

Microsoft®
Exchange 2000
Server

**Customizing Microsoft® Outlook® Web
Access**

White Paper

Table of Contents

Introduction	1
Important Note About Customizing Outlook Web Access.....	1
Microsoft Exchange 2000 Features	1
Introducing the Exchange Store	2
Outlook Web Access.....	3
Architecture.....	4
Accessing a Server.....	5
WebDAV	6
Server Configuration	7
Outlook Web Access User Management	7
Virtual Servers and Directories	8
Multilanguage Support.....	10
Security	11
Customization Overview	13
Outlook Web Access Components.....	13
Outlook Web Access Folder Structure.....	14
Virtual IIS Directories.....	14
Browser Compatibility	14
Using URLs to Access Data.....	15
Folder URL Access.....	16
URL Parameters.....	16
Global Address List Command Parameter	23
Default Folders	24
Hosting Within Frames	25
Standard Frame Layouts.....	25
Browser Compatibility with IFrame	29
Navigation Bar	30
Outlook Web Access Navigation Bar Components.....	31
Shortcut Folder URLs.....	32
HTML Components	32
Changing Navigation Bar Functionality	34
Modification of the Navigation Bar	34
MenuBar Toolbar	47
Hide Folder Toolbars	47

Global Form Registration	55
Outlook Web Access Built-in Forms.....	55
Outlook Web Access Default Registrations	55
Forms Registry	56
Registering New Global Forms	56
Customized Form Header and Footers.....	61
Style Sheets	61
Outlook Web Access Style Sheets.....	61
Customization	62
Icon Customization	63
Default Outlook Web Access Views	64
Modifying the Default Views	64
Address Book Lookup View	71
Customization	71
Logoff Customization	71
Options Page Customization.....	74
Options POST.....	74
Changing the Options Page	76
Disabling the Multimedia Button on the Options Page.....	76
Help Files	77
Customization	78
Other Outlook Web Access Customizations	78
Hiding Folders	78
Removing Folders Completely	80
Removing Public Folders Access Completely.....	80
Changing the Default Content Expiration Time	81
Summary	82
Appendix A: Additional Resources	84
Web Sites and White Papers.....	84
Microsoft Knowledge Base Articles.....	84
Appendix B: Code for the Navigation Bar	84
Non-Internet Explorer 5 Browsers (United States English).....	84
Internet Explore 5 or Later Browsers (United States English).....	87
Appendix C: MailFolder Toolbar Code.....	92
Internet Explorer 5 or Later (United States English).....	92

Appendix D: Outlook Web Access Customization Example	94
Appendix E: Style Sheet Sections	95

Customizing Microsoft Outlook Web Access

White Paper

Published November 2001

For the latest information, please see <http://www.microsoft.com/exchange/>.

Introduction

Microsoft® Exchange 2000 Server extends back-end server and client applications to support the latest advances in Web technology. Microsoft Outlook® Web Access is a tightly integrated component of Exchange 2000 and an updated version of Outlook Web Access first introduced in Exchange Server version 5.0. Enhancements to the Exchange 2000 architecture significantly increase Outlook Web Access scalability and functionality.

This white paper provides an overview of Exchange 2000, the Outlook Web Access architecture, and the individual components that make up Outlook Web Access. This paper also provides developers with the processes to customize and extend Outlook Web Access for their own solutions.

Important Note About Customizing Outlook Web Access

This white paper describes changes developers can make to, among other objects, Outlook Web Access source .htc, .js, and .xsl files that exist in the Exchweb folder. Be aware that any changes you make to these objects will be undone or fail with the installation of service packs. Also, Microsoft Developer Support cannot support customization to these Outlook Web Access objects.

Microsoft Exchange 2000 Features

Exchange 2000 Server delivers next-generation messaging and collaboration for businesses of all sizes through:

- An enhanced platform for messaging and collaboration.
- Exchange store collaboration and applications.
- Anytime, anywhere communication.

Exchange 2000 takes full advantage of the capabilities of Windows® 2000 and is the first application to offer unified management of all messaging, collaboration, and network resources. Exchange also provides the scalability required to meet the changing needs of your enterprise. With the Exchange store (previously known as the Web Storage System) at its core, Exchange serves as the single place to create, store, access, and share all of the information your company possesses. Built-in

support for Web-standard protocols, including Extensible Markup Language (XML) and HTTP, also makes Exchange the ideal hosting platform for a new generation of applications that are transforming the workplace by automating many essential day-to-day business tasks. With Outlook Web Access, employees can have real-time access everywhere. By incorporating a wide range of emerging digital technologies, Exchange provides a comprehensive communication platform that enables everyone in your organization to stay in touch, using the broadest array of devices.

Introducing the Exchange Store

Microsoft Exchange 2000 introduces the Exchange store, a breakthrough in allowing developers to leverage collaborative technologies in their applications. The Exchange store:

- Provides collaborative services and programmability right after installation so that existing applications can be enhanced with collaboration or new collaborative applications can be built.
- Provides unmatched client and developer access through its support of a multitude of access protocols and object models.
- Allows developers to use their existing development skills and tools to build collaborative applications.

The Exchange store provides all of these capabilities through Web-based standards, such as HTML and XML, as well as industry standards such as ActiveX[®] Data Objects (ADO) and OLE DB. Figure 1 illustrates the many ways that you can access the Exchange store.

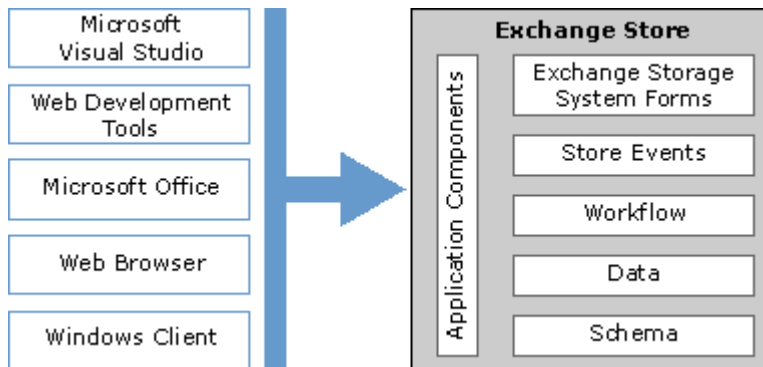


Figure 1 Different clients or development tools that can access the Exchange store

Accessing Data in Exchange 2000

Exchange 2000 introduces a number of new data access methods for retrieving and setting data. To quickly add or query data stored in Exchange, select the access method that best suits your application and skills. The different ways to access data in Exchange 2000 allow you to leverage your development tools and skills. Access methods include:

- Collaboration Data Objects (CDO) for Exchange 2000

- ADO 2.5
- OLE DB 2.5
- HTTP/XML
- MAPI

Outlook Web Access

Outlook Web Access is significantly enhanced in Exchange 2000. Used by remote, home, and roving users, Outlook Web Access provides Web browser access to e-mail, scheduling (including group scheduling), contacts, and collaborative information stored in Microsoft Exchange Storage System folders. When used with Microsoft Internet Explorer 5 or later, Outlook Web Access provides the most functionality because those browsers support dynamic HTML (DHTML), HTML components (HTC), and Extensible Stylesheet Language (XSL) that operate on XML data retrieved from the server. However, Outlook Web Access can be used with any browser that supports frames and scripting. Specific areas of improvement in Exchange 2000 include:

- A look and feel that is more like Microsoft Outlook. When used with Internet Explorer 5 or later, users also get drag-and-drop capability, pop-up menus, toolbars, true hierarchy control, and rich-text editing.
- Non-script based rendering of all core HTML, resulting in a significant increase in server performance.
- Rendering embedded messages in the body of an e-mail message.
- "In-frame" or single browser window navigation for HTML 3.2 clients.
- Embedded OLE object rendering.
- Name resolution against personal contacts as well as Active Directory™ directory services.
- More options available through user interface, such as out-of-office settings.
- Improved calendaring, including read-only delegate access functionality.

Figure 2 illustrates Outlook Web Access in Exchange 2000.

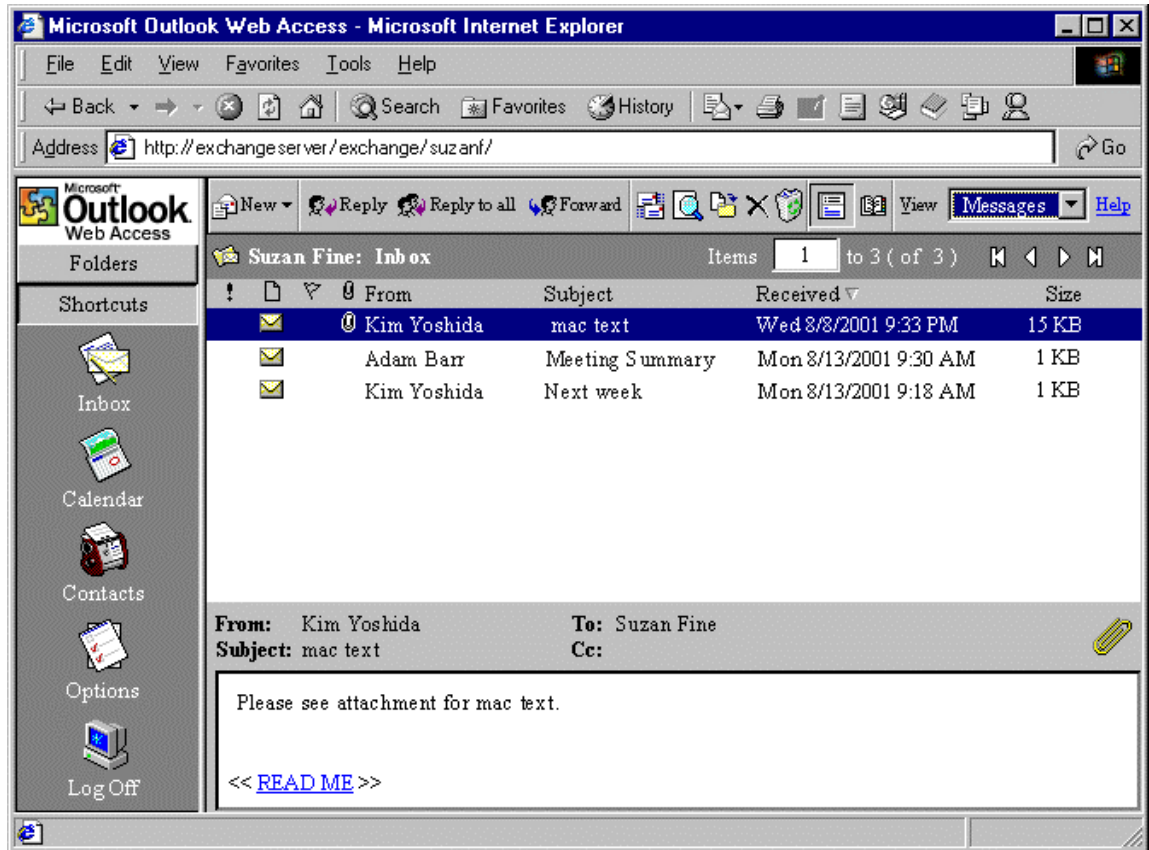


Figure 2 Outlook Web Access

Architecture

Outlook Web Access in Exchange 2000 is substantially different from the version introduced in Exchange 5.0. Outlook Web Access in Exchange 5.5 used Active Server Pages (ASP) to communicate with the server running Exchange. The ASP application used CDO version 1.2 and MAPI. The effective number of users for each Exchange 5.5 server was limited by the overhead needed to support interpreted scripts in ASP and to maintain MAPI sessions within ASP. In this context, Outlook Web Access could be deployed on a computer running Microsoft Internet Information Services (IIS), separate from a server running Exchange 5.5,.

Outlook Web Access in Exchange 2000 does not use MAPI to communicate with the mailbox store and no longer uses ASP or interpreted scripts for client access. Client access continues to use HTTP; however, Outlook Web Access is now built into the Exchange store and uses IIS to receive requests and send responses. Therefore, the Exchange store is tightly integrated with IIS.

Figure 3 illustrates the enhanced architecture of Outlook Web Access in Exchange 2000.

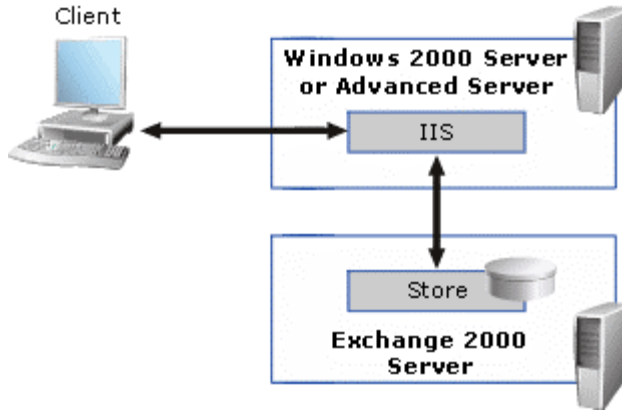


Figure 3 Basic architecture of Outlook Web Access

IIS, which is integrated into Microsoft Windows 2000, handles incoming HTTP requests from Web browsers and sends HTTP responses from Exchange 2000 Server or Outlook Web Access. IIS receives a client request, looks at the namespace or path specified in the request, and passes the appropriate information for the context of the URL back to the Web browser. If the server contains the Exchange 2000 database, Outlook Web Access uses a high-speed channel to access the mailbox store. If the server is a front-end server, the Exchange front-end proxy uses HTTP to forward the request to a back-end server.

Accessing a Server

Clients direct requests to Outlook Web Access using specific URLs, such as `http://owa.contoso.com/exchange`. Often the URL directs the client to the user's mailbox. However, URLs are not limited to addressing a mailbox. You can address most functions and components of the client by defining a specific URL.

You can open a specific folder by typing the name of the folder after the mailbox name. For example, to open a calendar, type the path to the user's mailbox followed by `/calendar`, as in `http://owa.contoso.com/exchange/username/calendar`. Likewise, you can access the Contacts folder directly by typing the path to the user's mailbox followed by `/contacts`. The language used for folder names is determined the first time a user logs on. If an English language client is used, the folders are Inbox, Calendar, Contacts, Drafts, and so on. If a Spanish language client is used, the folders are *Bandeja de Entrada*, *Calendario*, *Contactos*, *Borradores*, and so on.

There are two ways for a user to connect to a mailbox with Outlook Web Access:

- Implicit logon** Implicit logon is provided as a shortcut to logging on. You can only use implicit logon the first time you access a mailbox in a session, and only when you omit the user name from the first request (for example, `http://server/exchange/` or `http://server.contoso.com/exchange/`). For implicit logon to work through a front-end server, you must enable authentication. Implicit logon does not require configuration on a back-end server. After the user is authenticated, the authentication information is used to look up the mailbox associated with the user in Active Directory and to look up the back-end server on which the mailbox is located. The URL is then updated with the user name and sent to the correct back-end server. Implicit logon is only useful with

Outlook Web Access; specialized HTTP clients generally do not use this technique.

- **Explicit logon** Explicit logon includes the user name in the URL (for example, `http://server/exchange/username/` or `http://server.contoso.com/exchange/username/`). You must use explicit logon when the front-end server is not configured to authenticate users or when a user attempts to access a mailbox that is not his or her own (but to which he or she has access). With explicit logon, when the front-end server receives a request from a client, the user name is extracted from the URL. The user name is then combined with the Simple Mail Transfer Protocol (SMTP) domain name associated with the virtual directory or virtual server (this combination is made when the virtual directory or virtual server is created or when the default recipient policy's primary SMTP address is modified). This combination constructs a fully qualified SMTP address, which is a searchable property of a mail-enabled user object. The front-end server then looks up this address in Active Directory and determines which back-end server has the mailbox associated with the address. The request is then forwarded to that back-end server, which processes the request as if it came directly from the client. The response the request generates is returned to the front-end server, which returns the URL path, unchanged, to the client.

URLs are not limited to only accessing folders. You can open any item and perform many functions by using explicit URL addressing. Leveraging the basic functions of named URLs can provide fast integration of Exchange with corporate intranet sites. For more information, see the "[Using URLs to Access Data](#)" section later in this document.

WebDAV

Internet Explorer 5, Microsoft Office 2000, and Windows 2000 clients support an extension to the HTTP/1.1 protocol known as Web-based Distributed Authoring and Versioning (WebDAV). WebDAV allows richer manipulation of data on the server than does HTTP/1.1 alone, allowing the client to work with data in a more intuitive manner. WebDAV can accommodate all types of content, so you can use it with many types of documents. You can also use WebDAV to create anything you store in a file. WebDAV includes the following features:

- **Overwrite protection (file locking)** WebDAV allows users to write, edit, and save shared documents without overwriting another person's work, regardless of which software program or Internet service they use.
- **Namespace management** Namespace management allows users to manage Internet files and directories; for example, users can move and copy files and directories. The process is similar to the way users manage word-processing files and directories on a local computer.
- **Property (metadata) access** The WebDAV properties feature is an efficient means of storing and retrieving metadata. Metadata is information about a Web document such as the author's name, the copyright, the publication date, and keywords that Internet search engines use to find and retrieve relevant documents.
- **Client side rendering** WebDAV uses XML to transfer the data from the server to the client, allowing Outlook Web Access to move the task of HTML rendering

from the server to the client. Moving the HTML rendering from the server to the client essentially distributes the processing and, as a result, increases the capacity of the server.

- **Backward compatibility** Exchange supports standard HTTP methods (GET, POST, PUT, DELETE, OPTIONS) and adds new methods for WebDAV-specific functionality. Because WebDAV is an extension of HTTP, existing methods are not changed.

Internet Engineering Task Force (IETF) Request for Comments (RFC) 2518 defines WebDAV. For more information about WebDAV, see the WebDAV Resources Web site at:

<http://www.webdav.org/>

Server Configuration

By default, Exchange 2000 is configured to allow access to users' mailboxes and the default public folder tree through HTTP. However, you can configure the server to provide customized access for Web-based clients. You can specify items such as:

- Which users can access the server from a Web browser.
- Which authentication methods to allow.
- Which public folders hierarchies are exposed to users.

To configure the server for Web-based clients, use the Exchange System Manager console and the Active Directory Users and Computers console in Microsoft Management Console (MMC). The changes you make are stored in Active Directory and then applied by the appropriate server running Exchange.

Note The virtual Web servers and directories that you create using System Manager also appear in the Internet Services Manager console. The configuration changes you make in System Manager are written to Active Directory and are replicated periodically to the IIS Metabase on the server running Exchange, thereby overwriting the changes made to similar items using Internet Services Manager. Only use Internet Services Manager to make changes to items or settings that are not available in the Exchange Administration tool.

Outlook Web Access User Management

As previously mentioned, Outlook Web Access for all users is enabled in Exchange 2000 by default. To enable or disable Outlook Web Access for a user, you must use the Active Directory Users and Computers MMC snap-in.

To enable or disable Outlook Web Access for a user

1. On the **Start** menu, point to **Programs**, point to **Microsoft Exchange**, and then click **Active Directory Users and Computers**.
2. On the **View** menu, click **Advanced Features**.
3. Click to expand the organization object, and then click **Users**.

4. In the details pane, right-click the user for whom you want to enable or disable Outlook Web Access, and then click **Properties**.
5. Click the **Exchange Advanced** tab, and then click **Protocol Settings**.
6. In **Protocols**, select **HTTP**, and then click **Settings**.
7. In **HTTP Protocol Details**, to enable Outlook Web Access for the user, select **Enable for mailbox**. To disable Outlook Web Access, click to clear **Enable for mailbox**.

Virtual Servers and Directories

Virtual servers allow you to create separate Web server instances for internal and external users, for different departments within a company, or for users with different security requirements.

To create an HTTP virtual server in Exchange System Manager

1. On the **Start** menu, point to **Programs**, point to **Microsoft Exchange**, and then click **System Manager**.
2. Double-click **Servers**.
3. Double-click the appropriate Exchange server.
4. Expand **Protocols**.
5. Right-click **HTTP**, point to **New**, and then click **HTTP Virtual Server**.

When you create an HTTP virtual server, there are three tabs to configure:

- **General** Use this tab to configure the virtual server identification (host header, IP address, port), number of available client connections, and logging. If one domain (for example, contoso.com) needs different configuration settings from a second domain (for example, adatum.com), you can add both domains to the same virtual server.
Note The combination of identification values for each virtual server (host header, IP address, TCP port) must be unique.
- **Access** Use this tab to configure the type of authentication used to access secure content and to configure whether the server allows Anonymous access to shared content.
- **Security** Use this tab to configure security settings for this Exchange Administration object in the Active Directory. You can limit which level of administrator may modify these settings. This tab does not affect client connections.

Connecting to a Virtual Server

A browser connects to a specific virtual server by specifying `http://IP Address:Port` or `http://host header name:Port`. You do not need to specify the port if it is the default TCP port 80 or TCP port 443 for HTTPS or SSL connections. The host header name must have at least one of the following:

- A registered host record in DNS.
- An entry in the client's host file.
- The same name as the server's computer name if the connection occurs on an intranet.

If the server is placed on the Internet, the domain name must be registered with an Internet naming authority.

When IIS receives an HTTP request, IIS extracts the virtual server name from the host header in the request to determine which virtual server receives the request. If the virtual server name matches the host header of a virtual server, the request is directed to that server. Otherwise, the default Web server handles the request.

Disabling Virtual Servers

With Outlook Web Access in Exchange 5.5, you could use the **General** tab to enable or disable all HTTP access for Exchange. Exchange 2000 has similar functionality. To stop, start, or pause each virtual server, right-click the virtual server object in System Manager and click the appropriate option.

Notes

- You can only administer the default Exchange virtual server from Internet Services Manager.
- If you stop the default Exchange virtual server, you also stop the IIS default Web server. If you want this Web server to be available, but you want to eliminate Exchange access, you can remove the Exchange, Exadmin, and public virtual directories. You can also configure security to disable access. Right-click the object, and then click **Delete** to remove the directories. Removing virtual directories effectively disables management of public folders on that server through HTTP.

For each virtual server, you can configure multiple virtual directories to point to different public folders or to the private mailbox store. Ensure that each virtual server has a corresponding DNS alias to provide named access to the virtual server. When you create a private mailbox virtual directory, System Manager lists the primary SMTP domains of the available recipient policies. Listing the domains allows administrators to assign a virtual directory to users who were modified by a recipient policy. Virtual directories are similar to the public folder shortcuts used in previous versions of Exchange. The only exception is that custom public folder virtual directories are not automatically exposed to the user unless the Outlook Web Access user interface is customized to take the public folder virtual directories into account.

Front-End Servers

To configure a front-end server in Exchange 2000, select the **This is a front end server** check box in the server's **Properties** dialog box, and then restart the Exchange and IIS services or restart the computer. This check box is available only if you used Exchange 2000 Enterprise Server to install Exchange. You need an additional Exchange Server license to add a front-end server to a deployment.

When you configure a front-end server, the HTTP, POP3, and IMAP4 components of Exchange redirect all traffic to a back-end server that contains the user's mailbox or relevant public folder database. The Exchange store remains intact on the server and is available to MAPI clients; however, HTTP, POP3, and IMAP4 do not access the Exchange store on a front-end server. You must move mailboxes to a back-end server. You also must maintain a mailbox store if SMTP is being used on a front-end server. If a public store exists on the front-end server, move all folders contained therein to a back-end server and delete the public store.

Configure front-end servers immediately after you install Exchange 2000 Server.

Multilanguage Support

Outlook Web Access in Exchange 2000 presents a user interface in the following languages:

- English
- French
- German
- Japanese
- Italian
- Spanish
- Chinese (Traditional)
- Chinese (Simplified)
- Korean
- Czech
- Danish
- Dutch
- Finnish
- Greek
- Hungarian
- Norwegian
- Polish
- Portuguese (Portugal)
- Portuguese (Brazil)
- Russian
- Swedish
- Turkish

Security

The section describes the options available to secure Outlook Web Access in your Exchange organization.

Authentication

A number of options are available for Outlook Web Access authentication. Choosing the appropriate method of authentication is usually dependent on the capabilities of the client operating system and the specific security policies.

The default authentication methods for Outlook Web Access in a single-server environment are Basic and Integrated Windows authentication. You set authentication on the HTTP virtual servers configured for Outlook Web Access.

Note Outlook Web Access will not have a logoff button until Exchange 2000 Service Pack 2. In earlier versions of Exchange 2000, to log off the session, the user must close the browser. For more information about how to build this functionality into Outlook Web Access, see the "[Logoff Customization](#)" section.

Use one of the following methods for authentication:

- **Method 1: Basic** Uses clear text to perform a simple challenge and response.
- **Method 2: Integrated Windows authentication** Uses the security facilities available in the Windows platform.
- **Method 3: Anonymous** Provides access to public folders that are intended for general access.
- **Method 4: Secure Sockets Layer (SSL)** Although not an authentication method, SSL provides a secure communications channel that can be used in combination with any of the above methods.

For more information about authentication methods in a front-end and back-end server configuration, see the white paper *Exchange 2000 Front-end and Back-end Topology* on the Exchange site at <http://go.microsoft.com/fwlink/?LinkId=4721>.

Enabling User Principal Name Option

Windows 2000 includes the ability to log on by using a user principal name, for example Suzan.Fine@contoso.com. To enable users to log on to Outlook Web Access using the user principal name, configure IIS and Exchange to accept basic authentication and set the default domain to " \" (without the quotation marks). Use the user principal name logon option to mask infrastructure in a hosted environment, such as by not displaying Windows 2000 domain names. The user principal name logon option requires a user to remember only his or her e-mail address and password to log on.

Use the Internet Information Services MMC snap-in to complete the following steps on each back-end server and front-end server that will support user principal name logon requests.

To enable the user principal name logon option for Outlook Web Access users on the IIS side

1. On the **Start** menu, point to **Programs**, point to **Administrative Tools**, and then click **Internet Services Manager**.
2. Right-click the computer name, and then click **Properties**.
3. On the **Internet Information Services** tab, under **Master Properties**, verify that the **WWW Service** check box is selected, and then click **Edit**.
4. In **WWW Service Master Properties**, on the **Directory Security** tab, under **Anonymous access and authentication control**, click **Edit**.
5. Verify that the **Anonymous Access** and **Integrated Windows authentication** check boxes are cleared.
6. Select the **Basic Authentication** check box. Click **Yes** when prompted by the Internet Service Manager warning dialog box.
7. Beside the **Basic authentication** option, click **Edit**.
8. In the **Basic Authentication Domain** text box, type \ in the **Domain Name** text box, and then click **OK**.
9. In **Authentication Methods**, click **OK**.
10. In **WWW Service Master Properties**, click **OK**.
11. In **Inheritance Overrides**, select the **Default Web Site/public** and **Default Web Site/Exchange** check boxes, and then click **OK**.
12. In **Internet Information Services**, right-click the computer name, and then select **Restart IIS**.
13. In **Stop/Start/Reboot**, click **OK**.

After you complete the above procedure, use Exchange System Manager to complete the following steps on each back-end server and front-end Exchange server that will support user principal name logon attempts.

1. On the **Start** menu, point to **Programs**, point to **Microsoft Exchange**, and then click **System Manager**.
2. In **Exchange System Manager**, double-click **Administrative Groups**, double-click the appropriate administrative group, double-click **Servers**, double-click the appropriate Exchange server, expand **Protocols**, expand **HTTP**, and then expand **Exchange Virtual Server**.
3. Right click the **Exchange** virtual directory, and then click **Properties**.
4. In **Exchange Properties**, click the **Access** tab, and then click **Authentication**.
5. In **Authentication Methods**, clear the **Anonymous access** and **Integrated Windows authentication** check boxes.
6. Select the **Basic authentication** check box.
7. In the **Default domain** text box, type \.
8. For public folders, repeat steps 3-7 on the **public** virtual directory.

Customization Overview

Outlook Web Access is made up of various core components. To understand the later sections in this document, it is useful to identify these components.

Warning Any change made to the Outlook Web Access source .htc, .js, or .xsl files that exist in the Exchweb folder is unsupported.

Changes made to Wmtemplates.dll may cause Exchange 2000 Server to behavior unpredictably. Microsoft does not support Exchange 2000 if this core binary file is modified.

Outlook Web Access Components

By default, Outlook Web Access is installed with the installation of Exchange 2000. Exchange 2000 Server requires Windows 2000 Service Pack 1 (SP1) and Internet Information Services (IIS) 5.0 to be installed. Exchange 2000 Server SP2 requires Windows 2000 SP2.

Figure 4 illustrates the various components that make up Outlook Web Access.

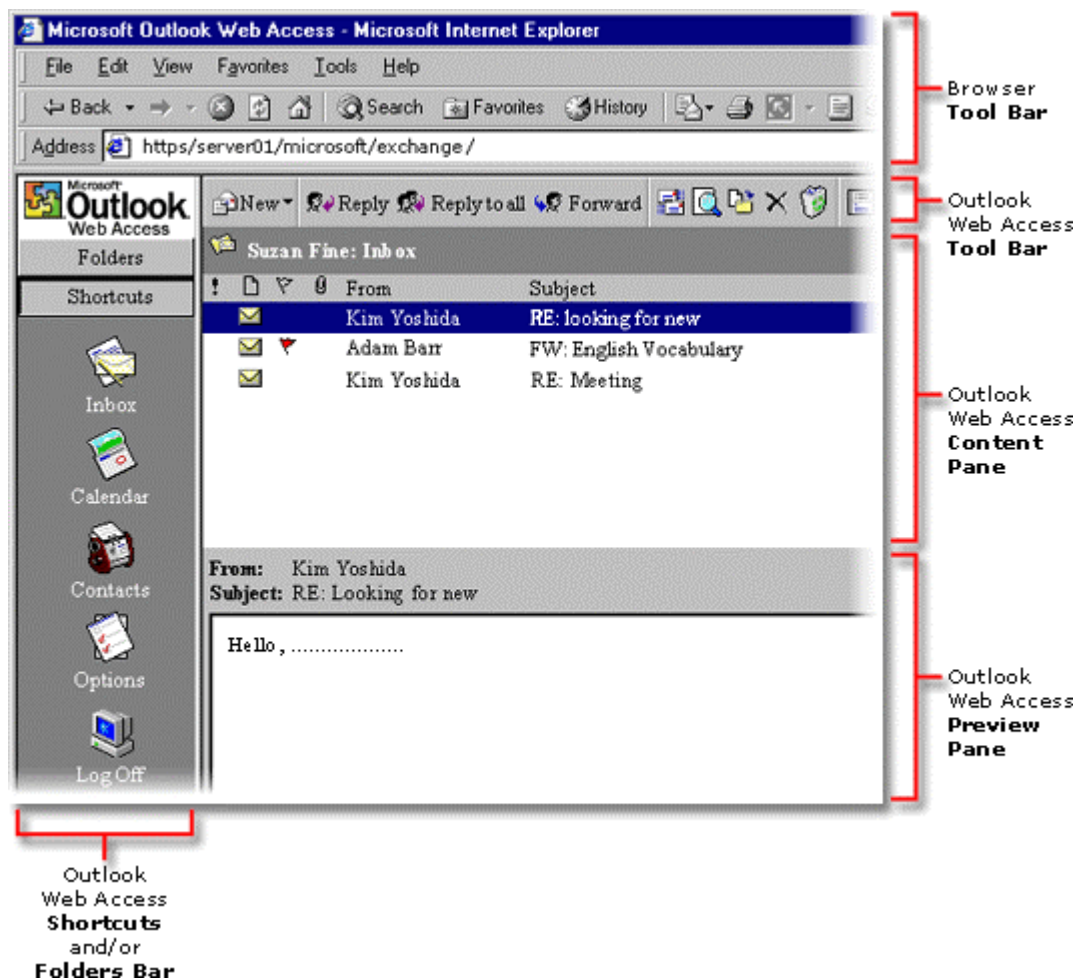


Figure 4 Outlook Web Access components

Outlook Web Access Folder Structure

Table 1 lists the directories and their contents that are relevant to Outlook Web Access. These directories are installed during Exchange 2000 setup.

Table 1 Outlook Web Access directories

Directory	Contains
Exchsrvr\Bin	Wmtemplates.dll, which defines the default templates used to render Outlook Web Access.
Exchsrvr\Exchweb\Bin	Exwform.dll, which handles custom form processing. Included with Exchange 2000 SP1.
Exchsrvr\Exchweb\Cabs	Contains the localized Exchange Multimedia Control and its associated .cab files.
Exchsrvr\Exchweb\Controls	Cascading Style Sheets (CSS), HTML Component, and client Microsoft JScript® libraries. This code is separated from the Outlook Web Access Internet Server Application Programming Interface (ISAPI) application for caching by IIS and the client.
Exchsrvr\Exchweb\Help	Localized versions of Outlook Web Access Help files.
Exchsrvr\Exchweb\Img	Graphic images used by Outlook Web Access.
Exchsrvr\Exchweb\Views	Extensible Stylesheet Language (XSL) files used to construct the standard Outlook Web Access folder views.

Figure 5 illustrates the folder structure of the Exchweb folder.

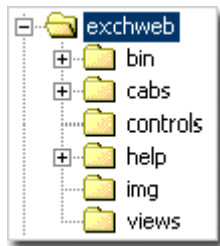


Figure 5 Outlook Web Access folder structure

Virtual IIS Directories

Exchange 2000 setup creates three virtual IIS directories that are used by Outlook Web Access and one virtual IIS directory that is used by the Exchange System Manager to administer public folders. Table 2 lists the virtual directories and their functions.

Table 2 Web virtual directories and their functions

Virtual directory	Function
Exchweb	Stores graphics and other ancillary files used by Outlook Web Access.
Exadmin	Used by Exchange System Manager to administer public folders.
Exchange	Contains the private mailbox stores.
Public	Contains the default public folders tree.

Browser Compatibility

Browser support is divided into two categories: “rich client” and all other clients.

Rich Client

As of September 2001, rich client functionality is available in Internet Explorer 5 or later for Windows, as well as in Internet Explorer 5.01 SP1 for UNIX. These browsers are referred to as the rich client because they provide users with complete Outlook Web Access functionality.

All Other Clients

Outlook Web Access in Exchange 2000 is designed to support Microsoft Internet Explorer 4.01 SP2, as well as Netscape Navigator 4.08 and later.

The non-Internet Explorer 5 clients provide support to any browser that supports HTML 3.2 and ECMA Script.

Using URLs to Access Data

One of the core functionality and accessibility changes in Exchange 2000 is that you can use a URL to access essentially every item within the Exchange store. Most Outlook Web Access customization is based on this concept.

For example, if you want to display Outlook Web Access, type the following URL in your browser, where *exchangeserver* is the name of your server running Exchange:

`http://exchangeserver/exchange/`

Figure 6 illustrates the mailbox of a user who is currently logged on to Outlook Web Access using the format shown above.

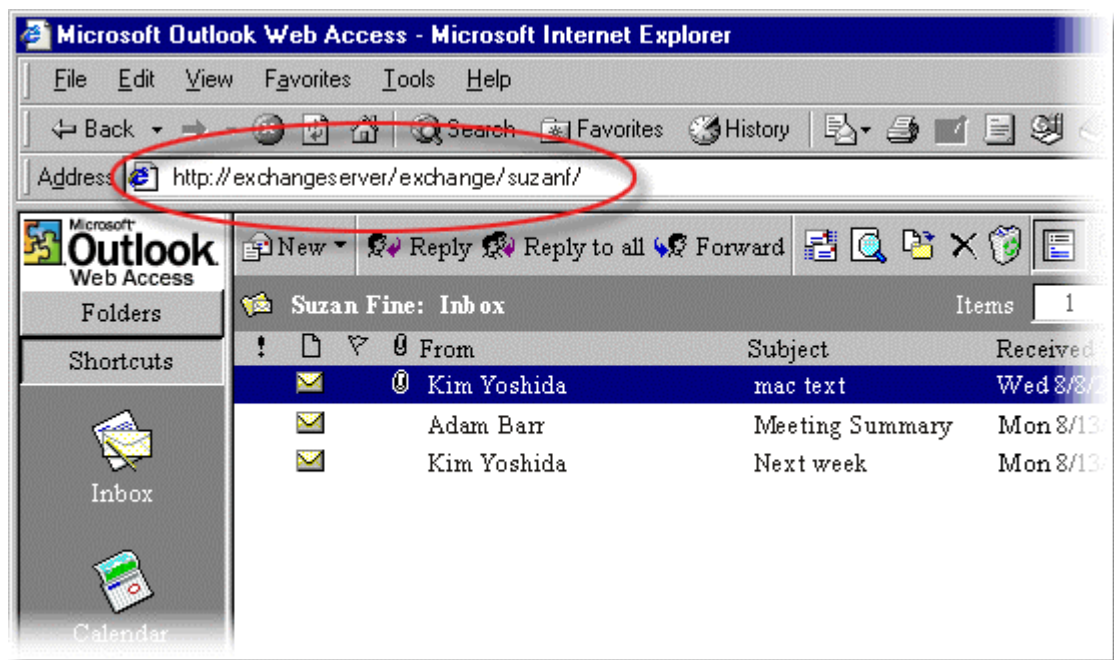


Figure 6 Default URL `http://exchangeserver/exchange/`

You can use URLs to access almost every component within Outlook Web Access, and you can individually address these components within the Exchange store. These components include the navigation bar, folder contents, views, and even the individual messages and items.

Important Many of the URLs used as examples in this white paper are case sensitive. If a URL does not work as expected, try switching the case of the first letter of the value.

Folder URL Access

The private folder hierarchy is located under the */exchange/username* virtual IIS directory. Users gain browser access to their personal folders stored on the server using this virtual directory.

For example, to display the Inbox folder, use the following URL, where *exchangeserver* is the name of your server running Exchange and *username* is the mailbox user alias:

```
http://exchangeserver/exchange/username/inbox
```

URLs for the other components are similar to the following URLs:

```
http://exchangeserver/exchange/username/notes
```

```
http://exchangeserver/exchange/username/calendar
```

```
http://exchangeserver/exchange/username/contacts
```

Essentially, you can address all folders using the format above. You can use similar syntax to address the public folder hierarchy, which is located under the public virtual IIS directory. For example, you can access the Sales folder with the following URL:

```
http://exchangeserver/public/sales
```

URL Parameters

You can pass specific parameters along with the URL. There are two categories of parameters:

- **Commands** Commands handle information.
- **Options** Options control how the information is viewed and acts on the command used.

The command parameter is specified by using the *?cmd* query string. For example, to display folder contents without the navigation frame, use the following URL:

```
http://exchangeserver/exchange/username/inbox/?cmd=contents
```

Commands

Table 3 lists some of the commands you can use to display specific folders.

Table 3 URL commands

Parameter	Description	Qualifies	Values	Internet Explorer 5 or later	Other
navbar	Displays the navigation bar. Used as left-most frame of Outlook Web Access frameset. Example: http://server/exchange/mailbox/?cmd=navbar			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
contents	Displays folder contents. Used as right-most frame of Outlook Web Access frameset. Outlook Web Access supports different views for mail, calendar, and contact folders. Example: http://server/public/discussion/?cmd=contents			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
page	Refers to a set of 25 items in a folder. Works with mail or contact folders. Example: http://server/public/discussion/?cmd=contents&page=2	?cmd=contents	Integer		<input checked="" type="checkbox"/>
view	Refers to a predefined folder view. Works with mail, calendar, and contact folders. Example: http://server/public/discussion/?cmd=contents&view=By%20Conversation%20Topic Example: http://server/exchange/mailbox/inbox/?cmd=contents&view=Messages&Page=10	?cmd=contents ?cmd=getView	Messages Unread Messages By Sender By Subject By Conversation Topic Unread by Conversation Topic Sent To Daily Weekly Monthly (Internet Explorer 5 or later only) Address Card Detailed Address Card Phone List	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
getView	Retrieves view descriptor. Example: http://server/exchange/mailbox/inbox/?cmd=getView&view=Sent%20To			<input checked="" type="checkbox"/>	

Parameter	Description	Qualifies	Values	Internet Explorer 5 or later	Other
d, m, y	<p>Displays the date, month, and year.</p> <p>Works with calendar folders. Shows the date, the week containing the date, or the month containing the date, depending on the selected view.</p> <p>Example: http://server/exchange/mailbox/calendar/?cmd=contents&view=daily&d=2&m=9&y=2003</p>	?cmd=contents ?cmd=new&type=appointment	d=[1-31] m=[1-12] y=[four digit year]	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
SortBy, SortOrder	<p>Selects view property and sort order.</p> <p>Works on message folder and contact phone list views.</p> <p>Example: http://server/exchange/mailbox/?cmd=contents&sortby=sender</p>	?cmd=contents		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
showfolders	<p>Shows folder list in contents views.</p> <p>Example: http://server/exchange/mailbox/?cmd=contents&showfolders=1</p>	?cmd=contents	0 - Hide subfolders 1 - Display subfolders		<input checked="" type="checkbox"/>
emptyDeletedItems	<p>Removes items from Deleted Items folder.</p> <p>Example: http://server/exchange/mailbox/?cmd=EmptyDeletedItems</p>			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
showDeleted	<p>Displays items recently deleted from a particular folder.</p> <p>Example: http://server/exchange/mailbox/Deleted%20Items/?cmd=showdeleted</p>			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
btnClose	<p>Displays setting for close button.</p> <p>Example: http://server/exchange/mailbox/Deleted%20Items/?cmd=showdeleted&btnClose=0</p>	?cmd=showdeleted	0 - Hide close button 1 - Show close button	<input checked="" type="checkbox"/>	
new	<p>Creates a new item. If left unqualified, creates the default item for the folder.</p> <p>Example: http://server/exchange/mailbox/calendar/?cmd=new</p> <p>Example: http://server/exchange/mailbox/contacts/?cmd=new</p>			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Parameter	Description	Qualifies	Values	Internet Explorer 5 or later	Other
newpost	Creates a new post item. Example: http://server/exchange/mailbox?cmd=newpost			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
open	Displays contents of a message, appointment/meeting, contact, contact distribution list, or post. Example: http://server/exchange/mailbox/folder/item.eml?cmd=open			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
type	Selects type of recurring appointment to open. Select master when doing a ?cmd=open on a specific instance to modify or view the appointment series. Select instance to open the specific instance of a series. Example: http://server/exchange/mailbox/Calendar/Weekly%7-31.eml?cmd=open&Type=Master	?cmd=open	master instance	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
editattach	Modifies attachment table on an item. Example: http://server/exchange/mailbox/item.eml?cmd=editattach			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
reply	Creates a reply to the sender of an item. The reply is staged in user's drafts folder, except in the case of public folders, where items with attachments are staged in the Outlook Web Access ScratchPad folder. Example: http://server/exchange/mailbox/item.eml?cmd=reply			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
replyall	Creates a "reply all" to the sender of an item. Example: http://server/exchange/mailbox/item.eml?cmd=replyall			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
forward	Forwards an item with attachments. Example: http://server/exchange/mailbox/item.eml?cmd=forward			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Parameter	Description	Qualifies	Values	Internet Explorer 5 or later	Other
edit	Modifies an unsent item. Example: http://server/exchange/mailbox/unsent_item.eml?cmd=edit				<input checked="" type="checkbox"/>
galfind	Displays the Find Names user interface. Example: http://server/exchange/mailbox/?cmd=galfind				<input checked="" type="checkbox"/>
dialog&template=	Displays a static dialog box, specified by the value of "template=". These pages are intended to be presented using window.showModalDialog in Internet Explorer 5.0 or later, and require specific JScript variables to be set in the parent document. Works on items or folders. Example: Refer to <i>openGAL()</i> function in WebClientUtil.js for an example of dlg_gal. Example: Refer to the function <i>idCalendarViewer_OnDoubleClickAppointment()</i> in Calendar.js for an example of dlg_openappt.		dlg_alert – displays alert message dlg_anr – Ambiguous Name Resolution dialog dlg_editname – Modify a recipient's name or e-mail address dlg_error – Displays error message dlg_gal – Displays Find Names dialog dlg_movecopy – Displays Move or Copy dialog dlg_msgoptions – Displays Message Option dialog dlg_newfolder – Displays New Folder dialog dlg_openappt – Displays "Occurrence or Series" prompt for opening recurring appointments dlg_recurrence – Display an appointment's recurrence dialog	<input checked="" type="checkbox"/>	
Accept	Accepts a meeting request. If the sender specifies "responses requested," the server sends a redirect to an acceptance response form. Example: http://server/exchange/mailbox/inbox/meeting.eml?cmd=accept			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Tentative	Tentatively accepts a meeting request. If the sender specifies "responses requested," the server sends a redirect to a tentative acceptance response form. Example: http://server/exchange/mailbox/inbox/meeting.eml?cmd=tentative			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Parameter	Description	Qualifies	Values	Internet Explorer 5 or later	Other
Decline	Declines a meeting request. If the sender specifies "responses requested," the server sends a redirect to a decline response form. Example: http://server/exchange/mailbox/inbox/meeting.eml?cmd=decline			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

For example, to display only the navigation bar (see Figure 7), use the following URL:

<http://exchangeserver/exchange/?cmd=navbar>

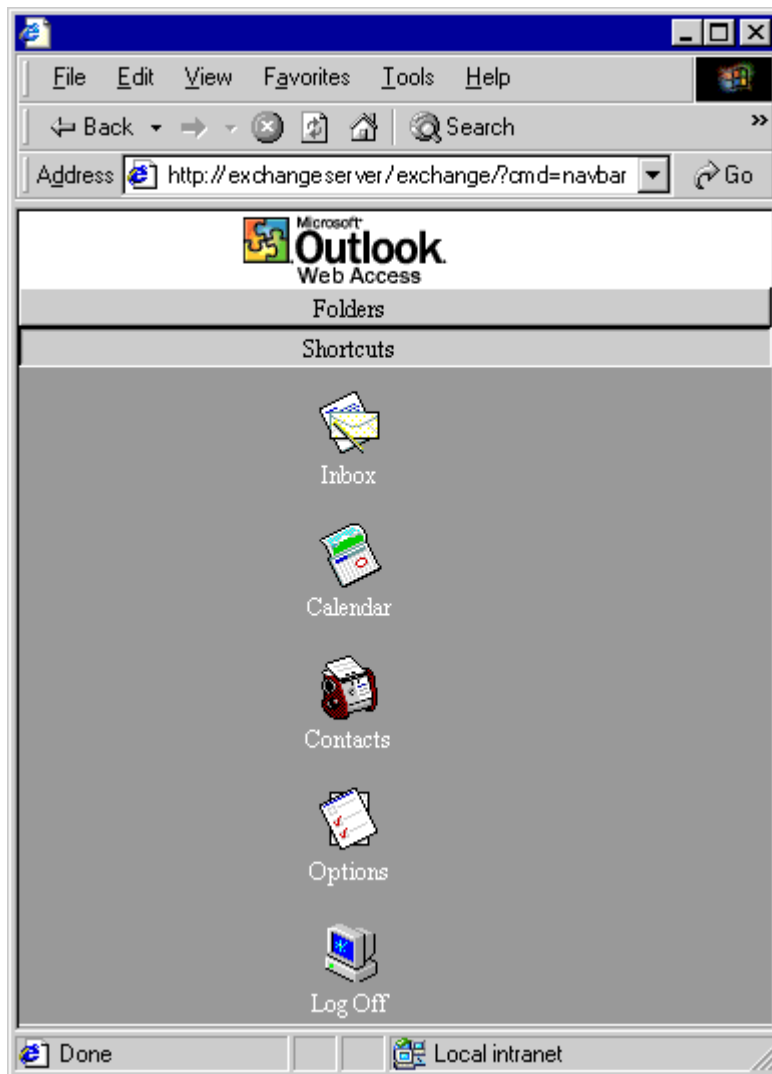


Figure 7 URL for the navigation bar

Similarly, to display the options page (see Figure 8), use the following URL:

<http://exchangeserver/exchange/?cmd=options>

Note This URL can be used only on the root folder of the mailbox; you cannot use this command on any subfolder such as Inbox, Calendar, and so on.

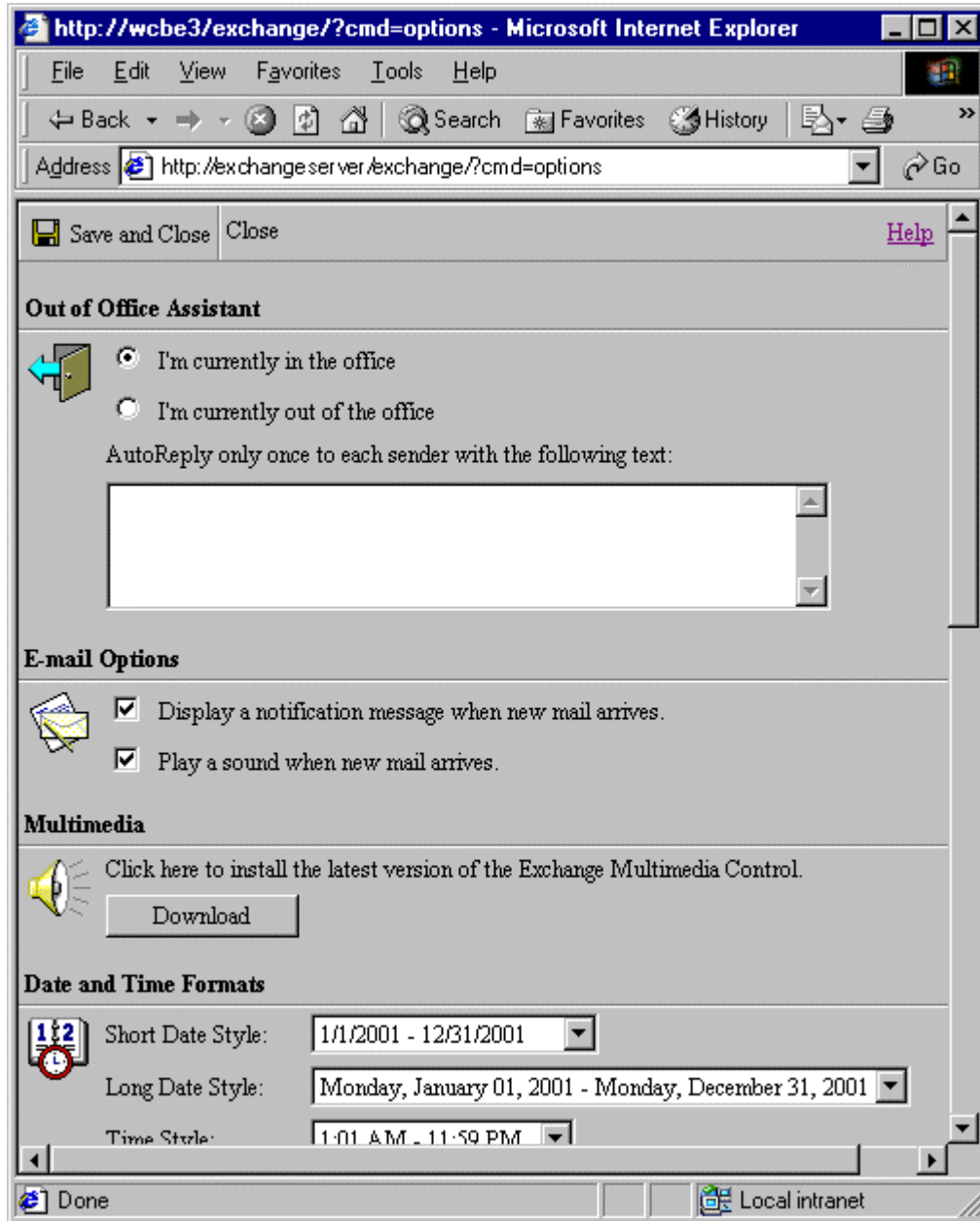


Figure 8 URL for the Options page.

To view a specific item in a folder, for example from the Inbox, use the following URL:

<http://exchangeserver/exchange/username/inbox/itemname?cmd=open>

To create a new calendar message, use the following URL:

<http://exchangeserver/exchange/username/calendar/?cmd=new>

To display a weekly calendar view (Figure 9), use the following URL:

<http://exchangeserver/exchange/username/calendar/?cmd=contents&View=weekly&d=22&m=1&y=2001>

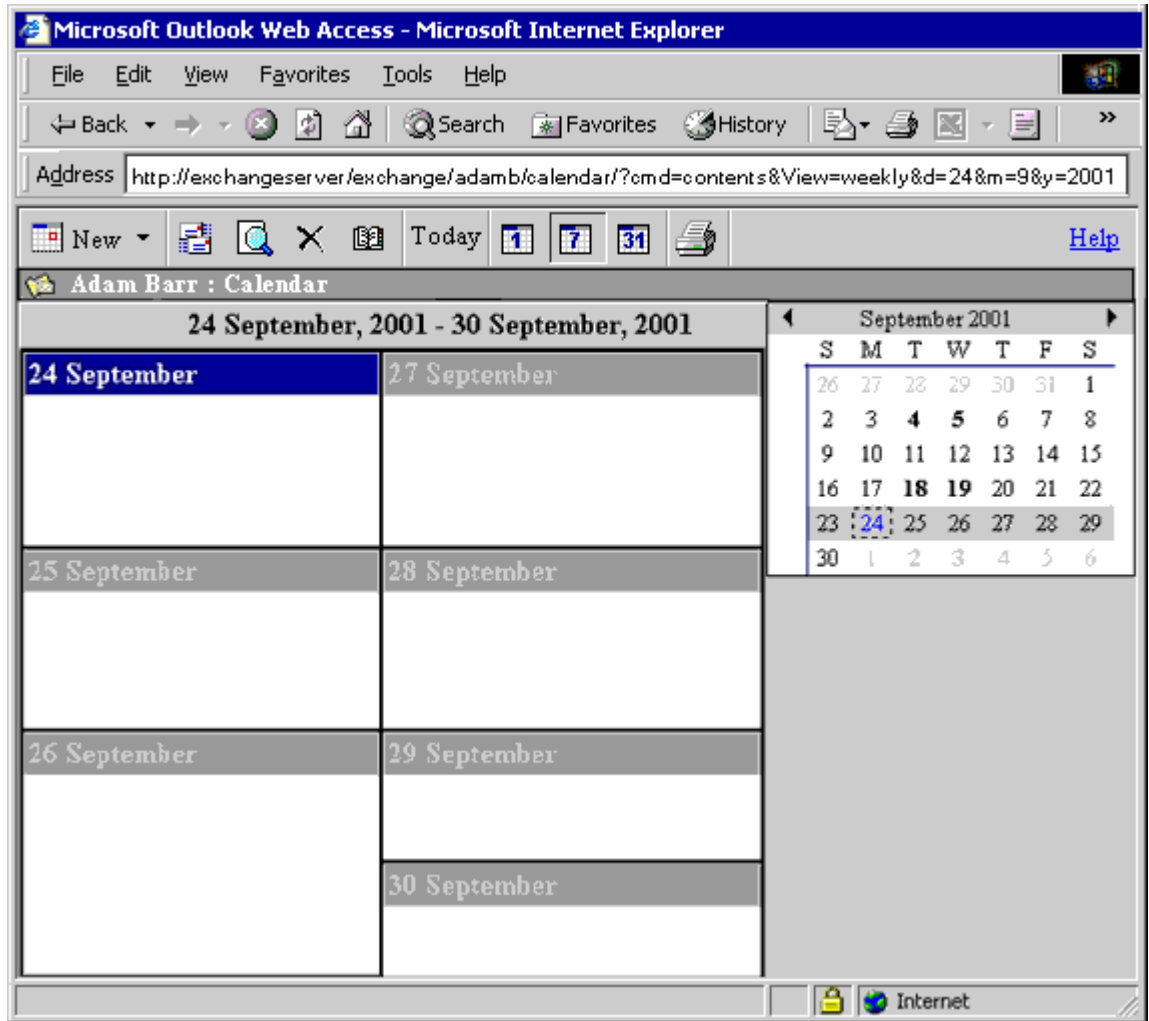


Figure 9 URL for a weekly calendar view

To display other calendar views, use the following URLs:

<http://exchangeserver/exchange/username/calendar/?cmd=contents&View=Monthly>

<http://exchangeserver/exchange/username/calendar/?cmd=contents&View=Weekly>

Global Address List Command Parameter

Table 4 lists a global address list command parameter you can use to customize Outlook Web Access.

Table 4 Global address list command parameter

Command (cmd)	Function
galfind*	Displays the global address list in earlier versions of the browsers. For rich clients along with additional parameters, this command returns search queries; for example, <code>?Cmd=galfind&DN=fred&LN=&FN=&TL=&AN=&CP=&DP=&OF=&CY=.</code>

Default Folders

Each user's mailbox folder has a set of properties that you can use to retrieve URLs for well-known subfolders such as Inbox, Calendar, Drafts, and Tasks. Additionally, you can retrieve the Microsoft Exchange mail submission Uniform Resource Identifier (URI) that is used to send messages through WebDAV or the Exchange OLE DB (ExOLEDB) provider. Table 5 lists the URLs for the default folders.

Note The returned URLs are localized into the language that is used by the client.

Table 5 URLs for default folders

Property	Well-known folder (English)
urn:schemas:httpmail:calendar	<code>http://server/exchange/username/Calendar</code>
urn:schemas:httpmail:contacts	<code>http://server/exchange/username/Contacts</code>
urn:schemas:httpmail:deleteditems	<code>http://server/exchange/username/Deleted Items</code>
urn:schemas:httpmail:drafts	<code>http://server/exchange/username/Drafts</code>
urn:schemas:httpmail:inbox	<code>http://server/exchange/username/Inbox</code>
urn:schemas:httpmail:journal	<code>http://server/exchange/username/Journal</code>
urn:schemas:httpmail:notes	<code>http://server/exchange/username/Notes</code>
urn:schemas:httpmail:outbox	<code>http://server/exchange/username/Outbox</code>
urn:schemas:httpmail:sentitems	<code>http://server/exchange/username/Sent Items</code>
urn:schemas:httpmail:tasks	<code>http://server/exchange/username/Tasks</code>
urn:schemas:httpmail:sendmsg	Exchange Mail Submission URI
urn:schemas:httpmail:msgfolderroot	<code>http://server/exchange/username/</code> Mailbox folder (root)

To access these properties

1. Bind to the user's root mailbox folder. You can either request the server and user name from the client to construct this initial URL or search Microsoft Active Directory.
2. Use the values of these properties to bind to these folders in a mailbox folder.

For example (using Microsoft Visual Basic® Scripting Edition [VBScript]):

```
Set Rec = CreateObject("ADODB.Record")
Set Conn = CreateObject("ADODB.Connection")
Conn.provider = "ExOLEDB.DataSource"
Conn.Open "http://exchangeserver/exchange/user"
Rec.Open "http://exchangeserver/exchange/user", Conn
Set Flds = Rec.Fields
msgbox(Flds("urn:schemas:httpmail:msgfolderroot"))
```

Hosting Within Frames

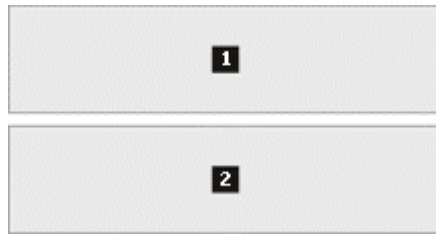
The previous section demonstrated how you can use URLs to access many Outlook Web Access components. This section describes how you can use this functionality to design and host Outlook Web Access components within HTML frames. Using frames opens up the possibilities for page design.

Note If the Navigation and Content panes are in separate frames, the frame that hosts the content pane must have an HTML Name property of **viewer**. This property ensures that the proper frame target for navigation changes is set.

Standard Frame Layouts

There are three popular frame layouts; each layout is illustrated in this section. Included after each layout is sample code to generate the frames, and how Outlook Web Access components might be used with the frameset.

Frame Layout 1



Sample frame1.asp

```
<frameset rows="100,*" frameborder="NO" border="0"
framespacing="0">

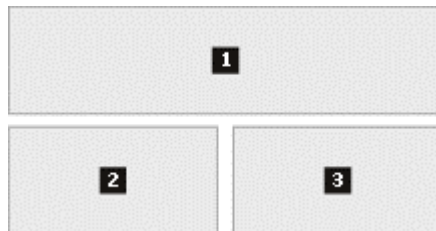
  <frame name="header" scrolling="NO" noresize src="header.htm"
border=0>

  <frame src="http://exchangeserver/exchange/"name="Outlook Web
Access" border=0>

</frameset>
```

Frame layout 1 could be used, for example, to display banner ads in frame 1 and the Outlook Web Access navigation bar and contents pane in frame 2.

Frame Layout 2



Sample frame2.asp

```

<%
on error resume next

Dim strLogonUser
Dim nIndex

strLogonUser = Request.ServerVariables("LOGON_USER")
nIndex = Instr(1, strLogonUser, "\")
If nIndex > 0 Then
    strLogonUser = Mid(strLogonUser, nIndex + 1)
End if
%>

<frameset rows="100,*" frameborder="NO" border="0"
framespacing="1">

    <frame name="header" scrolling="NO" noresize src="header.htm"
border=0>

    <frameset cols="150,*">

        <frame SCROLLING=yes name="NavBar"
src="http://exchangeserver/exchange/?cmd=navbar" target="viewer">

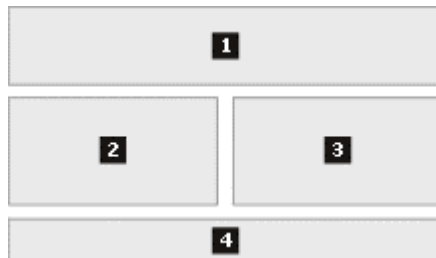
        <frame id="viewer" SCROLLING=yes name="Outlook Web Access"
src="http://exchangeserver/exchange/<%=strLogonUser%>/inbox/?cmd=
contents" frameborder=0 >

    </frameset>

</frameset>

```

Frame layout 2 could be used, for example, to display banner ads in frame 1, a customized navigation bar in frame 2, and a data view (such as message listing) in frame 3.

Frame Layout 3

Sample frame3.asp

```

<%
on error resume next

Dim strLogonUser

Dim nIndex

strLogonUser = Request.ServerVariables("LOGON_USER")
nIndex = Instr(1, strLogonUser, "\")
If nIndex > 0 Then
    strLogonUser = Mid(strLogonUser, nIndex + 1)
End if
%>

<frameset rows="100,*,100" frameborder="NO" border="0"
framespacing="1">

    <frame name="header" scrolling="NO" noresize src="header.htm"
border=0>

    <frameset cols="150,*">

        <frame SCROLLING=yes name="NavBar"
src="http://exchangeserver/exchange/?cmd=navbar" target="viewer">

        <frame id="viewer" SCROLLING=yes name="Outlook Web Access"
src="http://exchangeserver/exchange/<%=strLogonUser%>/inbox/?cmd=
contents" frameborder=0 >

    </frameset>

    <frame name="footer" scrolling="NO" noresize src="footer.htm"
border=0>

</frameset>

<HTML>
<HEAD>
</HEAD>
<BODY>
<HR>

```

Note For the above examples, the URL was generated using the user's Microsoft Windows NT® account name. Typically, an Active Directory Service Interface (ADSI) lookup should be used to find the e-mail address of the authenticated user and his or her e-mail address against Active Directory to ensure the proper URL is generated.

Frame layout 3 could be used, for example, to display banner ads in both frames 1 and 4, with a customized navigation bar in 2 and an Outlook Web Access data view in frame 3.

For an example of a customized Outlook Web Access layout, see "[Appendix D: Outlook Web Access Customization Example](#)" later in this document.

Browser Compatibility with IFrame

You can also display Microsoft Outlook Web Access components in an IFrame. IFrames are inline floating frames that provide positioning control on a Web page. For example, Web page authors can specify height and width parameters using IFrames. Figure 10 illustrates an example of Outlook Web Access hosted with an IFrame.

IFRAME is a standard HTML element as of the W3C HTML 4.0 specification. IFrames are supported in Microsoft Internet Explorer 4.0 or later and Netscape 6. For more information about IFrames, see the MSDN[®] Library.

Sample iframe.htm

```
<html>
<head>
</head>
<body>
<iframe name = "main",
src="http://exchangeserver/exchange/administrator/inbox?Cmd=conten
ts", width = 100%, height = 100%>
</iframe>
</body>
</html>
```

Note You must use Internet Explorer 4.0 or later to view this example.

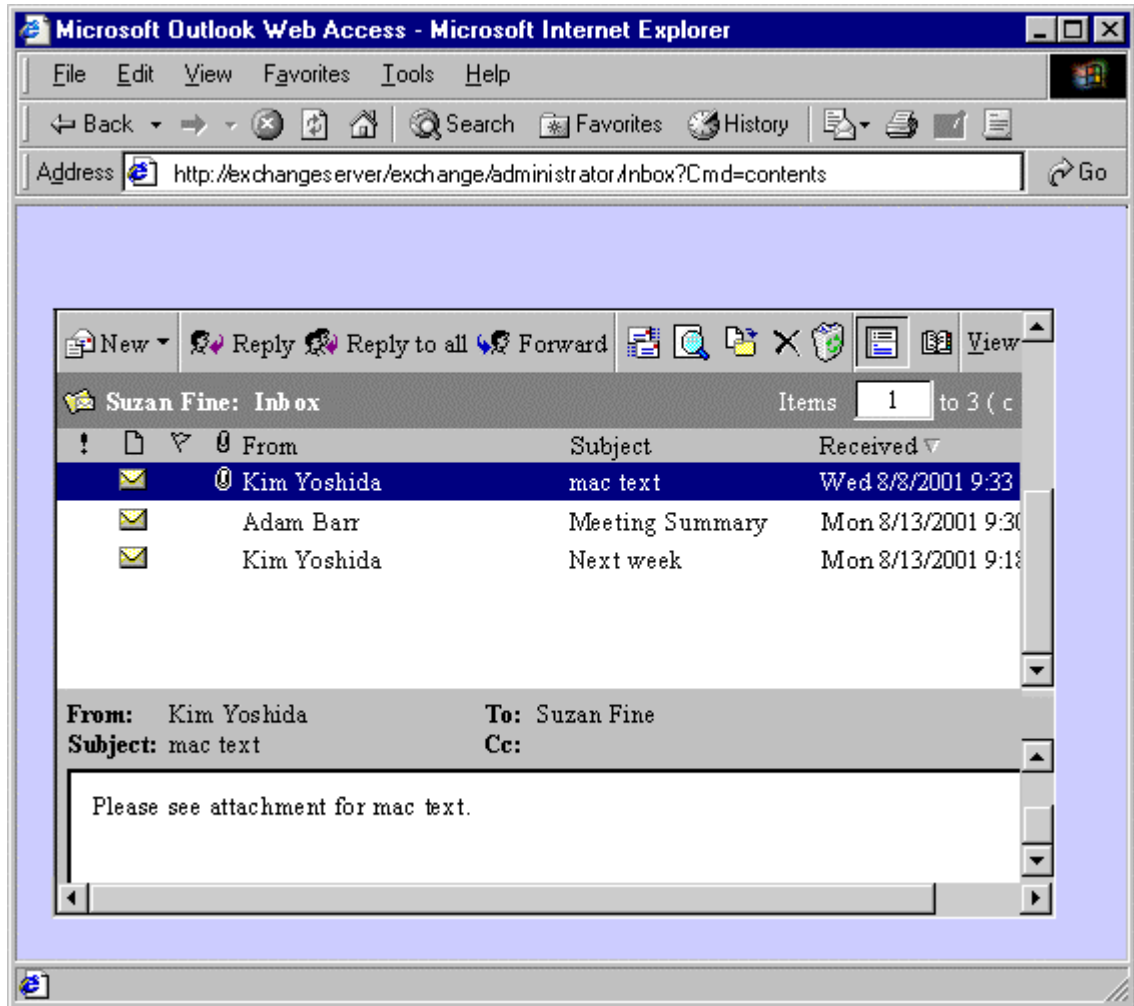


Figure 10 Outlook Web Access hosted with an IFrame

Navigation Bar

By default, the navigation bar in Outlook Web Access (located to the left of the main folder viewer) offers a number of shortcuts to your personal folders, as well as a listing of your private and public folders. The two views available in the navigation bar are **Shortcuts** and **Folders**. To switch between the two views, click the **Shortcuts** and **Folders** buttons at the top and bottom of the Outlook Bar.

Browsers other than Internet Explorer 5 or later display a **Shortcuts** view only. For these browsers, there is a **Folders** icon (see Figure 11). To display the folders in the Outlook Web Access viewing pane, double-click **Folders**.

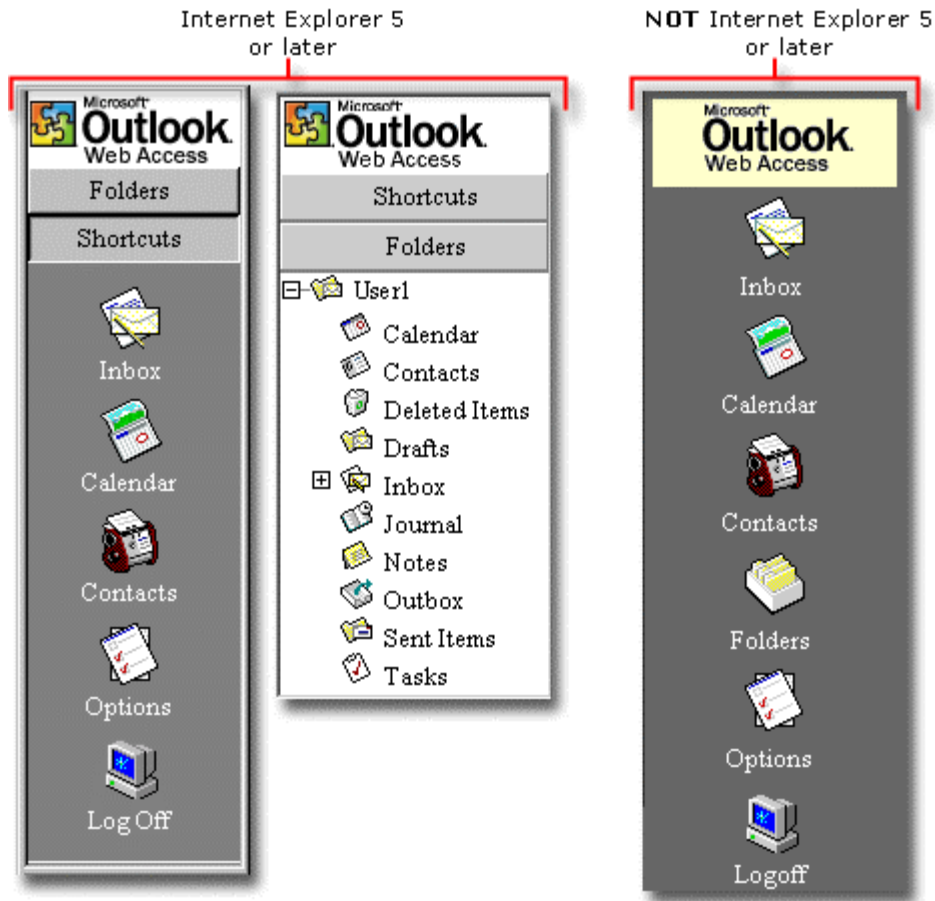


Figure 11 Outlook Web Access navigation bars

The Shortcuts view is similar to the Outlook Bar found in Microsoft Outlook (the MAPI client). The Shortcuts view provides quick links to commonly used items. Unlike Outlook, Outlook Web Access end users cannot customize the Shortcut view.

The Folders view typically shows both the current user's mailbox folder hierarchy and the Public Folder hierarchy, which are similar to the folder list in Outlook. The Folders view also contains a dropdown menu that is enabled by right-clicking an object.

Outlook Web Access Navigation Bar Components

The navigation bar consists of the following components:

- **Wmtemplates.dll** Contains the template for the navigation bar for both rich clients and earlier version clients. You can view this template in "[Appendix B: Code for the Navigation Bar](#)," later in this document. By default, this source is generated and localized using the Wmtemplates.dll.
- **/exchweb/controls/navbar.js** rich client only
- **/exchweb/controls/navbar.css** rich client only
- **/exchweb/controls/exchangetree.js** rich client only

- **Exchangetree.htc** rich client only
- **Dropmenu.htc** rich client only

Shortcut Folder URLs

Table 6 lists the default shortcut icons and URLs for the navigation bar.

Table 6 Default shortcut icons and URLs

Icon	URL
Inbox	http://exchangeserver/exchange/username/inbox/?cmd=contents&Page=1
Calendar	http://exchangeserver/exchange/username/calendar/?cmd=contents
Contacts	http://exchangeserver/exchange/username/contacts/?cmd=contents
Options	http://exchangeserver/exchange/username/?cmd=options
Folders*	http://exchangeserver/exchange/username/?Cmd=contents&ShowFolders=1

***Note** Earlier version (non-Internet Explorer 5 or later) clients only

HTML Components

Introduced in Microsoft Internet Explorer 5, HTML components (HTC) provide a mechanism to implement components in script as dynamic HTML (DHTML) behaviors. Saved with an .htc extension, an HTC is an HTML file that contains script and a set of HTC-specific elements that define the component.

Exchange 2000 uses HTCs to implement some of its DHTML behaviors. Two HTCs used in the navigation bar are the TreeView Control and the DropMenu Control.

TreeView Control

The Folders TreeView contained in the Outlook Bar Folders view bar is a good example of HTML Components and behaviors (see Figure 12). The TreeView HTC is contained in /exchweb/controls/exchangetree.htc.

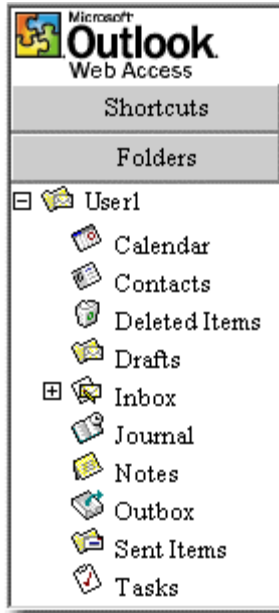


Figure 12 TreeView Control used in Outlook Web Access

DropMenu Control

To generate DropMenu functionality, right-click an object in the Folder Tree (see Figure 13). The DropMenu HTC is contained in /exchweb/controls/dropmenu.htc.

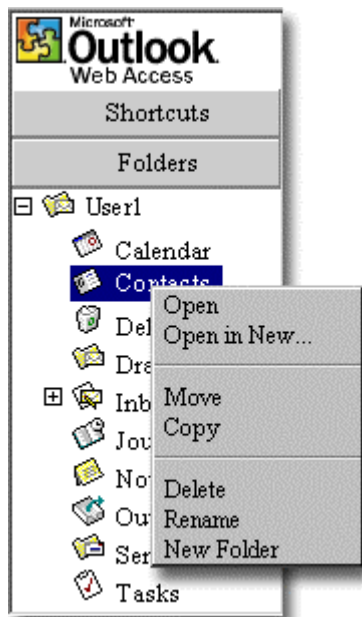


Figure 13 DropMenu control used in Outlook Web Access

You can examine the HTC files further to discover the exposed properties and methods for further customization.

Changing Navigation Bar Functionality

Some of the ways you may customize the navigation bar include:

- **Customization 1** Modify the navigation bar .js and .htc files (also known as behavior files).
- **Customization 2** Rewrite the navigation bar completely to take advantage of the Exchange 2000 ability to address URLs and the built-in HTML components (for example, the Exchange TreeView Control), and then build the new navigation bar into a frameset.
- **Customization 3** Perform customization 2, but, instead of building a frameset, register a global form in the forms registry against the **navbar** command.

Modification of the Navigation Bar

You can modify the navigation bar in a number of ways. This section describes some of those ways.

Adding Additional Folder Hierarchies

The following examples demonstrate how to add additional folder hierarchies to the Outlook Web Access navigation bar.

Customization of the Navigation Bar Source Code

The following script adds a second top-level hierarchy to the TreeView object **objtree** using the function **addHierarchy** (see Figure 14). Include this script with a customized version of the default navigation bar source found in "[Appendix B: Code for the Navigation Bar.](#)"

```
<script language=javascript>

objTree.addBehavior("/exchweb/controls/ExchangeTree.htc");
objTree.onreadystatechange = usrAddNodes;

function usrAddNodes()
{
    if (objTree.readyState != "complete")
        return(false);
    objTree.onreadystatechange = null;
    g_iAddCount=1;
    objTree.addHierarchy("http://greymage/exchange/administrator",true, "Administrator");
    objTree.addHierarchy("/secondTLH", false, "secondTLH");
}
```

```
</script>
```

The preceding code uses the **addHierarchy** method. This method takes three parameters:

- The root URL for the top-level hierarchy
- A Boolean value for expanded or not
- The display name to be displayed after the node is rendered

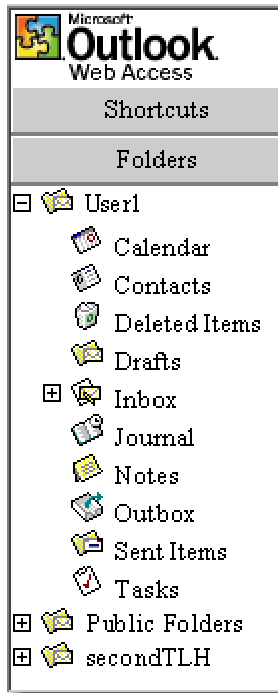


Figure 14 Navigation bar with an additional top-level hierarchy folder added

Customization of Navbar.js

The following script adds a second top-level hierarchy to the TreeView object, again using the function **addHierarchy**. However, in this script, the change is made to the Navbar.js file.

```
function initializeTree()
{
    if (objTree.readyState == "complete") {
        if (g_fPublicOnly){
            if (g_fPublicRoot)

objTree.addHierarchy(window.location,true,g_szPublicFoldersDN);
            else
                objTree.addHierarchy(window.location,true,null);    }
    }
```

```

        else {

objTree.addHierarchy(g_szUserBase, true, g_szUserDisplayName)
;

objTree.addHierarchy("/secondTLH", false, "DisplayName");}
treeProgress.style.display = "none"
var nHeight = window.document.body.clientHeight -
g_winDocAll['nbTableMain'].offsetHeight;
g_winDocAll['objTree'].style.height = nHeight;
}
}

```

Note The preceding script is only a section of the Navbar.js file. This section illustrates the changes made to include an additional top-level hierarchy folder.

Adding Buttons to the Navigation Bar

The Outlook Bar portion of the navigation bar is contained within the same table that contains the **idOutbarpane** object. By using the **navbar.js window.onload** function, you can modify this table.

To modify the table

1. Get the **idOutbarpane** object, which is the Outlook Bar pane object in the navigation bar.

```
Var objTb = g_winDocAll['idOutbarpane'];
```

2. Get the **InnerHTML** for the **idOutbarpane** object.

```
objTb.InnerHTML
```

3. Make changes to the HTML that generates the Outlook Bar pane.

For example, use the following script to add a logoff button.

```

function window.onload()
{
g_winDocAll = window.document.all;
if (!g_szUserBase)
{
return (false);
}
if (g_fPublicOnly)
{

```

```

treeProgress.style.display = "";
g_nFontHeight = g_winDocAll.idLoading.offsetHeight;
objTree.onreadystatechange = initializeTree;
objTree.addBehavior(g_szVirtualRoot +
"/controls/exchangetree.htc")
g_winDocAll['nbTableMain'].style.height = "auto";
with( g_winDocAll['idOutbarpane'] )
{
    style.height=0;
    style.display="none";
}
var nHeight = window.document.body.clientHeight -
g_winDocAll['nbTableMain'].offsetHeight;
g_winDocAll['objTree'].style.height = nHeight;
g_winDocAll['objTree'].style.display="";
}
var objTb = g_winDocAll['idOutbarpane'];
var iHeight = document.body.clientHeight - (25 + 30 + 30);
var sNavbar;

sNavbar = "<BR><A id='options' href='?cmd=logoff'
target='viewer' alt='Logoff'><IMG class='nbImage'
src='../..../exchweb/img/navbar-logoff.gif' alt='Logoff'></A><DIV
class='nbLabel'>Logoff</DIV>"

objTb.innerHTML = "<DIV id='nbTreeScroller'
style='overflow:auto; height:" + iHeight + "'>" +
objTb.innerHTML + sNavbar + "</DIV>";
}

```

Removing Public Folders from the TreeView Control

This section describes how to remove access to public folders for Outlook Web Access users in your organization.

Customization of the Navigation Bar Source Code

The following portion of code demonstrates removal of the `onaddnodes = "objTree_OnAddNodes ();"` event from the `objTree` container. Remove this event to remove the Public Folder node from being displayed by the TreeView Control.

```

        = "objTree_OnAddNodes()";
<DIV class='treeFolder' id="objTree" style="display:none;"
    onpickfolder      = "objTree_OnPickFolder()";
    onaddhierarchy    = "objTree_OnAddHierarchy()";
    onfolderoperationcomplete =
"objTree_OnFolderOperationComplete()";
    imagePath        = "http://greymage/exchweb/img/"
    acceptLang       = "en-us">
</DIV>

```

Customization of navbar.js

You can also remove the Public Folder node by modifying Exchangetree.js and, in particular, by making changes in the **addHierarchy** function.

For example, to remove the public folder hierarchy, make the following changes to the **f_AddHierarchy** function.

```

//if (displayName=="Public Folders")
    if (displayName==g_szPublicFoldersDN)
    {
        return(null)
    }
    else
    {
        var node = f_AddFolder(objFolder, null);
    }

```

Note Changes are marked in red.

The following code removes all folders with "public" in the URL.

```

function mf_AddNode(szIMG, szCaption, fHasChildren, szURL,
objAfterElement, szContentClass)
{
    var tempNode = this.__idMasterNode;
    var newNode = null;
    var szPFSearch = "public"
    if(szContentClass == ""){
        szContentClass = "urn:content-classes:mailfolder";
    }
}

```

```

if(window.document.all(szURL) != null){
    return(null);
}
if(tempNode != null && szURL.search(szPFSearch) == "-1")
{
newNode = tempNode.children[0].cloneNode(true);
newNode.id = szURL;
newNode.url = szURL;
...

```

Registering in the Forms Registry Against the navbar Command

You can also register a global form in the forms registry for the **navbar** command. Register the form globally against the **navbar** command and against the **mailfolder**, **folder**, **calendarfolder**, **journalfolder**, and **contactfolder** content classes.

For more information about registering global forms, see the "[Global Form Registration](#)" section later in this document.

Figure 15 illustrates an example of using the non-Internet Explorer 5 or later navigation bar with a rich client (Internet Explorer 5 or later) content view. To get this view, register a global Nav.asp form against the **mailfolder**, **folder**, **calendarfolder**, **journalfolder**, and **contactfolder** content classes.

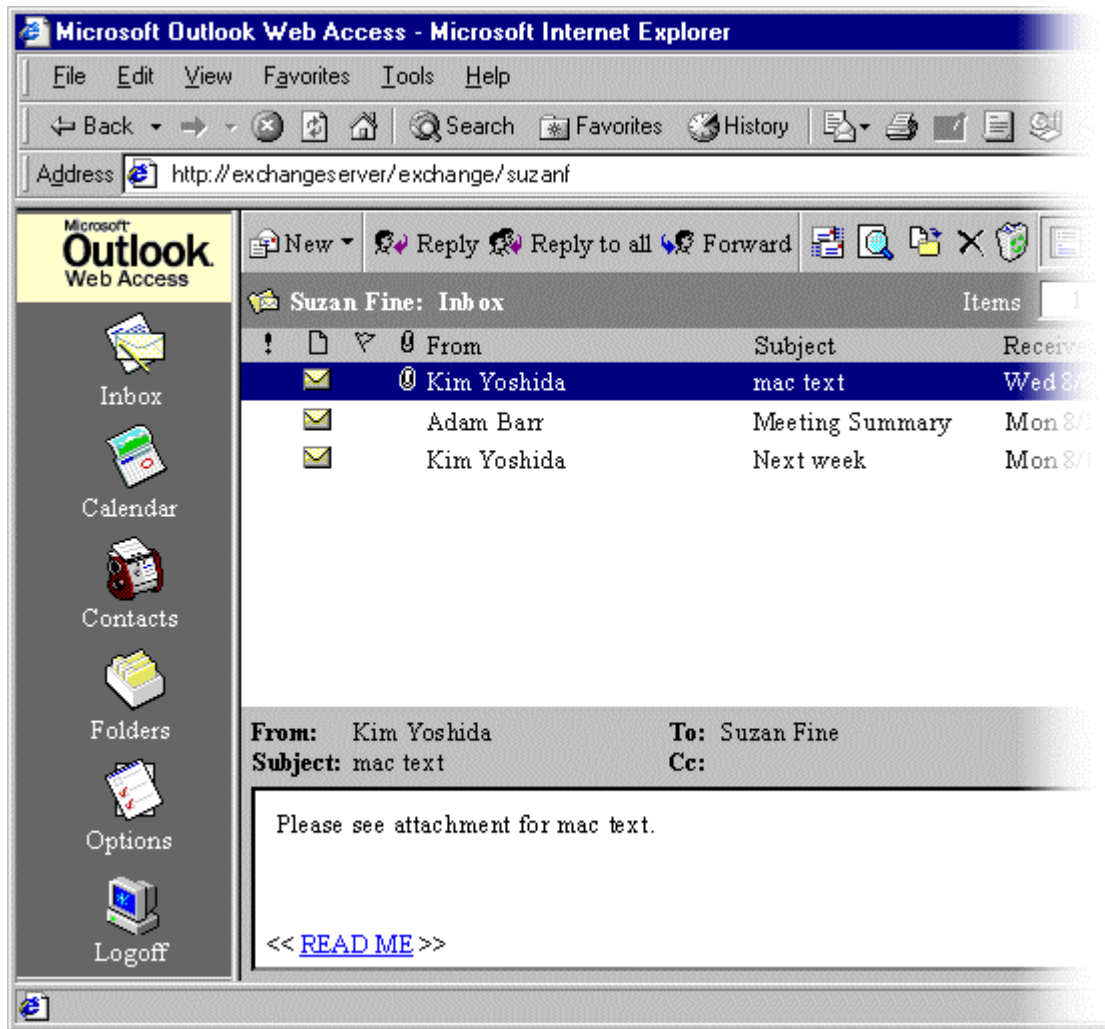


Figure 15 Outlook Web Access non-Internet Explorer 5 or later navigation bar with Internet Explorer 5 or later (rich client) content view

Sample Code for Nav.asp

Note The sample code below does not take globalized or renamed default folders into account. This code is included as an example and is not localized.

```
<META HTTP-EQUIV="Content-Type" CONTENT="text/html;
CHARSET=windows-1252">
<META HTTP-EQUIV="Pragma" CONTENT="no-cache">
<META HTTP-EQUIV="Expires" CONTENT="0">
<!--Copyright (c) 2000 Microsoft Corporation. All rights reserved.-->
<!--CURRENT FILE== "NON-IE5" "NON-WIN32" navbar -->
<HTML>
```

```

<HEAD>
<%
on error resume next
Dim strLogonUser
Dim nIndex

strLogonUser = Request.ServerVariables("LOGON_USER")
nIndex = Instr(1, strLogonUser, "\")
If nIndex > 0 Then
    strLogonUser = Mid(strLogonUser, nIndex + 1)
End if
%>
<TITLE>
None
</TITLE>
<STYLE type="text/css">
A:link      { color FFFFFFF;text-decoration none}
    A:visited  { color FFFFFFF;text-decoration none}
    A:active   { color FFFFFFF;text-decoration none}
</style>
<BASE href="http://lab01/exchange/<%=strLogonUser%>/" />
</HEAD>
<BODY bgcolor="666666" topmargin="0" leftmargin="0" vlink="0"
alink="0" link="0">
<TABLE bgcolor="666666" width="100%" border="0">
<TR>
<TD bgcolor="FFFCE" valign="top" align="center" height="38">
<IMG src="http://lab01/exchweb/img/logo-small.gif" width="91"
height="38" border="0">
</TD>
</TR>
<TR>

```

```

<TD valign="top" align="center" height="50"> <A
href="/Inbox/?Cmd=contents&Page=1" target="viewer" style='text-
decoration none'>

<IMG src="http://lab01/exchweb/img/navbar-inbox.gif" width="36"
height="36" alt="Go to inbox" border="0"><BR> <FONT color="white"
size="2">Inbox</FONT></A>

</TD>

</TR>

<TR>

<TD valign="top" align="center" height="50"> <A
href="/Calendar/?Cmd=contents" target="viewer" style='text-
decoration none'>

<IMG src="http://lab01/exchweb/img/navbar-calendar.gif" width="36"
height="36" alt="Go to calendar" border="0"><BR> <FONT color="white"
size="2">Calendar</FONT></A>

</TD>

</TR>

<TR>

<TD valign="top" align="center" height="50"> <A
href="/Contacts/?Cmd=contents" target="viewer" style='text-
decoration none'>

<IMG src="http://lab01/exchweb/img/navbar-contacts.gif" width="36"
height="36" alt="Go to contacts" border="0"><BR> <FONT color="white"
size="2">Contacts</FONT></A>

</TD>

</TR>

<TR>

<TD valign="top" align="center" height="50"> <A
href="http://lab01/exchange/<%=strLogonUser%>/?Cmd=contents&ShowF
olders=1" target="viewer" style='text-decoration none'>

<IMG src="http://lab01/exchweb/img/navbar-public.gif" width="36"
height="36" alt="Go to my folders" border="0"><BR> <FONT
color="white" size="2">Folders</FONT></A>

</TD>

</TR>

<TR>

<TD valign="top" align="center" height="50"> <A
href="http://lab01/exchange/<%=strLogonUser%>/?Cmd=options"
target="viewer" style='text-decoration none'>

```

```

<IMG src="http://lab01/exchweb/img/navbar-options.gif" width="36"
height="36" alt="Go to options" border="0"><BR> <FONT color="white"
size="2">Options</FONT></A>

</TD>

</TR>

</TABLE>

</BODY>

</HTML>

```

Modifying the DropMenu Control

The context menu that is enabled only in the rich client introduces advanced functionality within Outlook Web Access. To view the built-in navigation bar and the XML that defines the default DropMenu control, see "[Appendix B: Code for the Navigation Bar](#)" later in this document.

The dropdown menu is defined using the following XML template:

```

<WM:DROPMENU id=idTreeMenu HOLDSTYLE=true class="nbDropMenu"
menuName="treeMenu" >
  <WM:MENURROW>
    <WM:MENUIITEM type="default">
      <WM:MENUIITEMIMG/>
      <WM:MENUIITEMCAPTION>label</WM:MENUIITEMCAPTION>
      <WM:MENUIITEMSCRIPT>
idTreeMenu.hide();customfunc();</WM:MENUIITEMSCRIPT>
    </WM:MENUIITEM>
  </WM:MENURROW>
</WM:MENURROW>
...
</WM:MENURROW>
</WM:DROPMENU>

```

Note **customfunc()** refers to the function to be called when the context menu item is selected.

You can easily extend the template above to build a custom dropdown menu. Figure 16 illustrates a sample of the standard navigation bar with a customized **New Team Project** function.

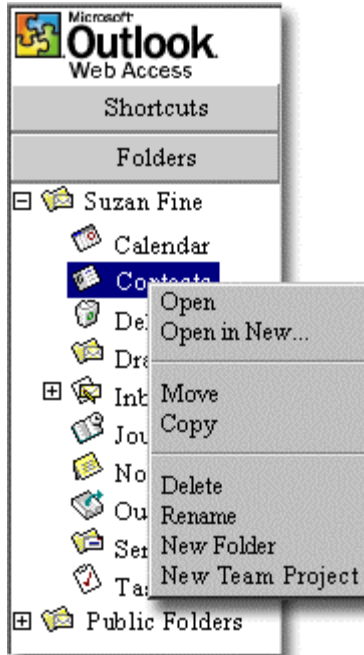


Figure 16 Dropdown menu with New Team Project function

Removing the Folders Section of the Navigation Bar

Figure 17 illustrates an example of the navigation bar without a Folders section.

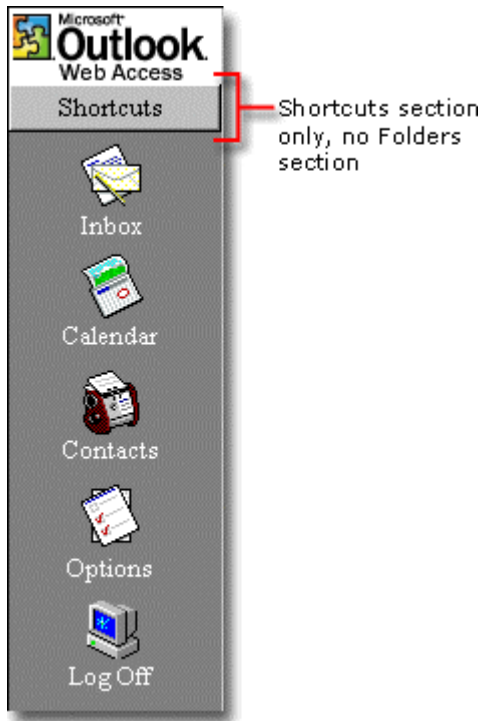


Figure 17 Navigation bar without a Folders section.

You can remove the Folders section of the navigation bar by hosting the following code in a frameset or by registering a form in the forms registry against the **navbar** command.

```

<META HTTP-EQUIV="Content-Type" CONTENT="text/html; CHARSET=utf-8">
<META HTTP-EQUIV="Pragma" CONTENT="no-cache">
<META HTTP-EQUIV="Expires" CONTENT="0">

<!--Copyright (c) 2000 Microsoft Corporation. All rights reserved.-->
<!--CURRENT FILE== "IE5" "WIN32" navbar -->
<HTML>
<?XML:NAMESPACE PREFIX="WM"/>
<HEAD>
<LINK rel="stylesheet" type="text/css"
HREF="http://exchangeserver/exchweb/controls/navbar.css">
<SCRIPT language="javascript">
var g_fPublicOnly          = false;
var g_fPublicRoot          = false;
var g_szUserDisplayName    = "user";
var g_szUserBase           = "http://exchangeserver/exchange/user" + "/";
var g_szFolder              = ".";
var g_szVirtualRoot        = "http://exchangeserver/exchweb";
var g_szImagePath          = g_szVirtualRoot + "/img/";
var g_szPublicFolderUrl    = "http://exchangeserver/public/";
var g_szPublicFoldersDN    = "Public Folders";
</SCRIPT>
<script>
var L_RenameCurrentFolder_Text          = "Rename the current folder
to:"
var L_ConfirmDelete_Text                = "Are you sure you want to
delete the current folder?"
var L_errCommunicationfailure_ErrorMessage = "Couldn't complete
request, communications error."
var L_errLogonFailure_ErrorMessage      = "Login failed or cancelled."
var L_errAccessDenied_ErrorMessage      = "Access denied."

```

```

var L_errPreconditionfailure_ErrorMessage = "Operation failed. An item
with the same name already exists in the destination folder."

var L_errInternalServerError_ErrorMessage = "An internal server error
occurred."

var L_errBadGateway_ErrorMessage = "Outlook Web Access does
not currently support this type of move/copy action."

var L_ErrorIllegalCharacter_Text = "The folder name entered
contains illegal characters."

var L_errServiceNotAvailable_ErrorMessage = "The service is no longer
available."

var L_errCrossDomainAccessDenied_ErrorMessage = "Access is denied.
Your browser security settings will not allow access to this folder from this
page."

var L_ErrorItemNotFound_ErrorMessage = "The item could not be
found. It may have been deleted."

</script>

<SCRIPT
src="http://exchangeserver/exchweb/controls/navbar.js"></SCRIPT>

<BASE href="http://exchangeserver/exchange/administrator/">

</script>

</HEAD>

<BODY id="outbar_mainbody" topmargin="0" leftmargin="0"
oncontextmenu="return(event.ctrlKey);" onselectstart="return(false);"
scroll="no" style='background-color:appworkspace;'>

<TABLE id=nbTableMain class='nbTableMain' style="height:100%;" rows=2
cols=1 cellpadding=0 cellspacing=0 >

<TR><TD class='treeBrand' >

        <DIV class='treeOFLOW'><IMG
src="http://exchangeserver/exchweb/img/logo-ie5.gif" style="padding:0;"
border=0></DIV></TD></TR>

<TR height="100%"><TD>

<TABLE class='nbTable' rows=4 cols=1 cellpadding=0 cellspacing=0 >

<TR><TD id="show_navbar" TABINDEX=0 class="nbFlybar" NOWRAP
onkeydown="flybar_keydown()" onclick="ToggleTab(this.id)">

        <DIV class='treeOFLOW'>Shortcuts</DIV>

</TD></TR>

<TR style="height:100%"><TD id="idOutbarpane" valign=top style="TEXT-
ALIGN:center;">

```

```

        <A id="inbox" href="./Inbox/?Cmd=contents&Page=1"
target="viewer" alt="Go to inbox"><IMG class="nbImage"
src="http://exchangeserver/exchweb/img/navbar-inbox.gif" alt="Go to
inbox"></A><DIV class="nbLabel">Inbox</DIV>

        <BR>

        <A id="calendar" href="./Calendar/?Cmd=contents" target="viewer"
alt="Go to calendar"><IMG class="nbImage"
src="http://exchangeserver/exchweb/img/navbar-calendar.gif" alt="Go to
calendar"></A><DIV class="nbLabel">Calendar</DIV>

        <BR>

        <A id="contacts" href="./Contacts/?Cmd=contents" target="viewer"
alt="Go to contacts"><IMG class="nbImage"
src="http://exchangeserver/exchweb/img/navbar-contacts.gif" alt="Go to
contacts"></A><DIV class="nbLabel">Contacts</DIV>

        <BR>

        <A id="options" href="?Cmd=options" target="viewer" alt="Go to
options"><IMG class="nbImage"
src="http://exchangeserver/exchweb/img/navbar-options.gif" alt="Go to
options"></A><DIV class="nbLabel" >Options</DIV>
</TD></TR>
</TABLE>
</TD></TR>
</TABLE>
</BODY>
</HTML>

```

MenuBar Toolbar

Most objects in Outlook Web Access have a toolbar. The toolbar provides additional functionality for the end user, depending on its context. Figure 18 illustrates the default Inbox toolbar.



Figure 18 Default Inbox toolbar

Hide Folder Toolbars

You can use either of the following methods to remove toolbars from folder views:

- Use HTML Document Object Model (DOM), style sheets, and some script. This method works for rich client browsers only.

- Register a new form in the forms registry for the default folder content classes.

Style Sheets and Script

Use the following code to remove the toolbar from the Calendar view. See Figure 19 and Figure 20 for before and after examples.

```
<html>
<head>
<script>
function window.onload(){
    var oFrame = document.frames("calendarView");
    if( oFrame.document.readyState == "complete" )
    {
        var oStyles = oFrame.document.createStyleSheet();
        oStyles.addRule( ".tdFolderPath", "display:none" );
        oStyles.addRule( "#idToolbar", "display:none" ); }
    }
</script>
</head>
<body>
<IFRAME id="calendarView" src=
    "http://exchangeserver/public/calendar/?cmd=contents"> </body>
</html>
```

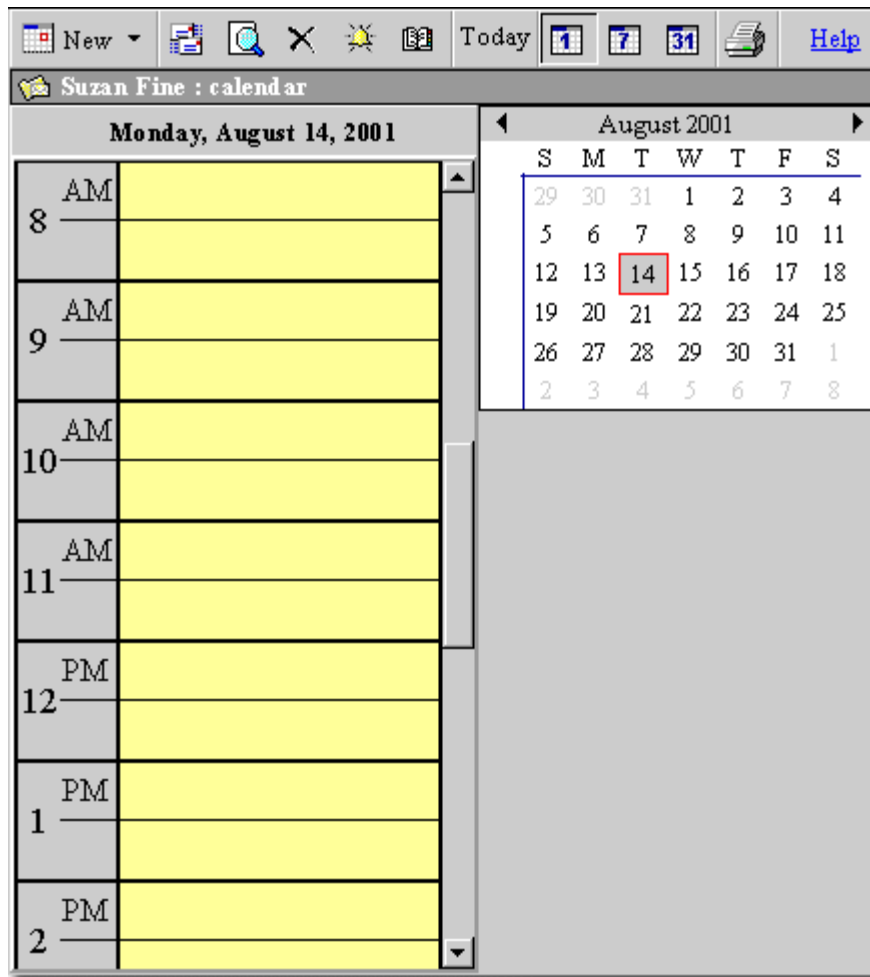


Figure 19 Before: Calendar view with toolbars

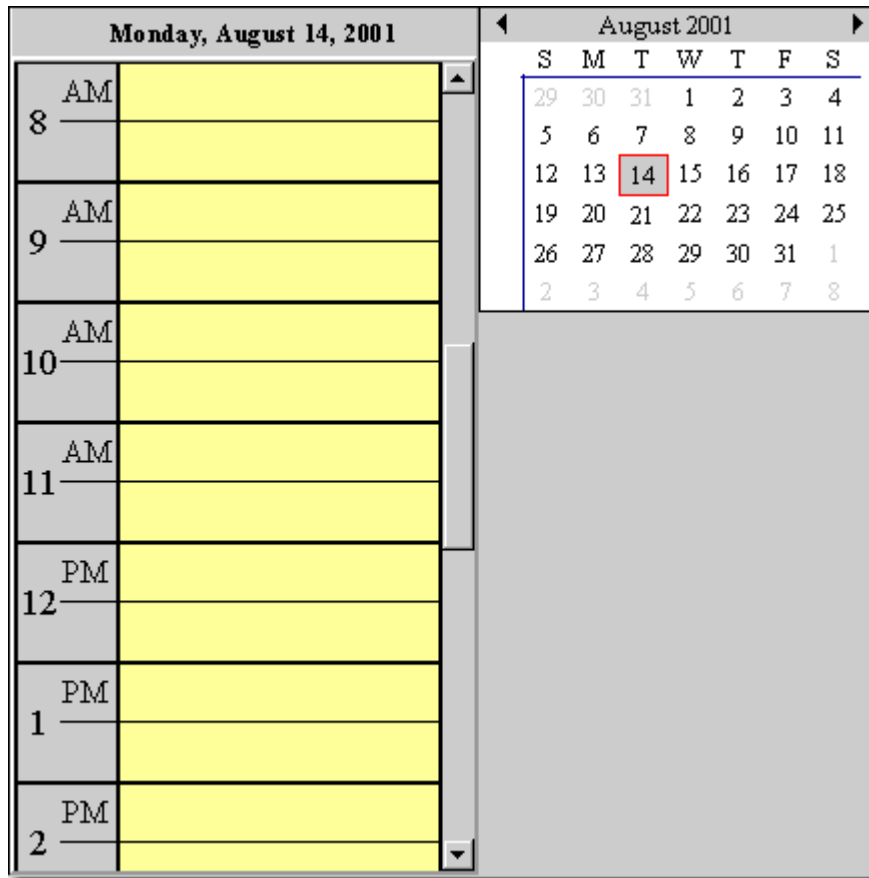


Figure 20 After: Calendar view with no toolbars

Note The page that hosts this HTML must be accessed within the relative namespace, meaning both pages should be hosted on the same server.

To remove or customize the toolbar, you must know the specific ID assigned to the toolbar and its components. In the example above, one of the IDs required is seen in the source for the contents:

```
<TABLE class="cbToolbar" id="idToolbar" cellpadding=0 cellspacing=0>
```

The following samples demonstrate this concept.

Note Place the following files in a folder on the server running Exchange, and then browse to Default.htm. Be sure to change the URL path in each of the sample files to match your server and user name.

Default.htm

```
<html>
<frameset cols="200,*">
    <frame src="nav.htm">
    <frame name="view" src="sample0.htm">
</frameset>
```

```
</html>
```

Nav.htm

```
<html>
<body>
<p><a target="view" href="./sample0.htm">Default view</a></p>
<p><a target="view" href="./sample1.htm">Hide toolbar</a></p>
<p><a target="view" href="./sample2.htm">Hide toolbar, date
picker</a></p>
<p><a target="view" href="./sample3.htm">Hide compose and delete
buttons</a></p>
</body>
</html>
```

Sample0.htm

```
<html>
<body>
<h1>Default <i>?cmd=contents</i> View</h1>
<IFRAME id="calendarview"
SRC="http://exchangeserver/exchange/administrator/calendar/?cmd=conten
ts" width="600px" height="400px">
</body>
</html>
```

Sample1.htm

```
<html><head>
<script>
function window.onload(){
    var oFrame = document.frames("calendarview");
    if( oFrame.document.readyState == "complete" )
    {
        var oStyles = oFrame.document.createStyleSheet();
        oStyles.addRule( ".tdFolderPath", "display:none" );
        oStyles.addRule( "#idToolBar", "display:none" );
    }
}
</script></head>
```

```

<body>
<h1>Hide toolbar</h1>
<IFRAME id="calendarview"
SRC="http://exchangeserver/exchange/administrator/calendar/?cmd=conten
ts" width="600px" height="400px">
</body>
</html>

```

Sample2.htm

```

<html><head>
<script>
function window.onload(){
    var oFrame = document.frames("calendarview");
    if( oFrame.document.readyState == "complete" )
    {
        var oStyles = oFrame.document.createStyleSheet();
        oStyles.addRule( ".tdFolderPath", "display:none" );
        oStyles.addRule( "#idToolbar", "display:none" );
        oStyles.addRule( "#idCalendarControl", "display:none" );
        oStyles.addRule( "#divCalStatus", "display:none" );
    }
}
</script></head>
<body>
<h1>Hide toolbar, datepicker</h1>
<IFRAME id="calendarview"
SRC="http://exchangeserver/exchange/administrator/calendar/?cmd=conten
ts" width="600px" height="400px">
</body>
</html>

```

Sample3.htm

```

<html><head>
<script>
function window.onload(){
    var oFrame = document.frames("calendarview");

```

```

if( oFrame.document.readyState == "complete" )
{
    var oStyles = oFrame.document.createStyleSheet();
    oStyles.addRule( ".tdFolderPath", "display:none" );
    oStyles.addRule( "#idCalendarControl", "display:none" );
    oStyles.addRule( "#divCalStatus", "display:none" );
    oStyles.addRule( "#NEWITEM", "display:none" );
    oStyles.addRule( "#NEWMENU", "display:none" );
    oStyles.addRule( "#delete", "display:none" );
    oStyles.addRule( "#addressbook", "display:none" );
}
}
</script></head>
<body>
<h1>Hide selected items, date picker</h1>
<IFRAME id="calendarview"
SRC="http://exchangeserver/exchange/administrator/calendar/?cmd=contents" width="600px" height="400px">
</body>
</html>

```

Figure 21 shows the Calendar view with a customized toolbar.

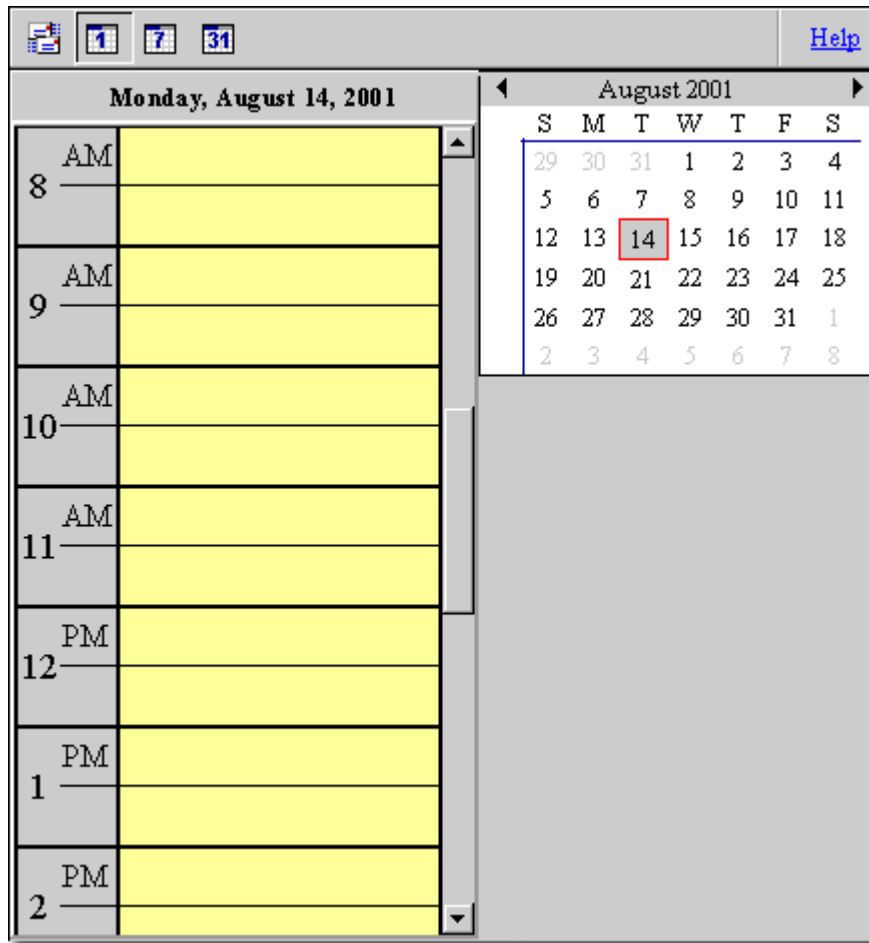


Figure 21 Calendar view with a customized toolbar

Register a Form for Folder Content Classes

Table 7 lists the forms and the default content classes for each form. For more information about registering the form, see the "[Global Form Registration](#)" section later in this document.

Table 7 Forms and default content classes

Form	Content Class
Calendar folder	urn:content-classes:calendarfolder
Contact folder	urn:content-classes:contactfolder
Journal folder	urn:content-classes:journalfolder
Folder	urn:content-classes:folder
Mail folder	urn:content-classes:mailfolder

When you register a custom form against these content classes, a custom page can be displayed for these folder types.

Global Form Registration

The forms registry enables custom-built applications to render an item or folder referenced in an HTTP request.

Outlook Web Access Built-in Forms

Outlook Web Access has a number of built-in forms that are hard-coded into the Exchange 2000 product and do not use the forms registry. These forms include the familiar forms used for Appointments, Contacts, and Messages (see Figure 22).

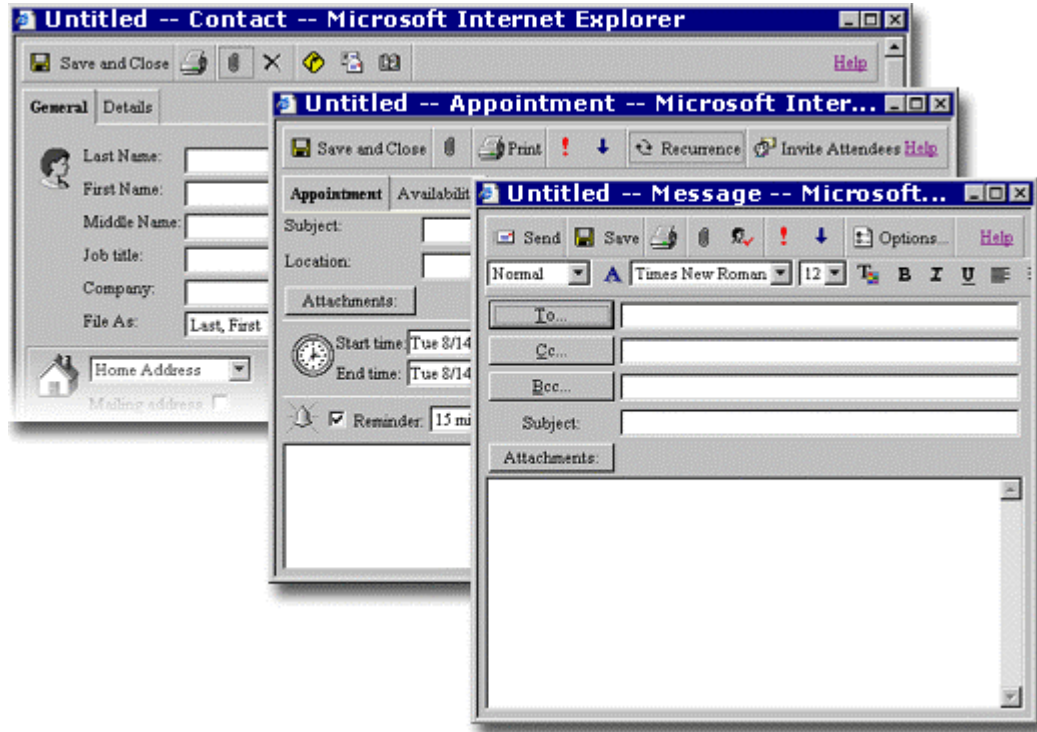


Figure 22 Default Outlook Web Access forms

Outlook Web Access Default Registrations

Table 8 lists the forms that are registered when Outlook Web Access is installed.

Table 8 Content classes

Form	Content Class
Message	urn:content-classes:message
Contact	urn:content-classes:person
Appointment	urn:content-classes:appointment
Calendar folder	urn:content-classes:calendarfolder
Contact folder	urn:content-classes:contactfolder
Folder	urn:content-classes:folder
Mail folder	urn:content-classes:mailfolder

Forms Registry

The Microsoft Exchange store forms registry provides a method for storing form registrations in the Exchange store. These form registrations can specify customized forms and actions based on a set of properties, thereby providing a way to register and query forms that are associated with an item of a particular content class. Registered forms that apply globally across the Exchange store are registered in Global Schema folders.

The following sections provide the locations of the Global Schema folders for the mailbox and public folder stores.

Exchange 2000 Server with No Service Pack Installed

The Global Schema folders for private mailbox stores are located at:

`http://exchangeserver/exchange/SystemMailbox{GUID}/NON_IPM_SUBTREE/Schema/`

The Global Schema folders for public folder stores are located at:

`http://exchangeserver/public/non_ipm_subtree/schema/`

Exchange 2000 Server with SP1

The Global Schema folders for private mailbox stores are located at:

`http://exchangeserver/exchange/SystemMailbox{GUID}/NON_IPM_SUBTREE/Schema-root/Microsoft/ExchangeV1/`

The Global Schema folders for public folder stores are located at:

`http://exchangeserver/public/non_ipm_subtree/schema-root/Microsoft/ExchangeV1/`

Note Server-wide form registrations are not supported. Registrations are per database only.

Registering New Global Forms

You must perform the following three steps to register new global forms in the private mailbox stores. This section details these three steps.

1. Determine the mailbox store globally unique identifier (GUID).
2. Allow access to the system mailbox.
3. Register the new form.

To perform these steps, use the following tools:

- Exchange System Manager MMC
- Exchange Explorer (available with Exchange Developer tools) or another method to register the form

Determine the Mailbox Store GUID

1. Use the Exchange System Manager MMC to locate the mailbox store against which you will register the forms (see Figure 23).

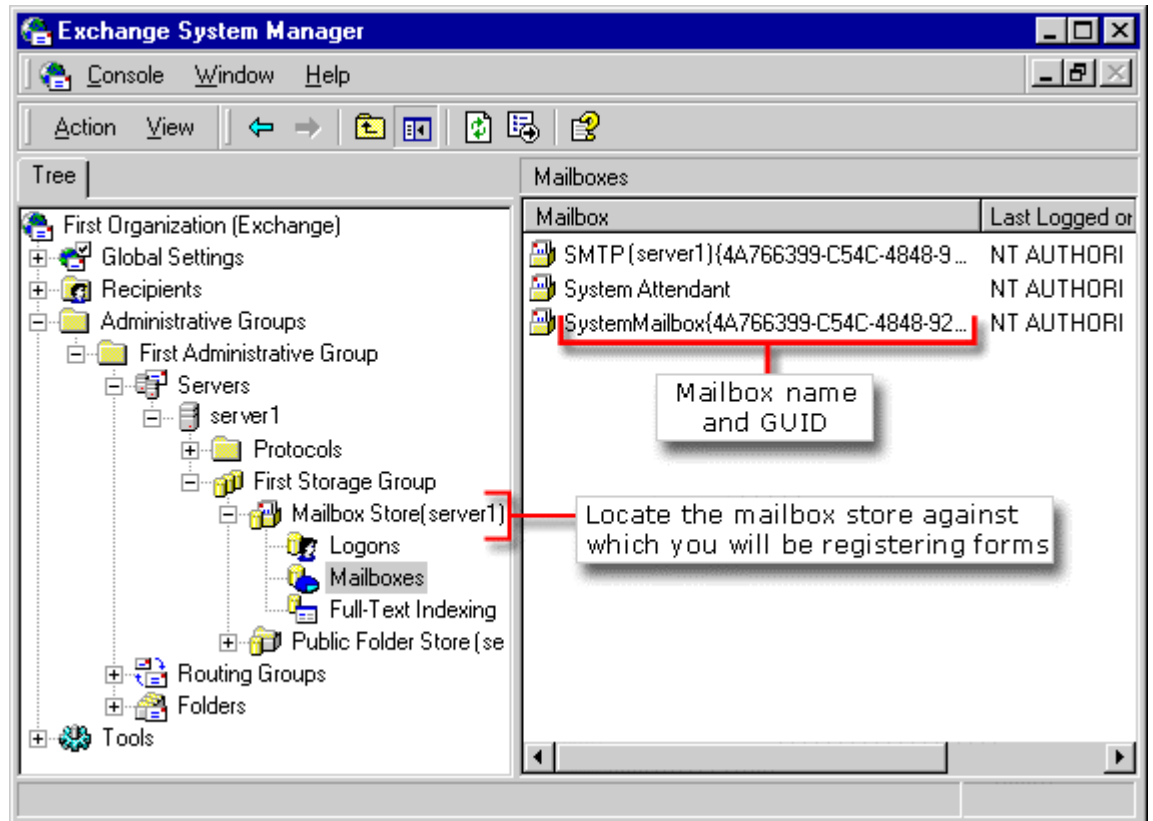


Figure 23 Exchange System Objects in Exchange System Manager

2. Under **Mailboxes**, in the **Mailbox** column, find the `SystemMailbox{GUID}` entry (see Figure 24). Use this mailbox name and GUID to register global forms against.

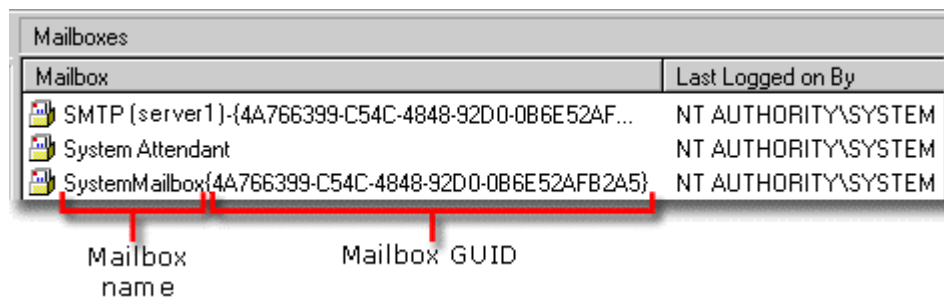


Figure 24 SystemMailbox name and GUID

In the example above, the GUID is {4A766399-C54C-4848-92D0-0B6E52AFB2A5}.

Note If you have more than one private mailbox store, you will have more than one `SystemMailbox{GUID}` combination. To select the appropriate mailbox store, look at the properties and find out with which mailbox store it is associated.

You can also get the mailbox store GUID by using the following VBScript:

```
set igetSG=CreateObject("Exoledb.StoreGuidFromUrl.1")
'Get the guid for the registration process

guid =
igetSG.StoreGuidFromUrl("http://exchangeserver/exchange/anystoreitem")
'anystoreitem is just that, any store item in the store will get you the 'store
url. It might be something like:
' "user1/inbox"

msgbox(guid)
```

Allow Access to the System Mailbox

If your logon account is the Administrator account or is a member of the Domain Admins or Enterprise Admins groups, you are denied access to all mailboxes other than your own, even if you otherwise have full administrative rights over the Exchange system. Unlike Exchange Server 5.5, all Exchange 2000 administrative tasks can be performed without granting an administrator rights to read other people's mail.

You can override this default restriction in several ways, but doing so should be in accordance with your organization's security and privacy policies. For more information about getting access to the system mailbox, see [Q262054](#), "XADM: How to Get Service Account Access to All Mailboxes in Exchange 2000," in the Microsoft Knowledge Base.

Register the New Form

You can register new forms in the following two ways:

- You can use Exchange Explorer.
- You can use code.

To register a form within a mailbox using Exchange Explorer

The following procedure shows how to register a new global form for the **mailfolder** content class urn:content-classes:mailfolder when you use the **new** command; for example:

```
http://exchangeserver/exchange/username/inbox/?cmd=new
```

Note This example uses Exchange 2000 SP1.

1. Start Exchange Explorer, log on, and then type the proper URL in the **Exchange store URL** box (see Figure 25); for example:

```
http://server01/exchange/SystemMailbox{GUID}/non_ipm_subtree/schema-root/
```




Figure 27 WSS Explorer Form Registration Wizard

4. Enter the Uniform Resource Locator (URL) for the form (for example, `http://exchangeserver/public/forms/form.htm`), and then click **Next**.
5. Enter or select an action that your form performs (for example, New) and click **Next**.
6. Select the browsers your form supports and click **Next**. For example, select "Any" if your form supports all browsers.
7. Select the data binding type (**Client** or **Server**), and then click **Next**.
8. Select the request method (**Any**, **POST**, or **GET**), and then click **Next**.
9. Enter or select a content class (for example, `urn:content-classes:mailfolder`), and then click **Next**.
10. Click **Finish**.

To register a form by using VBScript

You can also register the form by using the following VBScript:

```
Set oCon = CreateObject("ADODB.Connection")
oCon.ConnectionString = scrFolder
oCon.Provider = "ExOleDb.Datasource"
oCon.Open
```

```
Set oRec = CreateObject("ADODB.Record")
oRec.Open "defaultASPForm.frg", oCon, 3, 0
oRec.Fields("DAV:contentclass") = _
```

```

"urn:schemas-microsoft-com:office:forms#registration"
oRec.Fields("urn:schemas-Microsoft-com:office:forms#binding") = "server"
oRec.Fields("urn:schemas-microsoft-com:office:forms#cmd") = "new"
oRec.Fields("urn:schemas-microsoft-com:office:forms#contentclass") = _
    "urn:content-classes:mailfolder"
oRec.Fields("urn:schemas-microsoft-com:office:forms#contentstate") = "*"
oRec.Fields("urn:schemas-microsoft-com:office:forms#executeurl") = _
    "/exchweb/bin/exwform.dll"
oRec.Fields("urn:schemas-microsoft-com:office:forms#formurl") = _
    "http://exchangserver/user/form.htm"
oRec.Fields("urn:schemas-microsoft-com:office:forms#request") = "*"
oRec.Fields.Update
oRec.Close

```

Customized Form Header and Footers

The default forms for Outlook Web Access are not registered using the forms registry and cannot be easily customized. Modifying Outlook Web Access default forms is not supported by Microsoft and may cause data loss or corruption.

Style Sheets

Cascading Style Sheets (CSS) are design templates that provide augmented control over presentation and layout of HTML elements. CSS allows you to separate the way you design information from the HTML content by linking to a style sheet from your HTML file. This method allows you to make changes to a single file to change the appearance of multiple Web pages. CSS is used extensively in Outlook Web Access, in particular with the rich client experience.

For more information about Cascading Style Sheets, see the World Wide Web Consortium (W3C) Web site at:

<http://www.w3.org/Style/CSS/>

Earlier version browsers do not use CSS, so customizations of the CSS files does not affect presentation in those browsers.

Outlook Web Access Style Sheets

You can change the CSS files to modify the border style and colors of user interface elements, text font, size, color, and decoration used in Outlook Web Access. Do not change the CSS **position:absolute** and **position:relative** settings because these settings affect the page's functionality, as well as the page's appearance.

Table 9 lists the style sheets used by Outlook Web Access with Internet Explorer 5 or later.

Table 9 Outlook Web Access style sheets and functions

Style sheet	Outlook Web Access function
Calendarprint.css	Calendar style sheet used for printing calendar views.
Dlg_gal.css	Find Names Dialog style sheet.
Freebusy.css	Free Busy style sheet.
Navbar.css	Navigation Bar style sheet.
Olprint.css	Used in formatting when print.
Olstyle.css	General Outlook Web Access style sheet.
Exttoolbar.css	Controls the appearance of Outlook Web Access toolbars.
Toolbars.css	Controls the appearance of Outlook Web Access toolbars.

Customization

Figure 28 shows a customized view of Outlook Web Access. Font and color changes were made to the navigation bar style sheet and the toolbars style sheet.

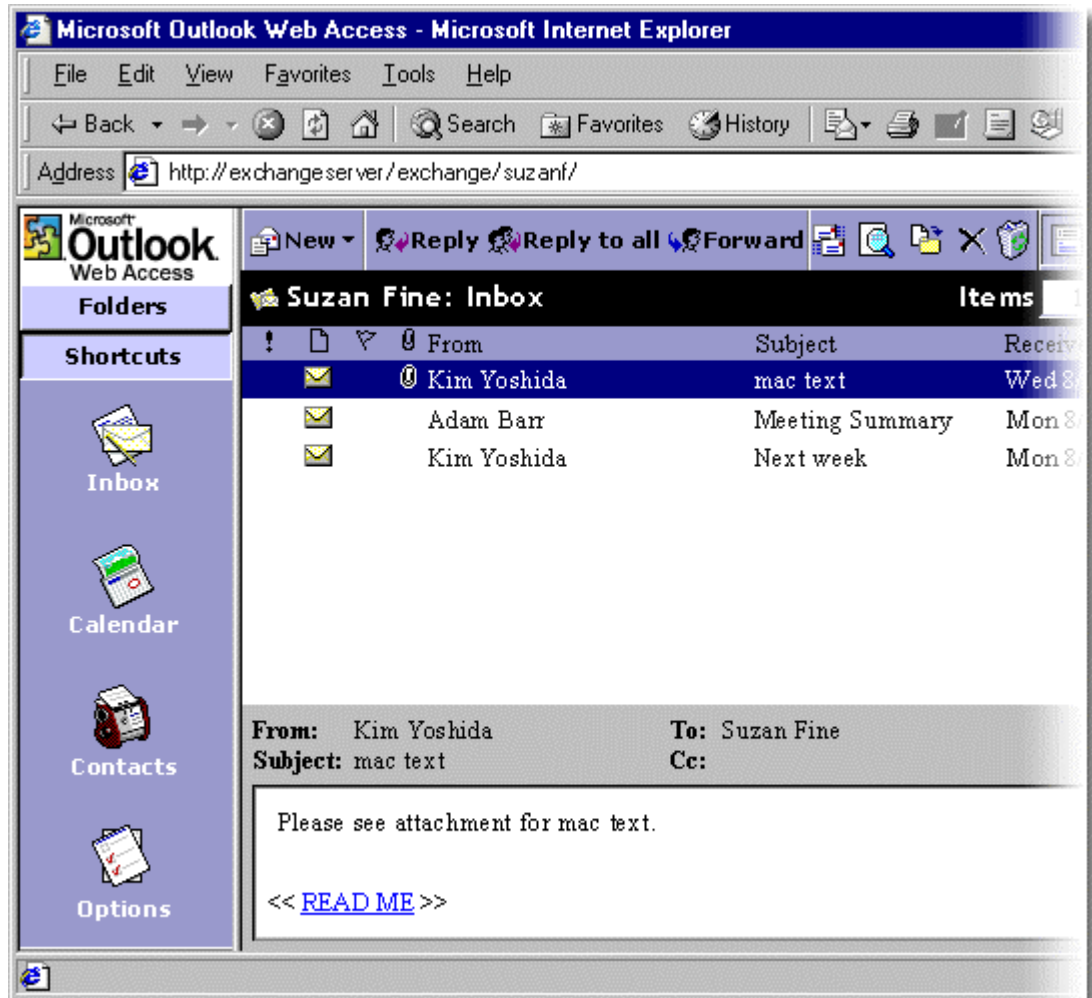


Figure 28 Outlook Web Access style sheet customization.

Icon Customization

The `exchweb/img` folder contains all the icons used for Exchange 2000. The icons all use the Graphics Image File 1989 (GIF89) format. If you add icons that will be used with the existing icons, the new icons should be the same size, resolution, and have the same transparency as the existing icons. Whenever possible, new icons should use the smallest possible set of Web-safe colors. To refer to the icons in your forms, use `` tags or create "img" elements using the browser DOM.

Note If you customize Outlook Web Access and include your own images, store the images in the same location.

Table 10 lists the naming prefixes for the items in the `exchweb/img` folder.

Table 10 Image prefixes and functions

Image prefix	Function
Navbar	Images associated with the navigation bar.

Image prefix	Function
Tool	Images associated with the toolbar.
View	Images associated with the view pane.
Icon	Miscellaneous icon images.
Icon-doc	Images associated with different document types, for example, Microsoft Word and Excel.
Icon-flag	Images associated with item flags.
Icon-contact	Images associated with contacts.
Icon-folder	Images associated with the TreeView control and folder views in earlier version of the clients.
Icon-msg	Images associated with message items.
Icon-mtgregq	Images associated with appointment items.
Form	Images associated with the built in Outlook Web Access forms, for example, appointment and message.

Default Outlook Web Access Views

Outlook Web Access uses three main XSL files to render folder views. Table 11 lists these .xsl files and the function of each one.

Table 11 XSL files and functions

View	Function
cardview.xsl	Used to transform contact data and provide Address Card and Detailed Address Card views.
v_cal.xsl	Used for client-side transformations of calendar data in IE5 calendar views.
standardview.xsl	Used for client-side transformations of tabular message folder and contact phone list views.

The view that is displayed depends on the version of the browser being used:

- **Internet Explorer 5 and later browsers view behavior** If the browser requesting a page with a view is Internet Explorer 5 or later, the Web client view behavior is used to display the view on the client side. The view behavior processes the HTML/XML view description on the client, queries the server for the appropriate data, and then displays this data using the DHTML support that exists in Internet Explorer 5 or later.
- **HTML 3.2 view behavior** If the browser requesting a page with a view is not Internet Explorer 5 or later, the server does not return the HTML/XML view description to the client. Instead, the server renders the view into an HTML 3.2-compatible table. This table is then returned to the client as static HTML.

Modifying the Default Views

The default views are hard-coded in Outlook Web Access and cannot be modified without creating a new view.

Use one of these two methods to create new views:

- **Method 1** Use the Outlook client to create new views for each folder. Use the **This folder, visible to everyone** setting.

- **Method 2** Create and use the Web Forms View Control (Wfview.htc), included with Exchange 2000 SP1.

Web Forms View Control

The Exchange Web Forms View Control—an HTML component (HTC) that works with Exchange—allows Web developers to easily display and customize their data. Web Forms View Control comes with Exchange 2000 SP1. The control provides an easy way to display data from the Exchange store in tabular or group-by views. You can attach the behavior in your Web document and provide a small view descriptor that describes which data fields to display and how they should appear.

The files used are:

- Wfview.htc (exchweb/controls folder)
- Wfview.js (exchweb/controls folder)
- Wfview.xsl (exchweb/views folder)

Several considerable differences exist between Outlook Web Access View Control and the Web Forms View Control. Web Forms View Control was modified, and it is now more generic and configurable. Some of the default behaviors (such as adding **NOWRAP** attributes to HTML table cells) were removed. Other features were added, such as check boxes and single-click navigation.

As with any HTML behavior, there are five ways you can reference a behavior from a page:

- **Inline Style** Attach the behavior by using the style attribute.

```
<DIV style="behavior:url(/exchweb/controls/wfview.htc)" ... >
```

- **Embedded Style** Attach the behavior to every DIV.

```
<STYLE>
DIV {BEHAVIOR:url(/exchweb/controls/wfview.htc);}
</STYLE>
```

- **Style Class** Define a style class to be used.

```
<STYLE>
msgViewer {BEHAVIOR:url(/exchweb/controls/wfview.htc);}
</STYLE>
```

- **Style Object** Attach to a style object using DOM.

```
<SCRIPT>
function window.onload()
{
    idMsgViewer.style.behavior = "url(/exchweb/controls/wfview.htc)";
}
```

```
</SCRIPT>
```

- **From a Script** Attach to a DOM object using scripting.

```
function window.onload() { objMsgViewerDiv.addBehavior("wfview.htc"); }
```

Table 12 lists the properties available for Web Forms View Control.

Table 12 Properties for Web Forms View Control

Property name	Read	Write	Default	Description
acceptLang	Y	Y		Indicates the language string sent by the browser, also known as the ACCEPT-LANGUAGE HTTP header. Used in the request header of the XML query.
Query	Y	N		The query property used by Excel.
viewDescriptor	Y	Y		An ID to an XML data island within the current document. An XML data island is a group of XML elements contained inside an HTML document.
viewDescriptorURL	Y	Y		A URL to a file containing only the view descriptor. View descriptors should not contain XML tags.
url	Y	Y		The URL used to query for data.
viewClass	Y	Y	/exchweb/views/wfview.xsl	The XSL file used to transform the data to HTML.
rowsPerPage	Y	Y	0	Number of rows displayed per page. If zero, all rows are retrieved and displayed on the same page.
linkSpec	Y	Y	%dataurl%/?cmd=open	Specifies the URL that the view uses to browse when items in the view are selected.
Target	Y	Y	_self	The target used when browsing to an item.
navType	Y	Y	0	0 for single-click navigation, 1 for double-click
Page	Y	Y		Specifies the page of the view to show.
Filter	Y	Y		Specifies the filter to be used to restrict the elements in the view. The filter specifies the SQL construct that follows the WHERE clause, properly escaped for inclusion as an XML element value.
filterAppend	Y	Y		Adds a clause to the Filter using AND. Must follow SQL syntax and must be properly escaped for inclusion as an XML element value.
searchSubfolders	Y	Y	0	Essentially a Boolean operator. If set to 1 , the search is performed over the specified folder and all available subfolders. This is also known as a "deep traversal." If set to 0 , the search operates on the contents of only the specified folder (shallow traversal).
Scope	Y	Y	SHALLOW TRAVERSAL OF ""	A property used by advanced users who want to do very specific searches. This property takes a SQL "FROM" clause. If this clause is set, the scope value is used as the FROM clause in the search. If both searchSubfolders and scope are set, scope is ignored and searchSubfolders takes precedence.

Property name	Read	Write	Default	Description
itemCount	Y	N		The number of items on the current page.
pageCount	Y	N		Total number of pages.

Table 13 lists the methods available for scripting the Web Forms View control.

Table 13 Methods

Method name	Parameters	Description
Update		Updates the entire view.
refreshPage		Refreshes the contents of just the current page.
gotoPage	Int	Changes the view based on the given page number.
nextPage		Changes to the next page, if one exists.
previousPage		Changes to the previous page, if one exists.
openMessage		Opens the currently selected message.
getSelectedXML		Returns XML for the currently selected rows.

Table 14 lists the events that the Web Forms View control can trigger.

Table 14 Events

Event name	When it occurs
OnSelectItem	When an item in the view is selected.
Onkeydown	When a key is pressed.
OnRefreshPage	When the page is refreshed.
OnPageChange	When the current page changes.
OnExpandGroup	When a group in a group-by view is expanded.
OnChangeSort	When the sort order is changed by the user.
Onerror	Whenever an error occurs.

The following sample view displays a list of messages and displays columns labeled **Selected**, **Importance**, **Item Type**, **Flag Status**, **Attachment**, **From**, **Subject**, **Received**, and **Size**.

Sample view (XML):

```
<view xmlns="http://schemas.microsoft.com/schemas/view"
xmlns:d="urn:uuid:c2f41010-65b3-11d1-a29f-00aa00c14882">
  <viewstyle style="font-size:large" cellpadding="0"
cellspacing="0">table-layout:fixed;width:100%;font-size:x-
small</viewstyle>
  <viewname>Messages</viewname>
  <column>
    <name>Selected</name>
    <prop>DAV:href</prop>
    <checkbox>1</checkbox>
    <sortable>0</sortable>
    <headerstyle>width:20px;text-align:center;vertical-
align:middle</headerstyle>
```

```

        <style>width:20px;text-align:center;vertical-
align:middle;font-weight:bold;background-color:window</style>
    </column>
    <column>
        <heading>Importance</heading>
        <prop>http://schemas.microsoft.com/exchange/x-priority-
long</prop>
        <type>i4</type>
        <bitmap>1</bitmap>
        <headerstyle width="2%">text-
align:center;cursor:hand</headerstyle>
        <style width="2%">text-align:center</style>
    </column>
    <column>
        <heading>Item Type</heading>
        <prop>http://schemas.microsoft.com/exchange/outlookmessageclass
</prop>
        <type>string</type>
        <bitmap>1</bitmap>
        <headerstyle width="3%">text-
align:center;cursor:hand</headerstyle>
        <style width="3%">text-align:center</style>
    </column>
    <column>
        <heading>FlagStatus</heading>
        <prop>http://schemas.microsoft.com/mapi/proptag/x10900003</pro
p>
        <type>i4</type>
        <bitmap>1</bitmap>
        <headerstyle width="3%">padding-left:3px;text-
align:center</headerstyle>
        <style width="3%">padding-left:3px;text-
align:center;cursor:hand</style>
    </column>

```

```

    <column>
        <heading>Attachment</heading>
        <prop>urn:schemas:httpmail:hasattachment</prop>
        <type>boolean</type>
        <bitmap>1</bitmap>
        <headerstyle width="2%">padding-left:3px;text-
align:center;cursor:hand</headerstyle>
        <style width="2%">padding-left:3px;text-
align center</style>
    </column>
    <column>
        <heading>From</heading>
        <prop>http://schemas.microsoft.com/mapi/sent_representing_name
</prop>
        <type>string</type>
        <headerstyle width="22%">padding-left:3px;text-
align:left;cursor:hand</headerstyle>
        <style width="22%">padding-left:3px;text-
align:left;background:green</style>
    </column>
    <column>
        <heading>Subject</heading>
        <prop>http://schemas.microsoft.com/mapi/subject</prop>
        <type>string</type>
        <style>padding-left:3px;text-align:left</style>
        <headerstyle>padding-left:3px;text-
align:left;cursor:hand</headerstyle>
    </column>
    <column>
        <heading>Received</heading>
        <prop>urn:schemas:httpmail:datereceived</prop>
        <type>dateTime.tz</type>
        <style width="22%">padding-left:3px;text-align:left</style>

```

```

        <headerstyle width="22%">text-
align left;cursor:hand</headerstyle>
    </column>
    <column>
        <heading>Size</heading>

    <prop>http://schemas.microsoft.com/mapi/proptag/x0e080003</prop>
    <type>i4</type>
    <style width="8%">padding-left:3px;text-
align right</style>
        <headerstyle width="8%">padding-left:3px;text-
align right;cursor:hand</headerstyle>
    </column>
    <filter>"http://schemas.microsoft.com/mapi/proptag/0x67aa000b" =
false AND "DAV:isfolder" = false</filter>
    <orderby>
        <order>
            <prop>urn:schemas:httpmail:datereceived</prop>
            <sort>DESC</sort>
        </order>
    </orderby>
    <headerstyle>background-
color:buttonface;cursor:hand</headerstyle>
    <rowstyle>background-color:blue;cursor:hand</rowstyle>
    <altrowstyle>background-color:hotpink</altrowstyle>
</view>    <BODY>
        <DIV id = "idMsgViewer"
            class = "msgViewer"
            url = "http://jfsserver/exchange/user1/Inbox/"
            viewDescriptorURL = "http://jfsserver/viewdesc3.xml"
            acceptLang = "en-us">
        </DIV>

```

Address Book Lookup View

The **Find Names** dialog box allows you to search through your organization's global address list (GAL) for a particular recipient. After you locate a person through Find Names, his or her name can be added directly to your e-mail message.

Figure 29 illustrates the address book **Find Names** dialog box as it is displayed by Internet Explorer 5 or later.

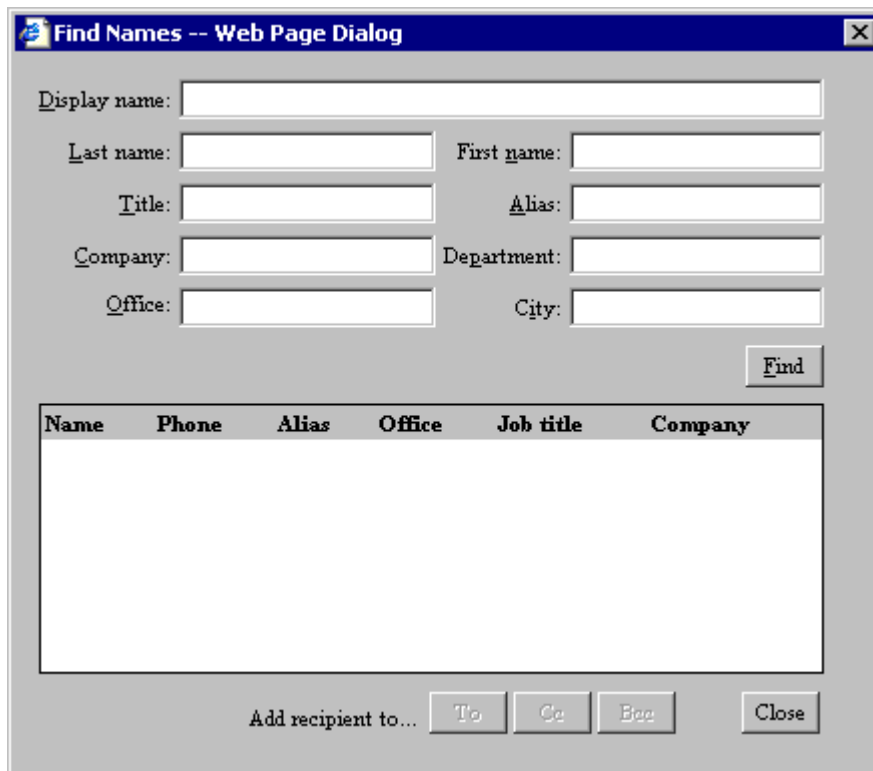


Figure 29 Address book **Find Names** dialog box

Customization

You cannot customize the address book **Find Names** dialog box.

Logoff Customization

Some Outlook Web Access deployment scenarios require control over the logoff process, thereby providing users a way to close the system and log off properly. Kiosks are a good example of why this type of control is important.

Figure 30 illustrates an example of the navigation bar with a **Log Off** button.



Figure 30 Navigation bar with **Log Off** button

Navigation Bar Customization

The following script is an example of how you can customize the navigation bar to include a prompt for logging off.

```

<script>
function logOutUser() {
    alert ("You are about to be prompted to close the
           browser window. Click yes to log out.");
    parent.window.close();
}
</script>

<html>
  <body onbeforeunload="logOutUser()">
    ...

  </body>
</html>

<BR>

  <A id="logoff" href="?Cmd=logoff"
  OnClick="top.g_fWarnOnLogOff=false" target="_parent" alt="Logoff"><IMG

```

```
class="nbImage" src="http://<exchangeserver>/exchweb/img/navbar-
logoff.gif" alt="Logoff"></A><DIV class="nbLabel">Logoff</DIV>
```

Outlook Web Access Logoff Page

Use the following sample HTML code to display a page that allows users to log off. Figure 31 illustrates the page that is generated by this code.

```
<HTML>
<HEAD>
<META HTTP-EQUIV="Content-Type" CONTENT="text/html; charset=iso-
8859-1">
<TITLE>Microsoft Outlook Web Access - Logoff</TITLE>
</HEAD>
<BODY scroll=auto bgcolor=#FFFFFF text=000000>
<table border="0" width="100%" height="100%">
  <tr valign="top" height="99%">
    <td>
      
      <font face="Verdana, Arial, Helvetica" size="2"><p>To complete the
log off process and prevent other users from opening your mailbox,<br>you
must <b>close</b> all browser windows and <b>exit</b> the browser
application.</p></font>
      <p><form><input onClick="javascript:window.close()" type="button"
value="Close" name="CloseWindow" style="font-family Verdana; font-
size:smaller; font-weight bold"></form></p>
    </td>
  </tr>
</table>
</body>
</html>
```

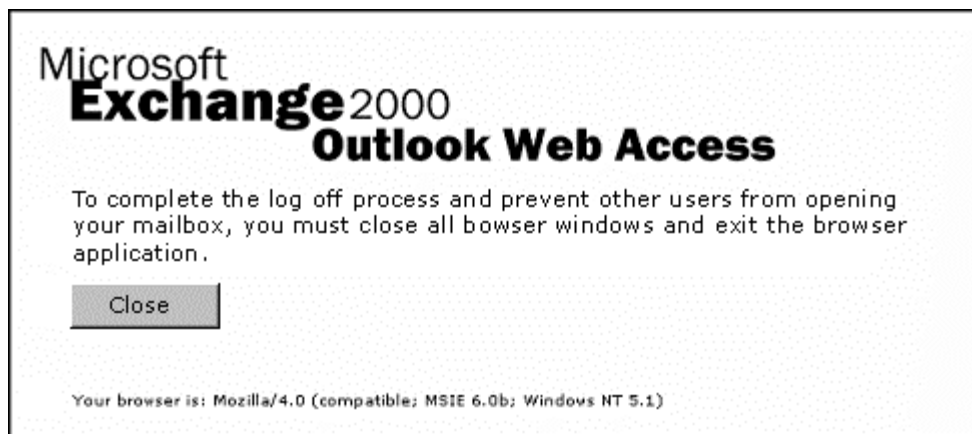


Figure 31 Sample log off screen

Options Page Customization

View the default Options page to learn how the Options page works or to use pieces of the code in a new Options page.

To view the default Options page code, see:

<http://exchangeserver/exchange/administrator/?cmd=options>

View the HTML source code for the page.

Options POST

The Options page updates the settings by sending an HTTP POST to the user mailbox with the following properties. If you wrap the fields in an HTML form, they are transmitted through HTTP as:

```
POST /exchange/username/ HTTP/1.1
Accept image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, */*
Referer http://server/exchange/username/?Cmd=options
Accept-Language en-ca
Content-Type application/x-www-form-urlencoded
Accept-Encoding gzip, deflate
User-Agent Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)
Host server
Content-Length 685
Connection Keep-Alive
Cache-Control no-cache
```

Cookie sessionid=08cf7dbc-f6f2-4ac3-b12a-8c5c871a9526,0x409

Authorization Basic cmVkbW9uZFxYXJpbWI6MXRvb011Y2g=

Cmd=options&OofState=0&OofReply=OOF+message&http%3A%2F%2Fschemas.microsoft.com%2Fexchange%2Fshortdateformat=yyyy-MM-dd&http%3A%2F%2Fschemas.microsoft.com%2Fexchange%2Flongdateformat=dddd%2C+MMMM+dd%2C+yyyy&http%3A%2F%2Fschemas.microsoft.com%2Fexchange%2Ftimeformat=h%3Amm+tt&http%3A%2F%2Fschemas.microsoft.com%2Fexchange%2Ftimezone=Pacific+Standard+Time&http%3A%2F%2Fschemas.microsoft.com%2Fexchange%2Fweekstartday=0&http%3A%2F%2Fschemas.microsoft.com%2Fexchange%2Fworkdaystarttime=60&http%3A%2F%2Fschemas.microsoft.com%2Fexchange%2Fworkdayendtime=1410&http%3A%2F%2Fschemas.microsoft.com%2Fexchange%2Ffirstweekofyear=3&http%3A%2F%2Fschemas.microsoft.com%2Fexchange%2Ffarcontactsfirst=0

Note Text-escaping was removed from the example above to make it more clear. The result is simply what happens with a submit on the HTML form that is the Options page.

The server-side code that handles the HTTP POST creates and sets the out-of-office information; otherwise, the code simply writes the schema properties illustrated in Table 15 as properties on the root folder (for example, POST /exchange/mailboxbase/).

Table 15 Schema properties and descriptions

Schema property	Description
Cmd=options	Activates Options post handler.
OofState	1=Turns out of office on, 0=Turns it off.
OofRepl	Out-of-office message. Up to 4 KB for the out-of-office message.
http://schemas.microsoft.com/exchange/shortdateformat	Short date format; uses y, m + d, other characters as separators, for example, yyyy-MM-dd.
http://schemas.microsoft.com/exchange/longdateformat	Long date format; same as above, for example, dddd, MMMM dd, yyyy.
http://schemas.microsoft.com/exchange/timeformat	Time format (use HH:mm TT); HH means leading zero, TT is A.M./P.M. indicator, other characters as separators.
http://schemas.microsoft.com/exchange/timezone	Time zone name. Corresponds to the CDO time zone structure. Exchange 2000 Service Pack 2 uses the Win32® National Language Support (NLS) time zone structures.
http://schemas.microsoft.com/exchange/weekstartday	Week start day. 0=Sunday, 6=Saturday
http://schemas.microsoft.com/exchange/workdaystarttime	Work day start time. Number of minutes after midnight, in half hour increments.
http://schemas.microsoft.com/exchange/workdayendtime	Work day end time. Number of minutes after midnight, in half hour increments.
http://schemas.microsoft.com/exchange/firstweekofyear	Week numbers in calendar. 0=Don't display, 1=Start counting Jan 1, 2=Start counting first 4-day week, 3=Start counting first full week.

Changing the Options Page

You can register a global form in the forms registry for the **options** command. For details about how to register global forms, see the "[Global Form Registration](#)" section earlier in this document.

Register the form globally against the **cmd** options and against the **mailfolder**, **folder**, **calendarfolder**, **journalfolder**, and **contactfolder** content classes. Figure 32 illustrates the result of changing the **options** command to display a new Options page.

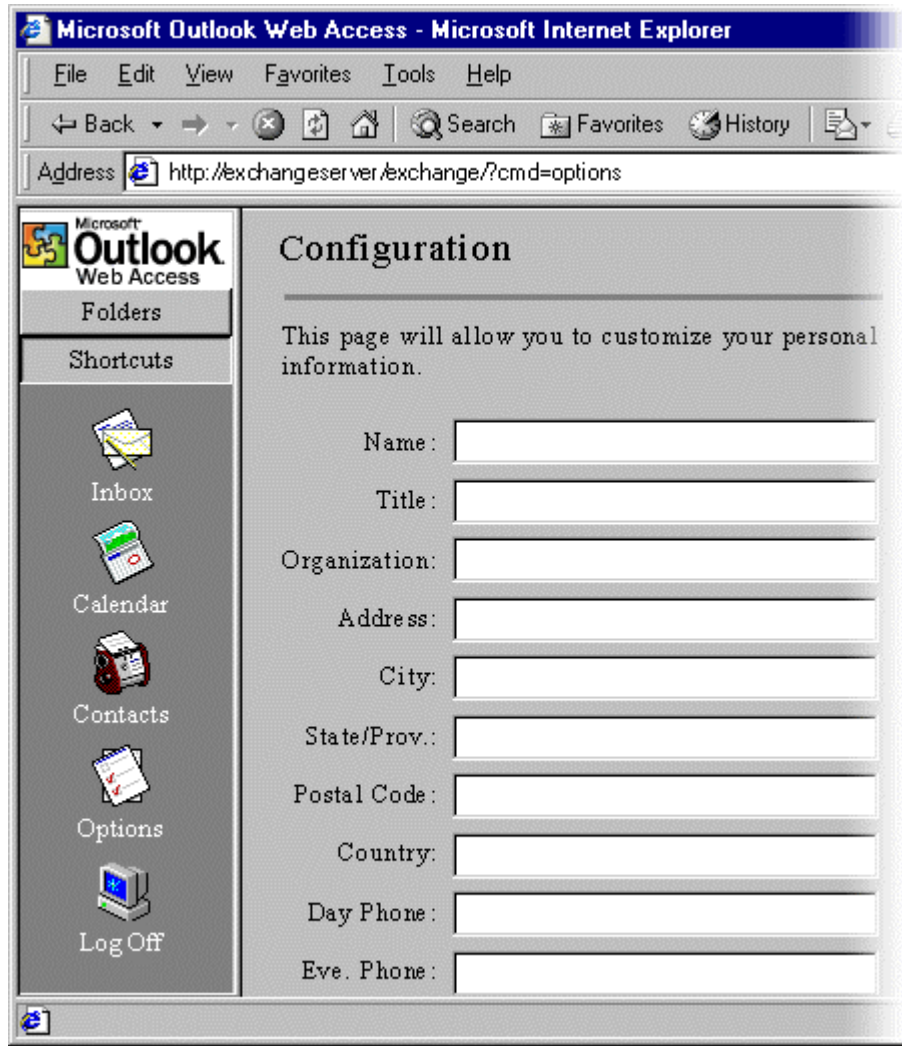


Figure 32 Sample options screen customization

Disabling the Multimedia Button on the Options Page

Warning Using Registry Editor incorrectly can cause serious problems that may require you to reinstall your operating system. Microsoft cannot guarantee that problems resulting from the incorrect use of Registry Editor can be solved. Use Registry Editor at your own risk. For information about

how to do this, see the Regedit.exe or Regedt32.exe online documentation. You should back up the registry before you edit it. If you are running Windows NT or Windows 2000, you should also update your Emergency Repair Disk (ERD).

To disable the Multimedia button on an Outlook Web Access server

1. Start Registry Editor (Regedt32.exe): click **Start**, click **Run**, type **regedt32**, and then click **OK**.
2. Locate the following registry key:
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\MSExchangeWEB\Outlook Web Access
3. On the **Edit** menu, click **Add Value**, and then add the following registry value:
Value Name DisableMultimedia
Data Type DWORD
Value 1
4. Click **Start**, point to **Program Files**, click **Administrative Tools**, and then click **Services**.
5. Right-click **World Wide Web Publishing Service (W3SVC)**, click **Restart**, and then click **OK**.

Help Files

Outlook Web Access provides default help files for users. The help files that are displayed depend on the language, what the user is doing, and the browser type.

Outlook Web Access stores the help files in the Ie5 folder for browsers running Internet Explorer 5 or later. Outlook Web Access stores the files in the Ie3 folder for non-Microsoft browsers and for Internet Explorer 3 or earlier (see Figure 33).

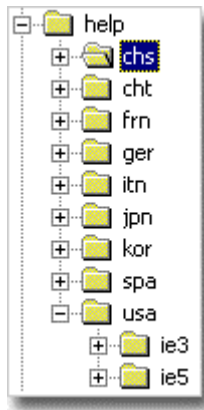


Figure 33 Folder structure for help files

Note The location of the help files is language specific.

Customization

To customize these help files, edit them in Microsoft FrontPage® or Microsoft Visual InterDev®.

Other Outlook Web Access Customizations

This section describes additional customizations that are available with Outlook Web Access. Read this section to learn how to hide and remove folders, remove access to public folders, and change the default content expiration time.

Hiding Folders

You can use either of the following methods to hide folders from users:

- Set the **DAV:ishidden** property on a folder
- Set the **DAV:defaultdocument** property on a folder

Note Both of these methods hide a folder to the casual user, but they do not completely remove or stop access to the data contained in those folders.

Set the DAV:ishidden Property

You can hide folders by setting the **DAV:ishidden** property. The following WebDAV script gives an example of this method.

```
PROPPATCH /exchange/administrator/inbox HTTP/1.1
```

```
Content-type text/xml
```

```
<?xml version="1.0"?>
<a:propertyupdate xmlns:a="DAV:">
  <a:set>
    <a:prop>
      <a:ishidden>1</>
    </>
  </>
</>
```

Note You can also use the Exchange Explorer tool to set the **ishidden** property on a folder.

Figure 34 illustrates an example of the navigation bar with all folders hidden except Inbox, Drafts, Deleted Items, Outbox, and Sent Items.

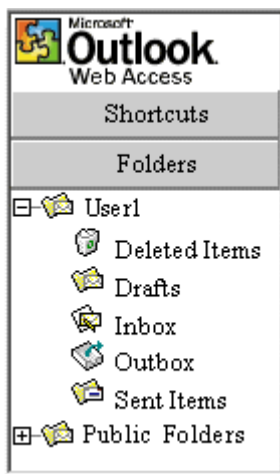


Figure 34 Navigation bar with hidden folders

Note Users can still navigate directly to the hidden folders using a direct URL. For example, <http://exchangeserver/exchange/user/calendar/?cmd=contents> still displays the Calendar Folder.

Hiding Folders Using the `DAV:defaultdocument` Property

You can also hide a folder by setting the **defaultdocument** property on the folder to another URL. The following WebDAV script gives an example of this method.

```
PROPPATCH /exchange/administrator/inbox HTTP/1.1
Content-type text/xml

<?xml version="1.0"?>
<a:propertyupdate xmlns:a="DAV:">
  <a:set>
    <a:prop>

    <a:defaultdocument>http://exchange/administrator/inbox/page.htm</>

  </>
</>
</>
```

Note Do not set this property on the root of a mailbox because the property interferes with core Outlook Web Access functionality.

Removing Folders Completely

You cannot directly remove Exchange folders such as Calendar, Tasks, and Inbox by using Outlook Web Access or the Outlook client. However, you can remove these folders by using code or tools such as Exchange Explorer.

Exchange 2000 Service Pack 2 (SP2) includes the ability to segment the Outlook Web Access UI, thereby allowing administrators to set a tier value to hide segments of the Web client UI.

Removing Public Folders Access Completely

To completely remove public folder access, use Exchange System Manager to remove access to the public virtual directory. Figure 35 illustrates the settings for the public folder properties. In the folder Properties dialog box, clear all the **Access Control** settings, and set **Execute Permissions** to **None**.

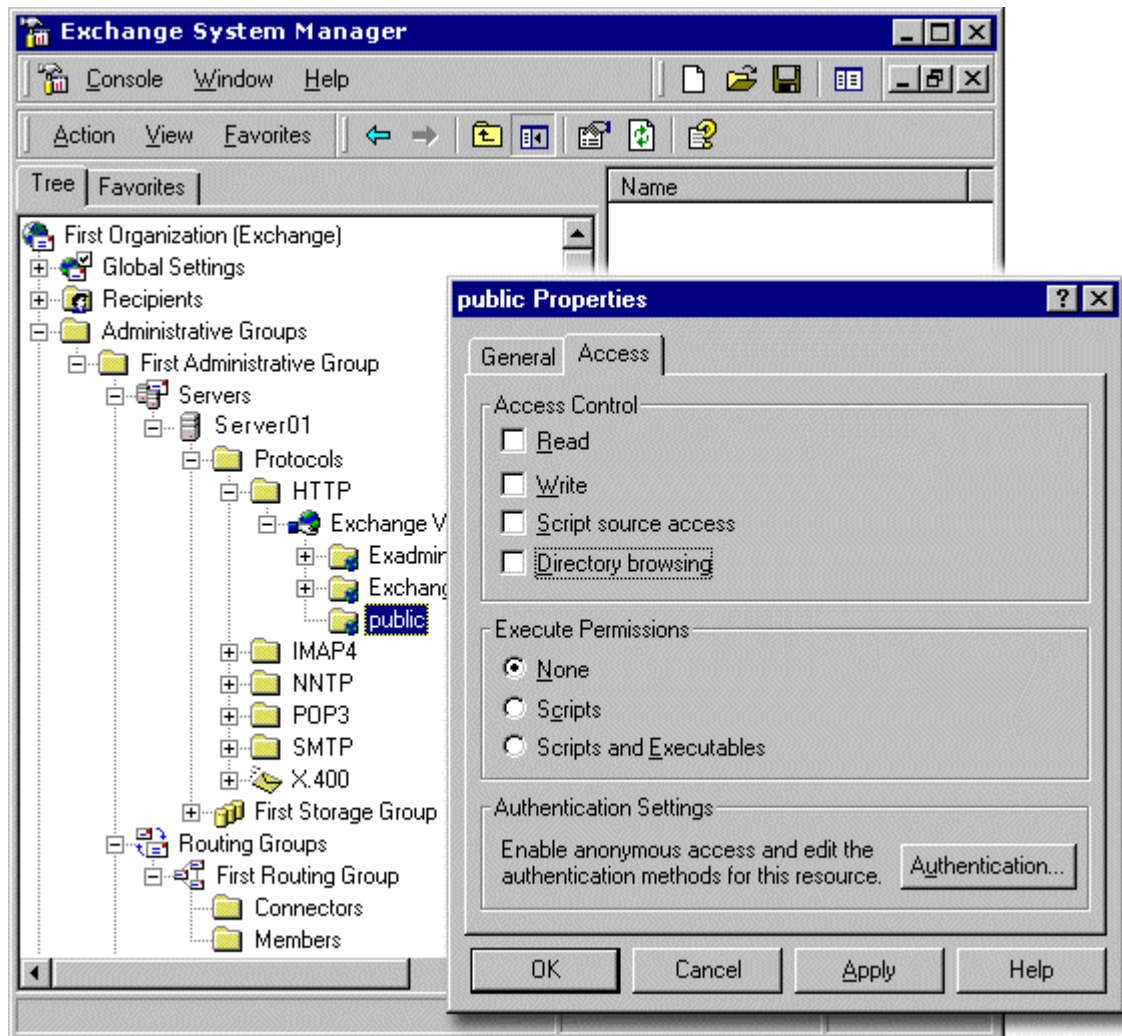


Figure 35 Exchange System Manager and removal of public folder virtual directory

Note Changing this property removes the Public Folder tree from the navigation bar TreeView control. All other access to the public folders results in a 403 Forbidden error. This change also breaks the free and busy schedule lookup.

Changing the Default Content Expiration Time

By default, the virtual IIS directory (/exchweb) has its content expiration time set to one day. To change this setting, use the Internet Information Services MMC to select the directory and view its properties. One reason you may want to change the default content expiration time is because you are making changes to Outlook Web Access components and want to ensure that the content expires faster than usual. If the rate of change is low on your pages, extend the content expiration time as long as possible. Extending the content expiration time is of greatest advantage to Internet Explorer 5 and later clients because these clients cache the .htc and .js files longer than other clients. Figure 36 illustrate the content expiration settings in IIS MMC.

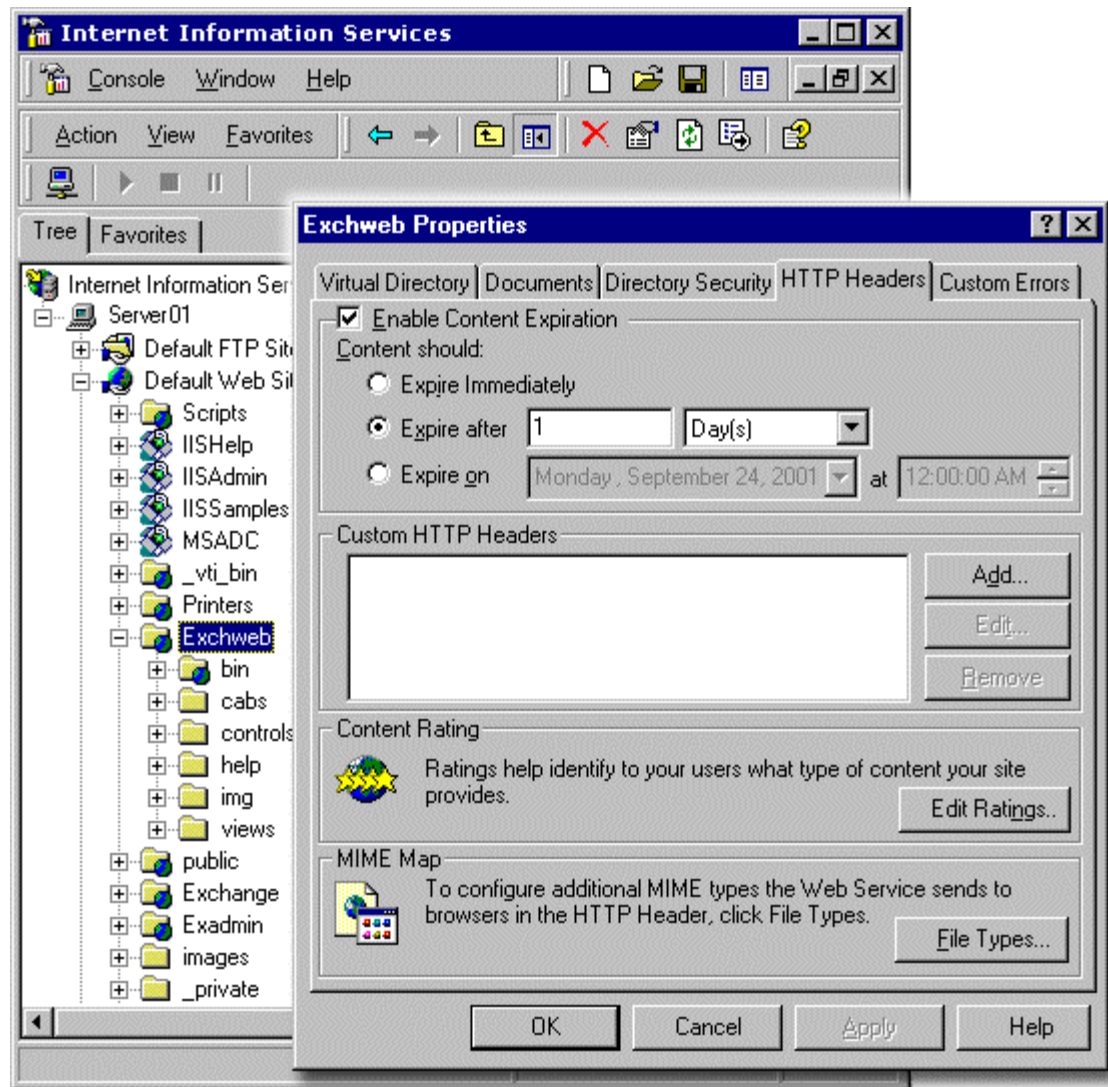


Figure 36 Content expiration settings in IIS MMC

Summary

Outlook Web Access is a tightly integrated component of Exchange 2000. Because of the many architectural changes in Exchange 2000, Outlook Web Access now provides significant scalability and increased levels of functionality. Exchange 2000 is built upon the Exchange store, an entirely new way to store data with many new access methods (the most profound method being direct URL access to data items and the Exchange store). A good understanding of the new Exchange store provides the knowledge you need to understand and modify the default behaviors and abilities of Outlook Web Access.

For more information: <http://www.microsoft.com/exchange/>

Did this paper help you? Please give us your feedback. On a scale of 1 (poor) to 5 (excellent), how would you rate this paper?

<mailto:exchdocs@microsoft.com?subject=Feedback: Customizing Microsoft Outlook Web Access>



Appendix A: Additional Resources

Web Sites and White Papers

- Microsoft Exchange 2000 Server
<http://www.microsoft.com/exchange/>
- MSDN Exchange Development
<http://msdn.microsoft.com/exchange>
- Web Storage System Forms: The Forms Registry
http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnmes2k/html/wssformsreg_v4.asp
- Maximizing the Performance of your Active Server Pages
<http://msdn.microsoft.com/workshop/server/asp/maxperf.asp>
- Exchange SDK Development Tools
<http://msdn.microsoft.com/downloads/default.asp?url=/downloads/sample.asp?url=/msdn-files/027/001/740/msdncompositedoc.xml>
- Developing Applications for the Exchange Storage System and Outlook Web Access
<http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnmes2k/html/wssdevaccess.asp>

Microsoft Knowledge Base Articles

The following Microsoft Knowledge Base articles are available on the Web at <http://support.microsoft.com/>.

- [Q259221 XADM: Security Tab Not Available on All Objects in System Manager](#)
- [Q262054 XADM: How to Get Service Account Access to All Mailboxes in Exchange 2000](#)
- [Q297121 XWEB: How to Hide the Change Password Button on the Outlook Web Access Options Page](#)

Appendix B: Code for the Navigation Bar

This appendix provides code that can be added to the navigation bar, both for rich clients (Internet Explorer 5 and later) and non-Internet Explorer 5 clients.

Non-Internet Explorer 5 Browsers (United States English)

```
<META HTTP-EQUIV="Content-Type" CONTENT="text/html;  
CHARSET=windows-1252">
```

```

<META HTTP-EQUIV="Pragma" CONTENT="no-cache">
<META HTTP-EQUIV="Expires" CONTENT="0">
<!--Copyright (c) 2000 Microsoft Corporation. All rights reserved.-->
<!--CURRENT FILE== "NON-IE5" "NON-WIN32" navbar -->
<HTML>
<HEAD>
<TITLE>
None
</TITLE>
<STYLE type="text/css">
A:link      { color  FFFFFFF;text-decoration  none}
      A:visited  { color  FFFFFFF;text-decoration  none}
      A:active   { color  FFFFFFF;text-decoration  none}
</style>
<BASE href="http://exchangeserver/exchange/user/" />
</HEAD>
<BODY bgcolor="666666" topmargin="0" leftmargin="0" vlink="0"
alink="0" link="0">
<TABLE bgcolor="666666" width="100%" border="0">
<TR>
<TD bgcolor="FFFCE" valign="top" align="center" height="38">
<IMG src="http://exchangeserver/exchweb/img/logo-small.gif" width="91"
height="38" border="0">
</TD>
</TR>
<TR>
<TD valign="top" align="center" height="50"> <A
href="./Inbox/?Cmd=contents&Page=1" target="viewer" style='text-
decoration  none'>
<IMG src="http://exchangeserver/exchweb/img/navbar-inbox.gif"
width="36" height="36" alt="Go to inbox" border="0"><BR> <FONT
color="white" size="2">Inbox</FONT></A>
</TD>
</TR>

```

```

<TR>

<TD valign="top" align="center" height="50"> <A
href="/Calendar/?Cmd=contents" target="viewer" style='text-
decoration none'>

<IMG src="http://exchangeserver/exchweb/img/navbar-calendar.gif"
width="36" height="36" alt="Go to calendar" border="0"><BR> <FONT
color="white" size="2">Calendar</FONT></A>

</TD>

</TR>

<TR>

<TD valign="top" align="center" height="50"> <A
href="/Contacts/?Cmd=contents" target="viewer" style='text-
decoration none'>

<IMG src="http://exchangeserver/exchweb/img/navbar-contacts.gif"
width="36" height="36" alt="Go to contacts" border="0"><BR> <FONT
color="white" size="2">Contacts</FONT></A>

</TD>

</TR>

<TR>

<TD valign="top" align="center" height="50"> <A
href="http://exchangeserver/exchange/user/?Cmd=contents&ShowFolders=
1" target="viewer" style='text-decoration none'>

<IMG src="http://exchangeserver/exchweb/img/navbar-public.gif"
width="36" height="36" alt="Go to my folders" border="0"><BR> <FONT
color="white" size="2">Folders</FONT></A>

</TD>

</TR>

<TR>

<TD valign="top" align="center" height="50"> <A
href="http://exchangeserver/exchange/user/?Cmd=options" target="viewer"
style='text-decoration none'>

<IMG src="http://exchangeserver/exchweb/img/navbar-options.gif"
width="36" height="36" alt="Go to options" border="0"><BR> <FONT
color="white" size="2">Options</FONT></A>

</TD>

</TR>

</TABLE>

</BODY>

```

```
</HTML>
```

Internet Explore 5 or Later Browsers (United States English)

```
<META HTTP-EQUIV="Content-Type" CONTENT="text/html; CHARSET=utf-8">
```

```
<META HTTP-EQUIV="Pragma" CONTENT="no-cache">
```

```
<META HTTP-EQUIV="Expires" CONTENT="0">
```

```
<!--Copyright (c) 2000 Microsoft Corporation. All rights reserved.-->
```

```
<!--CURRENT FILE== "IE5" "WIN32" navbar -->
```

```
<HTML>
```

```
<?XML:NAMESPACE PREFIX="WM"/>
```

```
<HEAD>
```

```
<STYLE>
```

```
WM\:DROPMENU
{behavior:url(http://exchangeserver/exchweb/controls/dropmenu.htc)}
```

```
</STYLE>
```

```
<LINK rel="stylesheet" type="text/css"
HREF="http://exchangeserver/exchweb/controls/navbar.css">
```

```
<SCRIPT language="javascript">
```

```
var g_fPublicOnly          = false;
```

```
var g_fPublicRoot          = false;
```

```
var g_szUserDisplayName    = "user";
```

```
var g_szUserBase           = "http://exchangeserver/exchange/user" + "/";
```

```
var g_szFolder              = ".";
```

```
var g_szVirtualRoot        = "http://exchangeserver/exchweb";
```

```
var g_szImagePath          = g_szVirtualRoot + "/img/";
```

```
var g_szPublicFolderUrl    = "http://exchangeserver/public/";
```

```
var g_szPublicFoldersDN    = "Public Folders";
```

```
</SCRIPT>
```

```
<script>
```

```
var L_RenameCurrentFolder_Text = "Rename the current folder to:"
```

```

var L_ConfirmDelete_Text           = "Are you sure you want to
delete the current folder?"

var L_errCommunicationfailure_ErrorMessage = "Couldn't complete
request, communications error."

var L_errLogonFailure_ErrorMessage   = "Login failed or cancelled."
var L_errAccessDenied_ErrorMessage  = "Access denied."

var L_errPreconditionfailure_ErrorMessage = "Operation failed. An item
with the same name already exists in the destination folder."

var L_errInternalServerError_ErrorMessage = "An internal server error
occurred."

var L_errBadGateway_ErrorMessage     = "Outlook Web Access does
not currently support this type of move/copy action."

var L_ErrorIllegalCharacter_Text     = "The folder name entered
contains illegal characters."

var L_errServiceNotAvailable_ErrorMessage = "The service is no longer
available."

var L_errCrossDomainAccessDenied_ErrorMessage = "Access is denied.
Your browser security settings will not allow access to this folder from this
page."

var L_ErrorItemNotFound_ErrorMessage = "The item could not be
found. It may have been deleted."

</script>
<SCRIPT
src="http://exchangeserver/exchweb/controls/navbar.js"></SCRIPT>
<BASE href="http://exchangeserver/exchange/user/">
</HEAD>
<BODY id="outbar_mainbody" topmargin="0" leftmargin="0"
oncontextmenu="return(event.ctrlKey);" onselectstart="return(false);"
scroll="no" style='background-color:appworkspace;'>
<TABLE id=nbTableMain class='nbTableMain' style="height:100%;" rows=2
cols=1 cellpadding=0 cellspacing=0 >
<TR><TD class='treeBrand' >
      <DIV class='treeOFLOW'><IMG
src="http://exchangeserver/exchweb/img/logo-ie5.gif" style="padding:0;"
border=0></DIV></TD></TR>
<TR height="100%"><TD>
<TABLE class='nbTable' rows=4 cols=1 cellpadding=0 cellspacing=0 >
<TR><TD id="show_navbar" TABINDEX=0 class="nbFlybar" NOWRAP
onkeydown="flybar_keydown()" onclick="ToggleTab(this.id)">

```

```

        <DIV class='treeOFLOW'>Shortcuts</DIV>
    </TD></TR>
    <TR style="height:100%"><TD id="idOutbarpane" valign=top style="TEXT-
    ALIGN:center;">
        <A id="inbox" href="./Inbox/?Cmd=contents&Page=1"
        target="viewer" alt="Go to inbox"><IMG class="nbImage"
        src="http://exchangeserver/exchweb/img/navbar-inbox.gif" alt="Go to
        inbox"></A><DIV class="nbLabel">Inbox</DIV>
        <BR>
        <A id="calendar" href="./Calendar/?Cmd=contents" target="viewer"
        alt="Go to calendar"><IMG class="nbImage"
        src="http://exchangeserver/exchweb/img/navbar-calendar.gif" alt="Go to
        calendar"></A><DIV class="nbLabel">Calendar</DIV>
        <BR>
        <A id="contacts" href="./Contacts/?Cmd=contents" target="viewer"
        alt="Go to contacts"><IMG class="nbImage"
        src="http://exchangeserver/exchweb/img/navbar-contacts.gif" alt="Go to
        contacts"></A><DIV class="nbLabel">Contacts</DIV>
        <BR>
        <A id="options" href="?Cmd=options" target="viewer" alt="Go to
        options"><IMG class="nbImage"
        src="http://exchangeserver/exchweb/img/navbar-options.gif" alt="Go to
        options"></A><DIV class="nbLabel" >Options</DIV>
    </TD></TR>
    <TR style="height:1.5em;" ><TD id="show_folders" TABINDEX=0
    class="nbFlybar" NOWRAP onkeydown="flybar_keydown()"
    onclick="ToggleTab(this.id)">
        <DIV class='treeOFLOW'>Folders</DIV>
    </TD></TR>
    <TR><TD valign=top id=treeProgress NOWRAP class="nbTreeProgress"
    style="display:none;">
        <SPAN id=idLoading style="overflow:hidden">Loading...</SPAN>
    </TD></TR>
</TABLE>
</TD></TR>
</TABLE>
<DIV class='treeFolder' id="objTree" style="display:none;"
    onpickfolder = "objTree_OnPickFolder();"

```

```

        onaddhierarchy                                =
"objTree_OnAddHierarchy();"

        onfolderoperationcomplete =
"objTree_OnFolderOperationComplete();"

        onaddnodes                                    =
"objTree_OnAddNodes();"

        imagePath                                      =
"http://exchangeserver/exchweb/img/"

        acceptLang = "en-us">
</DIV>

<WM:DROPMENU id=idTreeMenu HOLDSTYLE=true class="nbDropMenu"
menuName="treeMenu" >
    <WM:MENURROW>
        <WM:MENUIITEM type="default">
            <WM:MENUIITEMIMG/>

            <WM:MENUIITEMCAPTION>Open</WM:MENUIITEMCAPTION>

            <WM:MENUIITEMSCRIPT>idTreeMenu.hide();MenuSelect("Open");</
WM:MENUIITEMSCRIPT>
        </WM:MENUIITEM>
    </WM:MENURROW>
    <WM:MENURROW>
        <WM:MENUIITEM type="default">
            <WM:MENUIITEMIMG/>
            <WM:MENUIITEMCAPTION>Open in
New...</WM:MENUIITEMCAPTION>

            <WM:MENUIITEMSCRIPT>idTreeMenu.hide();MenuSelect("OpenNewWi
ndow");</WM:MENUIITEMSCRIPT>
        </WM:MENUIITEM>
    </WM:MENURROW>
    <WM:MENUDIVIDER/>
    <WM:MENURROW>
        <WM:MENUIITEM type="default">
            <WM:MENUIITEMIMG/>

```

```

    <WM:MENUITEMCAPTION>Move</WM:MENUITEMCAPTION>

    <WM:MENUITEMSCRIPT>idTreeMenu.hide();MenuSelect("Move");</
WM:MENUITEMSCRIPT>
        </WM:MENUITEM>
    </WM:MENUROW>
<WM:MENUROW>
    <WM:MENUITEM type="default">
        <WM:MENUITEMIMG/>

    <WM:MENUITEMCAPTION>Copy</WM:MENUITEMCAPTION>

    <WM:MENUITEMSCRIPT>idTreeMenu.hide();MenuSelect("Copy");</W
M:MENUITEMSCRIPT>
        </WM:MENUITEM>
    </WM:MENUROW>
<WM:MENUDIVIDER/>
<WM:MENUROW>
    <WM:MENUITEM type="default" state="inactive">
        <WM:MENUITEMIMG/>

    <WM:MENUITEMCAPTION>Delete</WM:MENUITEMCAPTION>

    <WM:MENUITEMSCRIPT>idTreeMenu.hide();MenuSelect("Delete");</
WM:MENUITEMSCRIPT>
        </WM:MENUITEM>
    </WM:MENUROW>
<WM:MENUROW>
    <WM:MENUITEM type="default">
        <WM:MENUITEMIMG/>

    <WM:MENUITEMCAPTION>Rename</WM:MENUITEMCAPTION>

    <WM:MENUITEMSCRIPT>idTreeMenu.hide();MenuSelect("Rename");
</WM:MENUITEMSCRIPT>
        </WM:MENUITEM>

```

```

</WM:MENURROW>
<WM:MENURROW>
    <WM:MENUITEM type="default">
        <WM:MENUITEMIMG/>
        <WM:MENUITEMCAPTION>New
Folder</WM:MENUITEMCAPTION>

        <WM:MENUITEMSCRIPT>idTreeMenu.hide();MenuSelect("New
Folder");</WM:MENUITEMSCRIPT>
    </WM:MENUITEM>
</WM:MENURROW>
</WM:DROPMENU>
</BODY>
</HTML>

```

Appendix C: MailFolder Toolbar Code

Use the following code to customize toolbars in Outlook Web Access mail folders; use for rich clients (Internet Explorer 5 or later) only.

Internet Explorer 5 or Later (United States English)

```

<TABLE class="cbToolbar" id="idToolbar" cellpadding='0' cellspacing='0'>
  <TR>
    <TD class="tbDivider" NOWRAP>
      <A class="tbButton" name="cbButton" id="newitem"
overelements="newmenu" HREF="#"
          Title="New"><IMG align="absmiddle"
          src="http://exchangeserver/exchweb/img/tool-
newmail.gif">New</A><A class="tbButton"
          name="cbButton" id="newmenu" sticky=true
overelements="newitem" HREF="#" submenu='openMessageMenu'>
          <IMG align="absmiddle"
          src="http://exchangeserver/exchweb/img/tool-downarrow.gif"
          style='width:10px;'>
        </A>
    </TD>

```

```
<TD NOWRAP><A class="tbButton" name="cbButton" id="reply"
  HREF="#" Title="Reply"><IMG align=absmiddle
  src="http://exchangeserver/exchweb/img/tool-reply.gif">Reply</A></TD>
```

```
<TD NOWRAP><A class="tbButton" name="cbButton" id="replyall"
  HREF="#" Title="Reply to all"><IMG align=absmiddle
  src="http://exchangeserver/exchweb/img/tool-reply_all.gif">Reply to
  all</A></TD>
```

```
<TD class="tbDivider" NOWRAP> <A class="tbButton"
  name="cbButton" id="forward" HREF="#" Title="Forward"><IMG
  align=absmiddle src="http://exchangeserver/exchweb/img/tool-
  forward.gif">Forward</A></TD>
```

```
<TD NOWRAP><A class="tbButton" name="cbButton"
  id="checkmessages" HREF="#" Title="Check for New Messages"><IMG
  align=absmiddle src="http://exchangeserver/exchweb/img/tool-
  refresh.gif"></A></TD>
```

```
<TD NOWRAP><A class="tbButton" name="cbButton"
  id="movecopy" HREF="#" Title="Move/Copy"><IMG align=absmiddle
  src="http://exchangeserver/exchweb/img/tool-move.gif"></A> </TD>
```

```
<TD NOWRAP><A class="tbButton" name="cbButton" id="delete"
  HREF="#" Title="Delete"><IMG align=absmiddle
  src="http://exchangeserver/exchweb/img/tool-delete.gif"></A></TD>
```

```
<TD class="tbDivider" NOWRAP><A class="tbButton"
  name="cbButton" id="emptydeleteditems" HREF="#" Title="Empty Deleted
  Items Folder"><IMG align=absmiddle
  src="http://exchangeserver/exchweb/img/tool-empty.gif"></A></TD>
```

```
<TD NOWRAP><A class="tbButton" name="cbButton"
  id="previewpane" HREF="#" sticky=true Title="Show/Hide Preview
  Pane"><IMG align=absmiddle
  src="http://exchangeserver/exchweb/img/tool-preview.gif"></A></TD>
```

```
<TD class="tbDivider" NOWRAP><A class="tbButton"
  name="cbButton" id="addressbook" HREF="#" Title="Address Book"><IMG
  align=absmiddle src="http://exchangeserver/exchweb/img/tool-
  addressbook.gif"></A></TD>
```

```
<TD NOWRAP style="padding-left:5px;padding-right:5px;">
```

```
<U>V</U>iew
```

```
</TD><TD id="tdViewList" style="width:100%;">
```

```
<SELECT name="viewSelect" style="display:none;"
  onChange="changeView(viewSelect.value)" ACCESSKEY="v">
```

```
<OPTION selected
  VALUE="Messages">Messages</OPTION>
```

```
<OPTION VALUE="Unread%20Messages">Unread Messages</OPTION>
```

```
<OPTION VALUE="By%20Sender">By Sender</OPTION>
```

```

<OPTION VALUE="By%20Subject">By Subject</OPTION>
<OPTION VALUE="By%20Conversation%20Topic">By Conversation
Topic</OPTION>
<OPTION VALUE="Unread%20By%20Conversation%20Topic">Unread By
Conversation Topic</OPTION>
<OPTION VALUE="Sent%20To">Sent To</OPTION>

</SELECT>
</TD>
<TD width=100% ALIGN=RIGHT>
    <A class="tbButton" name="cbButton" id="help" href="#"
title="Help">
        <IMG align=absmiddle
src="http://exchangeserver/exchweb/img/tool-help.gif">
        <SPANid="idHelpLinkText"
style='display:none;'>http://exchangeserver/exchweb/help/USA/ie5/notevie
w.htm</SPAN>
    </A>
</TD>
</TR>
</TABLE>

```

Appendix D: Outlook Web Access Customization Example

Figure D1 illustrates a customized Outlook Web Access mailbox and message. The button colors and font are modified, and banner advertisements are added to the forms.

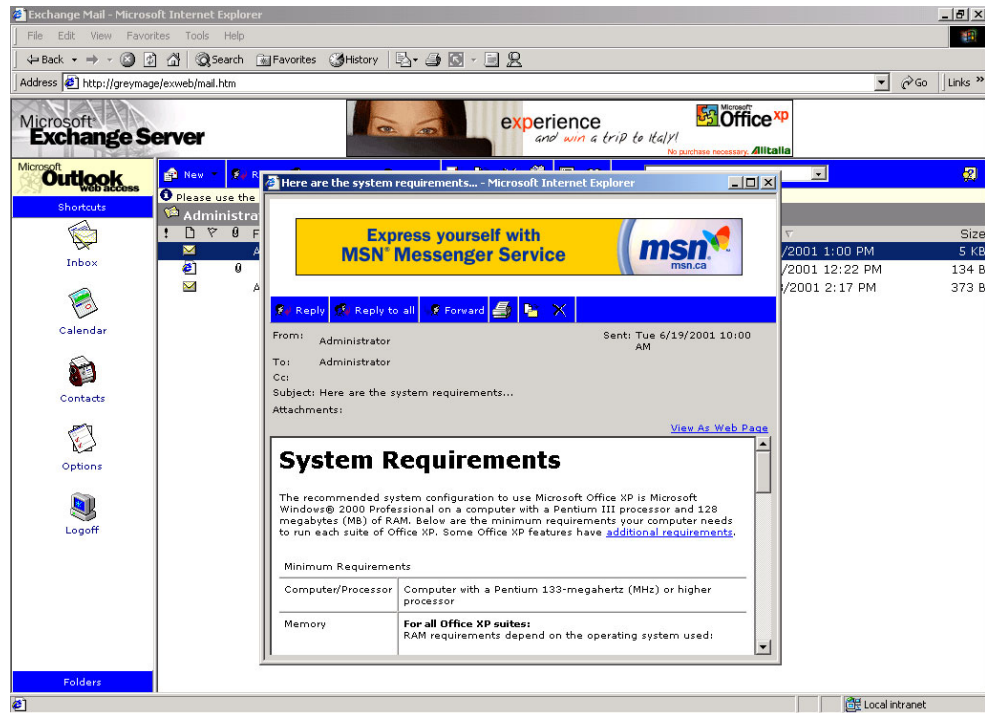


Figure D1 Example of customization

Appendix E: Style Sheet Sections

Use the Function column in each of the following tables to keep track of changes you make to the Outlook Web Access CSS styles. Each table lists the styles defined in the indicated CSS file.

Calendarprint.css

Table E1 lists the CSS styles defined in the Calendarprint.css file. These styles control the appearance of calendars printed using Outlook Web Access.

Table E1 Calendar.css styles

Styles	Function
.divCalendarViewer	
.divCalendarRightPane	
.dailydiv1	

Exttoolbar.css

Table E2 lists the CSS styles defined in the Exttoolbar.css file. These styles control the appearance of toolbars in Outlook Web Access.

Table E2 Exttoolbar.css styles

Styles	Function
.EXTOOLBAR	
.toolbar	

.formatbar	
.tbImg	
.tbDivider	
.tbButton	
.fbButton	

Freebusy.css

Table E3 lists the CSS styles defined in the Freebusy.css file. These styles control the appearance of information shown on the meeting attendee free and busy Outlook Web Access page.

Table E3 Freebusy.css styles

Styles	Function
.fb_behavior	
.fb_splitter_behavior	
.fbtime	
.fb_span_div	
.fb_color_div	
.fb_cell	
.fb_recip_col	
.fb_recip_row	
.fb_flags	
.fb_hdrs	
.fb_time	
.fb_title	
.fbtable	
.fbHeader	
.fbLegend	
.fbSlider	
.fbbody	
.fbrecips	
.cborder	
.cell	
.cell1	
.row	
.recipBg	
.tr	
.recipHeader, .recip	
.recip	
.recipData	
.attendees	
.busy, .tent, .oof, .unknown	
.busy, .busyClr	
.tent, .tentClr	
.oof, .oofClr	
.unknownClr	

Styles	Function
.unknown	
.fbLegend2clr, .fbLegend3clr, .fbLegend4clr, .fbLegend5clr	
.fbLegend2txt, .fbLegend3txt, .fbLegend4txt, .fbLegend5txt	

Olprint.css

Table E4 lists the CSS styles defined in the Olprint.css file. These styles control the appearance of material printed using Outlook Web Access.

Table E4 Olprint.css styles

Styles	Function
.cbToolBar, .trButtons, .divCalendarRightPane	
.fldLabel	
.fldText	
.fldInput	
.btnRecip	
.MSGBODY	
.txtBody	
.viewBody	
.MSGBODYFRM	
.tabdiv	
.tabdivHide	
.grpLEFT, .grpRIGHT, .grpTop, .grpBottom	
.calViewContainer1, .divCalendarView	
.dailyRowContainer	
.tblMsgBody	
.cbToolBar, .trButtons, .divCalendarRightPane	

Olstyle.css

Table E5 lists the CSS styles defined in the Olstyle.css file. These styles control the appearance of many areas in Outlook Web Access.

Table E5 Olstyle.css styles

Styles	Function
.noteBody, .previewBody	
.noteBody	
.btnRecip	
.btnRecipDiv	
.mainBody	
.msgViewer	
.msgPreview	
.msgPreview, .msgViewer	

Styles	Function
.msgHeader	
.msgHdrCN1	
.postHdr1	
.msgHdr_Post, .msgHdr_Note, .msgHdr_Appt	
.msgHdrP1	
.msgHdrN1	
.msgHdrN3	
.msgHdrP3	
.msgHdrN4, .msgHdrP4	
.previewHdr	
.preHdrN1, .preHdrC1, .preHdrP1, .preHdrR1, .preHdrA1	
.preHdrN3, .preHdrC3, .preHdrP3, .preHdrR3, .preHdrA3	
.preHdrN5, .preHdrA5, .preHdrP5	
.fldLabel	
.fldText	
.fldTextRecip	
.cbToolbar	
.tbDivider	
.tbButton, .tbButtonDown, .tbButtonRaise	
.tbButtonDown	
.cbSubmenuItem	
.tbDisabled	
.fbButton	
.cbSubmenu	
.cbSubmenuItem	
.AttachTable	
.AttachmentWell	
.viewBody	
.tblMsgBody	
.msgPreviewBody	
.txtBody	
.MSGBODY	
.MSGBODYFRM	
.FORMATBAR	
.rnStatusRow	
.rnStatusImg	
.rnStatusBar, .viewStatusBar, .rnOnlineBar	
.viewStatusBar	
.viewStatusImg	

Styles	Function
.rnOnlineImg	
.grpLEFT	
.grpRIGHT	
.grpTop	
.grpBottom	
.dailyRowContainer	
.dailyRow	
.dailyTimeNode	
.dailyTimeCaption	
.dailyAMPMContainer	
.dailyAMPNode	
.AllDayCell	
.dailyAllDayAppointment	
.divDailyScrolling	
.spanDailyTimesBlock	
.spanDailyAppointmentsBlock	
.monthlyViewContainer	
.weeklyViewContainer	
.dailyAllDayWell	
.appointment	
.dailyAppt	
.appointmentGrabBar	
.appointmentTopBar, .appointmentBottomBar	
.appointmentBody	
.weeklyNodeHeader	
.weeklyNodeBody	
.MonthlyAppointment	
.alldayMonthlyAppointment	
.monthlyviewslot	
.monthlyDayHeader	
.calctrlTitle	
.calctrlTable, .calctrlTitle	
.calctrlRow	
.calctrlDateCell, .calctrlWeeklyCell, .calctrlTitle	
.calctrlMonthDisplayName, .calctrlRow	
.calctrlRightNavButton, .calctrlMonthDisplayName	
.bdyView	
.tdViewContainer, .tbViewContainer, .divCalendarView	
.divCalViewStatus	
.divCalendarRightPane	
.divCalendarControl	

Styles	Function
.idCalCtrlStatus	
.divCalendarView	
.tdFolderPath	
.tdFolderPath A	
.divCalViewStatus	
.tabselected	
.tabsunselected	
.tabsnone	
.tabHeader	
.tabdiv	
.tabdivHide	
.datepicker_date	
.datepicker_time	
.apptTdDatePicker	
.fldrTitleBar	
.fldrTitleBar A	
.AttachTable, .cbToolbar, .cbSubmenu, .fldFont, .fldText, .msgHeader, .fldLabel,	
.trButtons, .msgViewer, .appointment, .rnOnlineBar, .viewStatusBar,	
.appointmentBody, .tabHeader, BODY	
.cbToolbar IMG, .cbSubmenu IMG	
.cbToolbar TD	
.monthlyDayHeader, .FORMATBAR, .cbToolbar, .btnRecip, .tbButtonRaise,	
.tbButton A: hover, .tbButton: hover	
.cbSubmenuItem A: hover, .cbSubmenuItem: hover	
.cbSubmenuItem SPAN	

Toolbars.css

Table E6 lists the CSS styles defined in the Toolbars.css file. These styles control the appearance of Outlook Web Access toolbars.

Table E6 Toolbars.css styles

Styles	Function
.tbContentElement	
.tbToolbar	
.tbButton	
.tbIcon	
.tbSeperator	
.tbMenu	

Styles	Function
.tbMenuItem	
.tbSubmenu	
.tbGeneral	
.tbHandleDiv	
.tbButtonMouseOverUp	
.tbButtonMouseOverDown	
.tbButtonDown	
.tbIconDown	
.tbIconDownPressed	
.tbMenuBodyOuterDiv	
.tbMenuBodyInnerDiv	
.tbMenuBodyTable	
.tbMenuSeperator	
.tbMenuSeperatorTop	
.tbMenuSeperatorBottom	
.tbMenuBlankSpace	
.tbSubmenuGlyph	
.tbMenuItemChecked	
.tbMenuItemIcon	
.tbMenuItemIconChecked	
.tbMenuItemIconMouseOver	
.tbMenuItemIconCheckedMouseOver	
.tbScriptlet	

Navbar.css

Table E7 lists the CSS styles defined in the Navbar.css file. These styles control the appearance of the Outlook Web Access navigation bar.

Table E7 Navbar.css styles

Styles	Function
.nbNavmenu	
.nbLabel	
.nbImage	
.nbFlybar	
.nbFlybarLabel	
.nbTableMain	
.nbTable	
.nbTreepane	
.nbTreeProgress	
.nbDropMenu	
.treeFolder	
.treeOFLOW	
.treeBrand	

The information contained in this document represents the current view of Microsoft Corporation on the issues discussed as of the date of publication. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information presented after the date of publication.

This White Paper is for informational purposes only. MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS DOCUMENT.

Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Microsoft Corporation.

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

Unless otherwise noted, the example companies, organizations, products, domain names, e-mail addresses, logos, people, places and events depicted herein are fictitious, and no association with any real company, organization, product, domain name, email address, logo, person, place or event is intended or should be inferred.

© 2001 Microsoft Corporation. All rights reserved.

Microsoft, Active Directory, ActiveX, FrontPage, JScript, MSDN, Outlook, Visual Basic, Visual InterDev, Win32, Windows, and Windows NT are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

The names of actual companies and products mentioned herein may be the trademarks of their respective owners.