

The making of MetroSphere, Part 6: Getting started with WebSphere Portal - Express

Skill Level: Introductory

[Nicholas Chase \(ibmquestions@nicholaschase.com\)](mailto:ibmquestions@nicholaschase.com)
Consultant
Backstop Media

04 Apr 2003

This tutorial is for developers and administrators who want to get familiar with WebSphere Portal in general, and WebSphere Portal - Express in particular. It explains basic concepts such as portlet use and administration, user administration, and the basics of creating portal pages.

Section 1. Introduction

Should I take this tutorial?

This tutorial is for developers and administrators who want to get familiar with WebSphere Portal in general, and WebSphere Portal - Express in particular. It explains basic concepts such as portlet use and administration, user administration, and the basics of creating portal pages.

Editor's update: The Web site MetroSphere.com is no longer live. However, this series provides accurate and relevant information for installing IBM WebSphere Portal.

What is this tutorial about?

This tutorial is part of the MetroSphere project, chronicling the creation of the MetroSphere.com community site. MetroSphere provides an easy, customizable

way for the technical community to keep up with what's going on in their particular fields. The main site will run on a Linux box using WebSphere Portal Enable, but because of procurement delays, we're getting familiar with the software by installing WebSphere Portal - Express on a Windows box.

We'll be using Portal - Express to set up a small site from which to coordinate Express development, but this tutorial will focus on taking a look at the capabilities of Portal - Express.

This tutorial covers:

- General user concepts: what can a user do with a WebSphere Portal?
- Managing users and groups: How does security work?
- Managing portlets: What are they, and how does the system see them?
- Portal settings: What can an administrator control about the site?
- Creating a new page: How can a user or administrator add and control content?
- Using the generic portlets: Is there an easy way to add standard content such as Web pages and servlets?
- Creating a cliplet: What is a cliplet, and how can an administrator make one?

Tools

In order to follow along with the steps in the tutorial, you will need access to a server running IBM WebSphere Portal - Express or one of the other WebSphere Portal offerings. It's not necessary to have physical access to the server, but you must be able to access it via a browser.

Section 2. Using a portal

What is a portal?

WebSphere Portal - Express (and the rest of the WebSphere Portal family) is designed to make it easy to create a portal site, so it's probably best to take a moment to discuss just what is meant by "portal."

In today's increasingly information-centered world, it's easy to become overwhelmed. Part of the success of weblogging, where a person posts links to interesting sites that they've found, is due to the fact that there is now so much information on the Web that it's impossible for one person to keep up, even if he or she is just trying to track a single topic. A portal site such as MetroSphere.com can make that easier by keeping a lot of information in one place, and the best portals enable the user to customize that information.

For example, a bank might create a portal that not only provides easy access to a user's account information, but also helps track their investments and provides breaking news on companies and industries in which they have a financial interest.

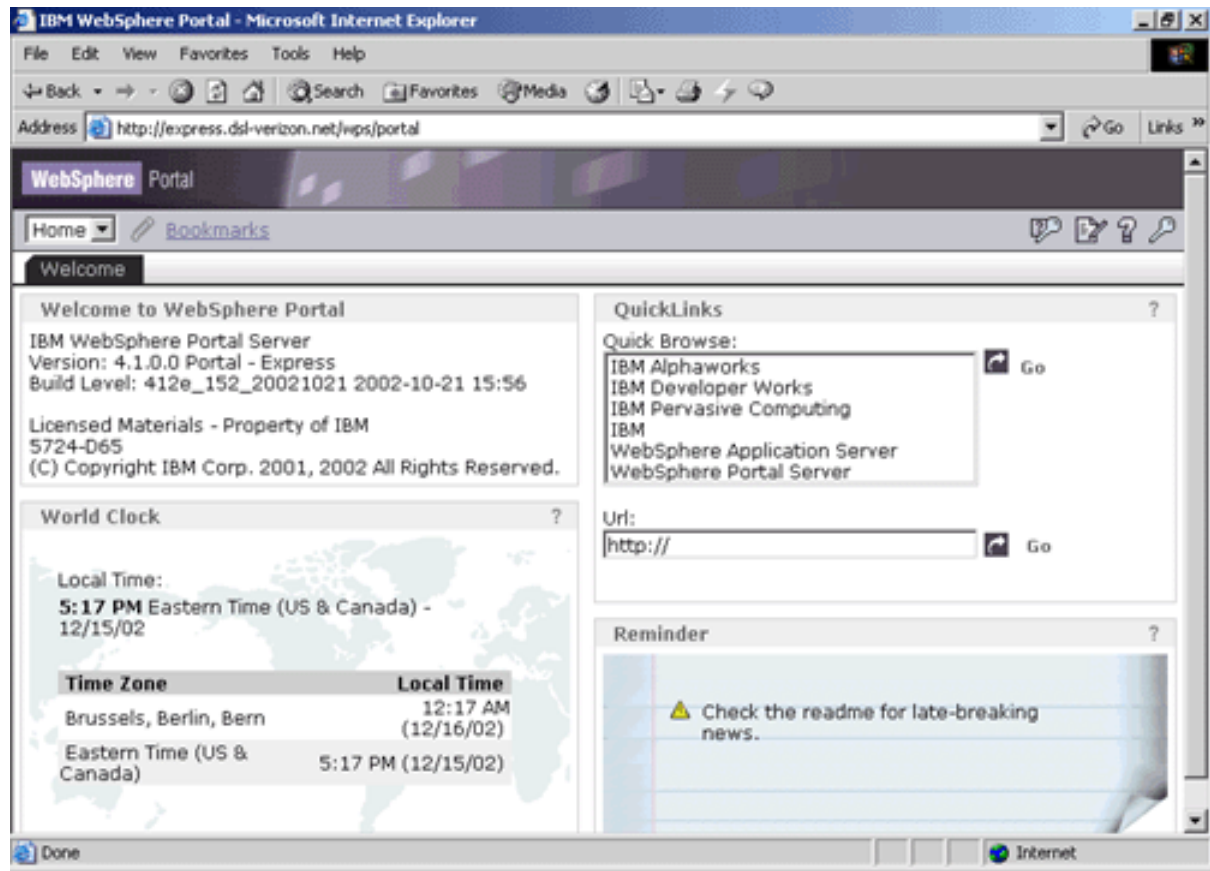
The portal might also enable a customer to choose what functions are on a page and where they should appear. The functions themselves are known as portlets.

What is a portlet?

First Sun introduced Java technology, and the small applications, called applets, that ran in the browser. Then people started running Java applications on the server, and those small applications were known as servlets. Now those small applications have been made to run within the framework of a portal, so they're known as portlets.

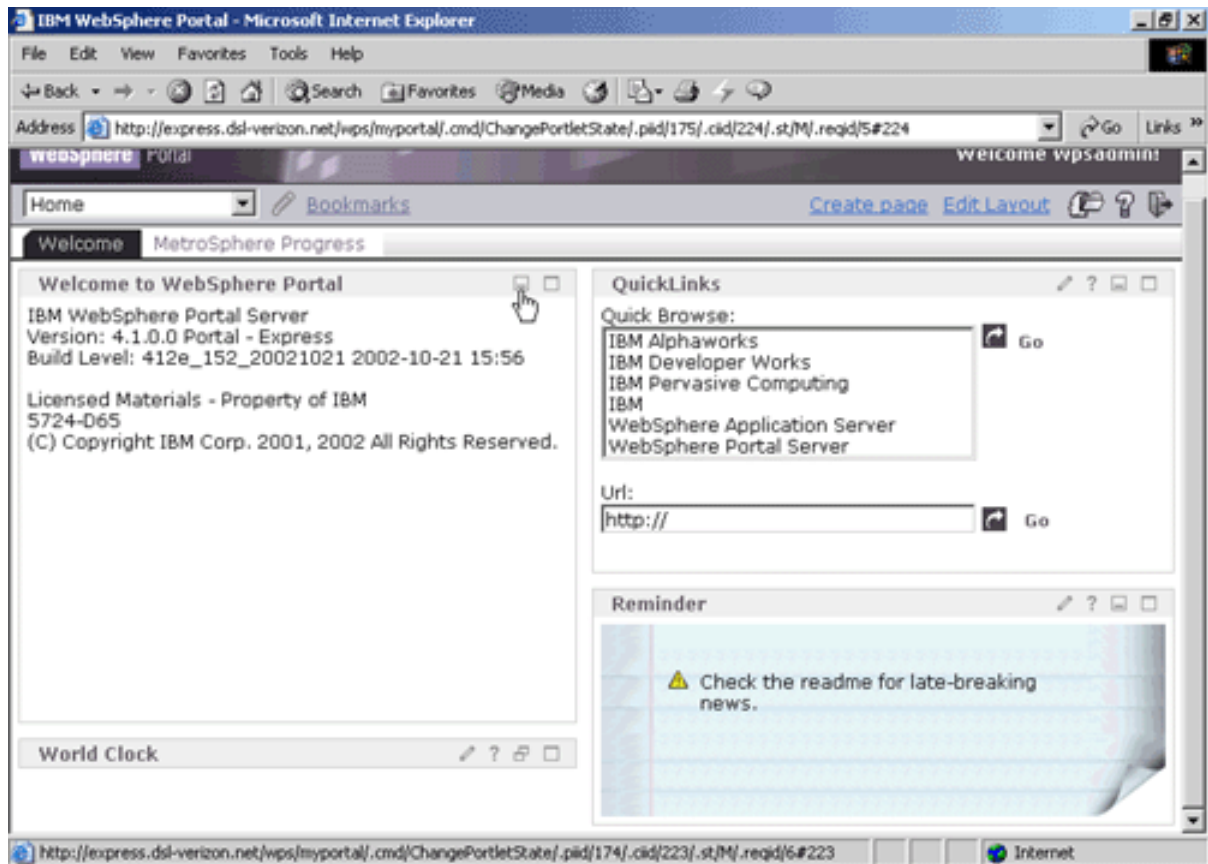
In WebSphere Portal V4.1, portlets are a special type of servlet. When a page is called, the server calls each of the portlets contained within it (in no particular order) and when they've all finished processing, it returns the page.

These portlets can be independent of each other, or they can communicate. For example, the default portal home page has four portlets -- **Welcome to WebSphere Portal**, **QuickLinks**, **World Clock**, and **Reminder** -- each of which is independent of the others.



Portlets as mini-windows

One interesting feature of portlets is that they can act, in many ways, like the individual windows of a GUI application. For example, when MetroSphere users log on to the portal, they will be presented with two or more options in the upper right-hand corner of each portlet. Like a traditional window, you can minimize a portlet so that only the title bar shows, leaving more room for other portlets, or you can maximize it, so that it takes up the entire window.

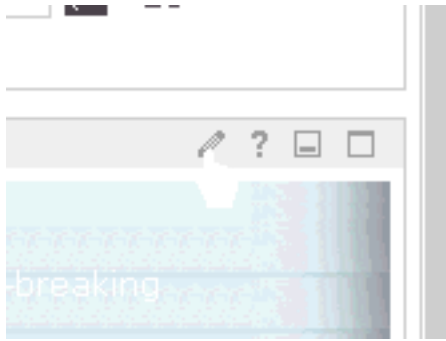


Portlets can also provide help capabilities, and some provide a means for editing, or interacting with the individual portlet, as shown next.

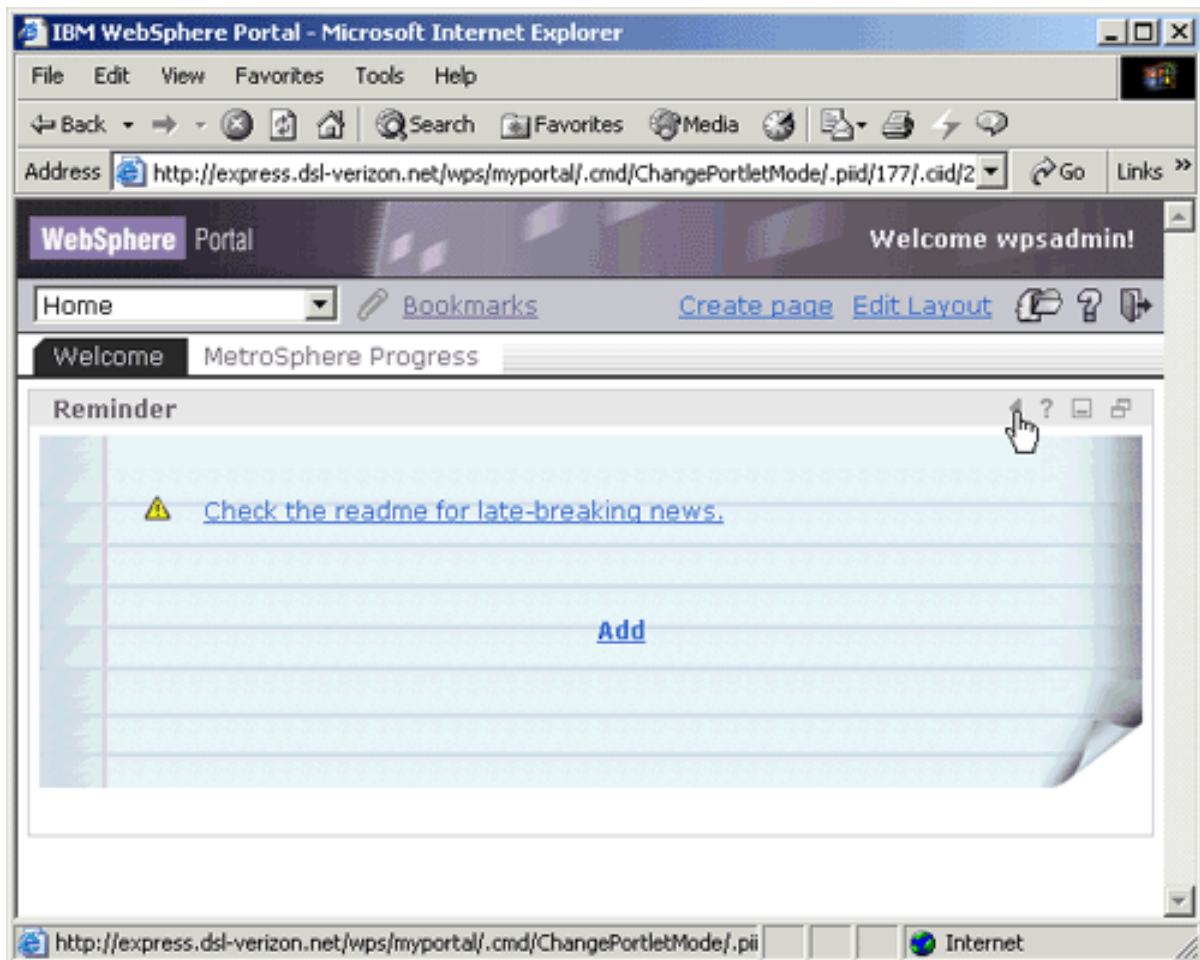
Interacting with portlets

Some portlets, such as the **Welcome** portlet on the default home page, are intended to simply display information, but that's not always sufficient. For example, we'll want MetroSphere.com users to be able to create portlets that point to specific syndicated news sources, so to do that, they'll need to be able to somehow interact with a portlet.

The **Reminder** portlet on the home page is an example of such a portlet, enabling users to enter their own reminders by clicking the pencil on the title bar. Doing so brings up a page that shows the portlet in *edit* mode, which in this case is provided by a separate JSP page that's called by the server.



Whenever a portlet is maximized, either because it's being edited or because the user clicked the **maximize** button, the user can go back to the previous page by clicking the portlet's **back** button. It's important to understand this, because in many cases, the portal server records the *state* of a particular page, so using the browser's **back** button doesn't always do the trick.



Pre-installed portlets

The Setup Manager for WebSphere Portal - Express V4.1 installs a number of portlets by default, though most of them are not added to any particular page. Virtually everything you do on a portal is accomplished through a portlet, and that includes the administrative functions. So as this tutorial will show, portlets implement all of the capabilities you will see. Other portlets installed in a basic Portal - Express system are:

- Banner Ad
- Bookmarks
- Calculator
- CSV File Viewer
- Document Search
- EIP Advanced Search
- EIP Federated Search
- EXCEL Document Viewer Portlet
- FileServer
- iNotesCalendarPortlet
- iNotesContactsPortlet
- iNotesMailPortlet
- iNotesNotebookPortlet
- iNotesToDoPortlet
- JSPServer
- Microsoft Exchange 2000 Calendar
- Microsoft Exchange 2000 Contacts
- Microsoft Exchange 2000 Mail
- Microsoft Exchange 2000 Notes
- Microsoft Exchange 2000 Tasks
- Moreover News
- MS Exchange Calendar Portlet
- MS Exchange Contacts Portlet
- MS Exchange Journal Portlet

- MS Exchange Mail Portlet
- MS Exchange Notes Portlet
- MS Exchange Tasks Portlet
- MyNotesCalendarPortlet
- MyNotesMailPortlet
- MyNotesTodoPortlet
- NotesDiscussionPortlet
- NotesMailPortlet
- NotesTeamroomPortlet
- NotesViewPortlet
- OCS News Feed
- PDF Document Viewer Portlet
- PowerPoint DocumentViewer Portlet
- QuickLinks
- QuickplacePortlet
- QuickPlacePortlet
- Reminder
- RichText Document Viewer Portlet
- RSS Portlet
- SametimePortlet
- ServletInvoker
- SQL Portlet
- Web Clipper HTML Template
- Web Clipper Multi-device Template
- WebPagePortlet
- Welcome Portlet
- WORD Document Viewer Portlet
- World Clock

- WPS Mail

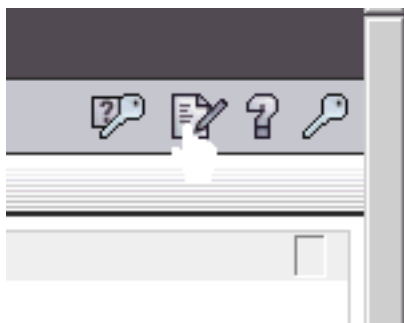
As new portlets are released, they are listed in the Portlet Catalog (see [Resources](#)).

Section 3. Users and groups, and security

Creating a new account

When you install Portal - Express, the `wpsadmin` user is automatically created, but it's not a good idea to do everything through that one user, particularly if you have more than one administrator. Instead, each person should have his or her own account with administrator privileges. To that end, I need to create an account for myself (and eventually one for Tom) and add it to the `wpsadmins` group.

The first step, creating the user, is easy, because the portal is set up so that users can create their own accounts. (If you need to remove that capability, it's as simple as editing a JSP page.) The process is called *self-care*, and users can easily enter and change their own information, or an administrator can do it from within the Portal Administration area. To start, the user clicks the **Start Registration** button.



As a developer or an administrator, you can control the items that appear on the registration page. Each item in the form is simply a traditional HTML form element with the attribute name prefaced by `wps.`, so the preferred language field is called `wps.preferredLanguage`. To add a new field to user records, simply add it to the form. For example, to add a new user attribute called `location`, simply add a form field named `wps.location` to the `PrepareEnrollement.jsp` file.

IBM WebSphere Portal - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Home Search Favorites Media Print Refresh

Address <http://express.dsl.verizon.net/wps/portal/.cmd/PrepareEnrollment> Go Links

Continue | Cancel

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:

Interests:

*Required field
Sports
Music
Politics

Done Internet

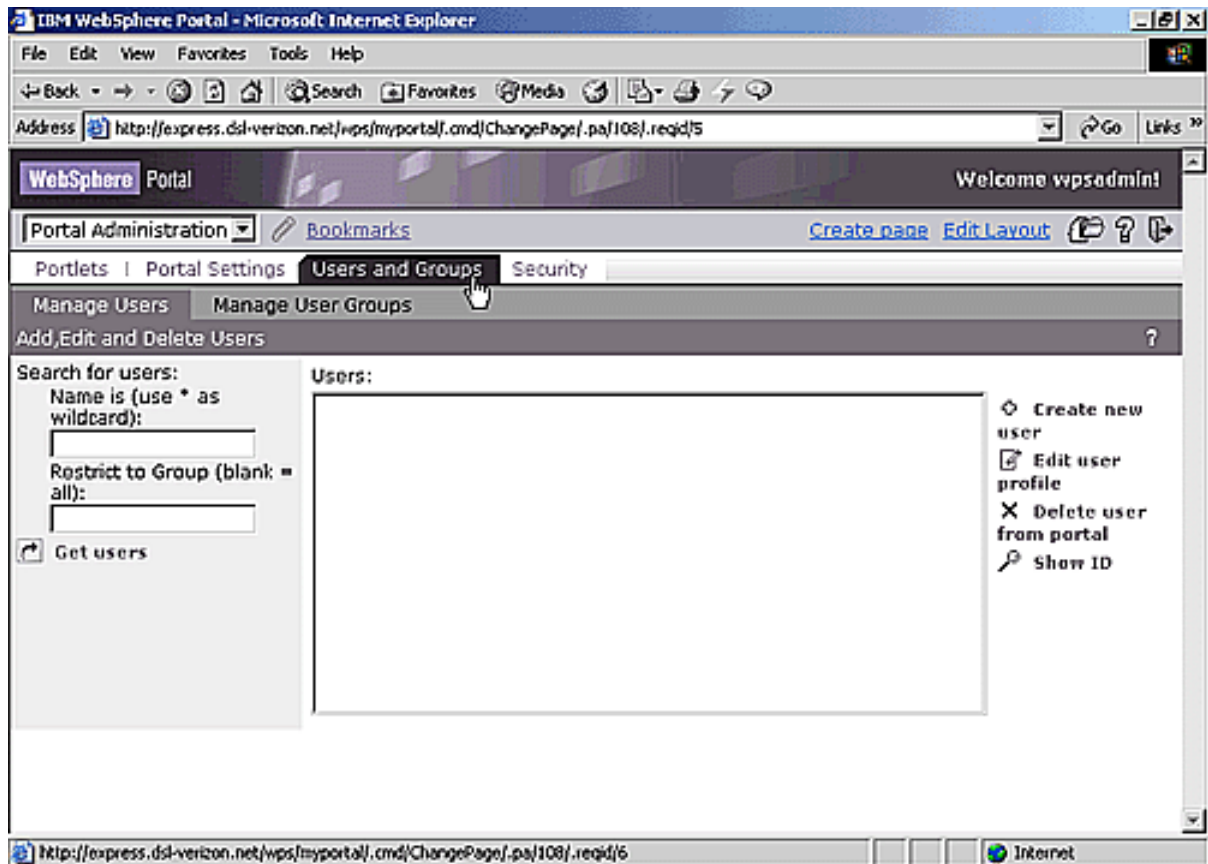
In fact, all of the registration and self-care pages are completely configurable. When we're ready to sign up MetroSphere.com users, we'll revamp the look and information on all of these pages, and as long as we name the fields right, the user's information will be properly stored in the database.

To complete the registration, click the **Continue** button, and then confirm the information by clicking **OK**.

Administering users

The first step in making the new user an administrator is to go to the Portal Administration section of the portal. Log in as `wpsadmin` -- the default password is simply `wpsadmin` -- and choose **Portal Administration** in the list of places in the upper left-hand corner of the window.

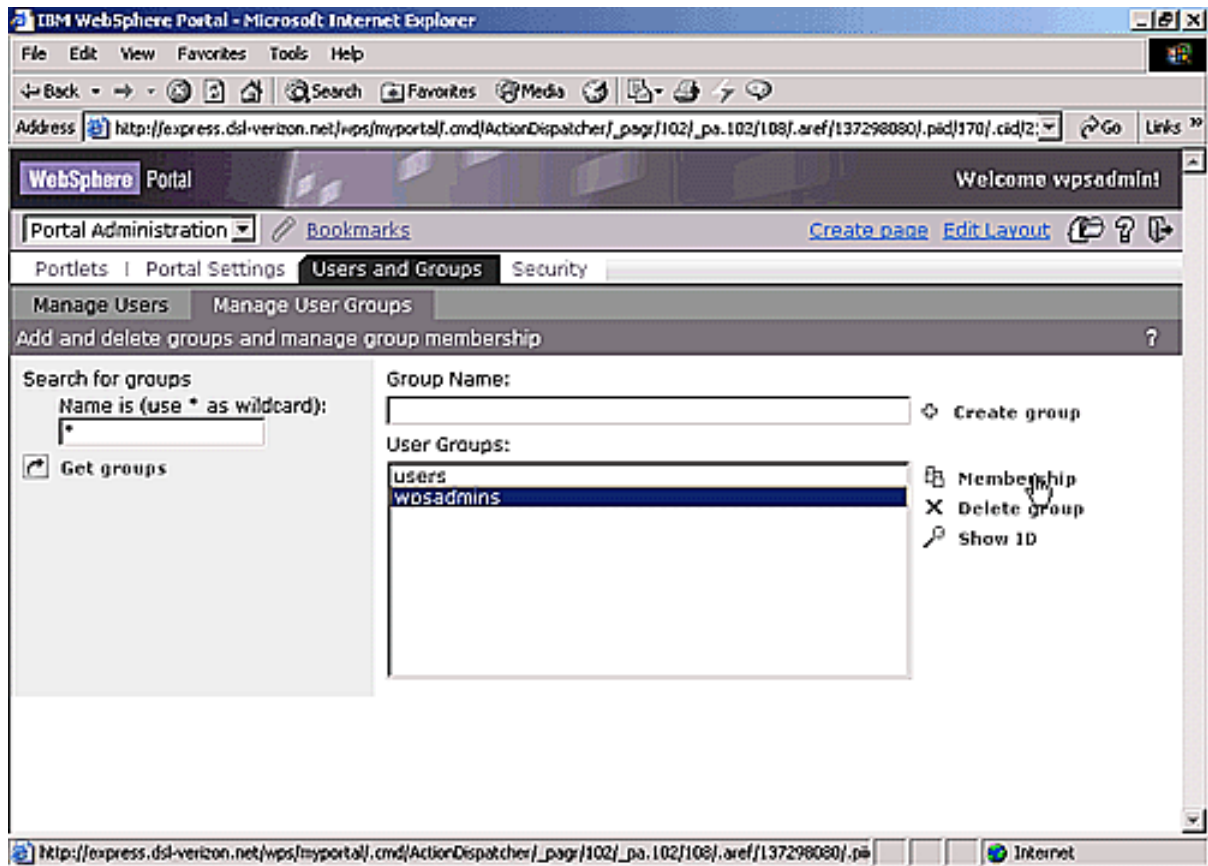
Choose **Users and Groups** to get to the user management area.



The natural temptation here would be to search for the new user, but it won't actually show up until it's added to at least one group. Click **Manage User Groups** to start the process.

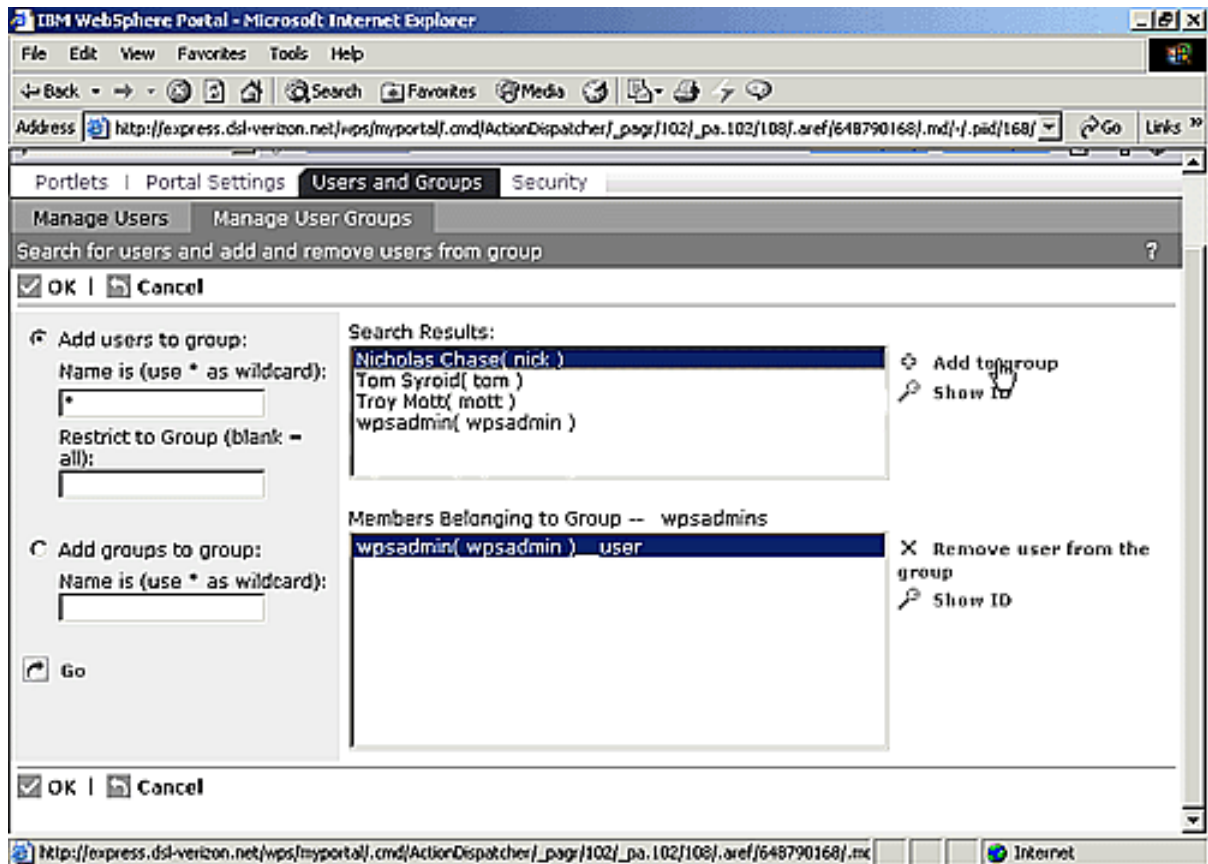
Get groups

From within the **Manage User Groups** portlet, do a search for all existing groups by entering an asterisk (*) in the **Name is** field and click the **Get groups** button. The page is then populated by all of the existing groups. Click **wpsadmins**, then click **Membership**.



Add the user to the group

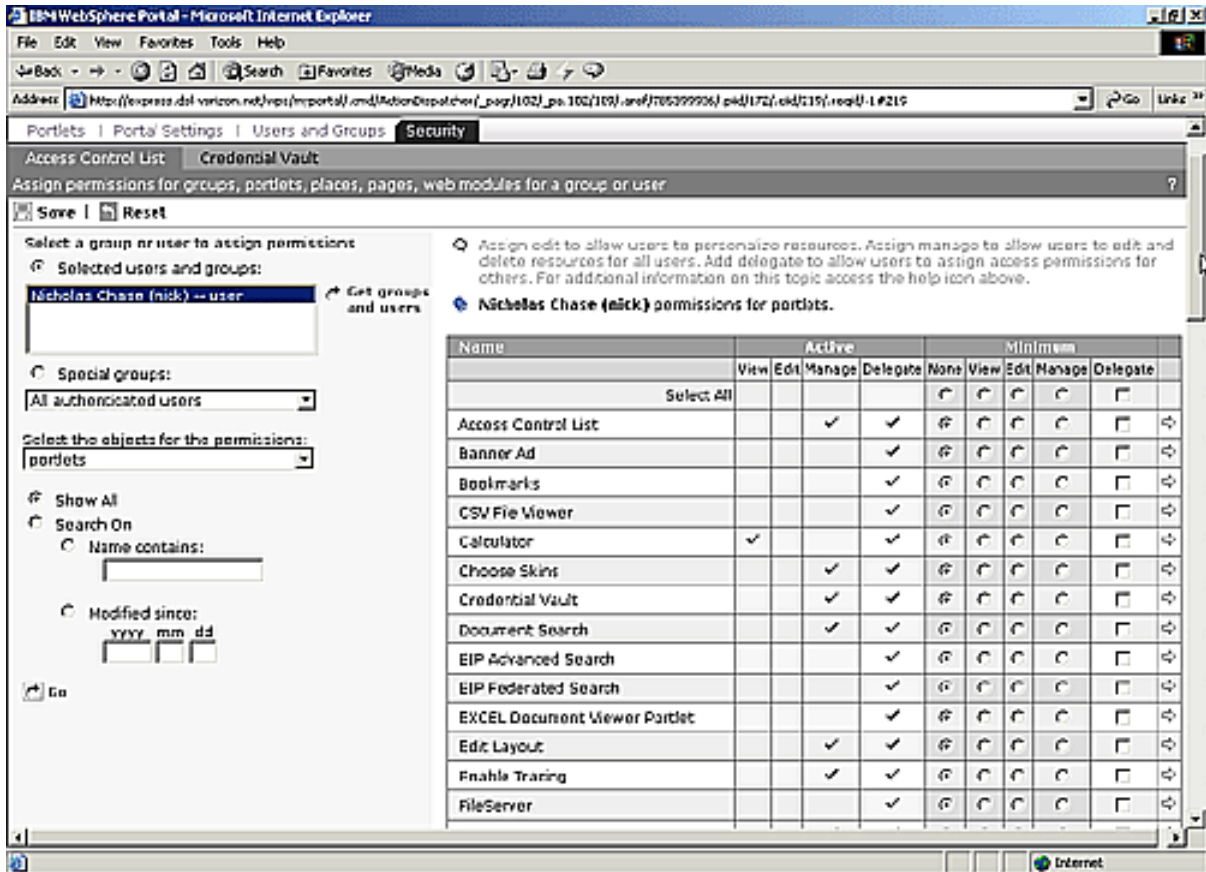
To add the new user to the group, either enter another asterisk (*) in the **Name is** field, or search for the name directly. Highlight the user (in this case, **nick**), and click **Add to group**.



You can also use this process to add the new user to the **users** group.

The Access Control List

It's probably obvious that we've created the new admin user so that it will have privileges to do the same types of things we can do with the `wpsadmin` user, but it might not be as obvious just how deep security runs through the entire WebSphere Portal project. Clicking the **Security** tab will bring you to the **Access Control List**, which shows the access rights a user has for a particular resource, whether that resource is a portlet, a page, or an entire group of pages such as **Portal Administration**.

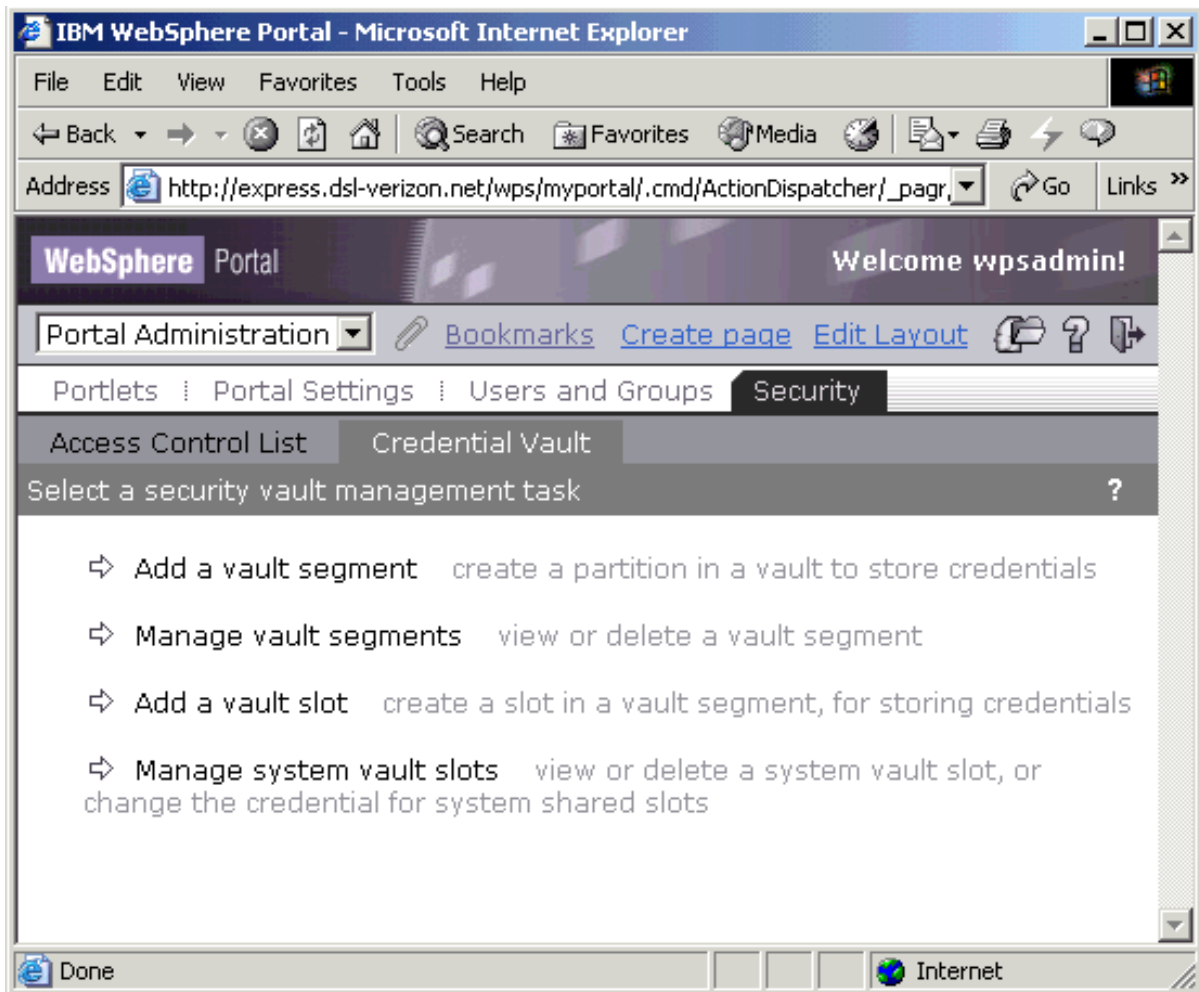


The Access Control List is a major topic unto itself, but here are some important facts to understand:

- What a user sees is completely dependent on information in the Access Control List. A user will see any resources for which he or she has **VIEW** privileges, and any for which he or she does not will simply not appear. This saves the developer from having to manage this directly. For example, MetroSphere users who aren't administrators won't even see the Portal Administration link.
- A user will have **MANAGE** privileges on any object he or she creates. For example a MetroSphere user who creates a new page will have the MANAGE privilege on it, while by default no other users will have any privileges on it.
- A user with the **DELEGATE** privilege on a resource can give another user any privilege up to and including the one he or she holds. For example, a MetroSphere user who creates a new page can give another user any privileges on it, but if he or she gives another user EDIT and DELEGATE privileges, that second user can only give VIEW or EDIT privileges to a third user.

The Credential Vault

The other option within the **Security** section is the **Credential Vault**. Because we're just taking a general look at the capabilities of Express, we won't spend a lot of time here, but understand that the Credential Vault provides a number of classes that you can build into your portlets in order to access external applications.



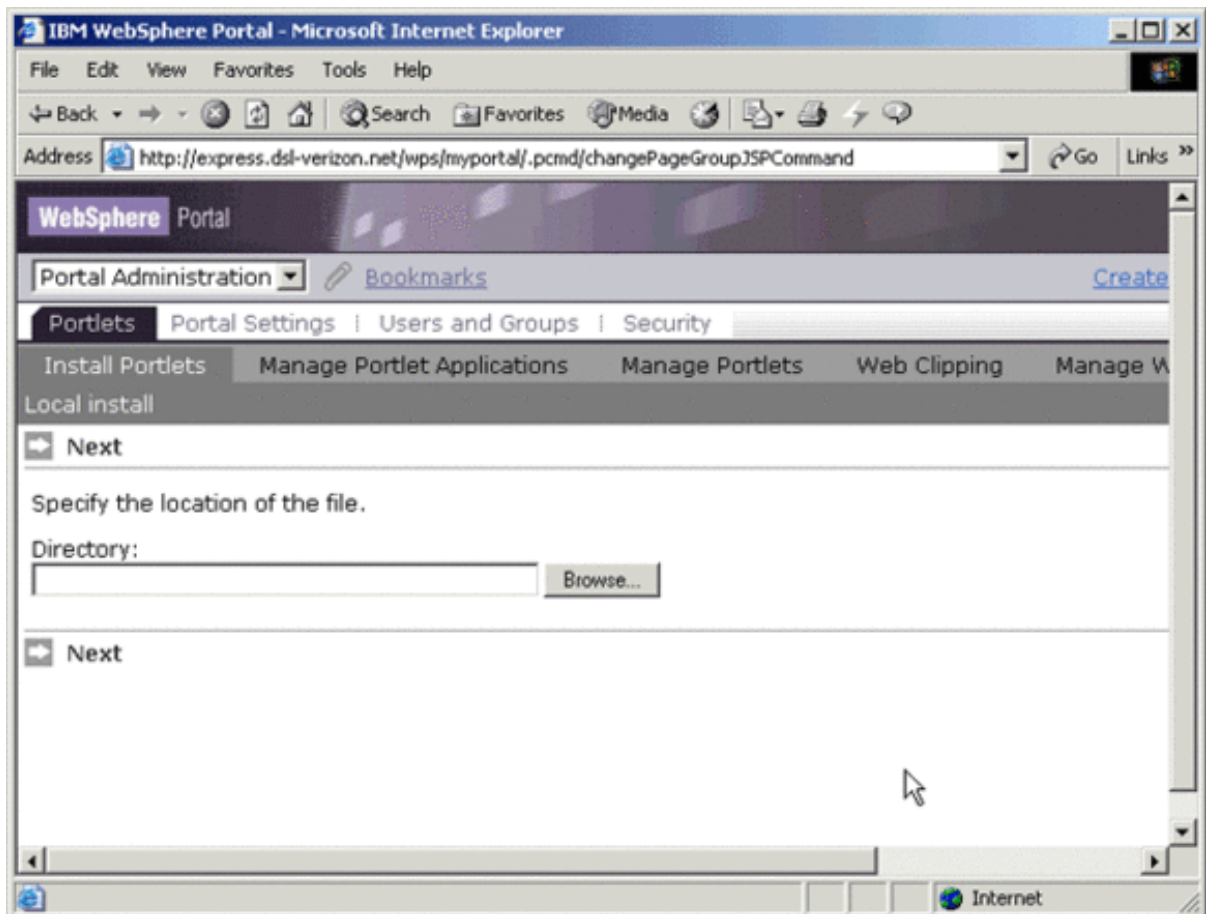
For more information on the Credential Vault, see the InfoCenter documentation. (See [Resources](#).)

Section 4. Managing portlets

Adding new portlets

This section will cover the administrative tasks we'll eventually need in order to manage the portlets we'll be creating and using as part of MetroSphere.com. Under **Portal Administration**, click the **Portlets** tab.

WebSphere Portal - Express has the same underlying structure as its cousins, so it can use portlets that have been developed for, say, Portal Enable, and vice versa. In addition to portlets from the Portlet Catalog (see [Resources](#)), you can write your own portlets using the Portlet API. In either case, the portlets are packages in a specially-constructed WAR file, which you can then install by uploading it through the browser.

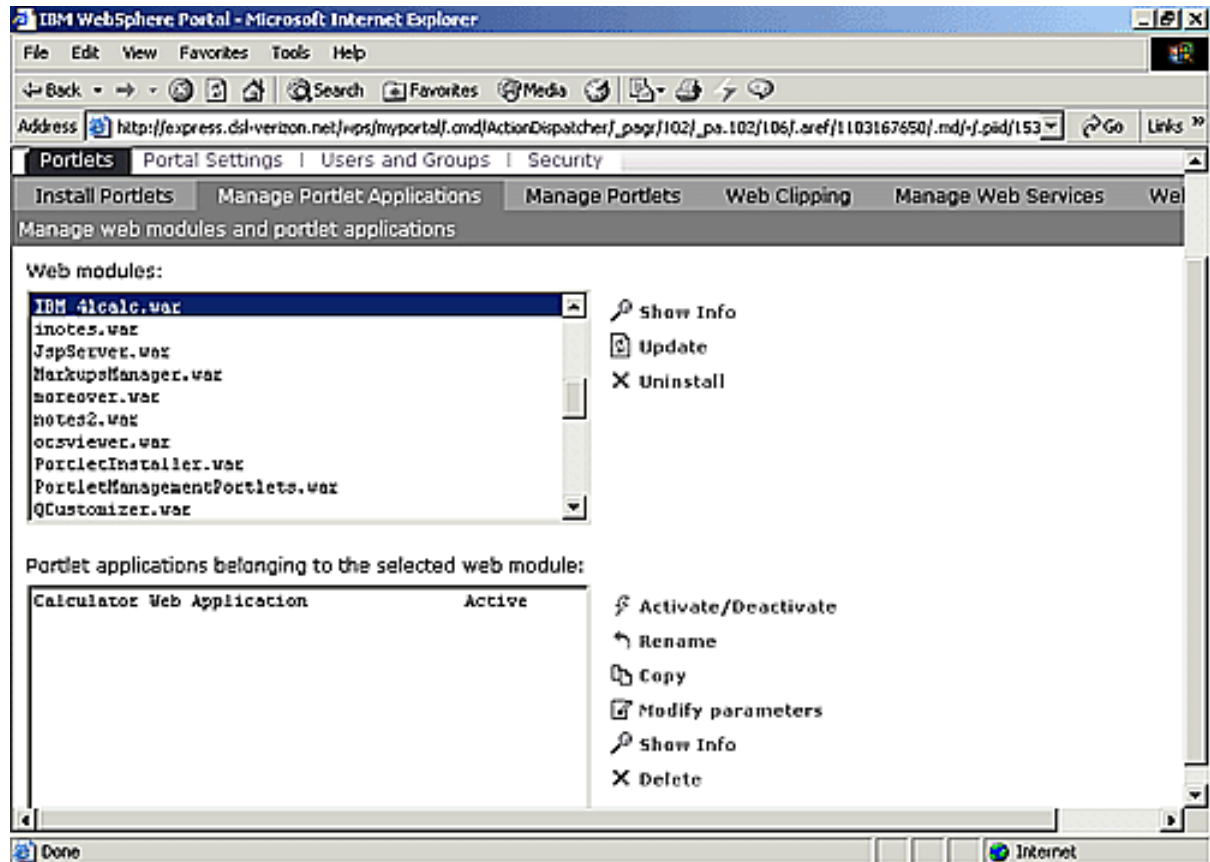


Portlet applications

The WAR file in which a portlet is deployed can actually contain much more than simple portlet classes. In addition to the fact that portlets generally also require support files such as Java Server Pages, a single WAR file, or Portlet Application,

can contain more than one discrete portlet.

Clicking **Manage Portlet Applications** gives you access to information about the WAR files themselves, where you can uninstall entire applications or simply activate or deactivate individual portlets within an application. To see what portlets are contained in an application, simply highlight it in the top field. The bottom field automatically updates to show the relevant portlets.

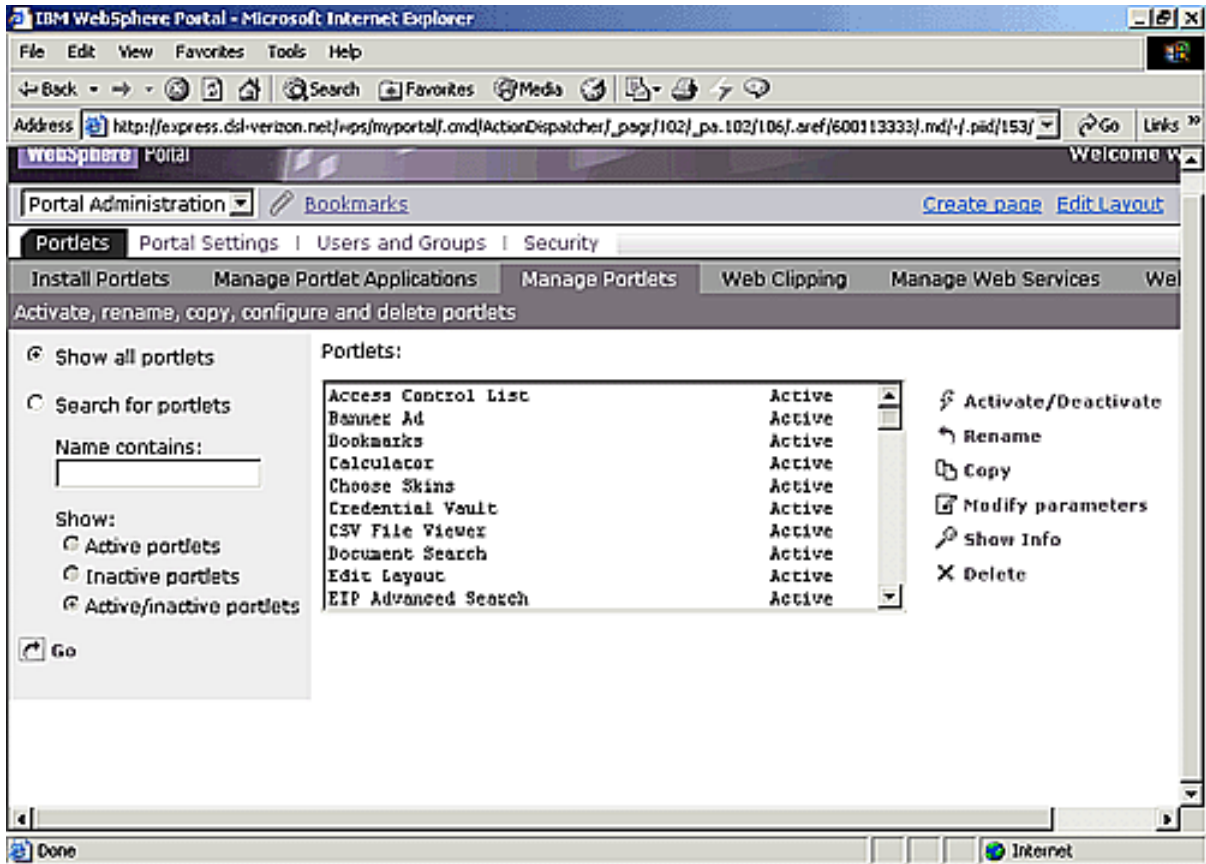


Individual portlets, however, are typically managed from the **Manage Portlets** page.

Manage portlets

Click **Manage Portlets** to see a list of the portlets available. If your system has a large number of portlets installed, you can use the search on the left-hand side to narrow down your choices.

At any given time, an individual portlet can be active or inactive. Deactivating a portlet prevents it from appearing on any page to which it's been added, but reactivating it adds it back in. This capability makes it easy to take portlets off-line in order to update them.



In [Using the generic portlets](#) we'll look at copying, renaming, and modifying the parameters for a portlet.

Web clipping

Another option in the Portlets Administration area is **Web Clipping**. A Web Clipping provides a way to add a portion of an external Web page to your site. For example, in [The Web Clipper portlet](#), we'll create a Web Clipping, or cliplet, that shows the content on the developerWorks home page. Later, when we set up the portal for our internal users, we can add this cliplet to the page, enabling us to easily keep up with new content that's been added each day.

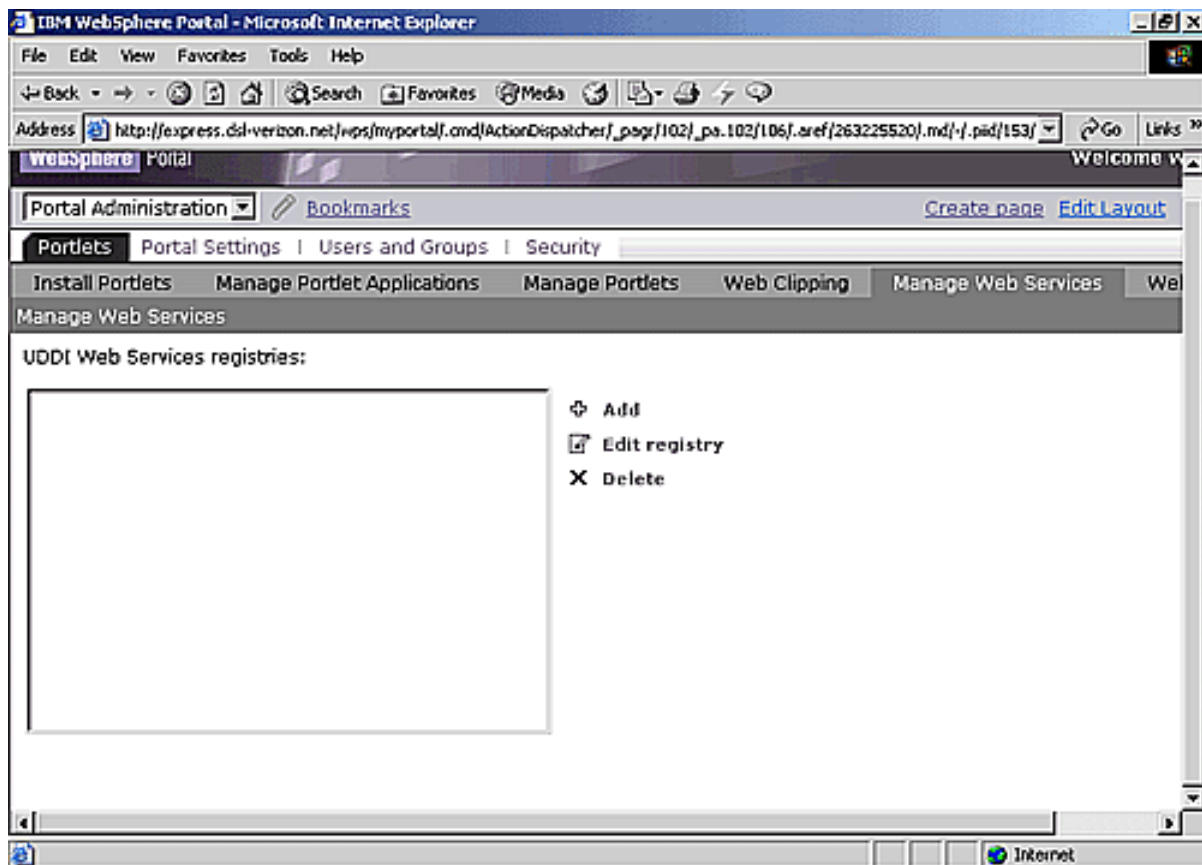
We'll cover Web Clipping in detail in [The Web Clipper portlet](#).

Manage Web services

One of the strengths of WebSphere Portal is the ability to seamlessly integrate with Web services. A Web service is an application that resides on a remote server, taking requests in a particular predictable format and returning results in a similar

predictable format. For example, we might create a portlet that accesses a Web service that returns the weather for a MetroSphere user's local area, or even the traffic. The information can then be seamlessly displayed on the page.

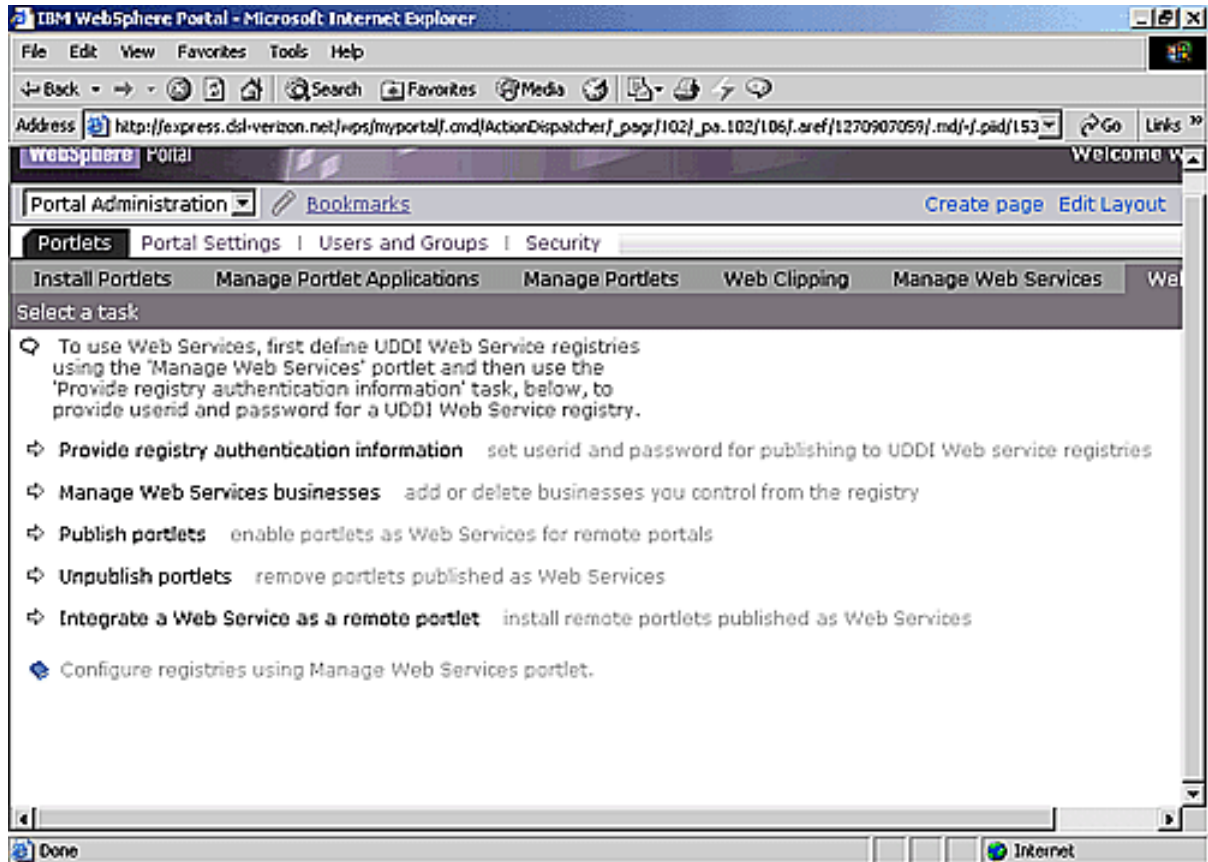
A UDDI registry is a collection of these services, searchable by company, industry, service, and other parameters. The **Manage Web Services** portlet provides the means for managing connections to one or more registries, which you can then search for specific information.



Web services

In addition to simply accessing Web services, WebSphere Portal enables you to publish portlets as Web services. For example, we might create a portlet that displays the MetroSphere top story for the day, then publish it as a Web service so that other sites and applications can integrate that information into their own applications and portals.

The **Web Services** portlet provides links to all of the administrative tasks involved in publishing portlets, registering them with UDDI registries, and making use of external portlets that have been published as Web services.



None of these portlets exist in a vacuum, of course. The next section will look at settings for the portal itself.

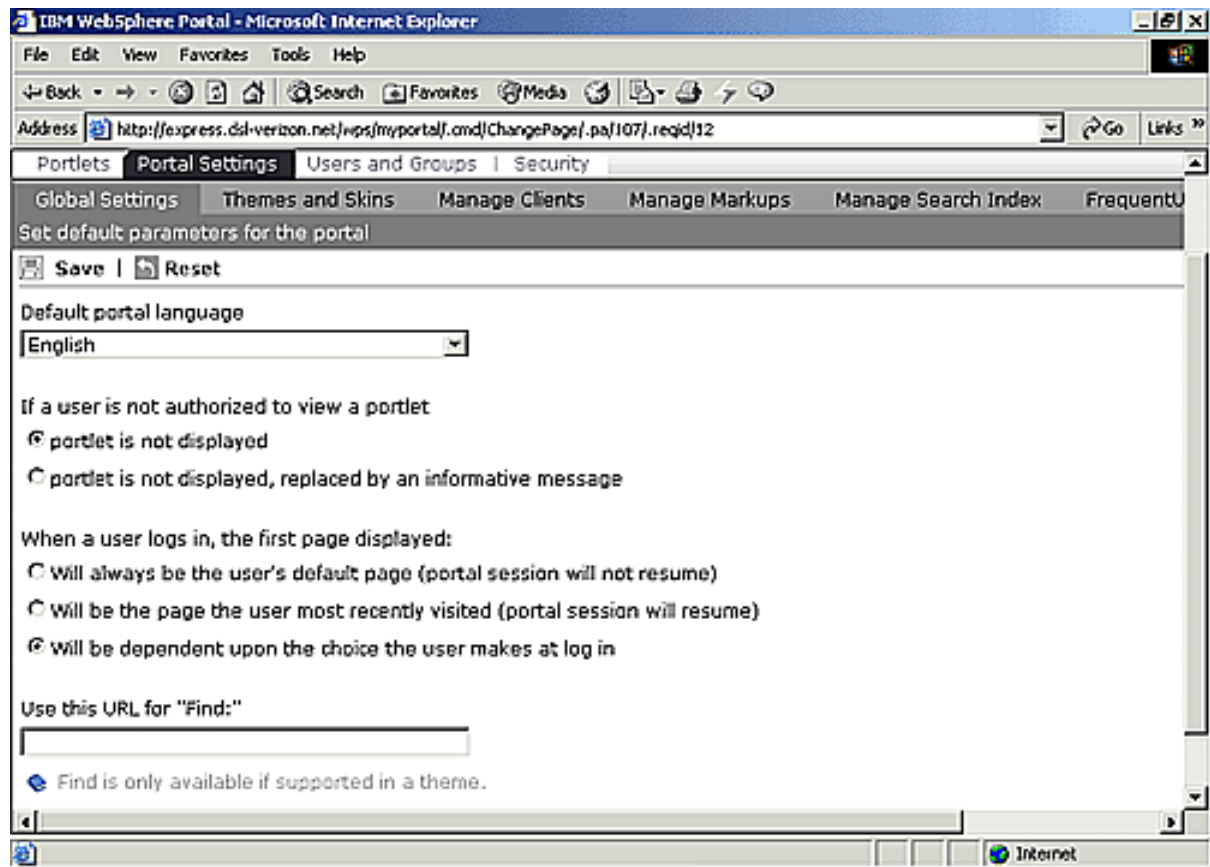
Section 5. Portal settings

Global settings

While portlets take care of the basic functionality provided by the portal, a number of different factors determine its appearance. The portal adapts to different browsers (displaying a different view on a mobile phone than a traditional browser), and different languages (displaying content in the user's native language, if it's available), and allows developers or users to choose a different theme, or look-and-feel, for their content.

The **Portal Settings** section of Portal Administration manages All of these capabilities.

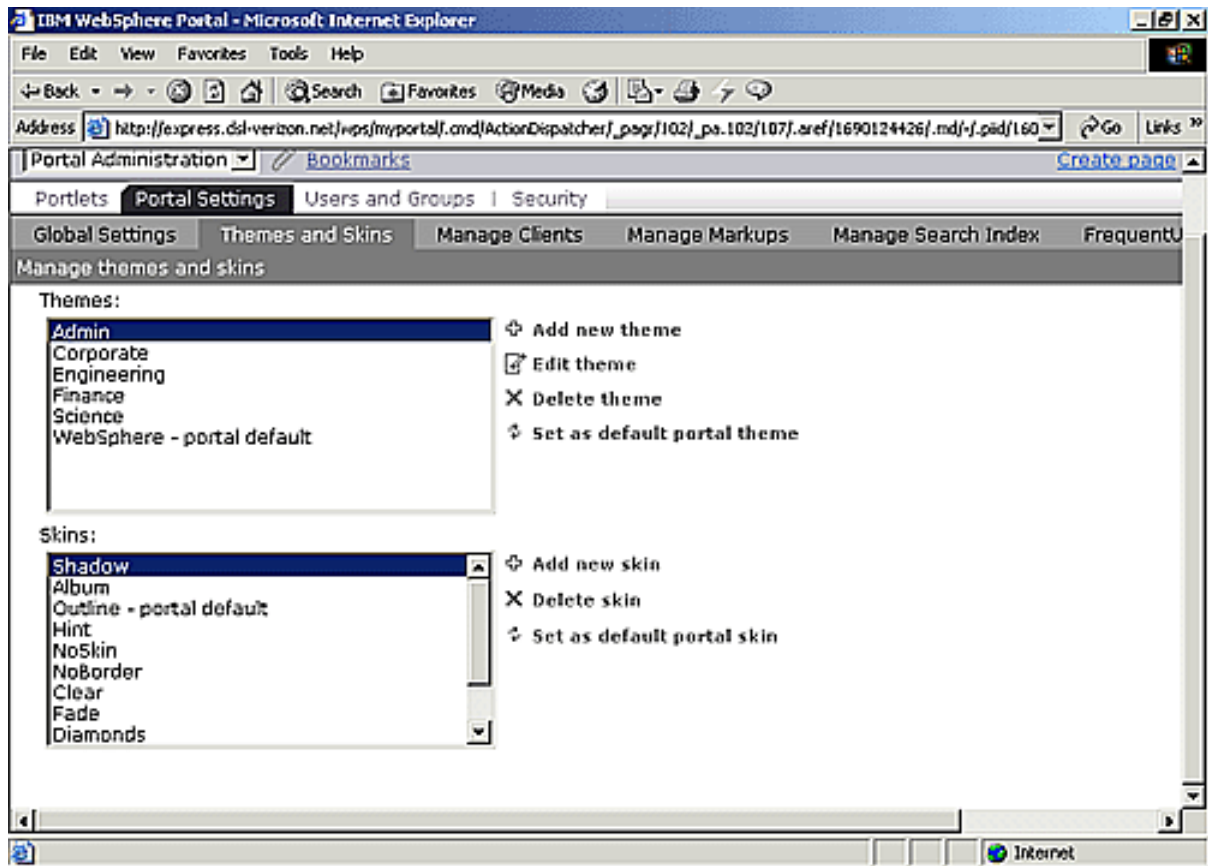
The **Global Settings** tab brings you to the portlet that controls the overall behavior of the portal.



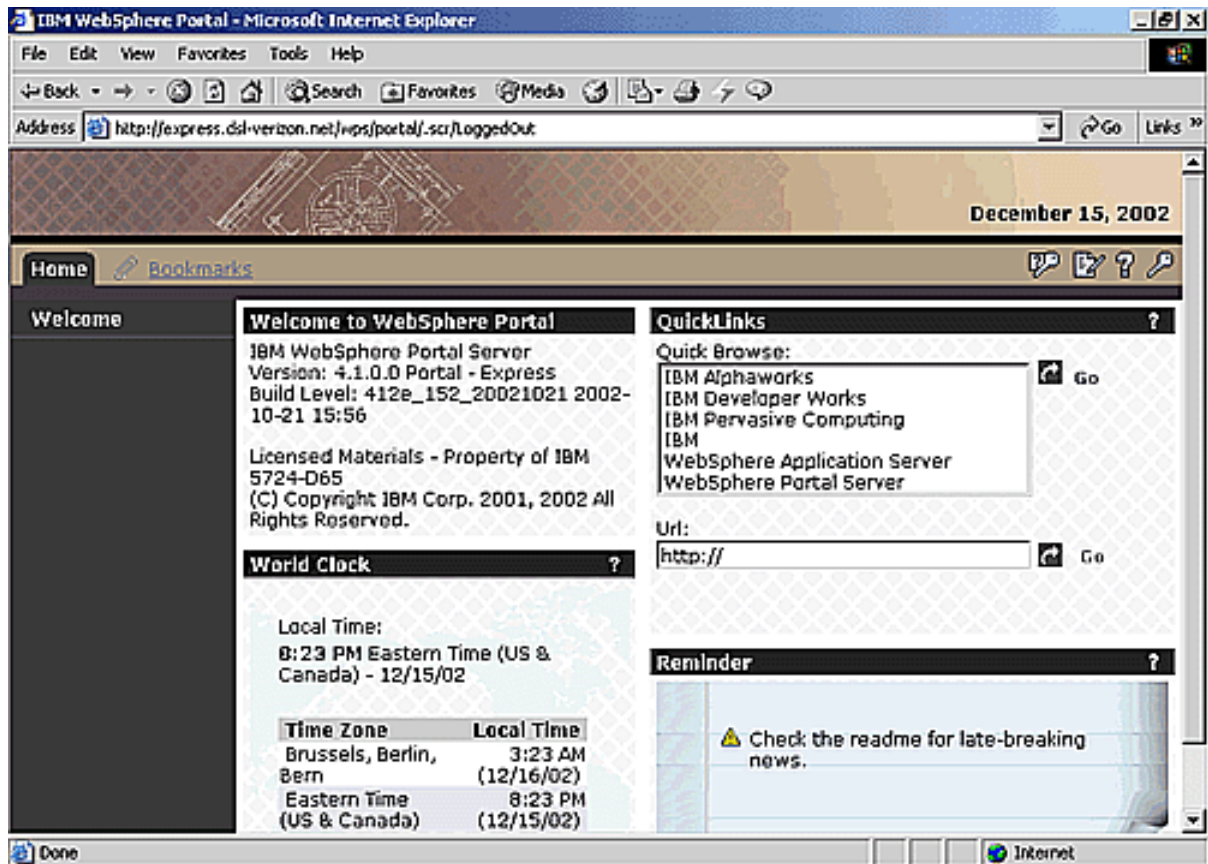
Themes and skins

The overall look of the portal is controlled by a theme or a collection of JSPs, graphics, and style sheets. Out of the box, Portal - Express comes with six different themes, with five of them suitable for public pages. You can easily create new themes using JSP pages and special tags, and add them to the portal. Click the **Themes and Skins** tab to see the themes that are currently available.

You can also associate each theme with one or more skins, which determine the appearance of individual portlets within a page.

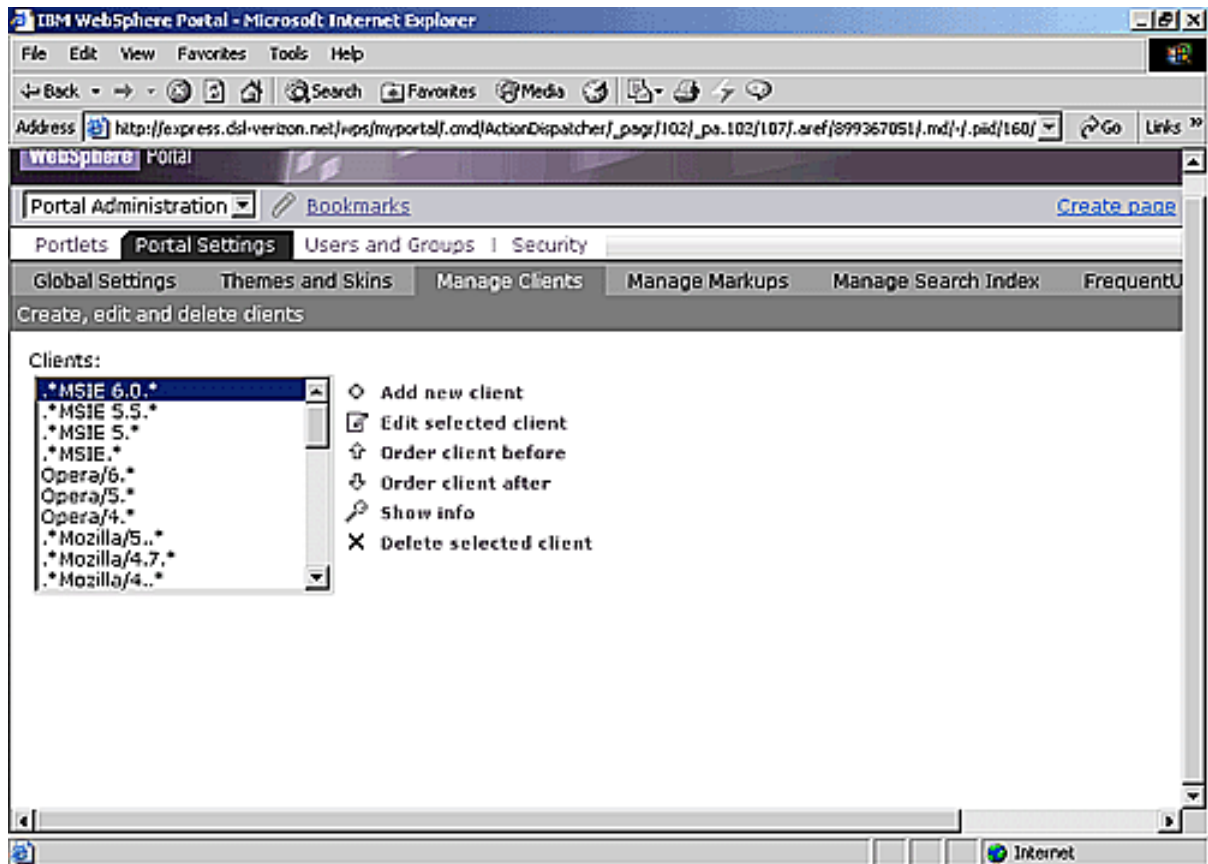


Changing the theme of a group of pages can completely change its look.



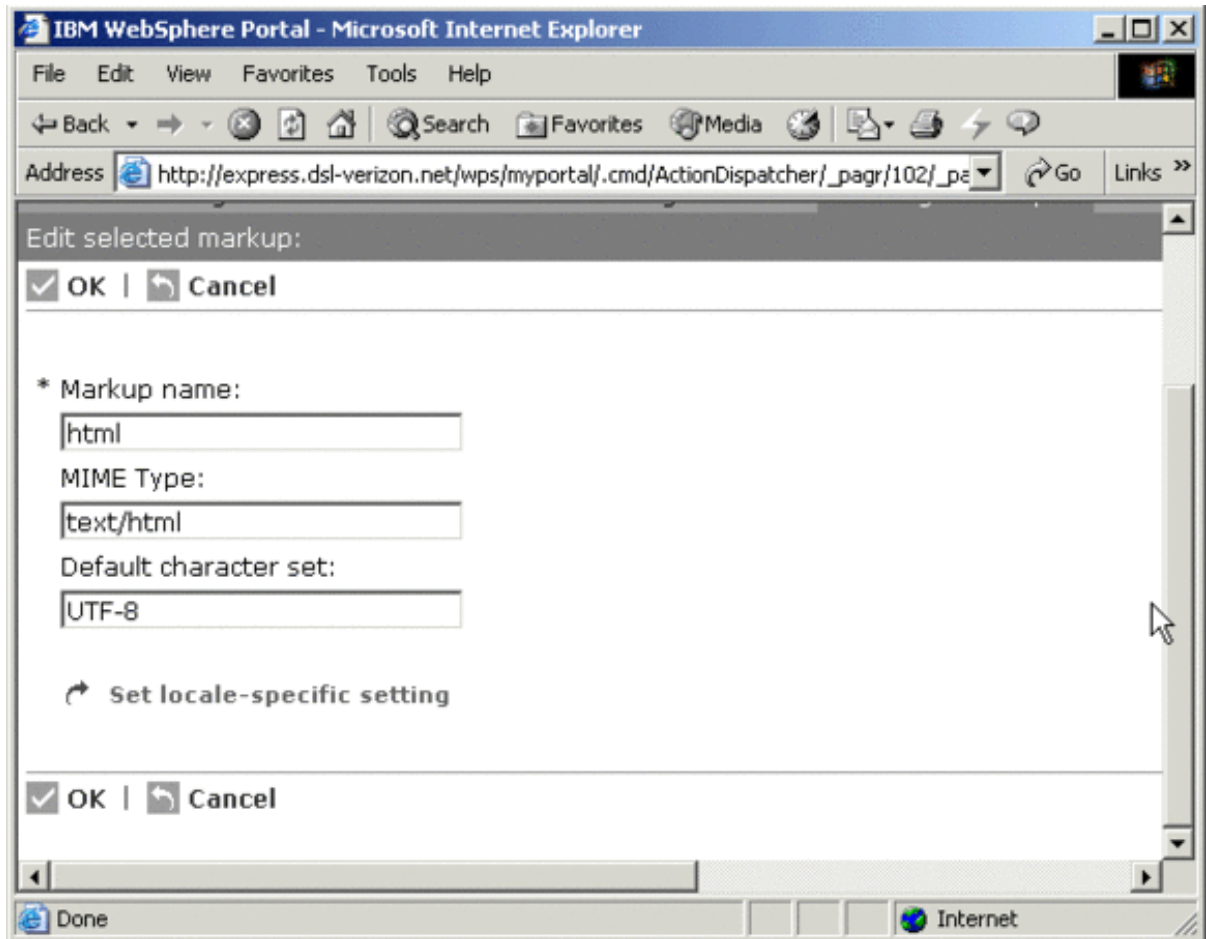
Manage clients

The portal can automatically deliver content in the right markup language, as long as it recognizes the browser, or user-agent, that's requesting the page. For example, any user-agent that contains the text `Nokia` should receive WML pages rather than HTML pages. The **Manage Clients** portlet enables you to add or edit clients for new markup languages or to reorder existing clients. The portal starts at the top of the list, and when it finds a match in the user-agent, it stops looking.



Manage Markups

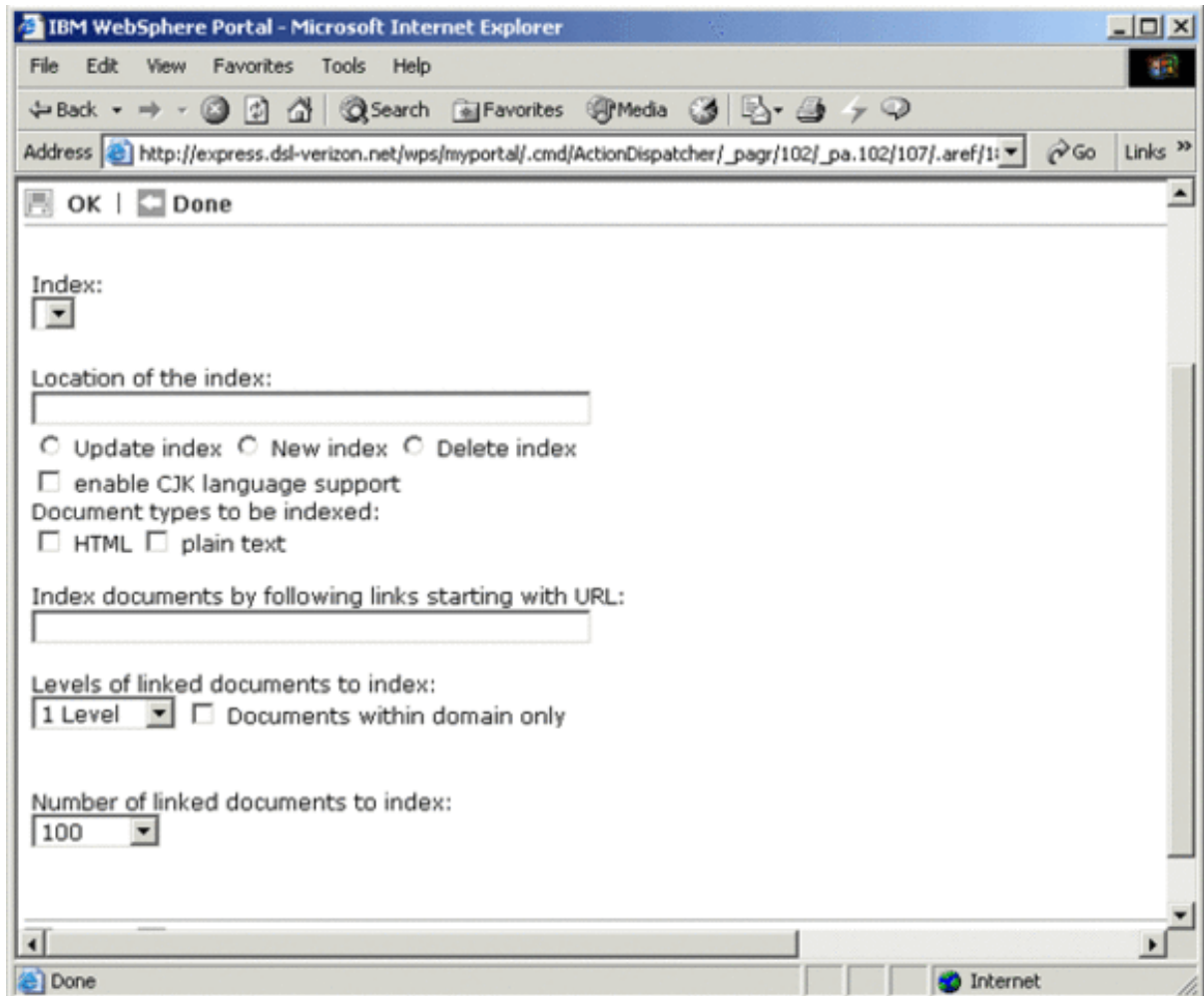
The actual translation between markup languages is handled by a piece of software called the Transcoding Server, but in order for it to work properly, the portal has to know how to deal with each one. Specifically, the **Manage Markups** portlet enables you to set the MIME-type and default character set for each markup, so the server can send back a response that the browser understands.



Manage search index

WebSphere Portal enables you to easily search data residing on the site. For example, MetroSphere users will be able to search for information on a particular technical topic of interest. In order to make that happen, we'll need to create a searchable index through the **Manage Search Index** portlet.

A fresh install doesn't have any search indexes defined, but once you create one, you can define the documents that should be indexed based on the start of their URL and how deeply linked they are from the starting page.

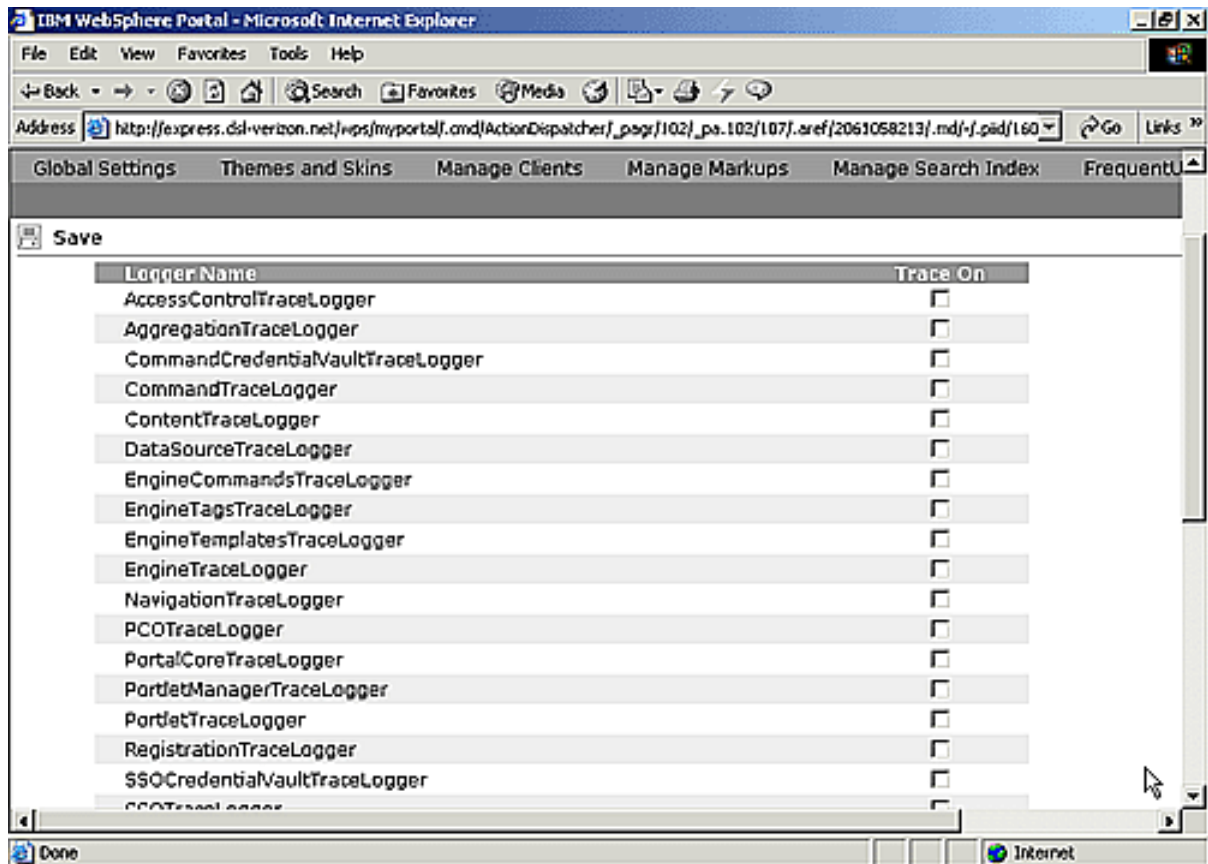


Frequent Users

The **Frequent Users** tab shows you the number of unique visitors to the site, so you have some idea how many seats you need to buy in order to be properly licensed.

Enable Tracing

The final entry for this section is Enable Tracing, which lets you set tracing for very specific functions of the portal. This function can be useful for developers trying to track down a problem with their portlets.



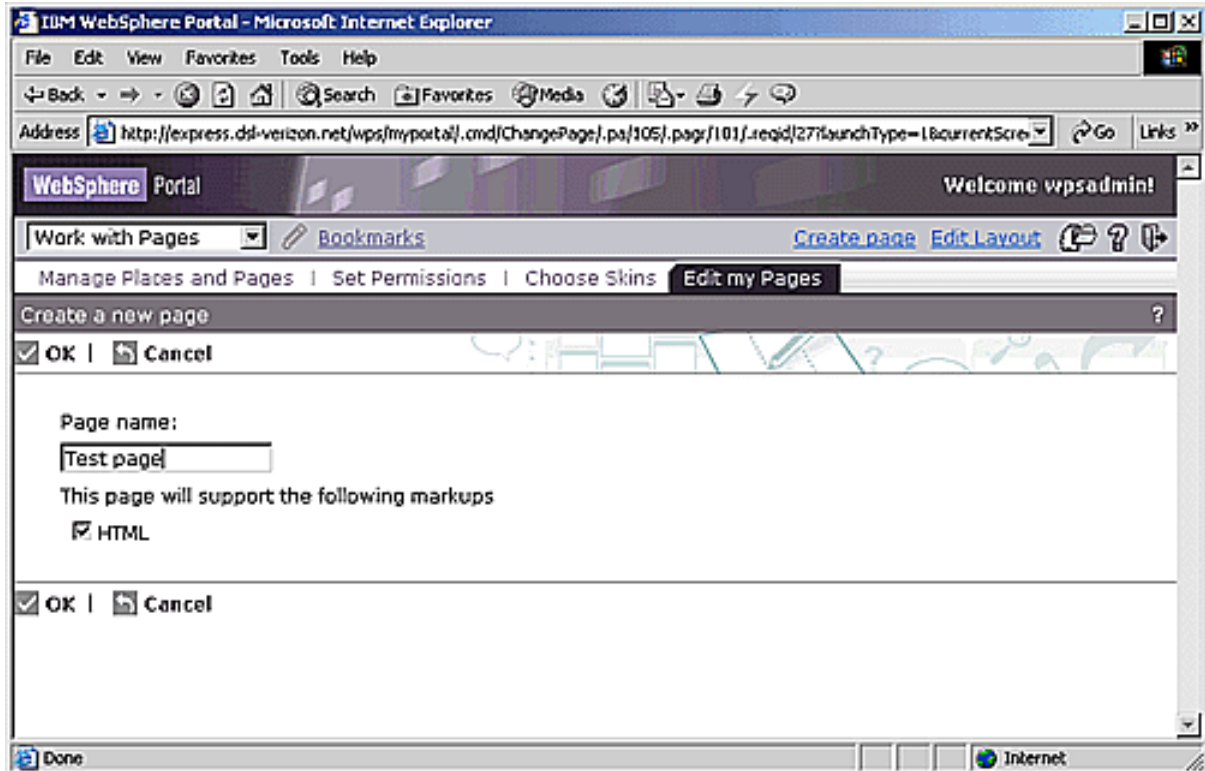
Now that you've seen all of the portal aspects that can be managed, let's create a new page.

Section 6. Creating a new page

Name the page

The name of the game here is flexibility, and WebSphere Portal makes it easy to add new pages and new content. It's important to note that both users and administrators can create new pages, and the user who creates a new page has full control over it. This section will look at the basics of creating a new page and adding portlets to it. (For more information on the Page Customizer, check the [Resources](#). The next piece in this series will be a tutorial detailing its use.)

To start, make sure you're logged in and on the home page. Then click the **Create page** link at the top of the page.

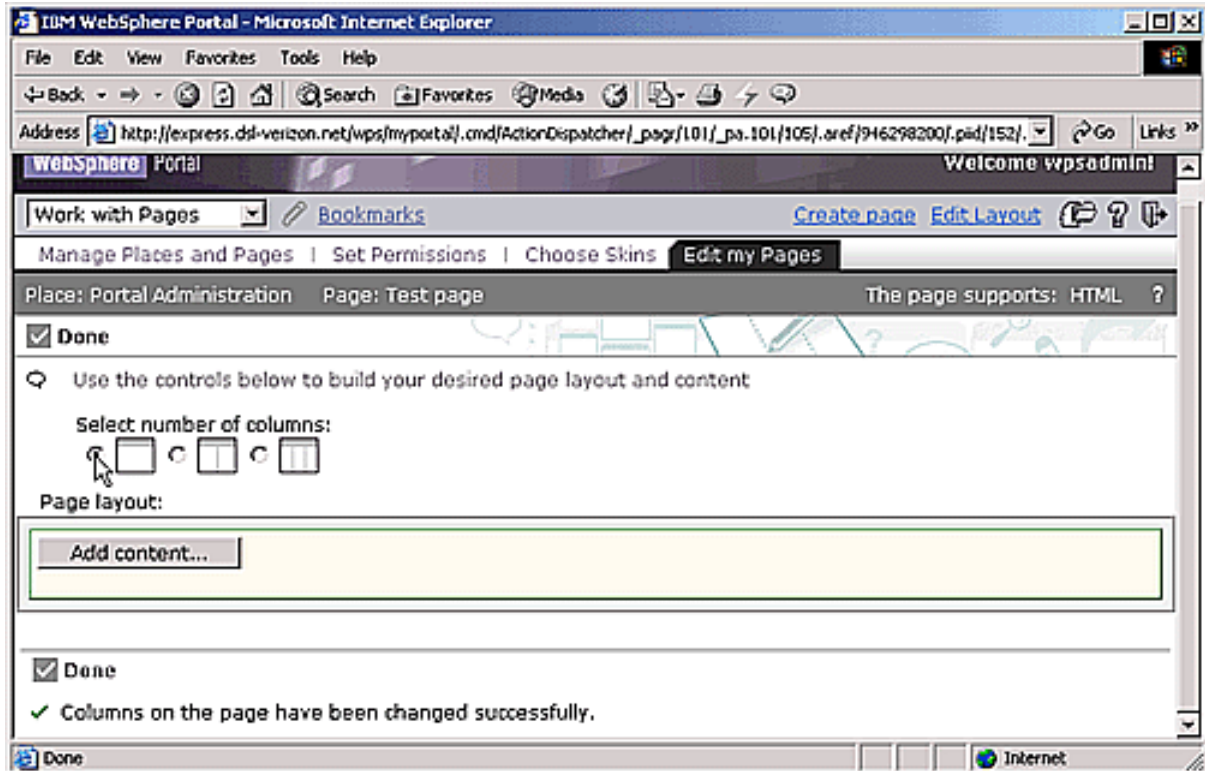


Each page has an identifying name, which must be unique within a group of pages. Give the page a name and click **OK**.

Choose the number of columns

Adding content to the page doesn't require any HTML knowledge. Instead, you create a page using the Web-based form. To start, choose the number of columns for the page. (An administrator can control the layout of the page more closely, using the **Edit Layout and Content** portlet, but the general **Edit Layout** portlet enables the user to choose a one-, two-, or three-column layout.

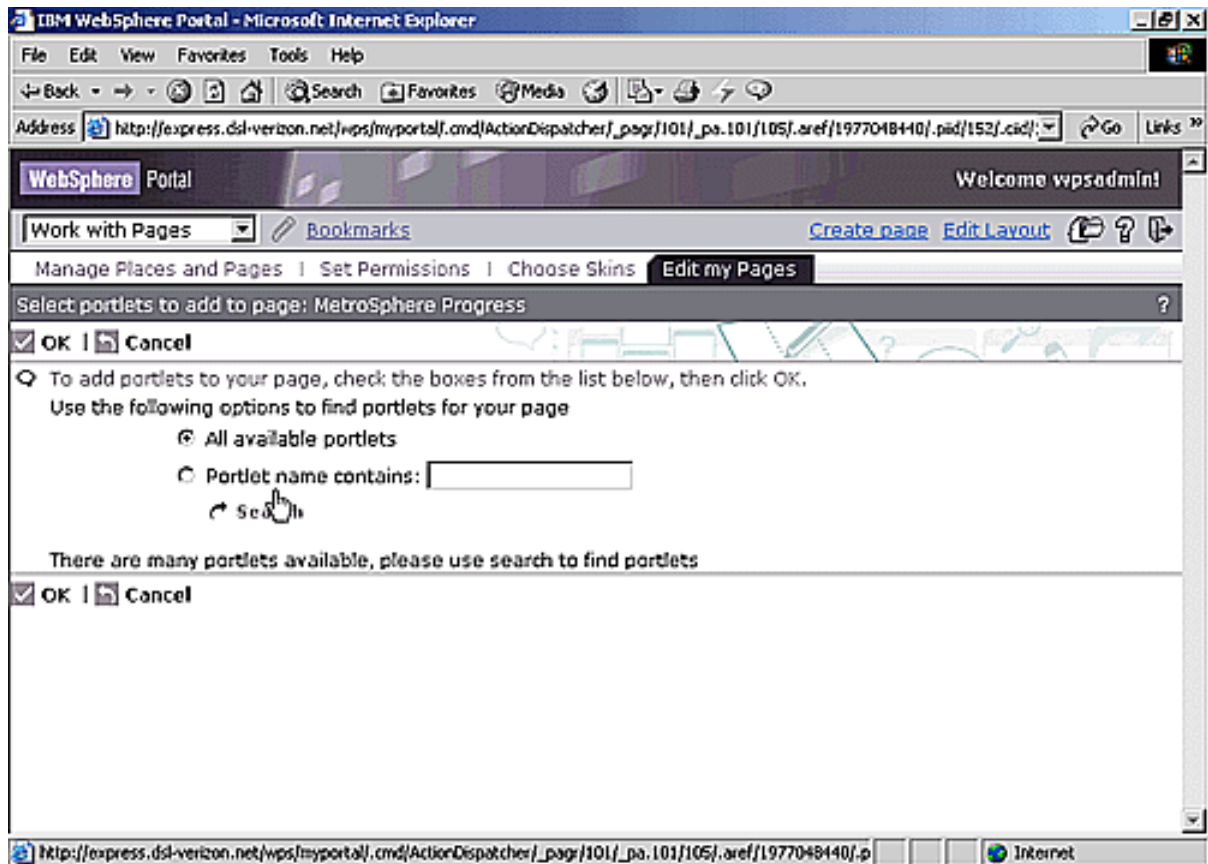
In order to keep this test page simple, use just one column.



To add a portlet to the page, click the **Add Content** button.

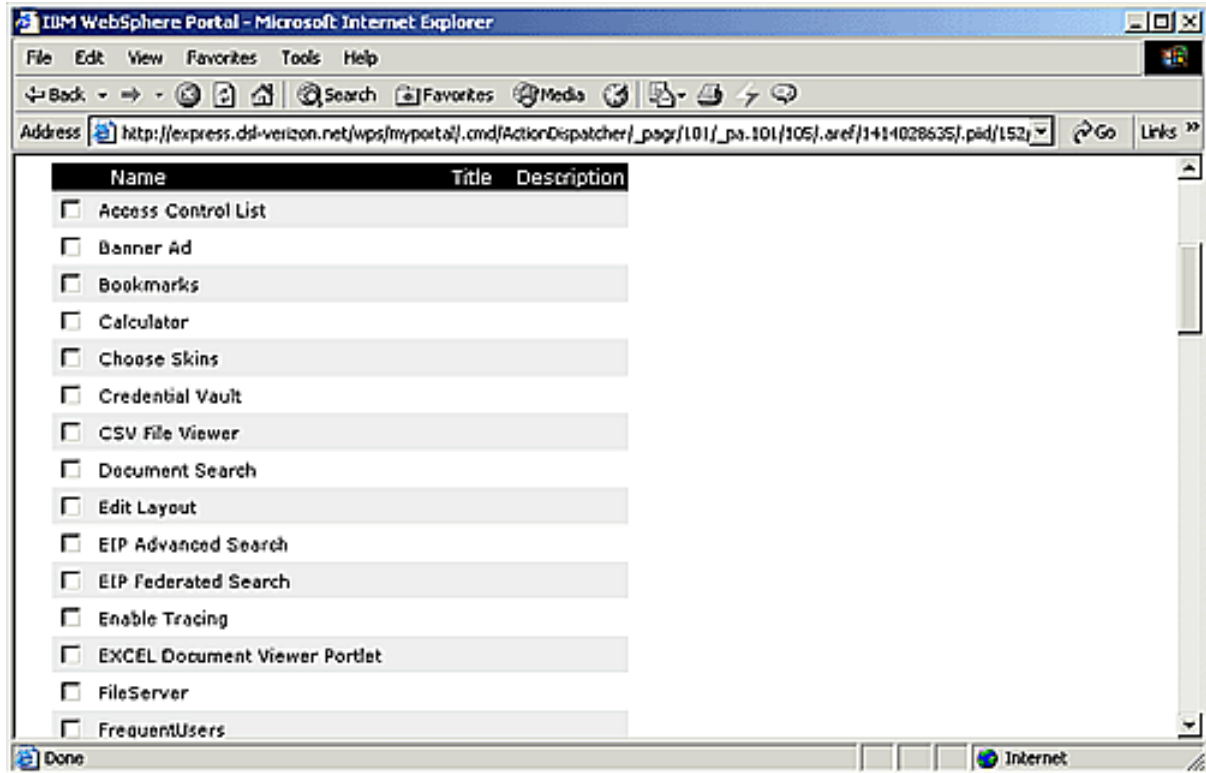
Search for portlets

The first step in adding a portlet to the page is to do a search of the available portlets. Leave **All available portlets** checked and click **Search** to see all of the potential choices.



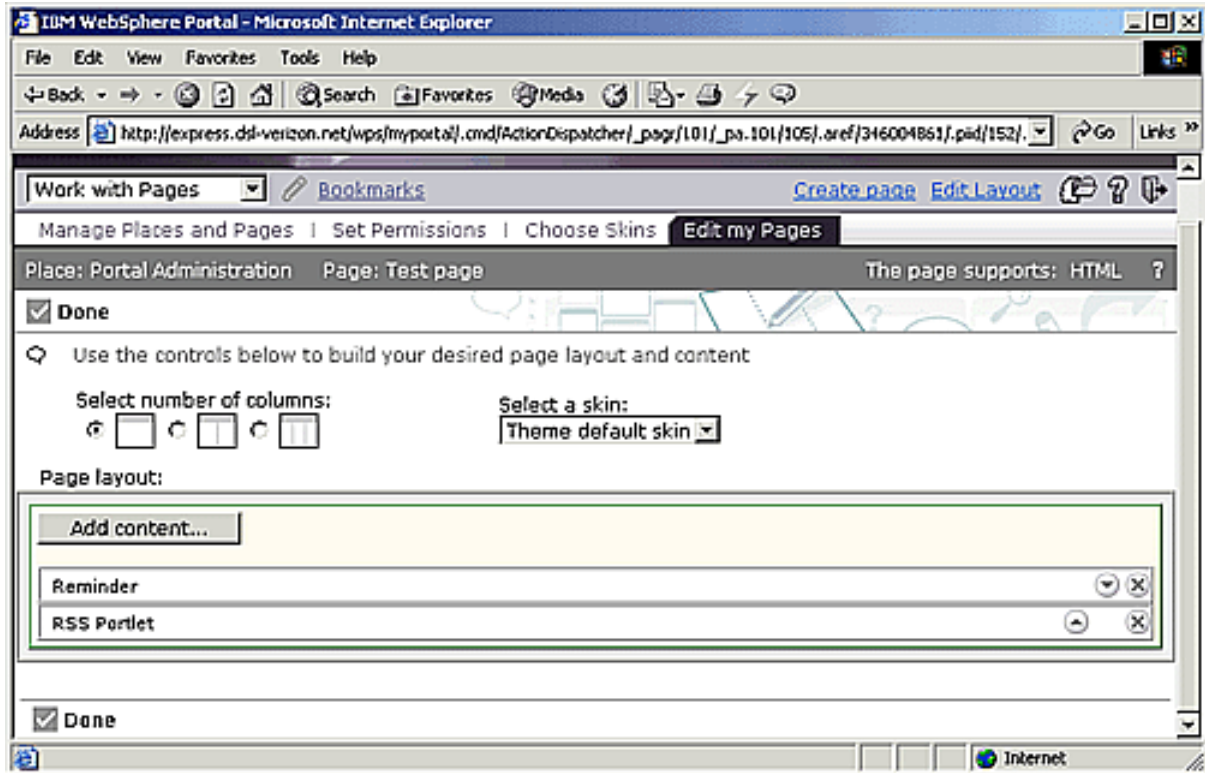
Choose portlets

Because we're logged in as an administrator, we can see not only the portlets that are intended for users, but also those that are used to administer the portal. Click the checkboxes next to the **Reminder** and **RSS Portlet** portlets and click **OK**.



Finalize layout

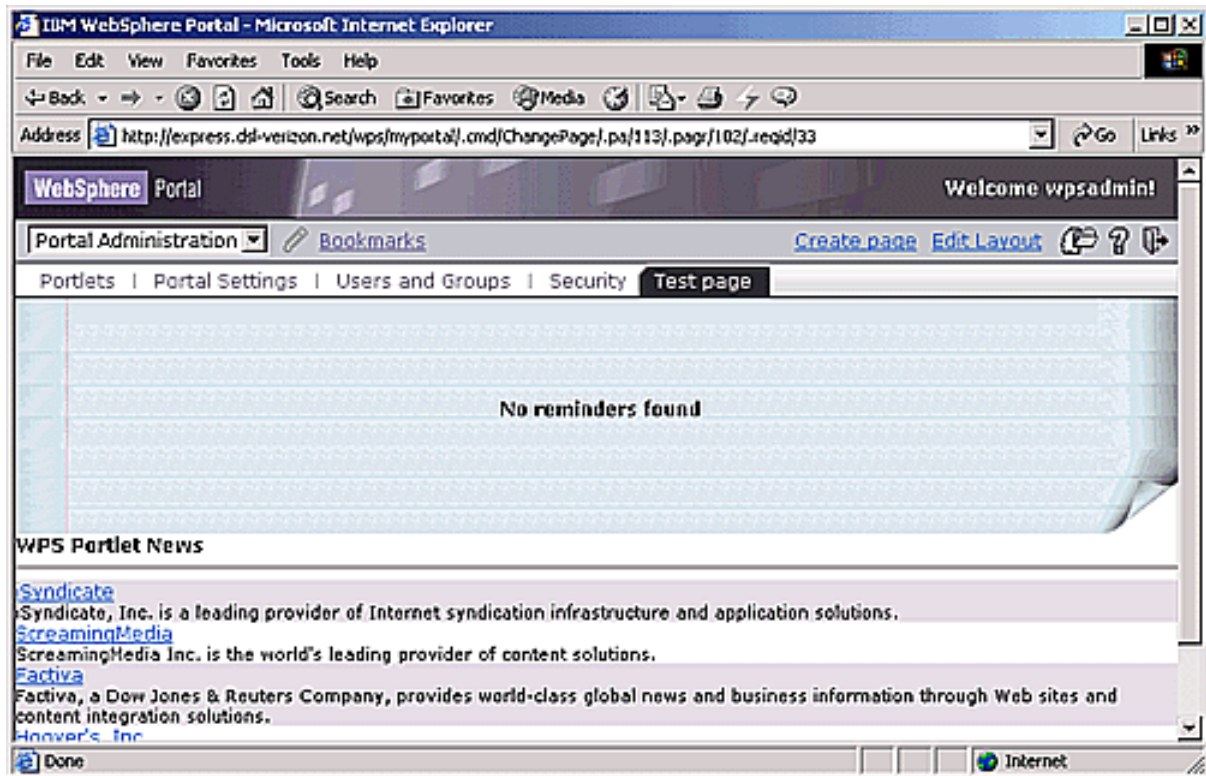
Once you've added the portlets to the page, use the arrows to move them around. In a single column, of course, you can only move them up and down in relation to each other. With multiple columns, you can move portlets from one column to another. You can also change the skin used for all of the portlets on the page.



When you're satisfied with the layout, click **Done**.

The final result

The final page shows the portlets you added to the page. If you're not happy with the layout, you can choose the **Edit Layout** link to make further changes.



Section 7. Using the generic portlets

What are generic portlets?

One way that WebSphere Portal makes it easy to add content to a page is by providing a number of *generic* portlets. These portlets let you quickly add static or dynamic content to the page without having to learn the Portlet API and build a new portlet.

The first type of generic portlet is the viewer. Portal - Express includes viewers for six different types of files:

- Comma-Separate Value (CSV) files
- Microsoft Excel
- PowerPoint
- Word documents

- PDF documents
- Rich Text Format (RTF) files.

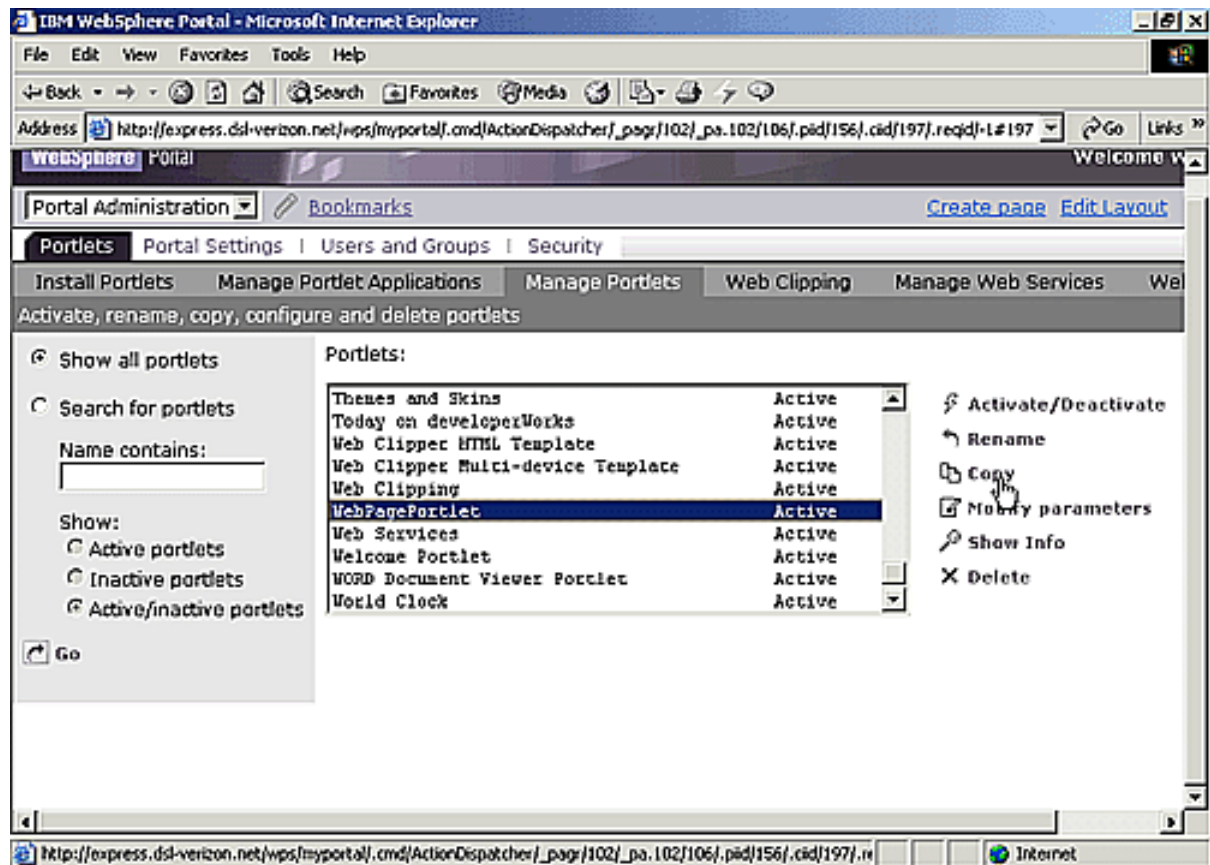
These portlets take the location of a static file as a parameter, and display the contents of the file.

The second type of generic portlet is a bit more complicated. These portlets provide a way to easily access dynamic contents such as servlets and JSP pages. These portlets include ServletInvoker, FileServer, JSPServer, SQL Portlet, and WebPagePortlet.

For the Express project, we'll create an instance of the WebPagePortlet that displays an HTML page with the current Message of the Day.

Creating a new portlet instance

To start the process, go to Portal Administration and click **Portlets -> Manage Portlets**. Highlight the **WebPagePortlet** and click **Copy**.



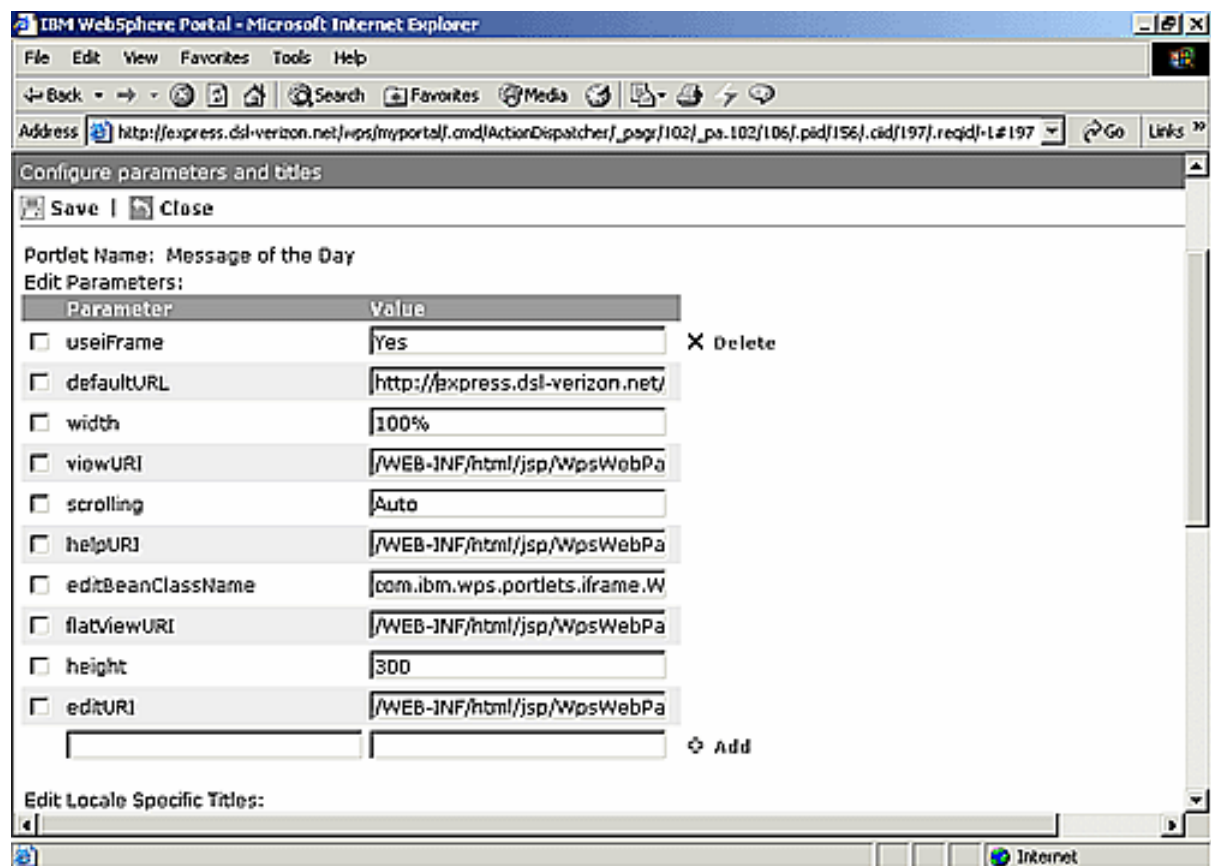
Fill in the dialog box with the name for the new portlet instance. Name the portlet

Message of the Day. After the copy's been made, highlight it and click **Modify parameters.**

Modifying portlet parameters

Every portlet can have parameters associated with it, but only some of them will be available to the user. However, as an administrator in the **Manage Portlets** section, you have access to all of the parameters. The changes that you make here will be the defaults for all users who add an instance of this portlet to their page.

In this case, the relevant parameter is the **defaultURL**, which defines the page that this servlet will retrieve and show. Note that the **WebPageServlet** won't work with a `file:///` URL; use an `http://` address instead.



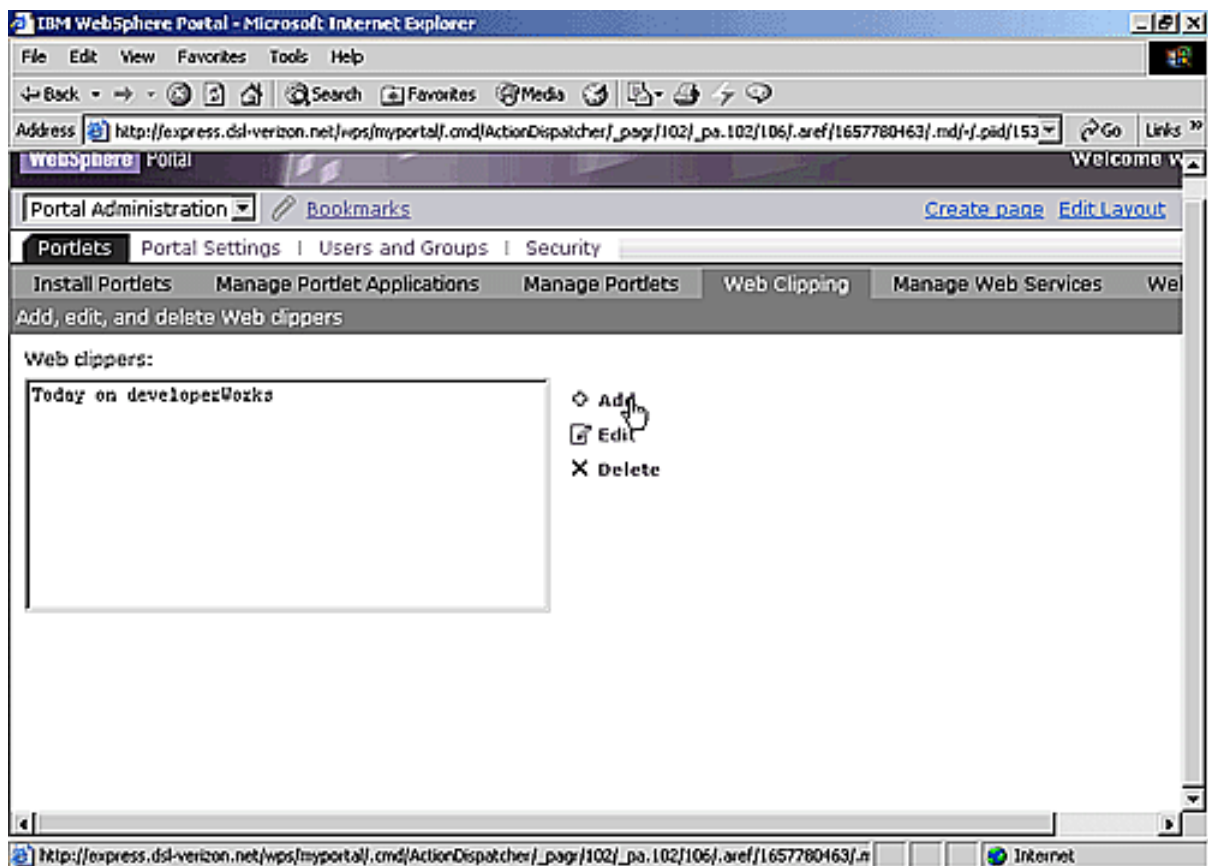
Once the new portlet exists, it can be added to any page just as the **Reminder** and **RSS Portlets** were added in the Add portlets step.

Section 8. The Web Clipper portlet

Create a new Web Clipper

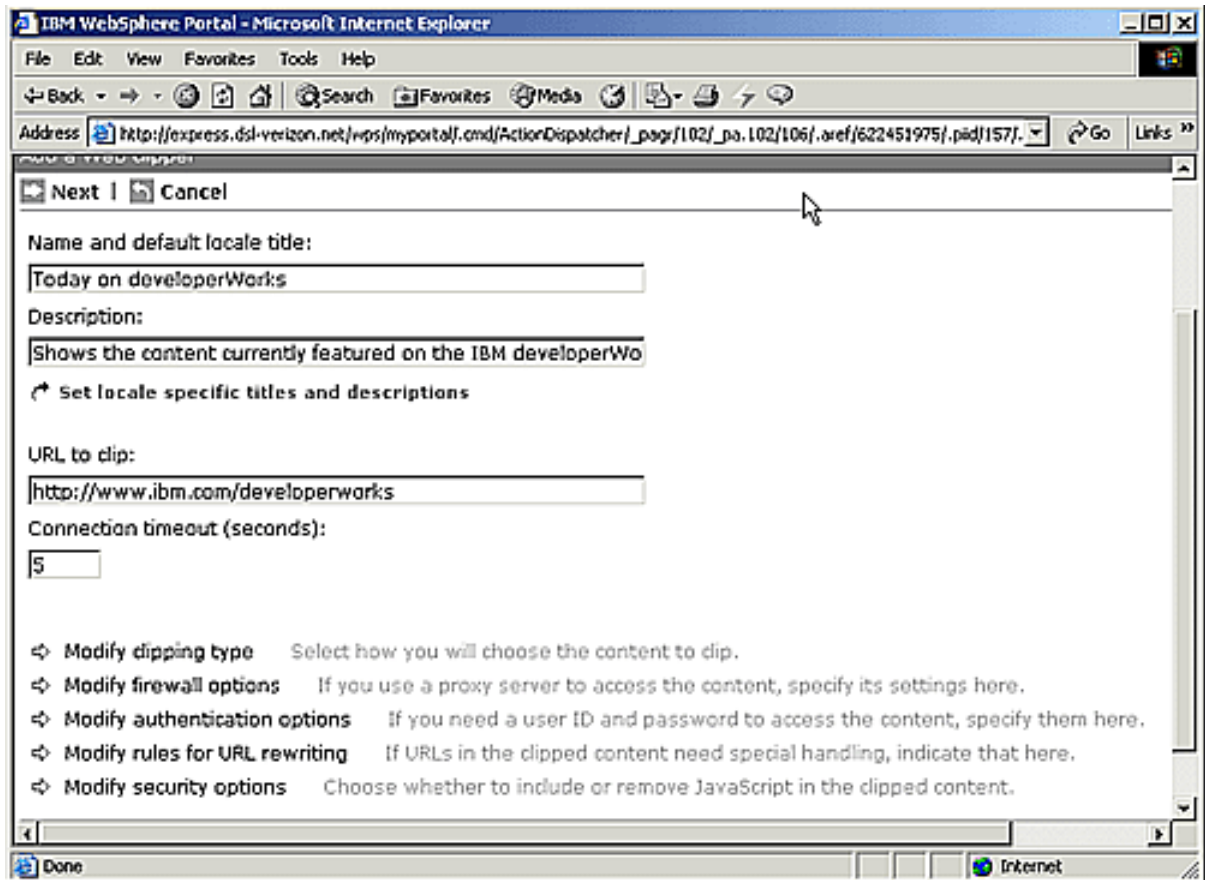
We touched briefly on cliplets in [Web clipping](#), but now you will see how to create a cliplet that can be added to a page. In this case, you're going to create a cliplet that shows the content featured on the home page for developerWorks.

To start, go to the **Portlets** section of **Portlet Administration** and choose **Web Clipping**. Click the **Add** link.



Specify the page

Next name the cliplet. This is the name that will appear in the listing of available portlets. The description appears as a *tool tip* when the cliplet has been added to a page layout and the mouse hovers over it. Next add the URL of the main page that contains the information that will become part of the cliplet.

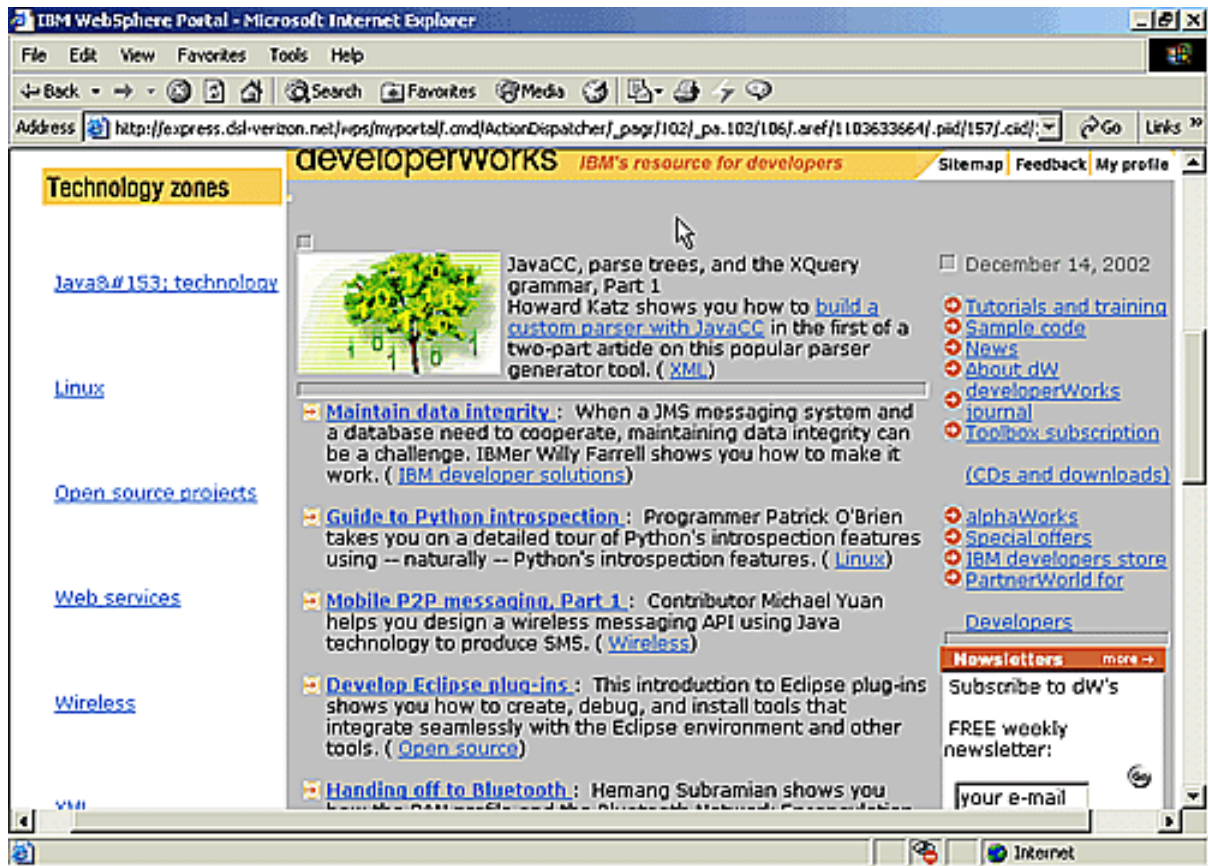


Click **Next** to move on.

Choose content

Next choose the content that should appear on the page. The portlet recognizes the content to display based on the structure of the original HTML page, so when the content changes (for instance, to add new articles) the cliplet should still recognize the information to display.

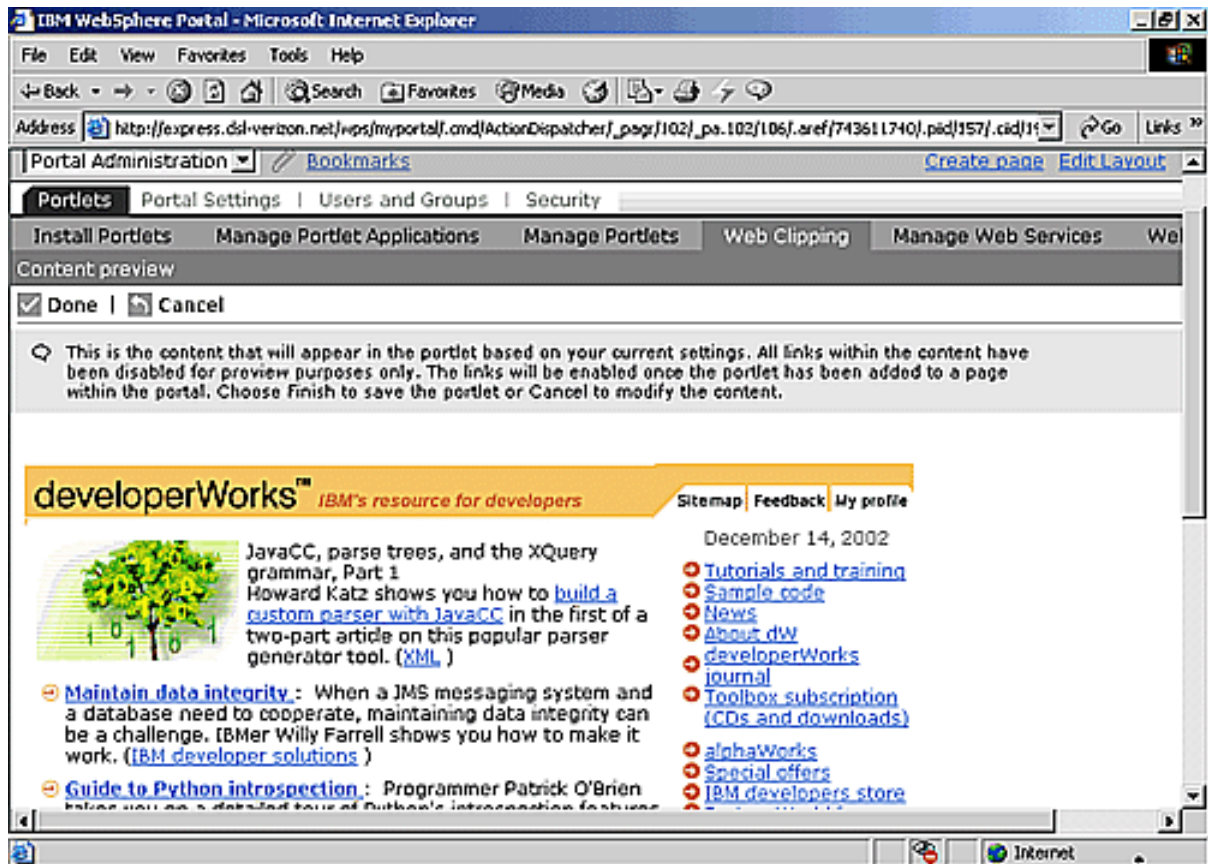
Click the content that should appear in the final cliplet. In this case, the table data cell that contains the main information for the page is selected.



Click **Next** to continue.

Preview the content

When you're satisfied with your choices, you can preview the final cliplet to make sure it appears the way you want it to. To go back and make more choices, click **Cancel**. Otherwise, to move on, click **Done**.



Like the **Message of the Day**, the **Today on developerWorks** cliplet can be added to a user's page.

Section 9. Summary

Recapping what you learned

In getting familiar with WebSphere Portal - Express, this tutorial covered the following topics:

- General user concepts: What can a user do with a WebSphere Portal?
- Managing users and groups: How does security work?
- Managing portlets: What are they, and how does the system see them?
- Portal settings: What can an administrator control about the site?

- Creating a new page: How can a user or administrator add and control content?
- Using the generic portlets: Is there an easy way to add standard content such as Web pages and servlets?
- Creating a cliplet: What is a cliplet, and how can an administrator make one?

Resources

- [Participate in the discussion forum for this content.](#)
- IBM's WebSphere Portal - Express involves a number of different topics. For more information, see some of the resources below.
 - Get up to speed on the MetroSphere project by checking out the [Series overview](#), which lists all the articles in this series.
 - Get more information about [WebSphere Portal](#) and check out [developer resources](#).
 - Find more information on the at the [WebSphere Portal - Express](#) information center.
 - Read the article [Using Credential Vault to Provide Single Sign-on for Portlets](#) for more information about the Credential Vault.
 - Get a taste of portal development by reading "[Develop portlets that use Web services to obtain data from remote systems](#)" (*developerWorks*, January 2003).
 - Learn about [writing portlets](#) to expand the capabilities of your portal server.

About the author

Nicholas Chase

Nicholas Chase has been involved in Web-site development for companies such as Lucent Technologies, Sun Microsystems, Oracle, and the Tampa Bay Buccaneers. He has been a high school physics teacher, a low-level radioactive waste facility manager, an online science fiction magazine editor, a multimedia engineer, an Oracle instructor, and the chief technology officer of an interactive communications company. He is the author of several books, including *XML Primer Plus* (Sams).