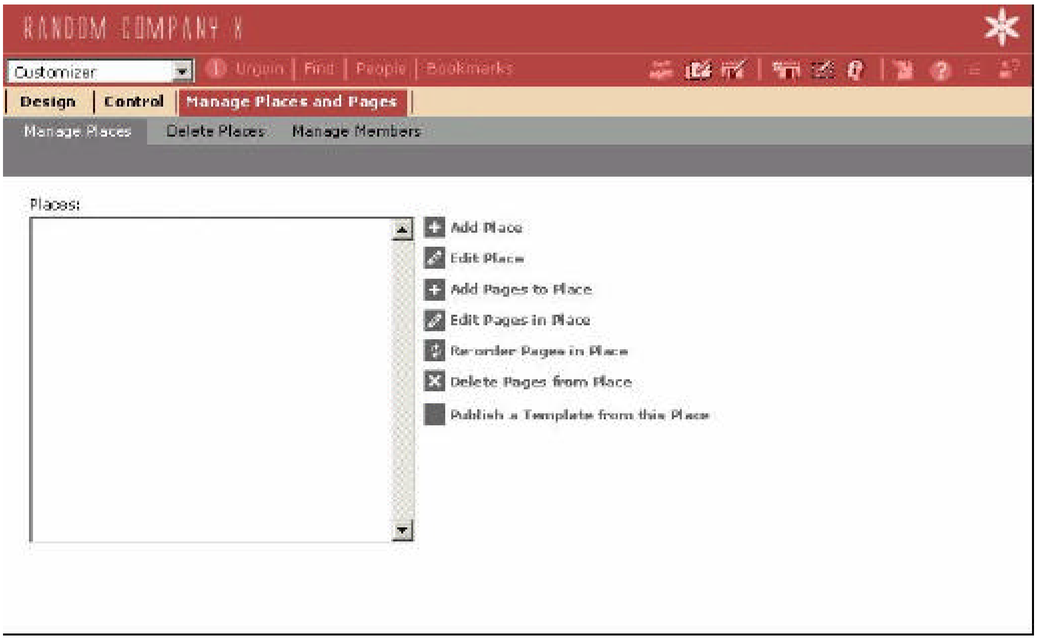


Figure 6.3: Managing places and templates

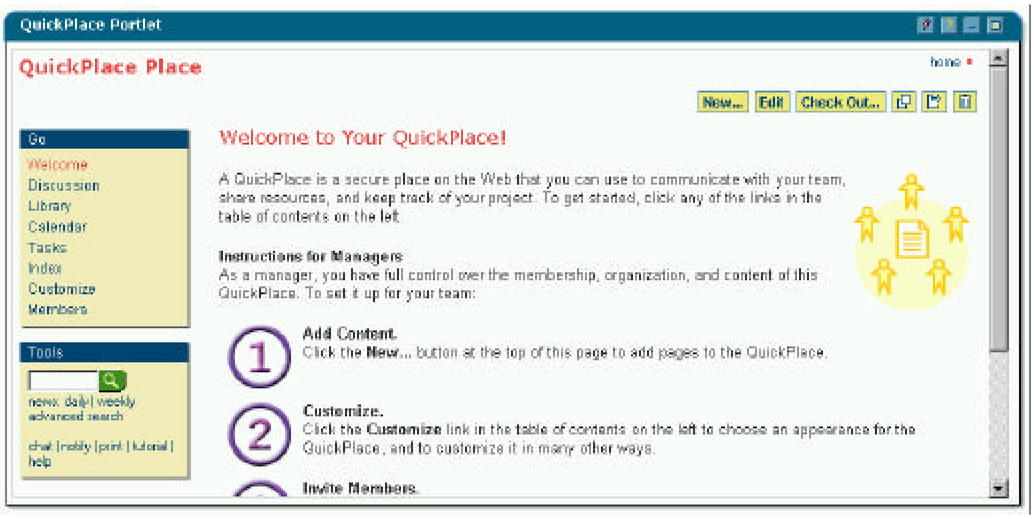
Related Products

WebSphere Portal Extend integrates tightly with Lotus' world-class collaboration and knowledge management products [10], including Lotus QuickPlace, Lotus Sametime, and Lotus Discovery Server by providing portlets and services to access these products seamlessly from the portal.

LOTUS QUICKPLACE

Lotus QuickPlace provides workspaces for sharing and organizing ideas, content and tasks. It includes tools to manage team projects and schedules, organize discussion threads, and share documents. When a new Community of interest place is created, the place manager can also create a new Lotus QuickPlace at the same time.

The Lotus QuickPlace user interface is shown as a portlet in the portal place. Users who access the portal can access the Lotus QuickPlace seamlessly, without requiring an additional login set. The integration between Lotus Collaborative Places and the Lotus QuickPlace includes both single sign-on and synchronization of members lists between the two.



Welcome to Your QuickPlace!

A QuickPlace is a secure place on the Web that you can use to communicate with your team, share resources, and keep track of your project. To get started, click any of the links in the table of contents on the left.

Instructions for Managers
As a manager, you have full control over the membership, organization, and content of this QuickPlace. To set it up for your team:

1 **Add Content.**
Click the **New...** button at the top of this page to add pages to the QuickPlace.

2 **Customize.**
Click the **Customize** link in the table of contents on the left to choose an appearance for the QuickPlace, and to customize it in many other ways.

Invite Members.

Figure 6.4: The Lotus QuickPlace portlet

LOTUS SAMETIME

Lotus Sametime provides instant messaging, shared white boards, and application sharing for electronic meetings. Sametime functionality is integrated into the portal for access to chat sessions and buddy lists, as well as people and place awareness. Awareness is the ability to tell who the place members are and to find out whether they are online, offline, or not available.

Sametime provides other services that can also be integrated through portlets: application sharing, white boarding, and online meetings.

LOTUS DISCOVERY SERVER

Lotus Discovery Server is a separately purchased product that creates expertise and knowledge maps by analyzing and categorizing documents. It creates profiles of users based on their document activity, including their topics of interest and their area of expertise.

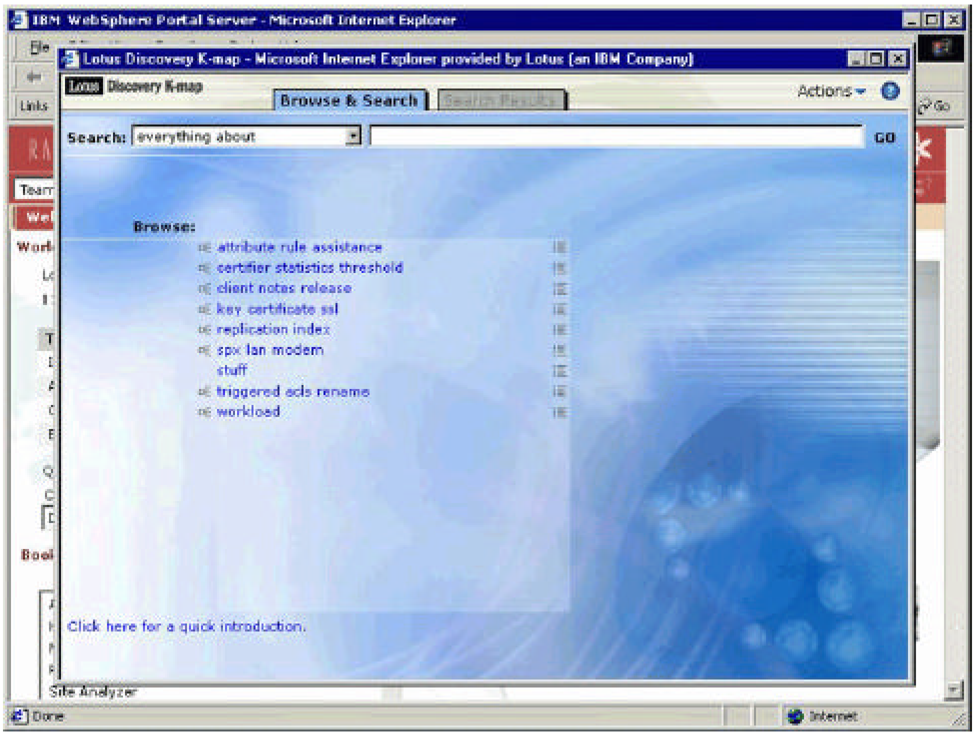
Discovery Server also examines user activity such as reading documents, responding to documents, timeliness of interactions, or links to specific documents. This way, Discovery Server can determine the relative proficiency of individuals to content categories. These proficiency indicators are called **affinities**, and they indicate the relative expertise of individuals to particular business areas of the organization.

The Discovery Server continually assesses the strength of affinities using metrics, ensuring that individuals with recent and high quality expertise ratings are found and presented to users seeking expertise when browsing the Discovery Server Knowledge Map interface.

Lotus Collaborative Places membership lists, bookmarks, and portlets are registered with Lotus Discovery Server so that they can be indexed, categorized, and searched.

When using Lotus Discovery Server, certain features are automatically enabled inside the portal. Additional menu options appear, such as "search for all documents" by the user selected or launching a profile of the user that is maintained and stored with the Discovery Server.

Lotus Discovery Server is purchased separately from WebSphere Portal.


IBM WebSphere Portal Server - Microsoft Internet Explorer
Lotus Discovery K-map - Microsoft Internet Explorer provided by Lotus (an IBM Company)
Lotus Discovery K-map
Browse & Search
Search Results
Actions
GO
Browse:
attribute rule assistance
certifier statistics threshold
client notes release
key certificate ssl
replication index
spx lan modem
stuff
triggered acls rename
workload
Site Analyzer
Done
Internet

Search: everything about

Click here for a quick introduction.

Figure 6.5: A knowledge map

Collaborative Components

Lotus Collaborative Components are building blocks (APIs and JSP tag libraries) for integrating the functionality of Lotus Domino, Lotus Sametime, Lotus QuickPlace, and Lotus Discovery Server into the portal. Developers can leverage the features of Lotus Domino, Lotus QuickPlace, Lotus Sametime, and Lotus Discovery Server, by using these components to add user interface extensions to their portlets and portal pages.

The collaborative services hide the configuration details of the Lotus products installed within an enterprise. Instead of working with more complex product API's, Lotus collaborative components provide developers with an easier method of integrating core collaborative features into any portal or portlet.

LOTUS COLLABORATIVE COMPONENTS	
Portal LDAP Service	Finds entries in the Domino LDAP directory, such as e-mail addresses, Domino mail servers, etc.
People Services	Provide online awareness capability so that portlets can determine who is online and what their status is. Also provides methods to find people and their profile information.
Environment Services	Provide information about the environment, such as the location, protocol, and version of Lotus QuickPlace Lotus Discovery Server, LDAP, or Lotus Sametime servers.
Domino Services	Provide Domino functionality, such as viewing documents, server and database listings; provides access to Domino.doc functionality.
Discovery Services	Make full text search, document and people search, expertise search, and other functions available to portlet developers.
QuickPlace Services	Provide methods for creating a new Lotus QuickPlace.
Community Services	Create, list, update, and delete workplaces and place templates; manage membership, bookmarks, contacts, and pages.

Table 6.6: Lotus Collaborative Components

The People Service provides an awareness function that is available through a custom Java Server Pages tag. Adding this custom tag creates an online status indicator and a popup menu of action choices that access various collaborative services. For example, you could start with a portlet that produces markup like this:

```
...
<sl>
    <li>Allen Adams</li>
    <li>John Jones</li>
    <li>Rita Roberts</li>
    <li>Susie Smith</li>
</sl>
...
```

Adding the person tag would lead to the following markup:

```
<%@ taglib uri="/WEB-INF/tld/people.tld" prefix="peopleservice" %>
...
<sl>
    <peopleservice:person><li>Allen Adams</li></peopleservice:person>
    <peopleservice:person><li>John Jones</li></peopleservice:person>
    <peopleservice:person><li>Rita Roberts</li></peopleservice:person>
    peopleservice:person><li>Susie Smith</li><peopleservice:person>
    </sl>
    ...
```

A great way to apply this idea is by adding the people awareness tags to a portlet that accesses an enterprise application, such as a PeopleSoft or Siebel function. The portal user might see something similar to this:

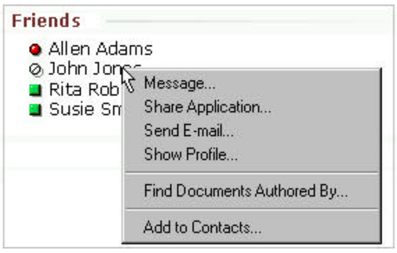


Figure 6.7: People awareness

The exact menu items depend on details about the specific place, the permission level of the current user, and what collaborative products and services are installed in the portal environment. All of the collaborative services except for the `PeopleService` tags can be used in both desktop and mobile portlets.

Portlets

The collaboration portlets include Lotus Notes e-mail, calendar, and to-do list portlets, plus Lotus Notes discussion, document library, and team room portlets.

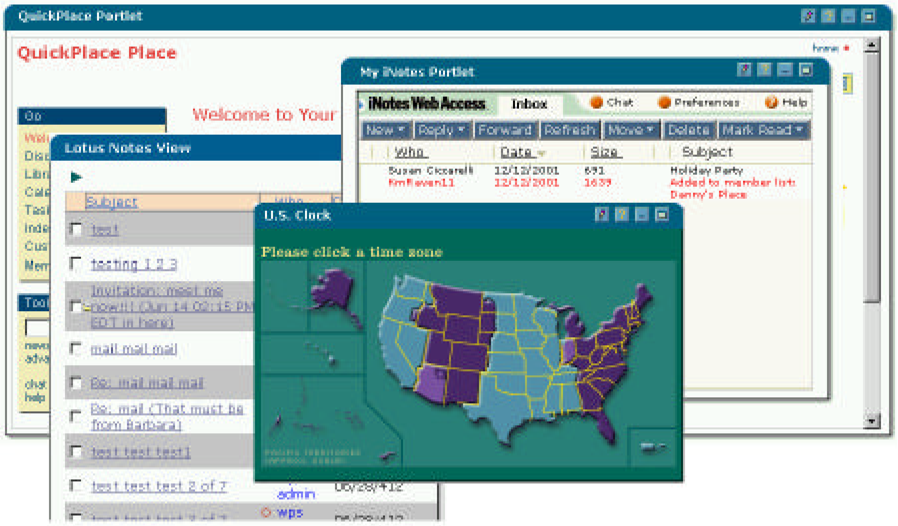


Figure 6.8: Collaboration portlets

The following table describes each portlet.

COLLABORATION PORTLETS	
My iNotes	Provides access to a Lotus iNotes server Welcome, Mail, Calendar, To Do List, Contacts, Notebook functions.
My Notes Calendar	Displays the user's calendar from their mail database. Users may choose to view 1,2,7, 14 or 31 days.
My Notes Mail	Displays the user's inbox from their mail database.
My Notes To Do	Displays the user's To Do list from their mail database.
Notes Discussion	Views Notes databases built with the Discussion Database Template.
Notes Document Library	Views Lotus Notes databases built with the Document Library Database Template.
Notes Mail	Views a user's inbox.
Notes View	Views Notes databases.
Place Catalog	Lists all the public and restricted workplaces that the user can access. You can use the Place Catalog to navigate to other workplaces.
Lotus QuickPlace	Displays a Lotus QuickPlace view inside the portlet.
Sametime Chat	Displays a Sametime chat window inside the portlet.
Team room	Views Notes databases built with the Team Room Database Template.
Web Page	The Web Page portlet will show the contents of an internet or intranet web page in a scrollable portlet.

Table 6.9: Collaboration portlet descriptions

Application Integration

Enterprise Applications for the Portal

By definition, a portal provides access to content, data and services located throughout the enterprise. These include not only predefined connectors and portlets, but also tools for creating additional connectors and portlets.

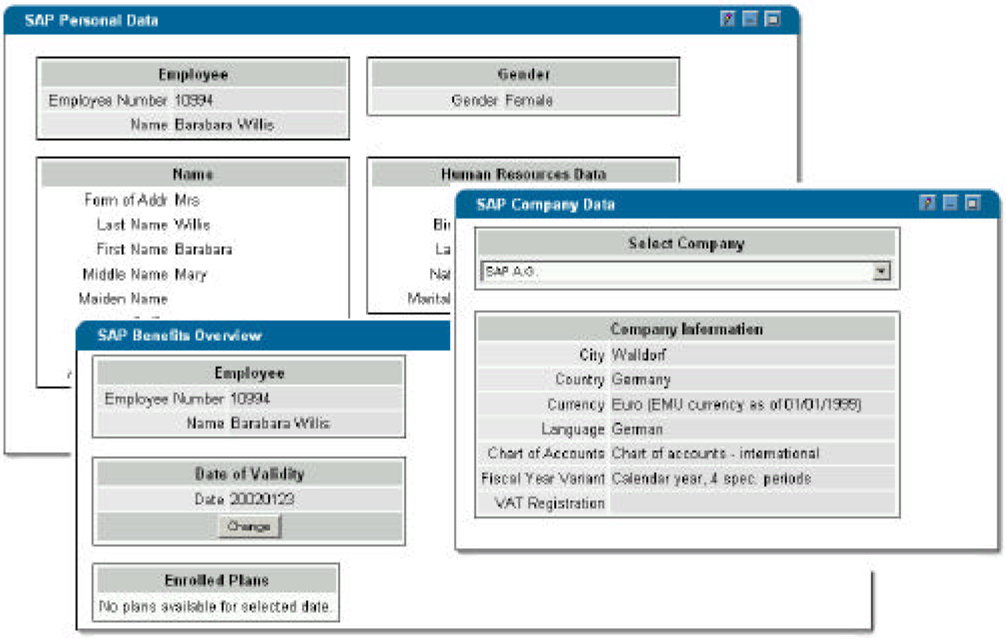
Enterprise Resource Planning (ERP) and Customer Relationship Management (CRM) systems are excellent candidates for portlets because efficient, personalized access to these functions provides measurable return on your portal investment. WebSphere Portal includes portlets that help you access a variety of ERP and CRM systems.

STANDARD JAVA CONNECTORS

In addition, IBM provides connectors to enterprise applications, using the Java Connector Architecture (JCA). JCA is a standard architecture for integrating J2EE applications with Enterprise Information Systems that are not relational databases. Each of these systems provides native APIs for identifying a function to call, specifying its input data, and processing its output data. The goal of the JCA is to achieve an independent API for coding these functions.

JCA also defines a standard Service Provider Interface (SPI) for integrating the transaction, security and connection management facilities of an application server with those of a transactional resource manager. Thus, JCA is a standards-based approach to managing connections, transactions, and secure access to enterprise application systems. IBM's JCA connectors provide access to systems such as SAP, PeopleSoft, J.D. Edwards, Oracle, CICS, IMS, and Host-on-Demand. Leveraging its CrossWorlds acquisition, IBM will also develop and integrate JCA connectors to many other systems.

WebSphere Studio Enterprise Integrator will provide a complete development and unit test environment for applications that use JCA connectors, web services, and micro flows. The Studio tools include support for Web Service Definition Language (WSDL), developer's versions of the connectors, the Web Services Invocation Framework (WSIF), and the micro flow engine.



SAP Personal Data

Employee
Employee Number 10994
Name Barbara Willis

Gender
Gender Female

Name
Form of Addr Mrs
Last Name Willis
First Name Barbara
Middle Name Mary
Maiden Name

Human Resources Data
Bir
La
Nat
Marital

SAP Company Data

Select Company
SAP A.G.

Company Information
City Walldorf
Country Germany
Currency Euro (EMU currency as of 01/01/1999)
Language German
Chart of Accounts Chart of accounts - international
Fiscal Year Variant Calendar year, 4 spec. periods
VAT Registration

SAP Benefits Overview

Employee
Employee Number 10994
Name Barbara Willis

Date of Validity
Date 20020123
Change

Enrolled Plans
No plans available for selected date.

Figure 7.2: Native SAP portlets

OTHER APPLICATION CONNECTORS

SUPPORTED SYSTEMS

- SAP
- PeopleSoft
- Baan
- Siebel
- Oracle Financials

The WebSphere Portal Application Access Feature (AAF) enables you to add data from your enterprise applications (including SAP, Baan, Siebel, PeopleSoft, and Oracle Financials) to portal pages. The Application Access Feature is optional, and is installed separately from the application server where the portal server is installed.

Data is retrieved through special connectors called **iViews**, and then displayed in a portlet. The iView connectors use technology from SAP Portals to send and receive XML data between the portal server and the iView server.

Predefined iViews are available for purchase from IBM; you can select from the predefined list or create your own. In addition, a few example iViews are shipped with Application Access Feature. The iView portlets only provide query access to the backend application, so their capabilities are much more limited than portlets based on JCA.

IVIEW PORTLETS

SAP

Aging Sales Quotes
Customer Average
Arrear Analysis
Customer Balances By
Fiscal Period
Customer Credit
Exposure
Customer Credit
Limited Exceeded
Employees Per Pay
Area
Exchange Rates
Expiring Purchasing
Contracts
Expiring Sales
Contracts
Internal Order by Area
Incomplete Sales
Orders
Past Due Sales Items
Planned Order List
Sales Hot List
Customer Bank Data
Employee Get List
Material Availability
Sales Order Per
Customer
Vendor Open Items
Shipments Due
Top 10 Back Orders
by Customer
Top Value Returns
Non Invoiced
Shipments

PeopleSoft

401K At Limit
401K Over Limit
Birth Date
Class Fill
Information
Cobra
Expirations
EEO Compliance
Monitoring
Interview
Schedule
Job Postings
Late Benefits
Enrollment
Late Increases
New Closed
Positions
New Hires
Payroll Summary
Report
Penetration
Report
Stale Requisitions
Training No
Show
Training Schedule
Turnover
Upcoming
Anniversaries
Vacation
Overdrawn
Vacation At Limit

Siebel

My Accounts
Activity Assigned
List
All Activities
All Assets
All Billable
Projects
My Opportunities
All Product
Defects
All Projects
All Service
Orders
Contact List
Critical Service
Request
High Severity
Product Defects
Incomplete
Deliveries
My To Do List
New Product
Notification
New Prospective
Accounts
Open Sales
Orders
Open Service
Request
Open Service
Request
Opportunities
Won
Opportunity
Forecast
Service Request
By Priority

Oracle Financials

View Expense
Report History
Purchase
Requisition User
Analysis
Inventory
Availability
Supplier Item
Summary
Invoices Received
Overdue Receipts
Retired Assets
Open Delivery
Schedules
All Assets
Asset Invoices
Delivery
Performance
Invoices and
Debit by PO
Number
Requisition
History

Baan

Contracts Not
Yet Returned
Expiring Sales
Contracts
Goods Waiting
for Approval
Past Due and
Current Sales
Orders
Past Due
Purchase Orders
Past Due Sales
Orders
Purchase Price
Variance
Revenue Drop
Sales Order Hot
List
Shipment to
Problem
Customers
Slow Moving
Items
Top Outstanding
Payments
Top Outstanding
Receipts
Top Returns
From Customer
Top Returns To
Supplier
Total
Outstanding
Receipts

Table 7.1: iViews

COMMERCE

Many portals need to include access to commerce functions, such as portlets that access a catalog of products and services, or portlets that present price, discount, or tariff information. Portals may provide these functions to consumers or to business partners, for procurement or electronic exchange that links buyers and sellers of goods.

In January 2002, IBM announced a new capability for integrating WebSphere Portal and WebSphere Commerce Server [9] to create commerce-enabled portals that provide consolidated, personalized access to commerce functions via the Web or wireless devices. This capability is available through service offerings from IBM and business partners.

The product roadmap for integrating commerce and portal functions includes integrating user management, authentication, single sign-on, and access control functions, using common implementations and APIs for each. In the first step, WebSphere Portal and WebSphere Commerce Server will share a common user and group management subsystem. This way, users register only once, and then all the user profile information (e.g. name, address, user id, password, user preferences, etc.) are consistent across different portlets.

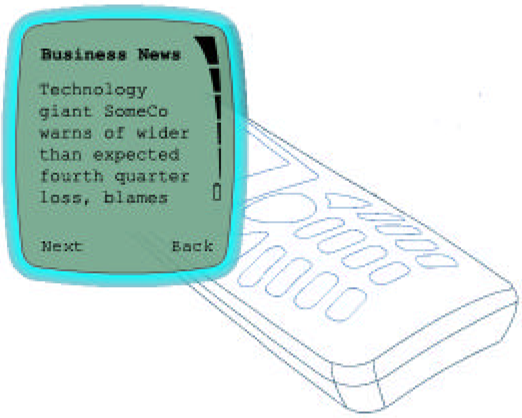
In addition, authentication for both the portal server and the commerce server will be based on WebSphere Application Server security, using either its built-in forms-based authentication, or else using its Trust Association Interceptor modules to accept the authentication that is done by an external authentication proxy server. Once the users' credentials are established, the portal server and commerce server will use these for any access controls required by the different portlets.

Mobile Portals

Universal Access from Any Device

Next generation portals will be accessible through more than just traditional desktop browsers. Access through handheld or mobile devices is becoming increasingly important. The portal server currently supports mobile devices by generating portal pages in three markup languages: HTML for desktop computers and some personal digital assistants, WML for WAP devices, which are typically mobile phones, and cHTML for mobile devices in the NTT DoCoMo iMode network.

Users can customize a unique home page for each device, selecting the content and applications that are most useful on the device. When the home page is requested, the page is produced by first detecting the type of device that is making the request, and then assembling the portlets which each render their contents in the appropriate markup language.

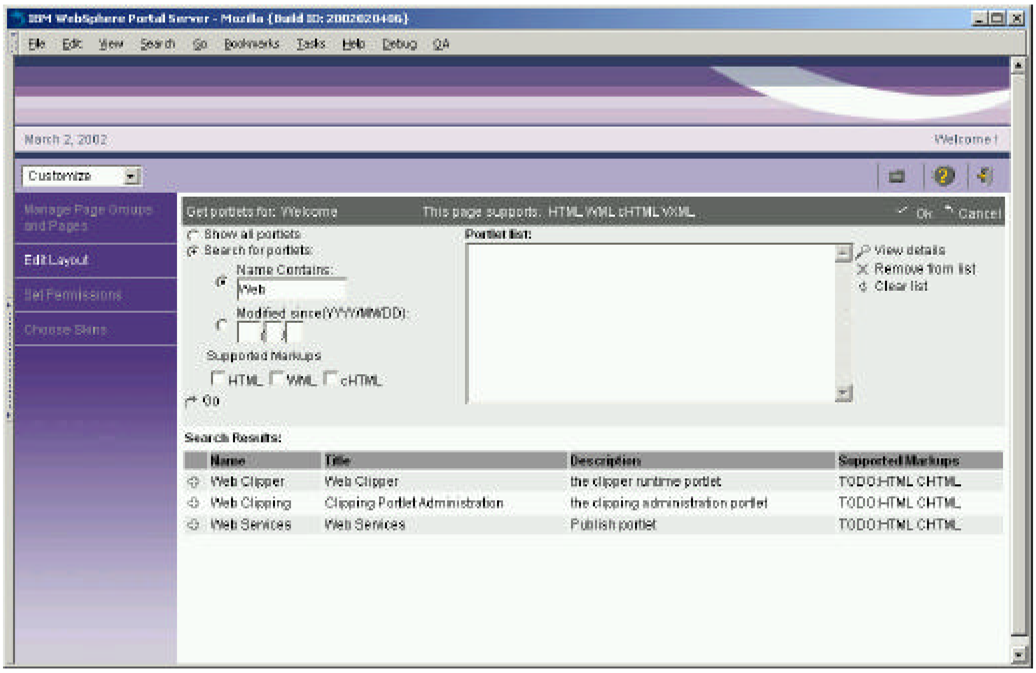


Business News

Technology giant SomeCo warns of wider than expected fourth quarter loss, blames

Figure 8.1: A mobile portlet

When a user customizes the home page for a particular device, the portlet selection list only shows portlets that can actually produce markup appropriate for that device. Thus, the list of available portlets for each device depends on what the portlets can actually do.



IBM WebSphere Portal Server - Mozilla {Build ID: 2002020406}
File Edit View Search Go Bookmarks Tasks Help Debug QA
March 2, 2002
Welcome !
Customize
Manage Page Groups and Pages
Edit Layout
Set Permissions
Choose Skins
✓ OK
Cancel

Get portlets for: Welcome   This page supports: HTML WML cHTML VXML

○ Show all portlets
● Search for portlets:
● Name Contains: Web
○ Modified since(YYYYMMDD):
Supported Markups
□ HTML □ WML □ cHTML
Go

Portlet list:

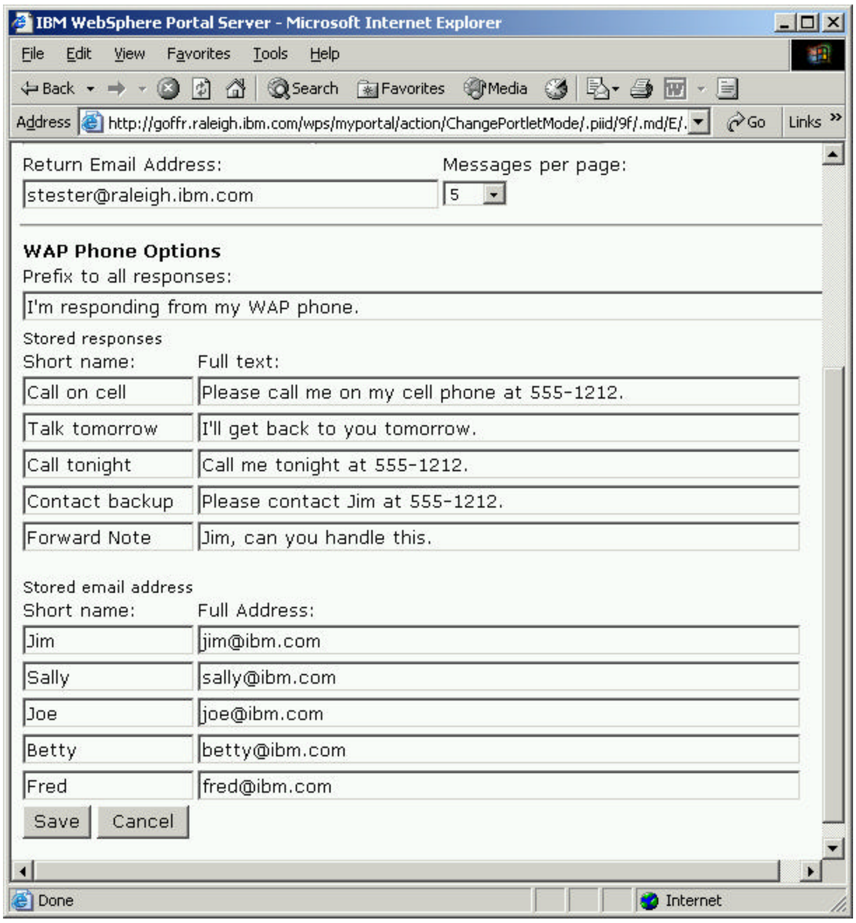
View details
Remove from list
Clear list

Search Results:

	Name	Title	Description	Supported Markups
⊕	Web Clipper	Web Clipper	the clipper runtime portlet	TODO:HTML CHTML
⊕	Web Clipping	Clipping Portlet Administration	the clipping administration portlet	TODO:HTML CHTML
⊕	Web Services	Web Services	Publish portlet	TODO:HTML CHTML

Figure 8.2: Selecting portlets based on their supported devices

Some portlets may be available for all the supported devices, while others may be available only on a single device. The user interface design of each portlet also varies from device to device, so that the user's experience can be made optimal on each device. Thus, the user's home page and each of the portlets might be very different on a mobile phone than they would be in a desktop browser.


IBM WebSphere Portal Server - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://goffr.raleigh.ibm.com/wps/myportal/action/ChangePortletMode/.piid/9f/.md/E/.
Go Links
Done
Internet

Return Email Address: stester@raleigh.ibm.com

Messages per page: 5

WAP Phone Options

Prefix to all responses: I'm responding from my WAP phone.

Stored responses

Short name:	Full text:
Call on cell	Please call me on my cell phone at 555-1212.
Talk tomorrow	I'll get back to you tomorrow.
Call tonight	Call me tonight at 555-1212.
Contact backup	Please contact Jim at 555-1212.
Forward Note	Jim, can you handle this.

Stored email address

Short name:	Full Address:
Jim	jim@ibm.com
Sally	sally@ibm.com
Joe	joe@ibm.com
Betty	betty@ibm.com
Fred	fred@ibm.com

Save Cancel

Figure 8.3: E-mail portlet settings optimized for mobile users

Supporting New Devices

CONTENT TRANSFORMATION

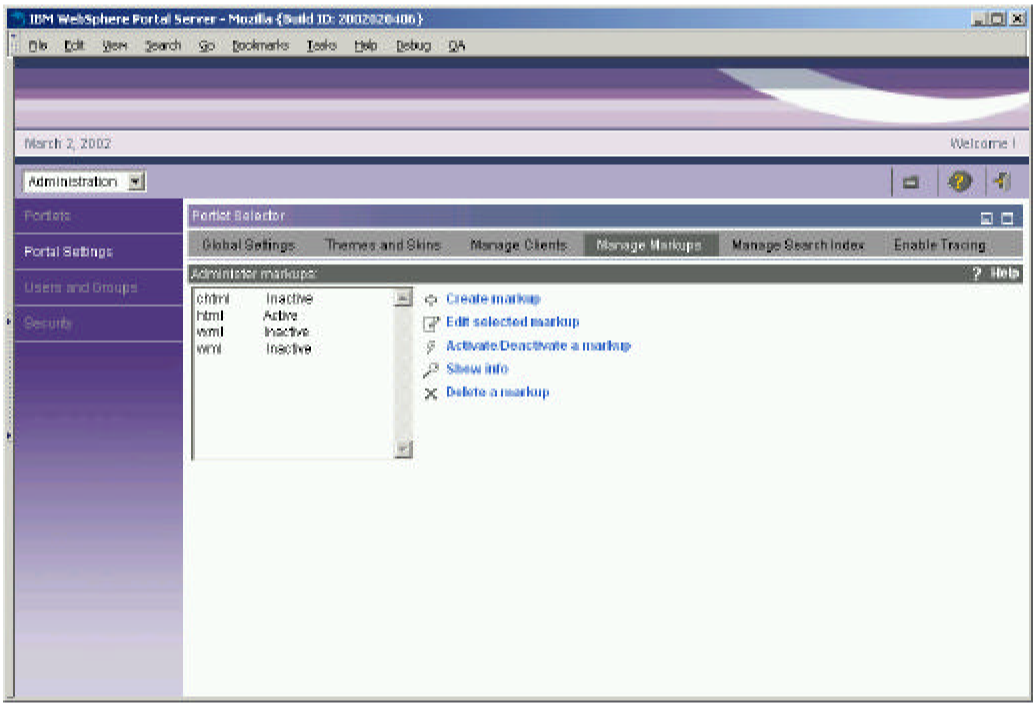
Adapting Content for new devices

The portal server supports several different markup languages so that portlets can render themselves for a variety of desktop and mobile browsers. Portlets that do not natively support the device markup can optionally be transformed using Transcoding technology [8]. This means that portlets can easily and automatically support mobile devices, even if the portlet developer did not explicitly support that device.

CLIENTS AND MARKUPS

The portal server's page aggregation subsystem supports several markup languages and recognizes certain browsers and mobile device user agent signatures, out of the box. If you would like to support additional markups or new devices, the framework is very easy to extend.

To support new browsers and devices, you add new markup and clients using the corresponding administration portlets. In the markups portlet, the markup name indicates the name of the folders that are used to store the page templates and the theme or skin files matching that markup language.

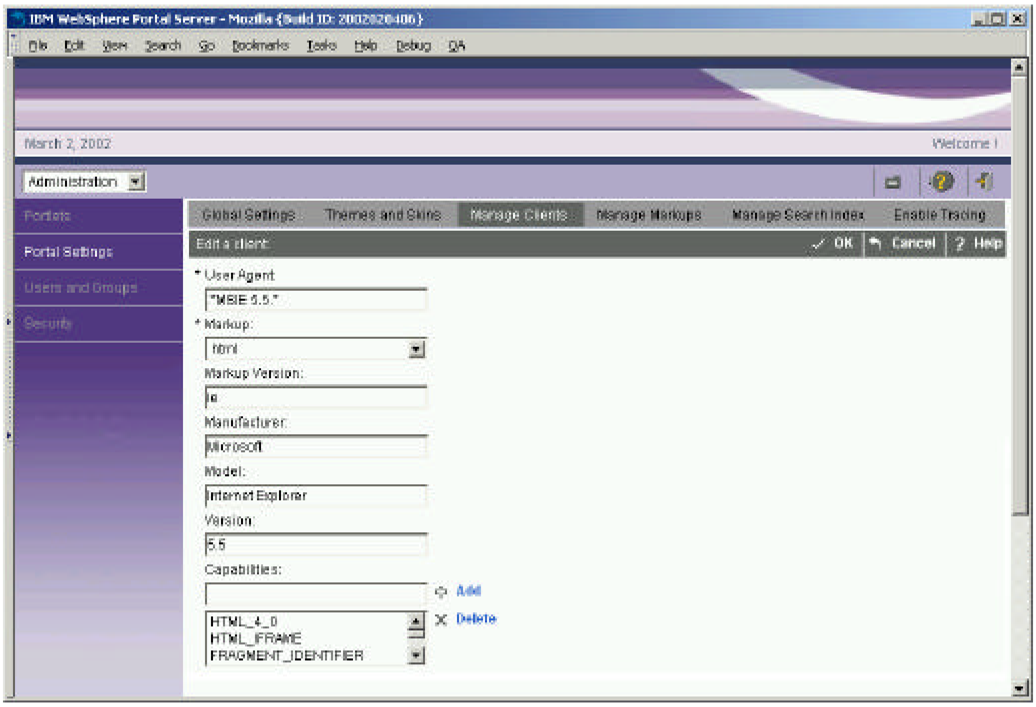


chtml	Inactive
html	Active
vxml	Inactive
wml	Inactive

Figure 8.4: Enabling the portal for new markups

To add a markup, create a new entry specifying the MIME type and the character set associated with that markup. You also need to add all the JSP templates associated with supporting a markup, such as new layouts, screens, skins, and style sheets.

When the portal server receives an HTTP request, it matches the values in the user agent header against known patterns that identify common browsers for desktops, mobile phones, and other devices. Entries for common clients are already set up, but you can add new ones using the clients portlet.



Edit a client

* User Agent: "MSIE 5.5."

* Markup: html

Markup Version: ie

Manufacturer: Microsoft

Model: Internet Explorer

Version: 5.5

Capabilities:

HTML_4_0
HTML_IFRAME
FRAGMENT_IDENTIFIER

Figure 8.5: Enabling the portal for new client browsers

The ordering of the client entries determines the order in which the patterns are applied against incoming HTTP requests. For further information, refer see the WebSphere Portal InfoCenter.

Glossary

Aggregation Process of building the portal page and all its portlets for presentation.

Authentication Determining whom the user is.

Authorization Determining what the user is allowed to do.

B2B Business to business. A business model that supports electronic transactions between businesses.

B2C Business to consumer. A business model that supports electronic commerce between a business and a consumer.

B2E Business to employee. A business model that supports electronic communications between a business and its employees.

Cache A buffer storage that contains copies of remote documents that are accessed by a portal; used to improve performance.

Card A WML document that provides the user-interface and navigational settings to display content on mobile devices.

Cascading Style Sheet A W3C standard mechanism for adding style (e.g. fonts, colors, spacing) to Web documents.

Certificate A cryptographic digital identity credential defined by the X.509 standard that can be used for authentication, digital signatures, secure exchange of encryption keys, etc.

CHTML Compact hypertext markup language. A format for publishing hypertext information on wireless devices.

Collaboration Immediate communication between individuals. Features of collaboration software include instant text messages, chat sessions, and shared documents.

Collaborative Filtering Personalization technology that employs recommendation engines that use advanced statistical models and other forms of intelligent software to extract trends from the behavior of Web site visitors.

Content Management Software designed to help businesses manage and distribute content from diverse sources.

Crawler A class of robot software that explores the World Wide Web by retrieving a Web document and following the links within that document. Based on the information gathered, a crawler creates indices for search engines.

Customization Modification of a portal page or portlet by an end user.

Deck An XML document that contains a collection of WML cards. See cards.

EJB Enterprise Java beans.

Enrollment The process of entering and saving user or user group information in a portal.

EIP Enterprise Information Portal. Software developed by IBM that provides tools for advanced searching, and content customization and summarization.

HTML Hypertext markup language.

HTTP Hypertext transport protocol.

HTTPS Secure hypertext transport protocol. See HTTP.

i-mode An Internet service for wireless devices.

ISP Internet service provider.

JDBC Java database connectivity.

JSP Java Server Pages.

LDAP Lightweight Directory Access Protocol.

LTPA Lightweight Third Party Authentication.

NewsML News markup language. An XML-based format for publishing news-related information.

NITF News industry text format. An XML-based format that defines the structure and content of news articles.

OCS channels Open content syndication channels. An XML-based format for syndicated content.

PDA Personal Digital Assistant. A small handheld computer that provides organizational features.

Personalization Personalization enables information to be targeted to specific users based on business rules and user profile information.

Pervasive computing The use of a computing infrastructure that supports information appliances from which users can access a broad range of network-based services, including Internet-based e-commerce services.

Portal A single, secure point of access to diverse information, applications, and people that can be customized and personalized. WebSphere Portal provides the required framework and functionality to develop and deploy many types of portals.

Portlet A Java application that is written to the portlet API and that runs within the portal server.

An area of content on a portal page that has a predefined role, such as retrieving the latest news headlines, driving a search engine, searching a database, viewing stock quotes, serving HTML files, or displaying a calendar.

RSS Rich site summary. An XML-based format for syndicated content.

SSL Secure sockets layer; a standard for encrypting TCP/IP connections.

Transcoding Content adaptation to meet the specific capabilities of a client device.

UDDI Universal Description, Discovery, and Integration, a registry of information about business entities and the web services they make available.

URI Uniform Resource Identifier.

W3C World Wide Web Consortium. A standards organization that develops interoperable technologies for the web, including specifications, guidelines, software, and tools.

WAP Wireless Access Protocol. An open, international specification for applications that are accessed by wireless devices.

WAR Web Archive File, a format distributing Java web applications.

WML Wireless Markup Language. A format for publishing and optimizing content on WAP-enabled devices.

XML Extensible Markup Language. A format for publishing and optimizing content on the worldwide web.

XSL Extensible Stylesheet Language. A format for transforming XML markup into other markups.

References

For more information

[1] WebSphere Portal, www.ibm.com/software/websphere/portal

[2] WebSphere Application Server, www.ibm.com/software/webservers/appserv

[3] IBM Tivoli Access Manager, www.tivoli.com/products/index/secureway_policy_dir

[4] WebSphere Personalization, www.ibm.com/software/websphere/personalization

[5] Lightweight Directory Access Protocol standard, www.ietf.org/rfc/rfc1777.txt

[6] Rich Site Summary resources, www.webreference.com/authoring/languages/xml/rss/

[7] WebSphere Studio, www.ibm.com/software/websphere/studio/

[8] WebSphere Transcoding Publisher, www.ibm.com/software/webservers/transcoding

[9] Integrating WebSphere Commerce and WebSphere Portal, www-3.ibm.com/software/info1/websphere/commerceportal.jsp

[10] Lotus product information, http://www.lotus.com/home.nsf/welcome/products

[11] WebSphere Site Analyzer, /www.ibm.com/software/websphere/siteanalyzer/

[12] IBM Enterprise Information Portal, www.ibm.com/software/data/eip/

[13] IBM Content Manager, www.ibm.com/software/data/cm/

[14] WebSphere Everyplace Server, www.ibm.com/pvc/products/wes_provider/index.shtml

[15] Web Services, www.w3.org/2002/ws/

[16] WebSphere Translation Server, http://wwwww.ibm.com/software/speech/enterprise/ep_8.html

[17] Accessibility resources and guidelines, www.w3.org/WAI

[18] Java Community Process, JSR 168, jcp.org/jsr/detail/168.jsp

[19] OASIS, Web Services for Remote Portals, www.oasis-open.org/news/oasis_news_01_28_02.shtml

Produced in the United States of America

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PAPER “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.

Information in this paper as to the availability of products (including portlets) was believed accurate as of the time of publication. IBM cannot guarantee that identified products (including portlets) will continue to be made available by their suppliers.

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

Any references in this document 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 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
4205 South Miami Boulevard
Research Triangle Park, NC 27709 U.S.A.