SharePoint Team Services Administrator's Guide

In This Document:

Introducing SharePoint Team Services and FrontPage Server Extensions 2002
SharePoint Team Services and FrontPage Server Extensions 2002 Architecture
Windows Security Model
FrontPage Server Extensions 2002 Security Under UNIX
SharePoint Team Services Requirements
FrontPage Server Extensions 2002 Requirements
Installing SharePoint Team Services
Installing FrontPage Server Extensions 2002Server Extensions 2002
Running Unattended Setup
Upgrading from FrontPage Server Extensions and Office Server Extensions
HTML Administration
Command-line Administration
Using the Administration Tools Remotely
Setting Configuration Properties
Configuring Properties for a Virtual Server
Setting Installation Defaults
Creating a New Virtual Server
Optimizing Web Site Performance
Managing Roles
Managing Users
Managing Web Site Permissions
Delegating Administration Tasks to Site Owners
Limiting User Accounts
Inviting Members to a Site
Managing the Database
Backing Up and Restoring Data
Managing Web Document Discussions and Subscriptions
Configuring Support for Database Integration with a Web Site

Monitoring Server Health

Managing Source Control

Analyzing Web Site Usage

Scheduling Timed Jobs

Searching SharePoint Team Web Sites and FrontPage-based Web Sites

Files and Permissions on Internet Information Service

Files and Permissions on Apache

Special Directories and Storage Locations

Introducing SharePoint Team Services and FrontPage Server Extensions 2002

SharePoint™ Team Services from Microsoft® offers a new way of working together. SharePoint Team Services provides both Web publishing and collaboration features to make communicating ideas and sharing information easier. SharePoint Team Services is a superset of Microsoft FrontPage® Server Extensions 2002, and includes all of the features available with the server extensions. In addition, SharePoint Team Services contains new workgroup features that create a rich environment for Web publishing and team communication. By using SharePoint Team Services, administrators can create, author, and administer Web sites that help a team organize and advance on a project.

> **Note** SharePoint Team Services is available on the Microsoft Windows® 2000 platform only. FrontPage Server Extensions 2002 are available on the Microsoft Windows NT®, Windows 2000, and UNIX platforms.

FrontPage Server Extensions 2002 are an update of FrontPage 2000 Server Extensions, and include new security features, such as roles and rights, and new features for monitoring server health and Web site usage.

SharePoint Team Services and FrontPage Server Extensions features

If you build and maintain Web sites for customers and coordinate with multiple authors, then the features included with both SharePoint Team Services and FrontPage Server Extensions 2002 can help you accomplish the following tasks:

- Manage Web sites either on the local server or remotely, by using HTML Administration pages or a quick command-line interface.
- Secure Web sites and grant authoring, browsing, site management, or other user

rights to authenticated users.

- Analyze site usage to find out who is viewing the site and how often.
- Apply versioning to files on a SharePoint team Web site or FrontPage-based Web site to ensure that users always have the latest copy of a file.
- Track errors on the server to help prevent site or server crashes.
- Upgrade FrontPage Web site management functionality with features like forms, search tools, and usage analysis.
- Use the rich features of Microsoft Office XP to add new documents and edit existing documents on the server running SharePoint Team Services.

SharePoint team Web site features

In addition to the features included in both SharePoint Team Services and FrontPage Server Extensions 2002, SharePoint team Web sites include unique features that give you the ability to do the following:

- Discuss and subscribe to documents on your SharePoint team Web site.
- Customize a SharePoint team Web site by adding or removing features, such as task lists or surveys.
- Send invitations to the new site and allow users to sign up for the site automatically.
- Create document libraries to find and view documents quickly.
- Create and customize team lists and allow each user to create customized views of the list.
- Add announcements, calendar information, surveys, and other special items to the Web site.
- Add users, and perform other Web administration tasks by using an easy HTML interface.

Note The additional features in SharePoint team Web sites require a Microsoft SQL Server™ or MSDE database. For more information, see SharePoint Team Services System Requirements.

How SharePoint Team Services and FrontPage Server Extensions work

SharePoint Team Services and FrontPage Server Extensions 2002 work in the following settings:

- Intranet sites, for users within an organization
- Internet sites, for users in different locations
- Extranet sites, for users in more than one organization

SharePoint Team Services and FrontPage Server Extensions 2002 use the Common Gateway Interface (CGI) or the Internet Server Application Programming Interface (ISAPI), which are near-universal Web server extension mechanisms. Both FrontPage Server Extensions and SharePoint Team Services work with Microsoft Internet Information Services (IIS) on the Windows platform, and FrontPage Server Extensions work with the FrontPage-patched Apache Web Server on UNIX.

Communication between a client computer and a Web server running SharePoint Team Services or FrontPage Server Extensions 2002 uses the same open HTTP protocol that Web browsers on a client computer use to interact with a Web server. No file-sharing access on the Web server computer is needed, and neither FTP nor telnet access is required. No proprietary file system sharing calls are necessary to use SharePoint Team Services.

When SharePoint Team Services or FrontPage Server Extensions 2002 is installed on a Web server, SharePoint team Web site or FrontPage-based Web authoring and administering functionality is available from any computer that has a level-4 browser (such as Microsoft Internet Explorer 4.0 or later, or Netscape Navigator 4 or later). Authoring and administration are also available from a SharePoint Team Services–compatible client program, such as the Office 2000 client or the Office XP client, whether the computer is on the Internet or on an intranet.

SharePoint Team Services for hosting companies

Hosting companies such as Application Service Providers (ASPs) and Internet Solution Providers (ISPs) look for low-cost, rich-featured solutions to offer their clients (either organizations or individuals). With SharePoint Team Services, you can provide clients with easy-to-build, user-friendly Web sites and straightforward Web management. SharePoint team Web sites include the following features:

- Quick and easy content creation and Web publishing
- Simple Web customization tools
- Effortless document management with lists and document libraries
- Rich collaboration features such as Web discussions and subscriptions
- Secure and remote administration from both the command line and HTML Administration pages
- Ability to delegate administration rights individually

  You can give clients the rights to add users without compromising your security or processes.

The following scenarios demonstrate how a hosting company might use SharePoint team Web site features.

Hosted site for a neighborhood association

A development company is working on plans to build a new mall in a suburban neighborhood. The neighborhood association wants to organize meetings to discover what impact the mall will have on its neighborhood.

The neighbors contact an ISP, which sets up a SharePoint team Web site for their association. The association chairperson posts documents with the zoning plans for the neighborhood to a document library, creates a membership list for the association in the Contacts page, and sets up meeting dates for neighborhood events on the Events page. Members are assigned tasks on the Tasks page to discover what other plans are being made and to post their findings to the Web site.

New neighbors add their e-mail addresses to the association roster from their Web browsers and contribute comments to the discussion, which are stored in the collaboration database. They participate in the team Web site from their browsers without having to install a SharePoint Team Services–compatible client, such as Office XP. If they do have a SharePoint Team Services–compatible client, however, they can create and save documents to the Web easily, with all the richness that desktop applications offer.

The board members set up a meeting with the development company to discuss their concerns, and the note taker posts the meeting notes to the document library on the Web site to keep the neighbors informed. The neighborhood Web site allows each neighbor to contribute and have his or her own concerns addressed.

Hosted extranet for an advertising campaign

An advertising agency is working on an ad campaign for a new client in a different region and wants to keep communication efficient To get a secure area for posting plans and discussing ideas, the agency signs up with an ISP that provides them with a SharePoint team Web site and delegates user management rights to the advertising agency manager.

The manager adds his team members and members of the client's team to the site. He posts the initial plan for the advertising campaign to a document library, and then subscribes to the document so that he is automatically notified when the client reviews the plan. The client manager posts the required time line, adds inline discussion items to the plan, and creates a list of key contacts within her organization on the Contacts page. The members of the client team contribute background material to use in the ad, discuss the plan by using inline discussions, and add their perspective on the company image.

The agency team reviews the comments and posts revised plans and draft ads. When the ad agency team posts these plans, a hyperlink is automatically added to the document library based on the document properties; the team does not have to change the navigation or open the site to get its content posted to the right place — the content is linked into the page automatically. The client reviews the ad and passes the plan to the legal team at corporate headquarters. The legal team reviews the plan and the drafts, and then signs off online.

SharePoint Team Services for organizations

If you have your own internal staff to set up and manage your Web sites, you can also benefit from the new features available in SharePoint team Web sites. For example, SharePoint team Web sites give you the following features:

- Better workgroup communication by using Web document discussions, subscriptions, tasks lists, and so on
- Better document management by using lists and document libraries
- Ability to use a SharePoint Team Services–compatible client, such as Office XP, to create great Web documents
- Quick customization by using a SharePoint Team Services–compatible Web page editor, such as FrontPage 2002
- Easy publishing and Web site management tools
- Ability to analyze site usage

Budget planning on an intranet

A large organization is planning a project for the next fiscal year. The information technology (IT) department creates an out-of-the-box SharePoint team Web site for the project and assigns Advanced Author rights to the project lead. The lead adds members of the team to the site and posts the plans and key dates for the project to the Web.

Team members contribute their own items, discuss any items that overlap, and talk about priorities for the project. Department managers subscribe to the planning documents to find out how the project is progressing and comment on the plans by using inline discussions stored in the collaboration database.

The team members continue to add information to the site, and the project lead summarizes the information into a proposal for the head of the company. Because the department heads have been following the process on the Web, the proposal is quickly reviewed and accepted, and the project is assigned the resources needed to go ahead. As the project progresses, the team members continue to use the Web site to track issues, communicate ideas, and coordinate tasks.

FrontPage Server Extensions and the FrontPage client

As with FrontPage 2000 and FrontPage 2000 Server Extensions, you get enhanced FrontPage 2002 functionality when you author content on a server that has FrontPage Server Extensions 2002 installed. You can edit FrontPage-based Web sites in the FrontPage client, and you can get access to the special FrontPage Server Extensions–enabled features, such as usage analysis, from the FrontPage client when FrontPage Server Extensions are installed.

SharePoint Team Services and SharePoint Team Services–compatible clients

When you use SharePoint Team Services with SharePoint Team Services–compatible client programs, such as Microsoft Office XP, or a SharePoint Team Services–compatible Web page editor, such as Microsoft FrontPage 2002, you get enhanced functionality in the client. When you author content on a server that has SharePoint Team Services installed, you can use templates in the SharePoint Team Services–compatible Web page editor to create many of the special pages available in a SharePoint team Web site. For example, you can add lists and document libraries to other Webs hosted by your server running SharePoint Team Services.

Not only can you use the special pages from SharePoint Team Services in your SharePoint Team Services–compatible Web page editor, but you can also use the editor to edit the pages in your SharePoint team Web site. For example, you can open a SharePoint team Web site list page in a SharePoint Team Services–compatible Web page editor, and edit the text introducing the list, apply a theme to the entire site, or adjust the site navigation. Many of these tasks are also available through the browser, but you get a much richer editing environment if you do your work in a SharePoint Team Services–compatible Web page editor.

SharePoint Team Services and FrontPage Server Extensions 2002 Architecture

The architecture for SharePoint™ Team Services, a new technology from Microsoft®, is based on the architecture for Microsoft FrontPage® Server Extensions 2002, but also adds a SQL Server™ or MSDE database to the mix to support new features such as Web document discussions and document libraries. FrontPage Server Extensions 2002 are an update to FrontPage 2000 Server Extensions, and the architecture has not changed significantly.

FrontPage Server Extensions architecture

The FrontPage client system communicates with a Web server via WinSock and TCP/IP. Wizards and custom applications on the server computer communicate with the FrontPage client using OLE automation.

The FrontPage client communicates with FrontPage Server Extensions using HTTP, the same protocol Web browsers and Web servers use to communicate. FrontPage implements a remote procedure call mechanism on top of the HTTP POST request, so that the FrontPage client can request documents, update the Tasks list, add new authors, and so on.

The Web server sees POST requests addressed to the FrontPage Server Extensions Common Gateway Interface (CGI) programs and directs those requests accordingly. FrontPage correctly communicates between client and server through proxy servers (firewalls).

Note FrontPage does not use the HTTP PUT request. As described in the HTTP specification, PUT sends a document to a Web server; however, few Web servers implement PUT. Therefore, the FrontPage client uses the universally implemented HTTP POST request for all communication with FrontPage Server Extensions.

FrontPage Server Extensions are accessed from the Web server using the Common Gateway Interface (CGI), the universal Web server extension mechanism. The implementation of CGI differs somewhat among Web servers and platforms. For example, on UNIX, the Web server invokes a CGI extension by running it in a separate process, whereas on Windows NT, Internet Information Services (IIS) supports the Internet Server Applications Program Interface (ISAPI), a CGI-style communication interface that incurs less overhead.

SharePoint Team Services architecture

The architecture for SharePoint Team Services is based on that for FrontPage Server Extensions; however, SharePoint Team Services relies on a database to provide new functionality. Also keep in mind that SharePoint Team Services does not follow the "create and then publish" model you may be used to with FrontPage Server Extensions. The moment you create a SharePoint team Web site it is live on the server; you do not need to publish the Web site to another server. You can still edit the Web site in a SharePoint Team Services–compatible Web page editor, such as FrontPage 2002, or add pages and documents to the site, but you do not need to publish your changes — they take effect immediately when you save the files.

The following diagram illustrates the SharePoint Team Services architecture in action.

Note that although a few features require Microsoft Internet Explorer (such as Export to Spreadsheet, which uses ActiveX® components), all content returned to the client computer is HTML.

Because SharePoint Team Services relies on both the file system and a database to track information about your Web site, you must be sure to keep these systems synchronized. For example, a list in a SharePoint team Web site is a combination of data in the database and HTML files. If you want to back up a list, you must back up both the data in the database and the HTML files. SharePoint Team Services provides tools to use for backing up and restoring the collaboration database, but you must back up and restore the file system by using your usual operating system tools. Be sure to back up the file system whenever you back up the database. Keeping your backups synchronized will help simplify the process if you ever need to restore your Web site. For more information about the collaboration database, see Managing the Database.

Windows Security Model

SharePoint™ Team Services, a new technology from Microsoft®, and Microsoft FrontPage®

Server Extensions 2002 rely on the security features of Microsoft Windows NT® (FrontPage Server Extensions 2002 only) and Microsoft Windows® 2000 to provide security for Web site content. There are two elements in Windows security:

- User authentication — the process used to validate the user account that is attempting to gain access to a Web site or network resource.
- File system security — the ability to control which users gain access to which files or folders in the file system.

In addition to these elements, SharePoint Team Services and FrontPage Server Extensions 2002 include a new security feature: user roles. With user roles, you do not have to control the file and folder permissions separately, or worry about keeping your local groups synchronized with your list of Web users. You use roles to give users permissions on your Web site, and use SharePoint Team Services and FrontPage Server Extensions 2002 administration tools to add new users directly. For more information about user roles, see Managing Roles in the SharePoint Team Services Administrator's Guide.

User authentication

When you use SharePoint Team Services or FrontPage Server Extensions 2002 on the Windows platform, user authentication is based on Internet Information Services (IIS) authentication methods. IIS provides five forms of user authentication:

- Anonymous authentication
- Basic authentication
- Integrated Windows authentication
- Digest Access authentication
- Certificate authentication

Anonymous authentication

Anonymous authentication provides access to users who do not have Windows NT server accounts on the server computer (for example, Web site visitors). IIS creates the anonymous account for Web services, IUSR_*computername*. When IIS receives an anonymous request, it impersonates the anonymous account.

Basic authentication

Basic authentication is an authentication protocol supported by most Web servers and browsers. Although Basic authentication transmits user names and passwords in easily decoded clear text, it has some advantages over more secure authentication methods, in that it works through a proxy server firewall and ensures that a Web site is accessible by almost any Web browser. If you use Basic authentication in combination with Secure Sockets Layer (SSL) security, you can add a layer of protection to the user names and passwords,

making your user information more secure.

Integrated Windows authentication

Integrated Windows authentication (also known as Windows NT Challenge Response) encrypts user names and passwords in a multiple transaction interaction between client and server, thus making this method more secure than Basic authentication. Disadvantages are that this method cannot be performed through a proxy server firewall, and some Web browsers (most notably, Netscape Navigator) do not support it. You can, however, enable both this method and Basic authentication at the same time.

Digest Access authentication

Digest Access authentication is similar to Basic authentication, except that a user's name and password are transmitted in a more secure format. This method requires IIS 5.0 or later on the server computer and Microsoft Internet Explorer 5 or later on the client computer. Digest Access authentication works with domain accounts only; you cannot use Digest Access authentication with local user accounts.

Certificate authentication

Certificate authentication (also known as Secure Sockets Layer security) provides communications privacy, authentication, and message integrity for a TCP/IP connection. By using the SSL protocol, clients and servers can communicate in a way that prevents eavesdropping, tampering, or message forgery. With SharePoint Team Services or FrontPage Server Extensions 2002, SSL ensures secure authoring across firewalls and ensures security during remote administration of SharePoint Team Services or FrontPage Server Extensions 2002. You can also specify that SSL be used when opening a SharePoint team Web site or opening or publishing FrontPage-based Web sites.

You choose the authentication method you want to use when you set up your Web server. You cannot change the authentication method by using the SharePoint Team Services or FrontPage Server Extensions 2002 administration tools; you must use the Internet Information Services administration tool for your server computer to change the authentication method.

> **Note** For more information about IIS authentication methods, see the topic **About authentication** in IIS 5.0 or IIS 4.0 online Help.

About firewalls

SharePoint Team Services and FrontPage Server Extensions 2002 support connectivity through firewalls. Depending on your configuration, you must make sure your firewall is open for the standard HTTP ports 80 and 443. When using a firewall, you must configure your Web sites with Basic Authentication because Integrated Windows Authentication cannot

pass through a firewall.

File system security

SharePoint Team Services and FrontPage Server Extensions 2002 rely on the Windows operating system to secure the file system for your Web sites. Microsoft Windows NT and Windows 2000 both support access control lists (ACLs) to secure files and folders.

> **Note** ACLs are supported only by the Windows NT File System (NTFS). Because SharePoint Team Services and FrontPage Server Extensions 2002 security is in part based on ACLs, you must use NTFS on the server computer that hosts IIS and SharePoint Team Services or FrontPage Server Extensions 2002.

The following ACLs can be given to accounts to control access to a file.

File ACL	Description
None	User has no access to a file.
Read (or **Read Data** in Windows 2000)	User can view data in a file.
Write (or **Write Data** in Windows 2000)	User can change data in a file.
Execute (or **Execute Data** in Windows 2000)	User can run a program file.
Delete	User can delete a file.
Change Permissions	User can change permissions on a file.
Take Ownership	User can take ownership of a file (note that the owner of a file also has all other permissions for that file).

The following ACLs can be given to accounts to control access to a folder.

Folder ACL	Description
None	User has no access to a folder.
Read (or **List Folder** in Windows 2000)	User can view file names and subfolder names in a folder.
Write (or **Create Files** in Windows 2000)	User can add files and subfolders to a folder.
Execute (or **Traverse Folder** in Windows 2000)	User can change to subfolders.

Delete (or **Delete subfolders and files** in Windows 2000)	User can delete subfolders.
Change Permissions	User can change permissions on a folder.
Take Ownership	User can take ownership of a folder (note that the owner of a folder also has all other permissions for that folder).

Note For more information about ACLs, see the topics **File permissions** and **Folder permissions** in the Windows NT 4.0 or Windows 2000 online Help.

Managing permissions manually

With FrontPage 2000 Server Extensions, you could bypass the built-in security management and set permissions manually on the content of a FrontPage-based Web site. With SharePoint Team Services and FrontPage Server Extensions 2002, the roles and permissions have been improved, and this functionality is no longer available.

Related links

For more information about users and roles in SharePoint Team Services and FrontPage Server Extensions 2002, see Managing Roles and Managing Users in the SharePoint Team Services Administrator's Guide.

The Microsoft Technet Web site includes white papers that describe IIS security in more detail. For an overview of IIS security, see The Basics of Security. For information about IIS and security issues, see Untangling Security.

FrontPage Server Extensions 2002 Security Under UNIX

In addition to the Microsoft® Windows® operating system with Internet Information Services (IIS), you can also use Microsoft FrontPage® Server Extensions 2002 on the UNIX platform, running Apache Web server with the FrontPage Apache patch. To secure your Web site, you use the built-in security features of UNIX, Apache, and FrontPage Server Extensions 2002.

UNIX security model

The Apache Web server maintains a list of users who have permission to use the Web server. This access file list is specific to the Web server and is separate from the list of users and groups who can log on to the computer interactively.

The Web server access file contains names and passwords for each user, and the list of files and Common Gateway Interface (CGI) scripts that each user is allowed to access. On Apache

servers, the user name and password file is called service.pwd, the group file is called service.grp, and the access file containing the information about which users can access files and CGI scripts in the .htaccess directory is named .htaccess.

In FrontPage Server Extensions 2002, there are two levels of security. First, the .htaccess files in the _vti_bin, _vti_aut, and _vti_adm directories tell Apache which users can execute the shtml.exe, author.exe, and admin.exe CGI scripts. Second, the FrontPage Server Extensions 2002 code checks the roles list to see if the authenticated user has permission to execute the requested FP-RPC command.

Multiple access files are often stored on a Web server. Each access file provides security for the directory containing it and for any subdirectories that do not contain their own access files. By creating access files at all levels of the Web server, users with varying levels of permissions can be given access to different areas of the server.

The Web server runs as a UNIX account, usually www, which limits which files the Web server can access. Any process that the Web server runs and any files that it opens must be available under the permissions of the same UNIX account. This arrangement can cause a security problem in a multihosted environment, because one user's CGI script might read files in another user's content area on the server. This problem is compounded when write access is supported, as it is with the FrontPage 2002 client. If a user's CGI script has sufficient permissions to write to a file in the user's own content area, the CGI script could write or overwrite files in other users' content areas.

To address this security issue (which is common to all CGI scripts, including the FrontPage Server Extensions 2002), FrontPage Server Extensions 2002 uses an SUID approach, so that the FrontPage Server Extensions 2002 code always runs as the UNIX account that owns the particular content area. For more information, see the FrontPage Apache patch section below.

Supported authentication methods

The Apache Web server supports Basic Authentication out of the box, and FrontPage Server Extensions 2002 support remote administration when Basic Authentication is used. When the server uses Basic Authentication, the Web browser (or the FrontPage 2002 client) prompts the user for a user name and password. The user name and password are then transmitted in clear text. (Some Web servers use an encryption technique known as MD5, but FrontPage Server Extensions 2002 do not support this method.)

FrontPage Server Extensions 2002 use Basic Authentication because all firewalls and proxy servers support it. Because Basic Authentication transmits passwords across the network in an easily decoded format, however, it is easy for network spies to steal passwords that are sent in this manner. If you are concerned about security, consider using Secure Sockets Layer (SSL) to encrypt all communications to and from your Web server.

Controlling user access

In FrontPage 2000, three types of users were defined for every FrontPage-extended web: administrators, authors, and browsers (site visitors). For FrontPage Server Extensions 2002, there are now five roles for users by default: Browser, Contributor, Author, Advanced Author, and Administrator. As in FrontPage 2000, permissions for these roles are cumulative — for example, all Authors also have Browser permissions.

Under UNIX, FrontPage contains four executable CGI scripts:

- Admin.exe for administrative tasks
- Author.exe for authoring webs
- Shtml.exe for browse-time components such as form handlers
- Fpadmcgi.exe for advanced administrative tasks

These scripts are installed in the /Usr/Local/Frontpage/Version5.0/Exes/ folder. FrontPage performs all authoring and administrative tasks by sending HTTP POST requests to the Apache Web server, which then invokes these scripts. There are also three directories created below the root of every FrontPage-extended web to contain .htaccess files, which determine which users are allowed to run the executables. These directories are:

- _vti_bin, which stores permissions for shtml.exe
- _vti_adm, which stores permissions for admin.exe and fpadmcgi.exe
- _vti_aut, which stores permissions for author.exe.

The _vti_adm and _vti_aut subdirectories are within the _vti_bin directory, as shown in the following example:

```
/FrontPage-extended web root
      /_vti_bin
            /_vti_adm
            /_vti_aut
```

FrontPage Server Extensions 2002 maintain these .htaccess files when you add and remove users for a FrontPage-extended web. These .htaccess files refer to a user/password file and a group definition file. The user name/password list for each FrontPage-extended web with unique permissions is stored in _vti_pvt/service.pwd, and the group definitions are stored in _vti_pvt/service.grp. FrontPage-extended webs that inherit permissions from their parent web use their parent web's service.pwd and service.grp files. FrontPage Server Extensions 2002 maintain the service.pwd and service.grp files automatically when you add and remove users for a FrontPage-extended web. For more information, see the Access Files section below.

Securing Web content

Setting secure permission modes for files and directories in a customer's content directory is

necessary to maintain Web server security. All files and directories must be owned by the customer's user identifier/group identifier to prevent other customers on the same server from being able to modify those files. The following table shows how Owsadm.exe sets file modes when you run it to configure a virtual server with FrontPage.

Scenario	Settings to use	

Single UNIX account per customer	files:
	directories:
	umask(022):
		`rw- r-- r--`
		`rw- r-x r-x`
		`--- -w- -w-`

For example, when configuring a virtual server with FrontPage, you run Owsadm.exe with the following options (note that this example assumes that multihosting is being used, and that the virtual host name is www.host.com):

```
owsadm.exe –o install –p www.host.com
-servconf /usr/local/www/conf/httpd.conf –xu unixuser
  –u webusername –pw webpassword
```

In this example, *unixuser* is the Unix account of the customer that owns this virtual server. Each customer must have his own UNIX account if you want strong security. All files and directories under the root content directory for the virtual server change ownership (chown) to this account and the account's group (by using chgrp).

Also, *webusername* and *webpassword* are the user name and password that must be used when remotely accessing the web site by way of the FrontPage 2002 client or when accessing HTML Administration Pages for the virtual server.

Later, you can use Owsadm.exe to change ownership and change mode (chmod) of existing content files in a FrontPage-extended web if necessary.

By default, a 022 umask is used when FrontPage Server Extensions 2002 are used to create new Web pages and other files. To set a custom umask, you must edit and recompile the source file for the FrontPage Apache patch, fpexe.c. Because you must edit the umask in the patch itself, you can only change the umask value for an entire server — you cannot set a separate umask value for each web. To change the umask, open the fpexe.c file and search for the line `umask(022)`. Owsadm.exe uses the umask that is set for the shell in which Owsadm.exe is run.

Access files

On Apache Web servers, an access file contains security-related directives that control which users and groups have access to the content on a server. An access file can also contain directives that specify which IP addresses have access to content.

The global access settings for the Web server are contained in the access sections in the httpd.conf file. Per-directory access files (named .htaccess) can be placed in directories on the Web server to implement per-directory access control. FrontPage Server Extensions 2002 security uses and relies upon .htaccess files that are placed in the content directories. FrontPage Server Extensions 2002 assume you do not also try to control access to FrontPage-enabled webs by using httpd.conf. You must enable .htaccess files in Apache by using the AllowOverride setting in httpd.conf, otherwise the default FrontPage security will not work and your Web server will be insecure. The default Apache setting for AllowOverride is All, which is the recommended setting for FrontPage-extended virtual servers. AllowOverride must allow at least AuthConfig, Limit, and Options.

Access file settings

FrontPage Server Extensions 2002 rely on .htaccess files to add FrontPage-extended web visitors with the proper permissions to the Web server account list, and to protect content and programs in FrontPage-extended webs.

FrontPage Server Extensions 2002 create a directory named _vti_pvt for the root web and for each subweb. In each FrontPage-extended web with unique permissions, the _vti_pvt directory contains the following two files:

- Service.pwd contains the list of users and encrypted passwords for the FrontPage-extended web
- Service.grp contains the list of groups, and the users within each group

The .htaccess file in the _vti_bin directory controls access to the browse-time FrontPage Server Extensions 2002 program Shtml.exe. If all users have browse permission, this setting is specified in the access file. Otherwise, the access file points to a list of users and groups with browse access. A separate .htaccess file in the _vti_adm directory sets permissions for the administrative program, Admin.exe. A third /htaccess file in the _vti_aut directory sets the permissions on the authoring program, Author.exe.

When you add a user to a role by using the FrontPage Server Extensions 2002 administration tools, FrontPage Server Extensions 2002 add the user's name and password to the single Service.pwd file and to the appropriate group in the service.grp file pointed to from the .htaccess files stored in the _vti_bin, _vti_aut, or _vti_adm directories. When an IP address restriction is set on browse access to an FrontPage-extended web, this restriction is added to the _vti_bin directory access file.

The set of access files for the non-CGI _vti directories created by FrontPage Server Extensions 2002 on an FrontPage-extended web is illustrated in the following table.

Web directories or content	Access list

root web	**Runtime Access List** browsers (GET, POST) authors (GET, POST) administrators (GET, POST) UNIX: rwxr-xr-x uid/gid
_vti_log *authoring logs* _vti_txt *text index* _vti_pvt *configuration*	**Miscellaneous Access List** browsers authors (GET, POST) administrators (GET, POST) UNIX: rwxr-xr-x uid/gid

FrontPage Apache patch

FrontPage supplies a patch to the Apache Web server that has two purposes: it eliminates the need to update the httpd.conf file when a subweb is created, and it securely enables the SUID functionality that the FrontPage scripts run as the UNIX account that owns a particular FrontPage web.

The patch to the Apache Web server intercepts each call that the FrontPage client makes to the server extensions executable files. It then performs security checks, sets user ID to the owner of the Web site (thus requiring SUID/SGID operation of the server extensions and the Web content), and invokes a central copy of the server extensions executable files.

The FrontPage Apache patch was optional for FrontPage 2000 but is required for FrontPage Server Extensions 2002, because it is more secure and allows users to create subwebs remotely.

Apache patch security strategy

The FrontPage Apache patch consists of two parts:

- The FrontPage Apache module, which intercepts requests from the FrontPage client to the FrontPage Server Extensions 2002 executable files, performs security checks, and redirects the request to the fpexe stub program, which is set to SUID root. By intercepting each request within the server itself, no script alias is required.
- The fpexe program, which accepts authoring requests from the FrontPage Apache module, performs additional security validation, changes the user ID of the Web server process to the owner of the FrontPage-extended web being authored, and then invokes the central copy of FrontPage Server Extensions 2002 executable files.

The fpexe stub program must be set SUID to root in order for it to change user IDs of the Web server process to the owner of the web. For that reason, numerous security checks are

performed in order to prevent this stub program from being exploited. Checks are performed to validate the following:

- Proper ownership and permissions are set for the fpexe stub program and its directory.
- Proper ownership and permissions are set for the server extensions executable files and their directories.
- The Web content area being authored has a valid user ID.
- The Web content area being authored has a valid group ID.
- Only the FrontPage Server Extensions 2002 CGI executable files are being executed, and not other CGI scripts on the computer.
- The environment variables (including the path) are cleaned and passed only if the variable is on an approved list.
- FrontPage Server Extensions 2002 executable files are being invoked only by the Web server.
- A 128-byte key value is passed to the fpexe program and validated, ensuring that only the Web server is able to run fpexe.

The 128-byte key value is generated dynamically when the Web server is initialized, and stored for validation purposes in a suidkey.* file. This file can be read and written only by "root" and is stored in a directory that is readable only by "root." The suidkey.* file can be written with root-only permissions because the Web server process runs as "root" during initialization, and does not switch to another user ID (such as "www") until after initialization is complete. The suidkey.* file name suffix is the process group ID of the Web server.

The contents of this dynamic key value are generated during Web server initialization based on a permutation of the output of the process status (**ps**) command, and are then marked as exclusive or (XOR'ed) with the contents of an administrator-controlled custom key file stored in /usr/local/frontpage/version5.0/apache-fp/suidkey. This custom key file must exist, be owned and readable only by "root," and contain at least eight bytes of data.

Because programs with user ID set to "root" are of concern to server administrators, Microsoft makes the source code of the FrontPage Apache module and the fpexe stub program available for review. The source code is extensively commented with an explanation of the checks that are performed and recovery actions to take if an error denoting an insecure configuration is logged by the FrontPage Apache module or fpexe. As part of your server's maintenance, change the contents of the custom key file regularly when restarting the server. This step protects the key value.

Note The SharePoint Team Services Administrator's Guide includes the source code for the FrontPage Apache module and the fpexe stuff program. For more information, see Source Code for fpexec.c and Source Code for the FrontPage Apache module.

How the key is validated

When the FrontPage Apache module processes a request to invoke the FrontPage Server Extensions 2002 executable files, the module performs preliminary validation of the request and redirects the request to the fpexe stub program. The 128-byte key value generated when the server was initialized is passed to fpexe through a pipe and thus is not visible in the program environment. The 128-byte key value is read by fpexe from the pipe, and then compared to the contents of the dynamically generated suidkey.* file that was created when the Web server was initialized.

Because the user ID of fpexe is set to "root," it has access to the contents of the suidkey.* file. If the suidkey.* file has correct permissions (readable only by "root" in a directory readable only by "root"), and if the 128-byte key value matches, then fpexe performs additional checks to validate the user ID, group ID, and ownership of the target FrontPage Server Extensions 2002 executable files.

If all checks pass, then fpexe switches the user and group IDs of the server process to those of the Web content owner, and then runs the FrontPage Server Extensions 2002 executable files. If any of these checks fail, an error is written to the Web server log and the server extensions are not run.

Note that the FrontPage Apache module's security checks do not replace the Web server's .htaccess file security system. Both systems work together to ensure security for the FrontPage-extended web. The Web server's .htaccess security protects remote access to the Web content by validating that the user of the FrontPage Server Extensions 2002 is a registered site visitor, author, or administrator of the web. In addition to this normal level of security checking, the FrontPage Apache module's security checks ensure that the fpexe program is not used to gain unauthorized root access to the Web server.

Setting FrontPage Server Extensions 2002 to SUID/SGID

When you use the FrontPage Apache patch, the execute permissions of the FrontPage Server Extensions 2002 executable files are automatically changed to Set User ID/Set Group ID (SUID/SGID). This setting forces the server extensions to be executed with the file-system permissions of the owner of shtml.exe, author.exe, and admin.exe. When suitable permissions for each customer's content area are applied, the FrontPage Server Extensions 2002 (and any other CGI scripts) for one customer are prevented from writing to the content areas of other customers.

Setting the FrontPage Server Extensions 2002 to SUID/SGID provides two benefits:

- It prevents users who have telnet access to the Web server computer from modifying content in a web that they do not own.
- It provides an extra level of operating system security on top of the security built into

FrontPage.

Setting the FrontPage Server Extensions 2002 to SUID/SGID for the owner of each FrontPage-extended web is the recommended configuration for the FrontPage Server Extensions 2002 on UNIX servers. Existing installations of the FrontPage Server Extensions 2002 earlier than version 3.0 should be upgraded to use SUID/SGID operation of the server extensions and unique ownership of the Web content. New installations of the FrontPage Server Extensions 2002 that are performed with the Owsadm utility or the fp_install.sh installation script by using the FrontPage Apache patch automatically configure the server extensions for SUID/SGID operation.

The FrontPage Server Extensions 2002 do not require root access at any time. However, the Owsadm utility and the fp_install.sh installation script should be run as "root," because they set execute privileges of the extensions to SUID. Because fp_install.sh is written as a shell script, you can review it before running it on the Web server. Also, if you want the Server Administration Pages to work, a special copy of fpadmcgi.exe (two copies of this file are installed at Setup) must be run as root.

Other security issues

When you work with Web servers, you must be aware of other possible entry points to your content. For example, you can secure all of your users, and set permissions on files, but if you allow users to perform certain operations, such as posting by way of File Transfer Protocol (FTP) or using Telnet to perform administration tasks, you might expose your Web server to security threats. With the FrontPage Server Extensions 2002 and the FrontPage Apache patch, many of these concerns are handled for you.

Using FTP with FrontPage Server Extensions 2002

You can use FTP and the FrontPage Server Extensions 2002 on IIS and on Apache servers that are running the FrontPage Apache patch (apache-fp); however, it is not recommended. If you do want to use FTP with FrontPage Server Extensions 2002, be sure to configure the FTP server to disallow write access to any directory beginning with _vti, so that these directories and their contents cannot be deleted or renamed. Also, be sure to configure the FTP server so that when a user connects to the server by way of FTP, the directory root in his or her FTP client is the same as the content root for his or her Web site.

Using Telnet with FrontPage Server Extensions 2002

You can use Telnet and FrontPage Server Extensions 2002 on IIS and apache-fp servers, bearing in mind the same restrictions as for FTP.

Checking for changed permissions

The shell script, set_default_perms.sh, which is installed in the /usr/local/frontpage/version5.0 directory, sets the permissions for the contents of /usr/local/frontpage/version5.0 and all nested directories and files. After these permissions are set, FrontPage does not change them. However, if you suspect that the permissions on any /usr/local/frontpage/version5.0 files or directories have been changed, you can reapply the correct permissions by running set_default_perms.sh. This script takes no command-line arguments.

About firewalls

FrontPage Server Extensions 2002 support connectivity through firewalls. Depending on your configuration, you must make sure your firewall is open for the standard HTTP ports 80 and 443. When using a firewall, you must configure your Web sites with Basic Authentication.

Managing permissions manually

With FrontPage 2000 Server Extensions, you could bypass the built-in security management and set permissions manually on the content of a FrontPage-extended Web site. With FrontPage Server Extensions 2002, the roles and permissions have been improved, and this functionality is no longer available.

SharePoint Team Services Requirements

Before you install SharePoint™ Team Services from Microsoft® on your Web server, make sure that you have installed the required hardware and software.

Hardware requirements

Hardware requirements vary depending on the operating system you use. The following table describes the hardware requirements for the supported operating systems.

Operating System	Requirements	Recommendations for ISPs
Microsoft Windows 2000 Server or Microsoft Windows 2000 Advanced Server	200 MHz Intel Pentium processor Minimum: 128 MB RAM Recommended: 192 MB RAM 70 MB free hard-disk space, plus 5 MB minimum for each provisioned Web site	1 GHz or higher Intel Pentium III 1 GB RAM 70 MB free hard-disk space, plus 5 MB minimum for each provisioned Web site

Microsoft Windows 2000 Professional	200 MHz Intel Pentium processor Minimum: 64 MB RAM Recommended: 128 MB RAM 70 MB free hard-disk space, plus 5 MB minimum for each provisioned Web site	

Note The security features of SharePoint team Web sites require the NTFS file system. Microsoft Windows® 2000 includes a conversion utility (Convert.exe) that you can use to convert an existing file allocation table (FAT) volume to NTFS — without losing data.

Software requirements

To run SharePoint Team Services, you must first install the following software on your server:

- Microsoft Windows 2000 Professional, Server, or Advanced Server edition with Microsoft Internet Information Services (IIS) 5.0 or later with the World Wide Web service installed (Windows 2000 Server only)
- Microsoft Internet Explorer 4.0 or later

– or –

Netscape Navigator version 4 or later

Note The security features of SharePoint team Web sites require the NTFS file system. Windows 2000 includes a conversion utility (Convert.exe) that you can use to convert an existing file allocation table (FAT) volume to NTFS — without losing data.

Database requirements

In addition to the operating system and Web server software, you must also have a database installed on your server to run SharePoint Team Services. The following databases are supported:

- Microsoft SQL Server™ 7.0 or later
- Microsoft Data Engine (MSDE) 7.0 or later

Note If you do not have SQL Server installed on your server, MSDE is installed automatically when you install SharePoint Team Services. If you are using the command-line interface to install SharePoint Team Services, or are using a script to run the installation, you can specify a separate server to use for your database.

Client requirements

Any Windows, Macintosh, or UNIX client can use SharePoint team Web site features, providing the client runs the following software:

- Microsoft Internet Explorer 4.0 or later, or Netscape Navigator 4 or later
- A client program, such as Microsoft Office XP (required if the user contributes documents to a Web site, but not necessary for browsing)

FrontPage Server Extensions 2002 Requirements

Before you install Microsoft® FrontPage Server Extensions 2002 on your Web server, make sure that you have installed the required hardware and software.

Windows hardware requirements

Hardware requirements vary depending on the operating system you use. The following table describes the hardware requirements for Microsoft Windows® operating systems.

Operating system	Requirements	Recommendations for ISPs
Microsoft Windows NT Server 4.0 with Service Pack 6a	166 MHz Intel Pentium processor 128 MB RAM 13 MB free hard-disk space	1 GHz or higher Intel Pentium III 1 GB RAM 13 MB free hard-disk space
Microsoft Windows 2000 Server	P6-200 MHz Intel Pentium processor 128 MB RAM 13 MB free hard-disk space	1 GHz or higher Intel Pentium III 1 GB RAM 13 MB free hard-disk space

Note On the Windows platform, the security features of FrontPage Server Extensions 2002 require the NTFS file system. Microsoft Windows NT® includes a conversion utility (Convert.exe) that you can use to convert an existing file allocation table (FAT) volume to NTFS — without losing data.

Windows software requirements

To run FrontPage Server Extensions 2002, you must first install the following software on your server:

- Microsoft Windows 2000 Server edition with Microsoft Internet Information Services

(IIS) 5.0 or later with the World Wide Web service installed

– or –

Microsoft Windows NT Server 4.0 with Service Pack 6a or later and Microsoft Internet Information Server (IIS) 4.0 or later with the World Wide Web service installed

- Microsoft Internet Explorer 4.0 or later

UNIX requirements

Several software platforms are supported for FrontPage Server Extensions 2002 on UNIX, described in the following table.

Platform	Operating System
Alpha	Linux 6.0 and 7.0 (MIPS)
Intel x86	BSDI 3.1, 4.0 FreeBSD 3.3, 4.0 Linux 6.0 and 7.0 (Red Hat Software) Solaris 2.7, 2,8 Tru64 Unix (Digital UNIX) 4.0.f, 5.0
PA-RISC	HP/UX 10.2, 11.0
RS6000, PowerPC	AIX 4.x
Silicon Graphics	IRIX 6.x
SPARC	Solaris 2.7, 2.8

Note SCO, AIX, and MIPS Linux are supported by Microsoft Technical Support on a Best Effort Basis.

For any of the platforms in this table, you must have between 30 and 70 MB of hard-disk space available for your installation of FrontPage Server Extensions 2002, plus enough hard-disk space to host the Web sites and subwebs your users will create. These requirements are guidelines only, and may vary depending on the platform and operating system.

To run FrontPage Server Extensions 2002 on UNIX, you must first install the following software on your server:

- Apache Web Server 1.3.14
- Microsoft Internet Explorer 4.0 or later

– or –

Netscape Navigator 4 or later

Client requirements

Any Windows, Macintosh, or UNIX client can use FrontPage Server Extensions 2002 features, providing the client runs the following software:

- Microsoft Internet Explorer 4.0 or later, or Netscape Navigator 4 or later
- Microsoft Office XP (required if the user contributes to a Web site, but not necessary for browsing) or Microsoft FrontPage® 2002

  Users can also contribute to a Web site by using Office 2000 or FrontPage 2000, but a richer editing environment is available when they use Office XP or FrontPage 2002 with FrontPage Server Extensions 2002.

Installing SharePoint Team Services

SharePoint™ Team Services, a new technology from Microsoft®, is available on the Microsoft Office XP CD (Professional with FrontPage and Enterprise editions), the Microsoft FrontPage® 2002 stand-alone CD, and as a Web download for members of the Web Presence Provider for Microsoft FrontPage program. The installation process is simple; you can run the installation and set up a working SharePoint team Web site without having to make many decisions. This simple setup process requires these conditions:

- The server computer has only one virtual server.

  If you have more than one virtual server, you must install SharePoint Team Services and then choose which virtual servers to extend.
- Microsoft SQL Server™ and Microsoft Data Engine (MSDE) are not installed on the server computer.

  If SQL Server or MSDE is installed, you must supply information about the database and extend the virtual server or servers separately.
- There is no Web site content on the virtual server.

  If you already have content, then you must decide whether to replace the home page for that content with the SharePoint team Web site home page, or to create the new home page as Sharepoint.htm.

If these conditions are met, you can run the SharePoint Team Services installation, answer a few questions, and begin working with the SharePoint team Web site in your browser right away. If your system does not meet these requirements, the installation process ends at the HTML Administration pages, where you can enter database information or decide which virtual server to use for your SharePoint team Web site. For example, if you are running Microsoft Windows® 2000 Server, and you already have more than one virtual server, you must choose which virtual servers to extend from HTML Administration pages after Setup.

The following table illustrates the installation paths available on Windows 2000 Server.

Virtual servers	Database	Result
Single	None or MSDE already installed	Virtual server is extended and a SharePoint team Web site is created.
Single	SQL Server 7.0 already installed	Virtual server is not extended — use HTML Administration pages to extend the virtual server and create a SharePoint team Web site.
Multiple	Any database configuration	Virtual server is not extended — use HTML Administration pages to extend the virtual server and create a SharePoint team Web site.

The following table illustrates the installation paths available on a computer with Windows 2000 Professional.

Database	Result
None or MSDE is already installed	The default virtual server is extended and a SharePoint team Web site is created.
SQL Server 7.0 already installed	Virtual server is not extended — use HTML Administration pages to extend the virtual server and create a SharePoint team Web site.

Note that you must have the appropriate hardware and software before installing SharePoint Team Services. For more information, see SharePoint Team Services Requirements.

Note The security features of SharePoint team Web sites require the NTFS file system. Windows 2000 includes a conversion utility (Convert.exe) that you can use to convert an existing file allocation table (FAT) volume to NTFS — without losing data.

If you try to install SharePoint Team Services to a disk drive formatted with FAT, your default virtual server might not be extended. Also, any sites you create on that virtual server will not be secure.

To install SharePoint Team Services from the Office XP or FrontPage 2002 CD

1. Insert the Office XP CD into your server computer's CD-ROM drive.
2. Navigate to the SharePt folder on the CD.
3. Double-click **Setupse.exe**.
4. Follow the steps in the Setup Wizard to install SharePoint Team Services.

When the Setup Wizard is complete, you have SharePoint Team Services installed on your

server computer, and depending on your configuration, the browser is open either to the home page of a working SharePoint team Web site on your default virtual server, or to HTML Administration pages, which you can use to extend a virtual server with SharePoint Team Services.

If you are a member of the Web Presence Provider (WPP) for Microsoft FrontPage program, you can also install SharePoint Team Services from a download page on the Microsoft Web site.

To install SharePoint Team Services from the download

1. Open your browser and navigate to the SharePoint Team Services download page.
2. Click **Download Now!**
3. Select the platform and language you want to download, and then click the download file name to begin the download.
4. Click **Run this program from its current location** to begin the installation now, or **Save this program to disk** to run later.

   If you choose to **Save this program to disk**, you must then locate and run the program from your hard disk later.
5. Follow the steps in the Setup Wizard to install SharePoint Team Services.
6. Use HTML Administration pages or the command line to extend the server with SharePoint Team Services.

Extending a server with SharePoint Team Services

The part of the installation process during which SharePoint Team Services is applied to a virtual server is called extending a server. This process can also create a SharePoint team Web site. When you install SharePoint Team Services using the simple installation method, your default virtual server is extended (and a SharePoint team Web site is created) automatically.

If you do not meet the conditions for the simple installation, you must extend your virtual server in one of three ways:

- By using HTML Administration pages.
- By using the **install** operation with the command-line tools.
- By running the command-line installation in unattended mode with an edited cfgquiet.ini file.

Each of these methods of extending the server produces slightly different results. When you use HTML Administration pages to extend a server, you have control over which settings are applied when the server is extended. By using HTML Administration pages, you can configure the following elements:

- Which virtual servers are extended
- Which collaboration database on which server to use
- Web document discussion settings
- Web subscription settings

When you use the **install** operation with the command-line tools, you still choose which virtual server to extend and specify database information, but the following settings are configured for you:

- Discussions are enabled and set to unrestricted.
- Subscriptions are disabled, but the folder subscriptions setting is turned on.

Because you must specify an e-mail server for subscriptions, you must also set the **SMTPHost** or **SendMailCommand** property, the **EnableNotifications** property, the **MailSender** property, and the **MailReplyTo** property when you turn on subscriptions. For more information, see Managing Web Document Discussions and Subscriptions.

When you extend the server, you also specify the type of site you want to create on the virtual server. The following table describes the site types you can choose from when you extend the server.

Site type	Description	Command line parameter
SharePoint-based Web site	This option gives you a new SharePoint team Web site that's ready to use.	SHAREPTHP
SharePoint-based Web site (preserves home page if one exists)	This option gives you a new SharePoint team Web site, but does not replace any existing home page you already have. You can use a SharePoint Team Services-compatible Web page editor, such as FrontPage 2002, to integrate your existing content with your new SharePoint team Web site.	SHAREPT
SharePoint-enabled blank Web site (this enables features such as document libraries and lists)	This option gives you SharePoint Team Services, but no Web site content. You can still create a Web site that uses the SharePoint team Web site features, such as a discussion board or survey.	COLLAB

Blank Web site (FrontPage Server Extensions only)	This option gives you a blank Web site extended with FrontPage Server Extensions 2002 and no collaboration options.	PUBLISH

To extend the server from the command line and perform other actions in a batch process at the same time, use unattended setup and configuration. For more information, see Running Unattended Setup.

Using HTML Administration pages to extend a server

You can extend a server by using HTML Administration pages. Doing so gives you the most flexibility to choose which options you want enabled when you extend your server.

To extend a server by using HTML Administration pages

1. Open the **Server Administration** page.
2. In the list of virtual servers, next to the virtual server you want to extend, click **Extend**.
3. In the **Administrator** box, type the user name for the administrator of the virtual server.
4. If necessary, in the **Database** section, enter the collaboration database settings to use.
5. In the **Site Type** area, select a site type.
6. Click **Submit**.

Using the command line to extend a server

You can extend a server by using the command-line tools, Owsadm or Owsrmadm. To do so, you use the **install** operation. The **install** operation takes the following parameters.

Parameter	Short Form	Description
Port	-p	The port number. On IIS 4.0 or later, this value can be an instance number, such as **/LM/W3SVC/1**. If missing, this parameter defaults to port 80.
Username	-u	If your server is part of a domain, the domain name and your user name. For example, **MyDomain\MyUserName**.
DatabaseServer	-ds	Name of the server on which the collaboration database is stored. For example, **MyServer**.

DatabaseName	-dn	Name of the collaboration database. For example, **MyServer_collab**. The default value is generated automatically from the machine name and the virtual server.
DatabaseUserName	-du	User name for the administrator of the collaboration database.
DatabasePassword	-dp	Password for the database administrator user name.
Siteprovision	-sp	Creates a default SharePoint team Web site when SharePoint Team Services is installed on the virtual server. Takes the following values: **SHAREPTHP** (the default value), **SHAREPT**, **COLLAB**, **PUBLISH**.

If any of these parameters are missing, default parameters are used, although you are always prompted for a database password when you enter a database user name on the command line.

> **Note** The database server name can be set using the **DatabaseConnection** property. If this value is not set, it defaults to the local computer name.

The following example shows the syntax for the **install** operation:

```
owsadm.exe –o install –p <port> -u <username> -ds <database server> -dn
<database name> -du <database username> -dp <database password> -sp <value>
```

The values for the *siteprovision* parameter determine what (if any) default content is placed on your extended web. If you use the **PUBLISH** value, the virtual server is extended with Microsoft FrontPage Server Extensions 2002. If you use the **COLLAB** value, the virtual server is extended with SharePoint Team Services and the collaboration database is created. If you use the **SHAREPT** value, a SharePoint team Web site is also created. If you use the **SHAREPTHP** value, the SharePoint team Web site is created and the team Web site home page replaces the current default home page.

Removing SharePoint Team Services

You can temporarily or permanently remove SharePoint Team Services on a particular server. To temporarily remove SharePoint Team Services so you can clean or restore your site, you use the **Uninstall** command. In this mode, much of the data about your site is preserved, so you can extend the virtual server again and return to your original configuration.

To permanently remove SharePoint Team Services, use the **Full Uninstall** option. Doing so also removes all of the standard SharePoint team Web site folders, and any other files and folders that SharePoint Team Services installed. All of the data about your site (excluding

actual site content) is deleted when you perform a full uninstall. You can use either HTML Administration pages or the command line to uninstall SharePoint Team Services.

Using HTML Administration pages to uninstall

You can use HTML Administration pages to uninstall SharePoint Team Services from a virtual server.

To uninstall SharePoint Team Services

1. Open the **Server Administration** page.
2. In the list of virtual servers, next to the virtual server you want to uninstall, click **Administration**.
3. Under **Administration**, click **Uninstall SharePoint Team Services**.
4. If you want to uninstall SharePoint Team Services completely, including the metadata about your Web site, next to **Full Uninstall**, click **Yes**.
5. Click **Uninstall**.

Using the command line to uninstall

You can uninstall SharePoint Team Services by using the command-line tools, Owsadm or Owsrmadm. To do so, you use the **uninstall** operation or the **fulluninstall** operation. The **uninstall** operation removes SharePoint Team Services from your server. The **fulluninstall** operation also removes the collaboration database and the standard SharePoint team Web site folders from your virtual server.

The **uninstall** and **fulluninstall** operations take the *port* parameter. The following example shows the syntax for the **uninstall** operation:

```
owsadm.exe –o uninstall –p <port>
```

If you want to perform a full uninstall, but you want to preserve the collaboration database, you can use the **–c nodeldb** parameter with the **fulluninstall** operation. The following example shows the syntax to use for preserving the database:

```
owsadm.exe –o fulluninstall –p <port> -c nodeldb
```

Installing FrontPage Server Extensions 2002Server Extensions 2002

Microsoft® FrontPage® Server Extensions 2002Server Extensions 2002 can be downloaded from the Microsoft Web site. There are separate downloads and different installation methods for Microsoft Windows® and UNIX operating systems.

Note You must have the appropriate hardware and software before installing FrontPage Server Extensions 2002Server Extensions 2002. For more information, see FrontPage Server Extensions 2002 Requirements.

Installing FrontPage Server Extensions 2002 for Windows

FrontPage Server Extensions 2002 are available as part of an installation of Microsoft's SharePoint™ Team Services, or as a separate downloadable file on the Microsoft Web site.

Note On the Windows platform, the security features of FrontPage Server Extensions 2002 require the NTFS file system. Windows 2000 includes a conversion utility (Convert.exe) that you can use to convert an existing file allocation table (FAT) volume to NTFS—without losing data.

If you try to install FrontPage Server Extensions 2002 to a disk drive formatted with FAT, your default virtual server might not be extended. Also, any sites you create on that virtual server will not be secure.

To install FrontPage Server Extensions 2002 from the Office XP CD as part of the SharePoint Team Services installation

1. Insert the Office XP CD into your server computer's CD-ROM drive.
2. Navigate to the SharePt folder on the CD.
3. Double-click **Setupse.exe**.
4. Follow the steps in the Microsoft Office XP Setup Wizard to install SharePoint Team Services.
5. Use HTML Administration pages or the command line to extend the server with a **Blank Web site (FrontPage Server Extensions only)**.

This site type gives you the FrontPage Server Extensions 2002 on that virtual server.

Note that if you have a single virtual server and do not have a database installed on your server computer, you automatically get a full installation of SharePoint Team Services when you run the SharePoint Team Services Setup wizard. This full installation includes the collaboration database and SharePoint team Web site. If this is not what you want, you can uninstall SharePoint Team Services from that virtual server and extend the virtual server with FrontPage Server Extensions 2002. You can also control what gets installed by running unattended Setup with an edited Setup.ini file. For more information about a quiet installation of FrontPage Server Extensions 2002, see Running Unattended Setup.

To install FrontPage Server Extensions 2002 from the download

1. Open your browser and navigate to the FrontPage Server Extensions 2002 download page.
2. Click **Download Now!**
3. Select the platform and language you want to download, and then click the download file name to begin the download.

4. Click **Run this program from its current location** to begin the installation now, or **Save this program to disk** to run later.

If you choose to **Save this program to disk**, you must then locate and run the program from your hard disk later.

5. Follow the steps in the Setup Wizard to install FrontPage Server Extensions 2002.

6. Use HTML Administration pages or the command line to extend the server with the FrontPage Server Extensions 2002.

Installing FrontPage Server Extensions 2002 for UNIX

The installation package for a UNIX platform contains the installation script, fp_install.sh, and a compressed TAR file containing all FrontPage Server Extensions 2002 files. The tar file is named fp50.*platform*.tar.Z, where *platform* is the UNIX platform on which FrontPage Server Extensions 2002 are being installed. For example, fp40.solaris.tar.Z is the compressed tar file for Solaris servers.

Note Previous versions of FrontPage Server Extensions supported several Web servers on the UNIX platform that are no longer supported for FrontPage Server Extensions 2002. Only the Apache Web server with the FrontPage patch applied is supported for FrontPage Server Extensions 2002. For more information about the supported platforms for FrontPage Server Extensions 2002, see FrontPage Server Extensions 2002 Requirements.

The UNIX version of the FrontPage Server Extensions 2002 is available as a download from the FrontPage Server Extensions 2002 download page on the Microsoft Web site.

To download the UNIX version of the FrontPage Server Extensions 2002

1. Open your browser and navigate to the FrontPage Server Extensions 2002 download page.
2. Click **Download Now!**
3. Select the platform and language you want to download, and then click the download file name to begin the download.

Features that are installed

FrontPage Server Extensions 2002 are installed in /usr/local/frontpage by default. If you install them in another location, you must provide a link from /usr/local/frontpage to the installation area. (The installation script creates this link for you.) If you do not install FrontPage Server Extensions 2002 in /usr/local, the directory must still exist. The installation script creates this directory if it is not on your server.

Along with FrontPage Server Extensions 2002, the following features are installed:

- The program owsadm.exe, a tool for administrating FrontPage Server Extensions 2002

and FrontPage-extended webs, in /usr/local/frontpage/version5.0/bin/owsadm.exe.

- The HTML Administration Forms, used to administer server extensions from any computer connected to the Internet that has access to the server, in /usr/local/frontpage/version5.0/admin/<LCID>.

  LCID is the locale ID; for example, 1033 for US English.

- The FrontPage Apache patch, which makes it possible to install a single copy of FrontPage Server Extensions 2002 without installing stub versions of the FrontPage Server Extensions 2002 executable files in each FrontPage-extended web.

  The FrontPage Apache patch binaries and source files are installed in /usr/local/frontpage/version5.0/apache-fp.

- Other server extensions files, such as message and configuration files.

Using the Setup script

The FrontPage Server Extensions 2002 installation on the UNIX platform consists of four steps (you might not perform all the steps):

1. FrontPage Server Extensions 2002 and other files are uncompressed, untarred, and saved to their locations on the server computer.
2. If your Web server has an earlier version of the server extensions, it is upgraded.
3. If your Web server has never had the server extensions before, the extensions are installed.
4. If you have one or more virtual servers, FrontPage Server Extensions 2002 are optionally installed on those servers.

The FrontPage Server Extensions 2002 installation script is fp_install.sh. You must be logged on as "root" to run this script.

To untar FrontPage Server Extensions 2002

1. Run fp_install.sh.
2. When prompted, back up the installation directory, the server configuration file directory, and any content before installing FrontPage Server Extensions 2002.
3. When prompted, enter a server extensions directory.

   By default, FrontPage Server Extensions 2002 are installed in /usr/local/frontpage/. You can accept the default location or specify another location. If you select another location, a symbolic link is created from /usr/local/frontpage/ to the directory you choose. The installation directory must be on a local partition, and the partition must not be mounted NOSUID.

4. When prompted, untar and uncompress the FrontPage Server Extensions 2002 tar file, fp50.*platform*.tar.Z.

   If the tar file is not in the current directory, you are prompted for its location.

Upgrading earlier versions

After upgrading all servers to FrontPage Server Extensions 2002, you can set up the security of your FrontPage-extended webs interactively, or have fp_install.sh generate a script to perform the operation.

If you choose the interactive option, fp_install.sh prompts you for the UNIX user name and group name of each root web and subweb that you have upgraded. For each FrontPage-extended web, fp_install.sh then performs the change ownership (chown) operation on the content in each Web to be owned by the specified user and group.

If you choose the script option, a Bourne shell script is generated and performs all the necessary chown operations by using owsadm.exe. Before running the script, however, you must edit it by entering the UNIX user identifiers and group identifiers you want to associate with each web.

Installing a root web and subwebs for the first time

To install a root web and add subwebs, you perform the following steps:

1. Untar FrontPage Server Extensions 2002 and install them on the root web of the Web server.

   Before installing, you are prompted for a web administrator name and password. You will need this name and password when you administer or author the web from a FrontPage client. After installing the root web, you are prompted for your computer's local character encoding and default language.

2. Create subwebs within the root web, and enter the administrator user name and password for each subweb you create.

   During installation of the stub FrontPage Server Extensions 2002 on a subweb, you are prompted for the subweb name. If the name is of the form ~*webname* (a "per-user" web), then "webname" is used as the name of the subweb's owner in the UNIX file system. If the name is not in this form, you are prompted for the name of the owner. You are also prompted for the name of the group owner.

For each FrontPage-extended web, fp_install.sh will then perform the chown operation on the content in the Web to be owned by the specified UNIX user and group.

FrontPage Server Extensions 2002 then do the following for each root web or subweb:

- Expand FrontPage-based components, such as Include and Substitution components.
- Create a hyperlink map of the FrontPage-extended web.
- Extract page titles.
- Extract base URLs.

Installing on virtual servers

After you have installed to a root web and subwebs, you are prompted to install FrontPage Server Extensions 2002 on any additional virtual servers. If you choose to install on another virtual server, you are prompted for your server configuration file. Based on this file, the script displays a list of virtual servers. Before installing, you are prompted for a web administrator name and password. You will need this name and password when you administer or author the Web from a FrontPage client. After installing the root web, you are prompted for your system's local character encoding and default language.

After you have installed to the virtual server, you are prompted to create subwebs below the root web. When a subweb is being created, you are asked for a name for the subweb. If the name is of the form ~webname (a "per-user" web), then "webname" is used as the name of the subweb's owner in the UNIX file system. If the name is not in this form, you are prompted for the name of the owner. You are also prompted for the name of the group owner.

For each subweb that you choose, you are also prompted for the administrator's name and password. For each FrontPage-extended web, fp_install.sh will then perform the chown operation on the content in each Web to be owned by the specified UNIX user and group.

FrontPage Server Extensions 2002 then do the following for each root web or subweb:

- Expand FrontPage-based components, such as Include and Substitution components.
- Create a hyperlink map of the FrontPage-extended web.
- Extract page titles.
- Extract base URLs.

Installing the FrontPage Apache patch

On the Apache Web server, FrontPage Server Extensions 2002 need to modify the Web server's configuration file to mark directories containing the FrontPage Server Extensions 2002 as executable. Because creating a subweb by using the FrontPage client requires the client to write to the Web server's configuration file on these servers, which the Web server process owner usually does not have permissions to do, administrators cannot create FrontPage-extended subwebs in this way. On these servers, the owsadm.exe utility or fp_install.sh must be manually run as "root" on the host computer to modify the Web server's configuration file and create subwebs.

For the Apache Web server, FrontPage supplies a patch to overcome this limitation. With this patch, copies of FrontPage Server Extensions 2002 executable files are not stored in each FrontPage-extended web, but are stored in one centralized location. This centralization of executable files makes it unnecessary to mark directories in each Web as executable, and makes it possible for the FrontPage client to create new FrontPage-extended webs without modifying server configuration files.

You can convert your current Apache Web server to the FrontPage-patched Apache Web

server in two ways:

- You can install one of the precompiled patched executables included in the FrontPage Server Extensions 2002 tar file. That method is described in this section.
- You can manually compile-in the provided patch source files into your current Apache server. That method is described below, under “Compiling and installing your own version of the Apache patch.”

The fp_install.sh script also upgrades your Apache server to the FrontPage-patched Apache server, and upgrades any FrontPage-extended Apache webs to use the FrontPage patch.

Important Before you install the FrontPage Apache patch, back up your current Apache server, FrontPage installation directory, server configuration file directory, and all Web content.

When you run the script, it performs the following functions:

- Checks to make sure the current server has not already been upgraded.
- Moves the old Apache daemon to the file httpd.orig.
- Copies the new FrontPage-patched Apache Web server to the correct directory.
- Creates a default custom key file as /usr/local/frontpage/version5.0/apache-fp/suidkey. This file undergoes chown and chmod operations to become only readable and writable by "root." The default custom key value is dynamically generated, but for the best protection this key value should be changed on a regular basis and the server restarted.

Next, you are prompted to upgrade any FrontPage-extended Apache servers to use the new, patched server. If you choose to upgrade, the script:

- Modifies the FrontPage configuration files in /usr/local/frontpage so that they refer to the new server.
- Calls the owsadm.exe utility to upgrade each Web content area.
- Sets up your security. You can choose to set up the security of your FrontPage-based webs interactively, or you can have fp_install.sh generate a script (fp_chown.sh) to set up security.

If you choose the interactive option, the script will prompt you for the UNIX user ID and group ID of each root web and subweb that you have upgraded. For each FrontPage-extended web, fp_install.sh will perform the chown operation on all the FrontPage-created directories and content in each Web to be owned by the specified user and group.

If you choose the script option to set up security, fp_install.sh will generate the script. Before running the script, however, you must fill in the UNIX user IDs and group IDs you want to associate with each web.

Finally, you are prompted to set the default permissions. If you choose "y," the script set_default_perms.sh is invoked. This script sets permissions on /usr/local/frontpage/

version5.0 and all directories below it to the defaults. For a complete listing of default FrontPage Server Extensions 2002 permissions, see Files and Permissions on Apache.

For FrontPage Server Extensions 2002 to work once the new server is installed, the FrontPage Apache stub, in /usr/local/frontpage/version5.0/apache-fp/_vti_bin/fpexe, must be owned by, and user ID set to, "root"; fp_install.sh does this for you.

After installing the FrontPage Apache patch, you must restart Apache.

Compiling and installing your own version of the Apache patch

Microsoft makes the source code of the FrontPage Apache patch file available. To compile an Apache Web server with any custom or nonstandard modules along with the FrontPage Apache patch, follow the directions in this section. If you do not want to compile your own Apache Web server, follow the directions under “Installing the FrontPage Apache patch” to install the precompiled FrontPage Apache patch.

Before compiling and installing the FrontPage Apache patch, you must first uncompress and untar the FrontPage Server Extensions 2002 file, a process which fp_install.sh does for you. When this is completed, the FrontPage Apache patch source file, fp-patch-apache_1.3.X, will be located in a directory underneath where the original file was untarred. This patch must be compiled and the new Apache daemon installed before installation of the FrontPage Server Extensions 2002 can continue.

To compile and install your own version of the FrontPage Apache patch

1. Copy fp-patch-apache_1.3.x into your Apache server directory (usually /usr/local/apache).
2. If you have compiled an Apache Web server using a previous version of the FrontPage Apache patch source file, use the GNU version of the patch utility to remove the previous version of the FrontPage Apache patch from your Apache server:

```
%patch –p0 –R < previous_FrontPage_patch_file
```

3. Using the GNU version of the patch utility, run the patch command on the FrontPage Apache patch source file:

```
%patch –p0 <fp-patch-apache_1.3.x
```

4. Create the makefile for the patched server.

For a server that uses the 1.3.0 or later directory structure:

```
% configure --add-module=mod_frontpage.c
```

5. Build the Apache server:

```
%make
```

6. Replace the old Apache server.

7. Run the script fp_install.sh as described in “Installing the FrontPage Apache patch.” This will upgrade FrontPage-extended webs and set permissions properly on content.
8. Start the new Apache server.

Extending a server with FrontPage Server Extensions 2002

After you set up FrontPage Server Extensions 2002 on your server, you must extend the server.

You can extend the server one of three ways:

- By using HTML Administration pages.
- By using the **install** operation with the command-line tools.
- By running the command-line installation in unattended mode.

If you want to extend the server from the command line and perform other actions in a batch process, you can use unattended Setup and configuration. For more information, see Running Unattended Setup.

Using HTML Administration pages to extend a server

Using HTML Administration pages gives you the most flexibility in choosing which options you want enabled when you extend your server.

To extend a server by using HTML Administration pages on Windows

1. Open the **Server Administration** page.
2. In the list of virtual servers, next to the virtual server you want to extend, click **Extend**.
3. In the **Administrator** box, type the user name for the administrator of the virtual server.
4. Click **Submit**.

On UNIX, only those virtual servers that appear in the administration port’s configuration file show up in the list of virtual servers. So, to extend one of those virtual servers, you can use the **Extend** link next to the virtual server in the list. If you want to extend a virtual server that is not in the list, you can use the **Extend a virtual server** link on the **Server Administration** page instead.

To extend a server by using HTML Administration pages on UNIX

1. Open the **Server Administration** page.
2. Click **Extend a virtual server**, or in the list of virtual servers, next to the virtual server you want to extend, click **Extend**.
3. In the **Server Configuration File Path** box, type the full path to the configuration

file for the server.

4. In the **Port** box, type the port for the virtual server.
5. In the **Host name** box, type the host name.
6. In the **Administrator user name** box, type the user name for the administrator of the virtual server.
7. In the **Administrator Account Password** box, type the password for the administrator account.
8. In the **UNIX User Account Username** box, type the user name for the account that owns the content on the file system.
9. In the **UNIX Group Account Name** box, type the name of the group that owns the content on the file system.
10. Click **Submit**.

Using the command line to extend a server

You can extend a server by using the command-line tools, owsadm or owsrmadm. To do so, use the **install** operation, which takes the following parameters.

Parameter	Short Form	Description
Port	-p	The port number. On Internet Information Server (IIS) 4.0 or later, this value can be an instance number, such as **/LM/W3SVC/1**. If missing, this parameter defaults to port 80.
Username	-u	If your server is part of a domain, the domain name and your user name. For example, MyDomain\MyUserName.
Siteprovision	-sp	Windows only. Specifies the type of site to provision. To install FrontPage Server Extensions 2002 from the SharePoint Team Services Setup program, use the **PUBLISH** value. If you are using the FrontPage Server Extensions 2002 download to install the FrontPage Server Extensions 2002, you do not need to specify the *siteprovision* parameter.

If any of these parameters are missing, default parameters are used instead. Note that if you are installing from the SharePoint Team Services Setup program, the default *siteprovision* parameter is **SHAREPTHP** (install SharePoint Team Services and create a SharePoint team Web site). In this situation, if you want to install only FrontPage Server Extensions 2002, you

must specify the **PUBLISH** value.

The following example shows the syntax for the **install** operation on Windows:

```
owsadm.exe –o install –p <port> -u <username> -sp publish
```

On UNIX, you do not need to use the *siteprovision* parameter with the **install** operation. You do, however, need to include the *xuser* and *servconf* parameters, and optionally, the *xgroup* parameter. The following example shows the syntax for the **install** operation on UNIX:

```
owsadm.exe –o install –p <port> -u <username> -servconf <path to server
configuration file> -xuser <UNIX user account name> -xgroup <UNIX group
account name>
```

Toolbox For a complete list of the parameters required for the **install** operation and their descriptions, see the Owscmdln.xls spreadsheet.

Removing FrontPage Server Extensions 2002

You can temporarily or permanently remove FrontPage Server Extensions 2002 on a particular server. To temporarily remove the extensions so you can clean or restore your site, use the **Uninstall** command. In this mode, much of the data about your site is preserved, so you can extend the virtual server again and return to your original configuration.

To permanently remove FrontPage Server Extensions 2002, use the **Full Uninstall** option. Doing so also removes any other files and folders that FrontPage Server Extensions 2002 installed. All of the data about your site (excluding actual site content) is deleted when you perform a full uninstall, so be sure that is what you want before you use the **Full Uninstall** option.

Using HTML Administration pages to uninstall

You can use HTML Administration pages to uninstall FrontPage Server Extensions 2002.

To uninstall FrontPage Server Extensions 2002

1. Open the **Server Administration** page.
2. In the list of virtual servers, next to the virtual server you want to uninstall, click **Administration**.
3. Under **Administration**, click **Uninstall FrontPage Server Extensions 2002**.
4. If you want to uninstall FrontPage Server Extensions 2002 completely, including the metadata about your Web site, next to **Full Uninstall**, click **Yes**.
5. Click **Uninstall**.

Using the command line to uninstall

You can uninstall FrontPage Server Extensions 2002 by using the command-line tools, owsadm or owsrmadm. To do so, you use the **uninstall** or **fulluninstall** operations. The **uninstall** operation removes FrontPage Server Extensions 2002 from your server. The **fulluninstall** operation also removes the data about your site (excluding actual site content) from your virtual server.

The **uninstall** and **fulluninstall** operations take the port parameter. The following example shows the syntax for the **uninstall** operation:

```
owsadm.exe –o uninstall –p <port>
```

Running Unattended Setup

You might prefer to install Microsoft's SharePoint™ Team Services or Microsoft® FrontPage® Server Extensions 2002 without any user intervention. During an unattended (or quiet) installation, the user is never prompted to supply information and never sees any messages.

You can perform a quiet installation from the command line for either SharePoint Team Services or FrontPage Server Extensions 2002. You can run an unattended Setup by itself or as part of a script for distribution across several servers. Any output from Setup is stored in the log file in the Temp directory of the Windows volume.

To perform the unattended Setup, use the **setupse.exe** command with the ***/q*** command-line option. The **setupse.exe** command relies on settings in a text file, Cfgquiet.ini, to determine which options are incorporated into the Setup. When you run the **setupse** command, you must specify the path to the Cfgquiet.ini file by using the *logfile* parameter in the command-line syntax. To specify a quiet install, you would use the following syntax:

```
setupse.exe /q logfile=<path to cfgquiet.ini>
```

The settings in the Cfgquiet.ini file can be edited to suit your environment. For example, you can edit Cfgquiet.ini to extend specific virtual servers when SharePoint Team Services is installed, or to specify a database server to use for collaboration data, or to extend the virtual servers with FrontPage Server Extensions 2002 instead of SharePoint Team Services.

You can use any text editor, such as Notepad, to edit the Cfgquiet.ini file. The file is located in the same folder as Setupse.exe and contains all of the default options for Setup.

The Cfgquiet.ini file is divided into Security Settings, Database Settings, and Mail Settings. Basic information, such as the server type, is at the top of the file. The configuration options take the format **NAME=value**. To change an option, you replace the default value with the value you want. The following example shows the **Discussions** option set to On:

```
Discussions=On
```

The following table lists the configuration options that can be specified in the Cfgquiet.ini file.

Configuration option	Description
NumberOfServers	Specifies how many servers to configure. Accepts a numerical value — for example, **1**.
IndexServer	Controls whether the indexing service is started automatically. Supported values are **On** and **Off**.
VRootMDPath	Specifies the metabase path for Web site to extend (Microsoft® Internet Information Services 4.0 or later servers only). For example, **/LM/W3SVC/1**.
Discussions	Controls whether Web document discussions are available. Supported values are **On** and **Off**.
Notifications	Controls whether Web subscriptions are available. Supported values are **On** and **Off**.
SiteProvision	Specifies the type of installation you want: SharePoint Team Services with a SharePoint team Web site (**SHAREPTHP**), SharePoint Team Services with a SharePoint team Web site that preserves the home page if one exists (**SHAREPT**), SharePoint Team Services with a SharePoint Team Services-enabled blank Web site (**COLLAB**), or a blank Web site (**PUBLISH**). Default is **SHAREPTHP**.
Security option	**Description**
AdminAccount	Specifies the administrator's account user name. A blank or absent value defaults to the Windows NT® Administrators group.
Database settings	**Description**
Server	Specifies the SQL Server™ computer name. A blank or absent value defaults to the Windows NT computer name.

DatabaseName	Specifies the name of the collaboration database. The default value is generated automatically from the machine name and the virtual server.
UserName	Specifies the SQL Server user account. A blank or absent value defaults to the SQL sa account.

Password	Specifies the SQL Server user account password. A blank or absent value defaults to a blank password, unless MSDE was recently installed. If that is the case, the default is the password for MSDE.
Mail settings	**Description**
EmailServer	Specifies the e-mail server to use for Microsoft FrontPage forms and notifications.
AuthorEmail	Specifies the e-mail address of person or alias to contact for assistance stopping or modifying notifications.
ContactEmail	Specifies the Reply To address for e-mail messages sent by SharePoint Team Services.

Upgrading from FrontPage Server Extensions and Office Server Extensions

You can upgrade to Microsoft's SharePoint™ Team Services or Microsoft® FrontPage® Server Extensions 2002 from previous versions of FrontPage Server Extensions and Office Server Extensions, including the following:

- FrontPage 97 Server Extensions
- FrontPage 98 Server Extensions
- FrontPage 2000 Server Extensions
- Office 2000 Server Extensions (upgrade to SharePoint Team Services only)

SharePoint Team Services is a new technology, but it is based in part on the core functionality of FrontPage 2000 Server Extensions and Office 2000 Server Extensions. This sharing of code and functionality makes it easy to upgrade to SharePoint Team Services without compromising your Web content. You can even upgrade from FrontPage Server Extensions 2002 to SharePoint Team Services without losing data. You can upgrade to FrontPage Server Extensions 2002 from any previous version of FrontPage Server Extensions, but not from SharePoint Team Services or from Office 2000 Server Extensions.

After you upgrade to SharePoint Team Services or FrontPage Server Extensions 2002, you use the new SharePoint Team Services tools (HTML Administration pages and the command-line tools) to manage your Web site.

During the upgrade process, some changes are made to your Web site metadata and background structure, and some content is added to your Web site to work with SharePoint Team Services or FrontPage Server Extensions 2002. The following changes are made:

- Users are added to SharePoint Team Services or FrontPage Server Extensions 2002 user roles.

When you upgrade, already existing members of the Web site are copied from FrontPage user groups to the SharePoint Team Services or FrontPage Server Extensions 2002 user roles. The original local groups are not deleted, in case you need to recover information. However, after you upgrade, you can no longer use the FrontPage client to manage users in these groups; if you do so, your changes will not be reflected in your SharePoint team Web site or FrontPage-based Web site. You must use the SharePoint Team Services or FrontPage Server Extensions 2002 tools to administer user accounts after an upgrade.

- Web metadata is copied into the SharePoint Team Services collaboration database.

When you upgrade to SharePoint Team Services, your FrontPage Server Extensions metadata is copied into the collaboration database. Note that the metadata is not deleted from FrontPage Server Extensions directories, in case you need to uninstall and still have a working FrontPage-based Web site.

- If you are upgrading to SharePoint Team Services, new content may be added to your Web site.

The new SharePoint team Web site pages are created when you upgrade. No existing content in your Web site is deleted; however, the home page may be renamed and saved as a backup if you choose to create a SharePoint team Web site when you upgrade.

The upgrade process has two parts: you first install SharePoint Team Services or FrontPage Server Extensions 2002, then you upgrade your Web sites on a per-virtual server basis. When you upgrade a virtual server, all of the Web sites and subwebs on that virtual server are upgraded. However, when you install SharePoint Team Services or FrontPage Server Extensions 2002 on a server computer, the existing virtual servers on that server are not automatically upgraded. You must specify which virtual servers to upgrade.

Side-by-side installing and upgrading

If you have only one virtual server (for example, Microsoft Windows® 2000 Professional allows only one virtual server), you upgrade the entire server at one time. If you have a Web server with multiple virtual servers, you can decide which virtual servers to upgrade and when. It is possible to upgrade only one virtual server with SharePoint Team Services or FrontPage Server Extensions 2002 without having to upgrade any others (which means you now have a side-by-side installation). This flexibility is useful when you:

- Incrementally upgrade your Web sites.
- Continue to support older clients and server extensions, such as FrontPage 2000 Server Extensions.
- Remove and then reinstall SharePoint Team Services or FrontPage Server Extensions 2002 to repair an installation.

In a side-by-side installation, you can keep previous versions of FrontPage Server Extensions or Office 2000 Server Extensions on any virtual server when you install SharePoint Team

Services or FrontPage Server Extensions 2002 to a new virtual server. SharePoint Team Services and FrontPage Server Extensions 2002 do not overwrite any existing server extensions code on the other virtual servers during installation or upgrade. Also, you do not have to perform any special installation steps to run different versions of the server extensions or SharePoint Team Services side-by-side.

If you decide to upgrade the other virtual servers later, you can simply use the **upgrade** command from the command line or from HTML Administration pages.

Side-by-side installations and the Microsoft Management Console

When you upgrade a server to SharePoint Team Services or FrontPage Server Extensions 2002, you end up with two sets of Server Extensions tasks in the Microsoft Management Console (MMC) for each upgraded virtual server. SharePoint Team Services and FrontPage Server Extensions 2002 do not use the MMC for administration tasks, aside from creating new virtual servers, but they still have placeholder tasks in the MMC.

You can turn off the old versions of the MMC commands for upgraded virtual servers for Internet Services Manager, so you only see commands for SharePoint Team Services and FrontPage Server Extensions 2002.

To disable a snap-in in Internet Services Manager

1. Click **Start**, and then click **Run**.
2. In the **Open** box, type **MMC**, and then click **OK**.
3. On the **Console** menu, click **Open**.
4. In the File name box, type **%WINDIR%\system32\inetsrv\iis.msc**, and then click **OK**.
5. On the **Console** menu, click **Add/Remove Snap-in**.
6. Click the **Extensions** tab.
7. Clear the **Add all extensions** check box.
8. Clear the **FrontPage Server Extensions** check box.
9. Click **OK**.

Using the command line to upgrade

After you have installed SharePoint Team Services or FrontPage Server Extensions 2002, you can upgrade your virtual servers from the command line. To upgrade a virtual server, you use the **upgrade** operation. The upgrade operation takes the following parameters.

Parameter	Description
-port	The port that you want to upgrade. Optional. Defaults to port 80.

-multihost	The name or IP address of a virtual server (multihost). Optional.
-databaseserver	SharePoint Team Services only. The server on which the collaboration database exists. Optional.
-databasename	SharePoint Team Services only. The name of the SQL Server or MSDE database used for SharePoint team Web site data. Optional.
-databaseuser	SharePoint Team Services only. The administrator user name for the SQL Server™ or MSDE database. Optional.
-databasepassword	SharePoint Team Services only. The password for the SQL Server or MSDE database. Optional.
-siteprovision	SharePoint Team Services only. Specifies whether to create a default SharePoint team Web site when SharePoint Team Services is installed on the virtual server. Takes the following values: **PUBLISH**, **COLLAB**, **SHAREPT**, **SHAREPTHP** (the default value). Not available on UNIX.
-xuser	FrontPage Server Extensions 2002 for UNIX only. Specifies the UNIX user account that owns the content on the file system. Required.
-xgroup	FrontPage Server Extensions 2002 for UNIX only. Specifies the UNIX group account that owns the content on the file system. Optional.

The following example shows the syntax to use to upgrade to SharePoint Team Services on port 80:

```
owsadm.exe –o upgrade -p 80 -ds Server1 -dn Server1_collab -du User1
-dp password –sp collab
```

Using HTML Administration pages to upgrade

You can also upgrade a virtual server by using HTML Administration pages.

To upgrade a virtual server to SharePoint Team Services

1. On the server computer, click **Start**, point to **Programs**, point to **Administrative Tools**, and then click **Microsoft SharePoint Administrator**.
2. In the list of virtual servers, next to the virtual server you want to upgrade, click **Upgrade**.

   Note that if you are upgrading from a prerelease version of SharePoint Team Services, you must click **Administration**, and then click **Upgrade virtual server with Microsoft SharePoint** to upgrade to the current version.
3. In the **Administrator user name** box, type the user name for the administrator of the virtual server.
4. If necessary, in the **Database** section, enter the collaboration database settings to use.
5. In the **Site Type** area, select a site type.
6. Click **Submit**.

If you are upgrading to FrontPage Server Extensions 2002 from a previous version of the server extensions, the steps are similar.

To upgrade a virtual server to FrontPage Server Extensions 2002 (Windows)

1. On the server computer, click **Start**, point to **Programs**, point to **Administrative Tools**, and then click **Microsoft SharePoint Administrator**.
2. In the list of virtual servers, next to the virtual server you want to upgrade, click **Upgrade**.

   Note that if you are upgrading from a prerelease version of FrontPage Server Extensions 2002, you must click **Administration**, and then click **Upgrade virtual server with FrontPage Server Extensions 2002** to upgrade to the current version.
3. In the **Administrator user name** box, type the user name for the administrator of the virtual server.
4. Click **Submit**.

On UNIX, you must also supply the administrator password and user and group account names.

To upgrade a virtual server to FrontPage Server Extensions 2002 (UNIX)

1. On the server computer, open the FrontPage Server Extensions 2002 **Server Administration** page.
2. In the list of virtual servers, next to the virtual server you want to upgrade, click **Upgrade**.

Note that if you are upgrading from a prerelease version of FrontPage Server Extensions 2002, you must click **Administration**, and then click **Upgrade virtual server with FrontPage Server Extensions 2002** to upgrade to the current version.

3. In the **Administrator user name** box, type the user name for the administrator of the virtual server.
4. In the **Administrator Account Password** box, type the password for the administrator account.
5. In the **UNIX User Account Username** box, type the user name for the account that owns the content on the file system.
6. In the **UNIX Group Account Name** box, type the name of the group that owns the content on the file system.
7. Click **Submit**.

Related links

If you need to roll back to the previous version, you can uninstall SharePoint Team Services or FrontPage Server Extensions 2002 on a virtual server. For more information, see Installing SharePoint Team Services and Installing FrontPage Server Extensions 2002.

HTML Administration

Microsoft's SharePoint™ Team Services and Microsoft® FrontPage® Server Extensions 2002 include HTML Administration pages to help administer your Web sites and servers. You can use these forms on the local server computer or from a remote computer connected to the Internet. You must have Administrator rights to use HTML Administration pages.

Note For FrontPage 2000 Server Extensions and Microsoft Office 2000 Server Extensions, you used the Microsoft Management Console (MMC) to perform certain administrative tasks. With SharePoint Team Services and FrontPage Server Extensions 2002, you now perform all administrative tasks from either HTML Administration pages or the command line.

The three types of administration pages are:

- Server Administration pages
- Virtual Server Administration pages
- Site Administration pages

Note You can also perform most administration tasks by using the command line. For more information, see Command-line Administration.

Server Administration pages

The Server Administration pages allow you to administer settings for your Web server. These pages, which are created during SharePoint Team Services or FrontPage Server Extensions 2002 Setup, replace the Microsoft Management Console (MMC) interface used for administering the Microsoft FrontPage® 2000 Server Extensions. You must be a member of the local Administrators group for the server computer to view the Server Administration pages.

With the Server Administration pages, you can:

- View the list of virtual servers on your server and find information about each one.

  For example, you can upgrade your virtual servers to SharePoint Team Services or FrontPage Server Extensions 2002.

- Specify default settings for the virtual servers and Web sites on the server.

  You can specify default database, Web document discussion, Web subscription, usage analysis, server health, e-mail, and security settings. For more information, see Setting Installation Defaults.

- Specify which rights are available to be assigned to roles and users.

  You can turn off specific user rights if you do not want them to be available for the Web sites on your server. For more information, see Managing Web Site Permissions.

- Reset user passwords.

  If a user forgets his or her password, you can reset it from the Server Administration pages. For more information, see Managing Users.

The Server Administration pages are stored on the nonstandard administration port that SharePoint Team Services or FrontPage Server Extensions 2002 created during Setup on the Windows platform, or that you created on the UNIX platform by using the **setadminport** operation on the command line while running as the root account. To view the pages from a server computer running the Microsoft Windows NT® or Microsoft Windows® 2000 operating systems, on the **Start** menu click **Programs**, click **Administrative Tools**, and then click **Microsoft SharePoint Administrator**. Or, from the browser on a remote computer, type the URL to the pages on the administration port. For example: http://servername:port/fpadmdll.dll on Windows and http://servername:port/fpadmcgi.exe on UNIX.

> **Note** You can also open the administration pages locally by using localhost on the **Run** command. On the **Start** menu, click **Run**, type http://localhost:*port*/fpadmdll.dll, and then click **OK**.

There are links from the Server Administration pages to the Virtual Server and Site Administration pages for certain tasks. For example, if you set Web discussions to be turned on at the server level, you can link to the Site Administration pages to change discussion settings for just that Web site. Because of security considerations, however, there are no links from the Virtual Server or Site Administration pages to the Server Administration pages.

Virtual Server Administration pages

The Virtual Server Administration pages help control settings for any extended virtual server on your server computer. By default, a newly created virtual server inherits settings from defaults set on the Server Administration pages. You use the Virtual Server Administration pages to change these default settings and to specify what settings to use for subwebs of the virtual server. You must be a member of the local Administrators group for the server computer to view the Virtual Server Administration pages.

With the Virtual Server Administration pages, you can:

- Upgrade the virtual server to SharePoint Team Services or FrontPage Server Extensions 2002.

  After you've installed SharePoint Team Services or FrontPage Server Extensions 2002 to your server, you can upgrade individual virtual servers to use SharePoint Team Services or FrontPage Server Extensions 2002. For more information, see Upgrading from FrontPage Server Extensions and Office Server Extensions.

- Uninstall SharePoint Team Services or FrontPage Server Extensions 2002 from that virtual server.

  You can uninstall SharePoint Team Services or FrontPage Server Extensions 2002 either temporarily or permanently. For more information, see Installing SharePoint Team Services or Installing FrontPage Server Extensions 2002.

- Change configuration settings.

  You can change e-mail settings, performance tuning settings, and other settings for your virtual server. For more information, see Configuring Properties for a Virtual Server.

- Change database settings.

  For SharePoint Team Services, you can back up and restore the database, schedule automatic backups, or change the database connection settings for a virtual server. For more information, see Managing the Database.

- Set user account limits (Windows only).

  You can set quotas for how many users can be added to a virtual server. For more information, see Limiting User Accounts.

You can only get to the Virtual Server Administration pages from the Server Administration pages. To open the Virtual Server Administration page for a particular virtual server, on the **Server Administration** page, next to the virtual server you want to administer, click **Administration**.

Site Administration pages

In addition to the Server and Virtual Server Administration pages, which control settings for

each server and virtual server, there are Site Administration pages that control settings for each Web site. You must have Administrator rights to the Web site to view the Site Administration pages.

With the Site Administration pages, you can perform the following tasks:

- Manage users and roles.

  You can add or remove users, edit roles, and change which role a user is a member of. For more information, see Managing Users and Managing Roles.

- Manage Web document discussion and Web subscription settings.

  For SharePoint Team Services, you can turn Web document discussions and Web subscriptions on or off for a site, or change settings, such as when subscription notifications are sent by default. For more information, see Managing Web Document Discussions and Subscriptions.

- Perform usage analysis tasks.

  For example, you can turn usage analysis log processing on or off or specify usage analysis settings, such as log data expiration times or other scheduling details. For more information, see Analyzing Web Site Usage.

- Manage server health features.

  You can detect potential problems on your Web site and repair them with the server health tools. You can also schedule the server health check to be performed automatically. For more information, see Monitoring Server Health.

- Create, merge, or delete subwebs and manage settings for subwebs.

  If you want to add a subweb, or manage the existing subwebs to your Web site, you can manage settings for the subweb from the Site Administration pages. For more information, see the Site Administration Help for SharePoint Team Services or FrontPage Server Extensions 2002.

Note that if you are administering a subweb, the administration tasks available on the Site Administration page for the subweb are a subset of those available for the root web of the virtual server.

The Site Administration pages are stored in the _vti_bin/_vti_adm virtual directory of the Web site. The portal page for each SharePoint team Web site includes a **Site Settings** link that takes you to these pages. Even if you do not have a SharePoint team Web site, you can get to these pages for SharePoint team Web sites or FrontPage-based Web site from the FrontPage 2002 client (on the **Tools** menu, click **Server**, and then click **Administration Home**) or by typing the URL to the pages directly in the browser. The path to the Site Administration pages looks like this on Windows: http://servername:port/_vti_bin/_vti_adm/fpadmdll.dll; and like this on UNIX: http://servername:port/_vti_bin/_vti_adm/fpadmcgi.exe. The DLL file for the administration pages is stored in the ..\Program Files\Common Files\Microsoft Shared\Web Server Extensions\50\isapi_vti_adm folder. On UNIX, the site-level

fpadmcgi.exe file is stored at /user/local/frontpage/version5.0/exes/_vti_bin/_vti_adm. If the user has run the **setadminport** command, then the server version of the fpadmcgi.exe file is stored at /usr/local/frontpage/version5.0/admin-exes.

Command-line Administration

Microsoft's SharePoint™ Team Services and Microsoft® FrontPage® Server Extensions 2002 introduce two new tools for command-line administration: Owsadm and Owsrmadm. These tools can help you administer SharePoint Team Services and FrontPage Server Extensions 2002 from either the local server (Owsadm.exe) or from a remote computer (Owsrmadm.exe).

Note For FrontPage 2000 Server Extensions and Microsoft Office 2000 Server Extensions, you used the Microsoft Management Console (MMC) to perform certain administrative tasks. With SharePoint Team Services and FrontPage Server Extensions 2002, you now perform all administrative tasks from either HTML Administration pages or the command line.

These tools provide a method for performing the SharePoint Team Services and FrontPage Server Extensions 2002 administration tasks to run once, to be used in batch files, or to be used within a script. Command-line tools have a more streamlined interface than HTML Administration pages, yet they retain full functionality. For example, you can use the command-line tools to:

- Manage users and roles.
- Detect potential problems with the server extensions installation and repair it.
- Set database connection properties (SharePoint Team Services only).
- Add, delete, or merge subwebs.
- Recalculate a Web site.
- Upgrade to the new version of the server extensions.
- Uninstall the server extensions for a particular Web site.
- Set properties for a subweb, root web, or all Web sites on a server.

Toolbox For a complete list of the operations you can perform by using the command-line tools, see the Owscmdln.xls spreadsheet.

Command-line tools are not interactive

Unlike Fpsrvadm.exe in the FrontPage 2000 Server Extensions, Owsadm and Owsrmadm are not interactive tools. With the Fpsrvadm tool, you typed a command and operation, and then you were asked for parameters. With Owsadm and Owsrmadm, you type the operation and parameters all at once, and you are not queried for further information. If a required parameter is missing, the operation fails, and you must type the operation and

parameters again.

This new behavior allows better flexibility for batching commands, since the tools do not prompt you for information after you have submitted a command. However, it does mean that you cannot perform queries, such as looking up the role assigned to a particular user. If you want a more interactive tool, try using HTML Administration pages. With HTML Administration pages you can perform most of the same operations as the command-line tools, but you can also see the options you have for those operations, and look up information about each Web site, user, server, and so on.

Using the command-line tools

The command-line tools are applications for the UNIX and Microsoft Windows NT® operating systems that offer a complete set of SharePoint Team Services and FrontPage Server Extensions 2002 operations. You can use the command-line tools from the command line or from batch files. Owsadm.exe must be run on the server computer, while Owsrmadm.exe can be run from a remote location (on Windows only). The Owsadm tool can be run even when the Web server is not running (although the Web server administration service for your server may need to be running). The Web server must be running before you can use the Owsrmadm tool.

To use the Owsadm tool, you must have machine administrator rights on the server computer. To use the Owsrmadm tool, you must have machine administrator rights on the server computer, and you must have a copy of the Owsrmadm tool on your computer.

When you invoke Owsadm, you supply an operation in the form **•operation CommandName** and a set of command-line parameters in the form **•parameter value**. Each parameter for the command line also has a short form that you can use instead of the full parameter name. For example, the following command upgrades port 80 of the virtual server, sample.microsoft.com with SharePoint Team Services and a SharePoint team Web site:

```
owsadm.exe –o upgrade –u username -p 80 -ds Server1 -dn Server1_collab
-du User1 -dp password –sp sharepthp
```

For example, on a UNIX-based FrontPage-patched Apache Web server, here is how you would use Owsadm.exe to install FrontPage Server Extensions 2002 on the new virtual server, example.microsoft.com:

```
owsadm.exe –o install –u username –pw password –p 80
-m example.microsoft.com
-s /usr/local/www/apache/conf/httpd.conf -xu UNIXUserName
-xg UNIXGroupName
```

The following table explains the command and parameters from this example.

Command or parameter	Explanation
-o install	Extends the specified port with FrontPage Server Extensions.
-u username	Specifies a user with administrator rights to the server computer.
-pw password	Specifies the password for the user.
-p 80	Specifies the port number.
-m example.microsoft.com	Specifies the domain name of the virtual server.
•s/usr/local/www/apache/conf/httpd.conf	Specifies the location of the Web server's configuration file.
-xu UNIXUserName -xg UNIXGroupName	Specifies the UNIX accounts to be used. The FrontPage Server Extensions 2002 will be set to read/execute access with the set user ID bit set to the given UNIX user name and group name.

Related links

For detailed information about each command-line operation and related parameters, as well as a list of all properties that you can set, see the Owscmdln.xls spreadsheet.

For more information about remote administration, see Using the Administration Tools Remotely.

SharePoint Team Services and FrontPage Server Extensions 2002 have configuration properties that you can set by using the command line or by using HTML Administration pages. For more information, see Setting Configuration Properties.

Using the Administration Tools Remotely

Administrators may be required to create a new Web site based on Microsoft's SharePoint™ Team Services or Microsoft® FrontPage®, change permissions for an author, or perform other administrative tasks from a site remote to the server computer being administered. With SharePoint Team Services and FrontPage Server Extensions 2002, remote administration can be accomplished by using the following:

- Web page–based HTML Administration pages from a computer connected to the Internet.

- Command-line Owsrmadm tool (which is based on the server extensions administration utility Owsadm) from a Microsoft Windows® or Windows NT® personal computer.

Because remote administration of SharePoint Team Services and FrontPage Server Extensions 2002 is inherently less secure than on-site administration, be sure to follow the security precautions discussed in this topic.

Remote administration architecture

The two methods of remote administration of SharePoint Team Services and FrontPage Server Extensions 2002 — HTML Administration pages and the Owsrmadm tool — have similar architectures. Both communicate with a program on the server computer — Fpadmdll.dll for Windows servers and Fpadmcgi.exe for UNIX servers. These programs, in turn, invoke fp5awel.dll to send commands to SharePoint Team Services and FrontPage Server Extensions 2002.

Remote administration of SharePoint Team Services and FrontPage Server Extensions 2002 using the Owsrmadm utility must be initiated from a client personal computer running Windows or Windows NT. The client must have Microsoft Internet Explorer 2.0 or later and communicate to the server via HTTP using WinInet. Owsrmadm passes its command line to Fpadmdll.dll on an Internet Information Services (IIS) Web server, or to Fpadmcgi.exe on Apache.

On the Web server computer, Fpadmdll.dll or Fpadmcgi.exe acts as the form handler for any of the SharePoint Team Services and FrontPage Server Extensions 2002 HTML Administration pages.

Remote administration security

When you administer a server remotely, a wider community of users is given greater access to the Web server from the Internet, which can create a security risk. During remote administration of SharePoint Team Services and FrontPage Server Extensions 2002, an unauthorized person could gain access to Web sites based upon SharePoint Team Services or FrontPage on your server and modify Web site settings — even delete Web sites. To prevent such tampering, the following precautions are recommended:

- Use a secured connection (such as Secure Sockets Layer provides) for communication between the client and the server.

  Since information and, in some cases, user names and passwords are communicated over the network during remote administration or authoring, a secured connection will prevent your data from being read by network traffic spies. To require SSL security for authoring and site administration tasks, you use the **Require SSL for authoring and administration** option on the **Set Installation Defaults** page of the **Server Administration** page, or set the **RequireSSL** property by using the **setproperty**

operation from the command line. For more information about using the **setproperty** operation, see Setting Configuration Properties.

- Grant access to Fpadmdll.dll or Fpadmcgi.exe by using the Web server's security system.

Requiring a user to log on with a secure administrator account on the Web server prevents unauthorized access. For Windows NT and Windows 2000, you must be authenticated as a Windows administrator to use the administration tools from the local computer.

- Require the use of a nonstandard HTTP port for accessing Fpadmdll.dll or Fpadmcgi.exe.

This precaution will make it much more difficult for network spies to guess the URL of HTML Administration pages or the remote administration programs. When you installed SharePoint Team Services or FrontPage Server Extensions 2002 on the Windows platform, a nonstandard administration port is created for you. On the UNIX platform, you must create the administration port by using the **setadminport** operation on the command line while running as the root account. Use this administration port to access Fpadmdll.dll and Fpadmcgi.exe.

Note You can use Owsadm.exe to change this administration port number. Do not use Internet Information Services to change the administration port, because that can break the shortcut to HTML Administration pages from the **Start** menu.

- Use IP address mask restrictions to prevent unauthorized computers from accessing the secure administration port.

Typically, all IP addresses not associated with the owner of the server running SharePoint Team Services or FrontPage Server Extensions 2002 are denied access.

Using the command-line tools remotely

When you use Owsrmadm.exe, you form the command line in the same way that you would with the Owsadm utility, adding the Owsrmadm arguments as in the following **Upgrade** command:

```
Owsrmadm.exe –adminusername UserAccount –adminpassword ereiamjh

-targetserver https://sample.microsoft.com:1439/fpadmdll.dll

-o upgrade –p 8234 –m sample.microsoft.com
```

Note the use of a secured connection and a nonstandard port. The sample command line above is for the Windows 2000 Server or Windows NT 4.0 Server platforms. For the Windows 2000 Professional and Windows NT 4.0 Workstation platforms, the address for the target server would be https://sample.microsoft.com:1439/_sharepoint/fpadmdll.dll. For the UNIX platform, the address for the target server would be https://sample.microsoft.

com:1439/fpadmcgi.exe.

Along with supporting most of the same commands as Owsadm.exe, Owsrmadm.exe includes the following additional arguments that set up the connection to the remote server.

Argument	Description	Sample values
-targetserver	The full URL of the server-side administration program Fpadmdll.dll on IIS servers and Fpadmcgi.exe on all other servers.	A URL string, such as: https://servername:port/fpadmdll.dll For UNIX Web servers, the URL will be of the form: https://servername:port/fpadmcgi.exe
-adminusername	The user name to authenticate for access to the administration program. This is not the same as the **-username** argument; it is the user name that is used to log on for access to Fpadmdll.dll or Fpadmcgi.exe.	A user name, such as: useraccount This can be left blank if you are using Windows NT Challenge/Response Authentication or if you want to be prompted for a password when using Basic or Digest Access authentication.
-adminpassword	The password used to authenticate access to the administration script. This is not the same as the **-password** argument; it is the password that is used to log on for access to Fpadmdll.dll or Fpadmcgi.exe.	A password, such as: 787abC This can be left blank if you are using Windows NT Challenge/Response Authentication.

When you install SharePoint Team Services or FrontPage Server Extensions 2002 on the Windows platform, an administration port is created to use for remote administration. On the UNIX platform, you must create the administration port by using the **setadminport** operation on the command line while running as the root account. To use the remote command-line tool, you run operations by using this port.

To connect to the administration port

- On the Windows 2000 Server and Windows NT Server version 4.0 platforms, use the following syntax on the command line before adding any operation:

```
Owsrmadm.exe -adminusername <useraccount>
–adminpassword <password> -targetserver <for example,
https://sample.microsoft.com:1439/fpadmdll.dll>
```

- On the Windows 2000 Professional and Windows NT 4.0 Workstation platforms, use the following syntax on the command line before adding any operation:

```
Owsrmadm.exe -adminusername <useraccount>
–adminpassword <password> -targetserver <for example,
https://sample.microsoft.com:1439/_sharepoint/fpadmdll.dll>
```

- On the UNIX platform, use the following syntax on the command line before adding any operation:

```
Owsrmadm.exe -adminusername <useraccount>
–adminpassword <password> -targetserver <for example,
https://sample.microsoft.com:1439/fpadmcgi.exe>
```

Using HTML Administration pages remotely

When you install SharePoint Team Services or FrontPage Server Extensions 2002, HTML Administration pages are installed to an administration port. You use these pages on the administration port to administer your server remotely. You can open HTML Administration pages from any client computer, provided you log in by using an account that has administrator access rights to the server.

To connect to the administration port by using the HTTPS protocol

- In the **Address** box of your browser, use the HTTPS protocol and type the address to your server's administration pages, including the secure server port number.

  For example, https://sample.microsoft.com:1439/fpadmdll.dll

After you connect to the remote HTML Administration pages, you can perform any of the administration tasks as if you were connected locally.

Changing the administration port

You can change the administration port for your server to a port that is easy to remember or that is a standard installation port number for your organization. To change the administration port, you use the **setadminport** operation. The **setadminport** operation takes the *port* parameter (specifying the new port number) and the *servconf* parameter (specifying a configuration file), the *username* parameter, and the *password* parameter on

UNIX.

Changing the administration port can only be done from the command line. You must use the Owsadm.exe tool on the server computer itself to change the administration port. To change the administration port on the Windows platform, use the following syntax:

```
Owsadm.exe –o –setadminport –p <port>
```

Note that if you set up SSL on the administration port, the owsadm.exe -o setadminport command sets the non-SSL admin port but not the SSL admin port. However, the **getadminport** operation returns the port number of the SSL port, rather than the non-SSL port. If you want to change the port number for an SSL administration port, you must use Internet Services Manager to do so. Be aware, however, that the shortcut to the **Microsoft SharePoint Administrator** may not work after you change the SSL administration port.

To change the administration port on the UNIX platform, use the following syntax:

```
Owsadm.exe –o –setadminport –u <username> –p <password>
–s <path to configuration file> –p <port>
```

Related links

For more information about using the command-line tools to administer SharePoint Team Services and FrontPage Server Extensions 2002, see Command-line Administration.

For detailed information about each command-line operation and related parameters, as well as a list of all properties that you can set, see the Owscmdln.XLS spreadsheet.

For more information about HTML Administration pages, see HTML Administration.

Setting Configuration Properties

You can configure several features of Microsoft's SharePoint™ team Web sites and Microsoft® FrontPage® Server Extensions 2002 by setting the values of server extensions properties. For example, you can set a property to:

- Specify an SMTP Server to use for subscription notifications.
- Set the expiration time for Web document discussion data (SharePoint Team Services only).
- Specify whether to log server extensions data.
- Require a Secure Sockets Layer (SSL) connection between the client and server.

Note For FrontPage 2000 Server Extensions and Office 2000 Server Extensions, you used the Microsoft Management Console (MMC) to perform certain administrative tasks, such as requiring SSL. With SharePoint Team Services and FrontPage Server Extensions 2002, you now perform all administrative tasks from either HTML Administration pages or the command line.

Many properties are included as options in HTML Administration pages for your server or virtual server. Properties can also be set from the command line or manually in the Windows registry or the UNIX configuration files.

> **Note** For a complete list of the properties you can set from the command line, see the **Properties** worksheet in the Owscmdln.xls spreadsheet.

You can set most properties for a single subweb, an entire virtual server, or for all virtual servers and subwebs on a server (globally). Not all properties are available for all levels; some are only global properties, and some are only applicable at the subweb level. For example, the **RequireSSL** property can be set for either the global- or virtual-server level, but not for a single subweb.

When setting properties at different levels, it is possible to introduce conflicting settings. However, if the same property is set at more than one level, SharePoint Team Services or FrontPage Server Extensions 2002 resolves the conflict automatically. The following hierarchy is used to settle conflicts between property settings at different levels:

- Subweb properties take highest priority.
- Virtual-server properties take second priority.
- Global properties take third priority.

What do you do if a particular subweb has a property set, and you want to set a virtual-server property and have it take precedence? You simply delete the property at the subweb level, and the subweb will revert to the virtual server's property automatically.

Using the command line to set properties

Three new operations have been added to the SharePoint Team Services and FrontPage Server Extensions 2002 command-line tools to help you set configuration properties: **GetProperty**, **SetProperty**, and **DeleteProperty**. With these operations, you can query for, set, or delete property values directly from the command line. Previously, many of these properties had to be set by editing either the Windows registry or a UNIX configuration file. Because the properties are available through the command-line tools, you can set configuration properties and perform other operations by using a batch file.

When you get, set, or delete a property, you must specify the level of the Web server to which the property applies. You specify the level you want for the property in the syntax of the command. The following table lists the parameters to use to specify the level of a property.

Parameter	Scope
<none>	Gets or sets the property globally. The property applies to all virtual servers and subwebs on the server computer.

-port	Gets or sets the property by virtual server. The property applies to a single virtual server.
-web	Gets or sets the property by subweb. The property applies to a single Web site of a virtual server. You must also specify the virtual server (by using the *port* parameter) when you specify a subweb. When you use the command line to get, set, or delete a property at the subweb level, you must type the property name as vti_*propertyname*. For example, to set **RequireSSL**, you would type vti_RequireSSL.

Setting a property

When you set a property, you must specify the property as a string, although some properties are interpreted numerically. You must also specify the **propertyname** (**•pn**) and **propertyvalue** (**•pv**) when you set a property. In the following example, the **RequireSSL** property is turned on globally:

```
owsadm.exe –o setproperty –pn RequireSSL –pv enabled
```

The **RequireSSL** property specifies whether the server extensions require a Secure Sockets Layer connection between the FrontPage client and the server. To turn on this requirement, set it to **enabled**. To turn off this requirement, set it to **disabled**.

Querying for a property

You can retrieve the current state of a property by using the **GetProperty** operation. You specify the **propertyname** and the **propertyvalue** is returned. For example, to see what **RequireSSL** is currently set to globally, you type:

```
owsadm.exe –o getproperty –pn RequireSSL
```

Deleting a property

If you need to delete a property, you can use the **DeleteProperty** operation. For example, if you want a virtual server to inherit the **MailReplyTo** property from the global settings, you delete the property from the per-virtual-server area. The following example shows how to delete the **MailReplyTo** property from a virtual server on port 1088:

```
owsadm.exe -o deleteproperty -p 1088 -pn MailReplyTo
```

You can delete properties at any level — global, per virtual server, or per web — just as you can get or set a property at any of those levels.

Toolbox For more information about the properties you can get or set by using the command-line tools, see the Owscmdln.xls spreadsheet.

Setting properties manually

You can still set properties directly in the Windows registry or the UNIX configuration files. The following table describes the locations of server extensions properties on Windows and UNIX.

OS and Level	Where set
Windows Global	In the Windows registry, at HKEY_LOCAL_MACHINE\Software \Microsoft\Shared Tools\Web Server Extensions\All Ports
Windows Virtual Server	In the Windows registry, at HKEY_LOCAL_MACHINE \Software \Microsoft\Shared Tools\Web Server Extensions\Ports\Port *instance number* For example: HKEY_LOCAL_MACHINE \Software\Microsoft\Shared Tools\Web Server Extensions\Ports\Port /LM/W3SVC/*nnn:*
Windows Subweb	In _vti_pvt/Service.cnf, in the subweb or root web.
UNIX Global	In /usr/local/frontpage/version5.0/frontpage.cnf
UNIX Virtual Server	On multihosted systems, in *hostname*:*port*.cnf where *hostname* is the fully qualified domain name of the server. On a single-host system, in we*nnn*.cnf where *nnn* is the Web server's port number. In either case, the file is in /usr/local/frontpage by default.
UNIX Subweb	In _vti_pvt/service.cnf, in the subweb.

Note SharePoint Team Services and FrontPage Server Extensions 2002 assign each property a default value internally. This is true even if the variable is not present in the registry (on Windows) or in the file frontpage.cnf (on UNIX).

Properties and HTML administration

Most properties that can be set from the command line are also available as options in HTML Administration pages. For example, the **RequireSSL** property can be set by using the **Require SSL for authoring and administration** option on the **Set Installation Defaults** page (in the **Server Administration** pages) or the **Change Configuration Settings** page (in the **Virtual Server Administration** pages). If you are mainly using HTML Administration pages to perform your administration tasks, most properties will be set for you when you

select options on those pages.

Related links

For more information about using the command line to administer SharePoint Team Services and FrontPage Server Extensions 2002, see Command-line Administration.

For more information about using HTML Administration pages to administer SharePoint Team Services and FrontPage Server Extensions 2002, see HTML Administration.

Configuring Properties for a Virtual Server

In previous versions of FrontPage Server Extensions and with Microsoft® Office 2000 Server Extensions, you set configuration options for a virtual server by using commands in Microsoft Management Console (MMC). With Microsoft's SharePoint™ Team Services and Microsoft FrontPage® Server Extensions 2002, most of these configuration options are now set by using HTML Administration pages or with the command line.

Virtual server configuration settings

After you have installed SharePoint Team Services or FrontPage Server Extensions 2002 and extended a virtual server, you can configure options for the virtual server. The following options can be set at the virtual server level and affect all sites on that virtual server: authoring options, mail settings, performance tuning options, client scripting options, security settings, and database settings (SharePoint Team Services only). If you do not change these configuration settings, the virtual server inherits the settings from the **Set Installation Defaults** administration page (in the Server Administration pages) or from the global settings on the command line. You can configure these settings by using either HTML Administration pages or the command line.

Authoring options

On the **Change Configuration Settings** page (in the Virtual Server Administration pages), you can turn off authoring for a virtual server by clearing the **Enable authoring** check box. Note that even with authoring turned off for a SharePoint team Web site, users can still add items to and update information on the Web site. They cannot, however, add new pages to the site.

On the command line, you use the **Authoring** property to turn authoring on or off for a virtual server. To turn authoring on, set the property to **ENABLED**. To turn authoring off, set the property to **DISABLED**.

Mail settings

Periodically, the server sends e-mail notification (such as for Web subscriptions, invitations, document library rules, and usage analysis notifications) or other messages to users of your

site or to you, the administrator. On the **Change Configuration Settings** page (in the Virtual Server Administration pages), you can specify the SMTP mail server to use for these messages in the **Mail Settings** section. You also can type the From and Reply-to address to use for these messages, and specify which mail encoding option and character set to use in the e-mail messages.

> **Note** The mail setting can also be configured for the entire server at once by setting configuration defaults on the **Set Installation Defaults** page (in the Server Administration pages). If you set configuration defaults for your server, any new virtual servers automatically inherit the settings, although Web site administrators can override these settings for their Web sites. For more information, see Setting Installation Defaults.

On the command line, you use the following properties to control mail settings for a virtual server.

Property	Description
MailCharSet	Specifies the character set to use. Can be set to any character set value, not just the values listed on the **Change Configuration Settings** page (in the Virtual Server Administration pages). The default settings for character sets vary depending on the feature sending the e-mail content. For subscription notifications, the default character set matches the HKEY_LOCAL_MACHINE\Software\Microsoft\Office\10.0\Web Server\ServerLanguage registry entry. This registry entry is set when SharePoint Team Services or FrontPage Server Extensions 2002 is installed, and matches the language that is installed. For usage analysis notifications, document library rules notifications, and invitations, the default character set is the same as the character set for the current Microsoft FrontPage® Web site. You can change this setting in FrontPage by using the **Web Setting** command (**Tools** menu). In the **Web Setting** dialog box, click the **Language** tab, and then select a character set in the **Default page encoding** box.
MailEncoding	Specifies the type of encoding to use. The default value is **8bit** unless mail is being sent using the "iso-2022-jp" character set, in which case the default value is **7bit**.

MailReplyTo	Specifies the e-mail address to use when replying to subscription notifications messages. The default for SMTP is user@host, where user is the current user account and host is the current host name.
MailSender	Sets the user name to use as the From account when sending e-mail. Specifically, it is used as the argument to the SEND FROM: command in SMTP. The default for SMTP is user@host, where user is the current user account and host is the current host name.
SendMailCommand	Sets the name of the program to which e-mail should be piped. Typically this is sendmail, but it could be any program.
SMTPHost	Sets the name of the IP address of a host running an SMTP daemon, such as sendmail on UNIX or SMTP Server on Microsoft Windows®. By default, SharePoint Team Services and FrontPage Server Extensions 2002 assume that the daemon is listening on port 25 (the standard for SMTP), but you can override this by appending ":xx" to the name, where the xx is the port to use. Normally, you set either the **SMTPHost** or **SendmailCommand** properties, but not both, because **SendmailCommand** takes priority over **SMTPHost**.

Performance tuning options

The size of the Web sites on a server affects the server performance. On the **Change Configuration Settings** page (in the Virtual Server Administration pages), you set the performance tuning to match the size of your Web site in the **Performance Tuning** section. You can use the default settings, or specify settings for the in-memory document cache, the include-file cache, the image-file cache, the full-text search index size, and the maximum cached document size.

On the command line, you use the following properties to control performance tuning for a virtual server.

Property	Description
CacheMaxDocMeta	Sets the maximum number of documents kept in the cache while SharePoint Team Services or FrontPage Server Extensions 2002 is performing write operations.

CacheMaxImage	Sets the maximum number of images whose HEIGHT and WIDTH attributes the server extensions will cache while recalculating hyperlinks or saving a page. If you set this variable globally or per virtual server, and Web sites on your server frequently contain more than 16 images, you should increase this number. If you set this variable on an individual subweb, do not set it higher than the number of image files in the web.
CacheMaxInclude	Sets the maximum number of included files on a page (that is, files included through the Include Page component) that SharePoint Team Services or FrontPage Server Extensions 2002 caches while recalculating hyperlinks or saving the page. Increase the value of **CacheMaxInclude** to the highest number of pages that are included in any page in your Web site, if that number is higher than the default value (**16**).
CacheMaxIncludeSize	Sets the maximum document size in kilobytes that SharePoint Team Services or FrontPage Server Extensions 2002 caches internally.
CacheMinDocMeta	Sets the maximum number of documents kept in the cache while SharePoint Team Services or FrontPage Server Extensions 2002 is doing read operations. This is the maximum number of documents whose property information, such as web parameters, you want to keep in active memory. When an author opens a document after the cache is full, the cache is cleared and starts with the most recently opened document.

TextMemory	Sets full-text indexing settings for the built-in Wide Area Information Server (WAIS) search engine on the UNIX platform. Setting this property to **0** turns off full-text indexing of the Web site. A non-zero value specifies the number of megabytes of RAM the server extensions are to use during text indexing for hash-tables and other data structures. For Web sites with less than 500 pages, set the value of this property to **1**. For webs with 500 to 5000 pages, set it to **2**. For webs with more than 5000 pages, set it to **4**.

Client scripting options

You can control which types of client scripting are allowed on your virtual server. On the **Change Configuration Settings** page (in the Virtual Server Administration pages), in the **Client Scripting** section, you use the Scripting language box to select the scripting language to use. SharePoint Team Services and FrontPage Server Extensions 2002 support either JavaScript or VBScript. You can also select **no scripting** if you do not want to allow any client scripting on your virtual server.

On the command line, you use the following properties to control client scripting for a virtual server.

Property	Description
AllowExecutableScripts	Controls client scripting for a virtual server. When **AllowExecutableScripts** is set to a non-zero value, SharePoint Team Services or FrontPage Server Extensions 2002 sets the executable bit on files within executable directories. When directories are marked as executable, all files within the directory will also be marked as executable. If authors are permitted to upload into executable directories (that is, the **NoExecutableCgiUpload** variable is set to **0**), then when **AllowExecutableScripts** is set to a non-zero value, authors can execute newly uploaded CGI scripts and ISAPI extensions. If **NoExecutableCgiUpload** is set to **0** and **AllowExecutableScripts** is set to **0**, authors can upload and use ASP and IDC files, but not CGI or ISAPI files.
ScriptLanguage	Sets the language for the scripts that are generated by SharePoint Team Services or FrontPage Server Extensions 2002 to enforce any data-validation settings an author has applied to form fields. Valid values are **VBScript**, **JavaScript**, or none.

Security settings

You determine which security settings to implement on all virtual servers. On the **Change Configuration Settings** page (in the Virtual Server Administration pages), you can change security options in the **Security Settings** section. You can track information about the authoring processes by selecting the **Log authoring actions** check box. When you select this check box, an Author.log file is created in the Web site's _vti_log folder. If you want all users to use Secure Sockets Layer (SSL) security, you can select the **Require SSL for authoring and administration** check box. Finally, you can specify whether or not to allow users to store executable files (such as EXE files) on your virtual server by selecting or clearing the **Allow authors to upload executables** check box. Note that if the **Allow**

authors to upload executables check box is cleared, then users cannot upload any files to any folders that are marked executable. It does not matter whether the files being uploaded are executable files or not.

> **Note** The security options can also be configured for the entire server at once by setting configuration defaults on the **Set Installation Defaults** page (in the Server Administration pages). If you set configuration defaults for your server, any new virtual servers automatically inherit the settings, although Web site administrators can override these settings for their Web sites. For more information, see Setting Installation Defaults.

On the command line, you use the following properties to control security settings for a virtual server.

Property	Description
Logging	Specifies whether to log authoring actions. If **Logging** is set to a non-zero value, SharePoint Team Services or FrontPage Server Extensions 2002 logs all authoring operations to the file Author.log in the _vti_log directory of the Web site. Each operation is recorded with the current time, remote host, author's user name, Web name, operation performed, and the per-operation data. In the event of a security breach, this log file can be analyzed for authoring activity on the Web site.
NoExecutableCgiUpload	Specifies whether users can upload executable scripts. When **NoExecutableCgiUpload** is set to a non-zero value, new CGI scripts cannot be run. On a UNIX server, when this key is set to a non-zero value, SharePoint Team Services or FrontPage Server Extensions 2002 will not set the execute bit on any CGI scripts that an author uploads to a Web site using FrontPage. On a Windows NT server, when this key is set to a non-zero value, SharePoint Team Services and FrontPage Server Extensions 2002 will not allow the file to be uploaded.
RequireSSL	Specifies whether to require Secure Sockets Layer (SSL) security for all authoring and site-level administrative actions. When **RequireSSL** is set to enabled (or any other value except disabled), the server extensions require a Secure Sockets Layer connection between the FrontPage client and the server.

Database options (SharePoint Team Services only)

You can change the database connection settings for SharePoint Team Services at the virtual

server level. To manage database settings, you use the **Change Database Connection** page (in the Virtual Server Administration pages). For more information about changing database settings, see Managing the Database. In addition to the connection settings, you can also schedule automatic backups of the database, or perform a backup or restore process. For more information about these tasks, see Backing up and Restoring Data.

Site configuration settings

In addition to the virtual server configuration settings, you can set feature settings for your Web site. These settings can be changed by using the **Site Administration** page or the command line.

The following feature settings can be set for each individual Web:

- User and role settings

  You can add or delete users and roles, change which roles a user is assigned to, and so on. For more information about user settings, see Managing Users. For more information about role settings, see Managing Roles.

- Web document discussion and subscription settings (SharePoint Team Services only)

  You can control whether Web document discussions and subscriptions are turned on and configure other options. For more information about Web document discussion and subscription settings, see Managing Web Document Discussions and Subscriptions.

- Usage analysis settings

  You can turn usage analysis on or off for your Web site and configure other options. For more information about usage analysis settings, see Analyzing Web Site Usage.

- Server health settings

  You can turn server health on or off for your Web site and configure other options. For more information about server health settings, see Monitoring Server Health.

Using HTML Administration pages to configure properties for a virtual server

You use the **Change Configuration Settings** page in the Virtual Server Administration pages to configure your virtual server.

To configure a virtual server

1. Click **Start**, point to **Programs**, point to **Administration**, and then click **Microsoft SharePoint Administrator**.
2. On the **Server Administration** page, next to the virtual server name, click **Administration**.
3. Click **Change configuration settings**.

4. On the **Change Configuration Settings** page, select the options you want, and then click **Submit**.

Using the command line to configure properties for a virtual server

To configure settings for a virtual server from the command line, you use the **SetProperty** operation with Owsadm or Owsrmadm utilities. You can set authoring, performance tuning, client scripting, mail settings, and security settings properties by using the command line. For example, to turn authoring off for a virtual server, you would set the **Authoring** property by using the following syntax:

```
owsadm.exe –o setproperty -p 1088 –pn Authoring –pv enabled
```

This syntax turns authoring off for the virtual server at port 1088.

Toolbox For a complete list of the operations you can perform by using the command-line tools, see the Owscmdln.xls spreadsheet.

For more information about setting properties from the command line, see Setting Configuration Properties.

Setting Installation Defaults

With Microsoft's SharePoint™ Team Services and Microsoft® FrontPage® Server Extensions 2002, you can set policies for settings on your server. These policy settings (with the exception of SQL Server database settings) are default settings that are inherited by the virtual server or subwebs on your server. As with other configuration settings, most of these settings can also be administered in some form at the virtual server and subweb level. This means that Web site administrators can change these settings for their Web sites. A specific default setting is only inherited if the Web site administrator has not changed that setting for his or her particular Web site.

The SQL Server™ database settings, however, behave differently. These settings are copied into each virtual server and cannot be changed at the subweb level. Only an administrator with rights to administer the server computer can alter the SQL Server database settings. Keep in mind that because the settings are copied, if you want to make a global change to the SQL Server database, you must make the change both in the server policy and to each virtual server's SQL database settings. Note that these database settings are available only with SharePoint Team Services.

To specify default settings for your server, you use the **Set Installation Defaults** page (in the Server Administration pages). On this page you can set default setting for the following items:

- Database settings (SharePoint Team Services only)

The collaboration database is used to support the SharePoint team Web sites and the collaboration features such as Web discussions and subscriptions. Specify whether you want to use a SQL Server database or an MSDE database. To use SQL Server, you specify the server name, the database administrator user name, and the database administrator password to use for all Web sites on your server. If you want to use MSDE, select the **Use local MSDE database server** check box (this option is not available if SQL Server was installed before you installed SharePoint Team Services).

Note that the SQL Server database settings are copied to the virtual server level. If you want to change the SQL Server database settings for all existing virtual servers on this server, you must change the settings here and also on the **Change Database Connection** page for each virtual server. Any new virtual servers will be extended with the SQL database settings from the server policy. Note that MSDE database settings are inherited by the virtual servers, not copied.

- Web document discussions (SharePoint Team Services only)

You can control whether users can discuss documents, and which documents they can discuss. You can also set discussion items to be automatically deleted after a certain period of time. Web site administrators can change these settings by using the **Site Administration** page for their Web site.

- Web document subscriptions (SharePoint Team Services only)

You can turn subscriptions on or off, determine whether subscriptions can be used for both folders and documents or documents only, and specify when subscription notifications are sent. Web site administrators can change these settings by using the **Site Administration** page for their Web site.

- Usage analysis

The usage analysis features use data from your Web server log. The log is processed for each virtual server individually. You can specify whether log processing is on or off for your server (Windows platform only), schedule the processing for daily, weekly, or monthly intervals (Windows platform only), specify whether to automatically delete the usage data after a certain period of time, and specify whether to process the log file data for full days only. Web site administrators can change these settings by using the **Site Administration** page for their Web site.

- Server health (Windows platform only)

Server health processes are also run on a per-virtual server basis. Specify whether server health is turned on or off for your server, and specify when the server health checks will be performed. Web site administrators can change these settings by using the **Site Administration** page for their Web site.

- Mail settings

Periodically, the server sends e-mail notification or other messages to users of your site or

to you, the administrator. You can specify the SMTP mail server to use for these messages. You also can type the From and Reply-to address to use for these messages, and specify which mail encoding option and character set to use in the e-mail messages. Server administrators can change these settings by using the **Change Configuration Settings** page (in the Virtual Server Administration pages). Web site administrators can also change the From and Reply-to addresses on the **Change Web Subscription Settings** page (in the Site Administration pages).

- Security settings

Use this section to determine which security settings to implement on all virtual servers. You can track information about the authoring processes by selecting the **Log authoring actions** check box. If you want all users to use Secure Sockets Layer security (SSL), you can select the **Require SSL for authoring and administration** check box. Finally, you can specify whether or not to allow users to store EXE files on your virtual server by selecting or clearing the **Allow authors to upload executables** check box. Server administrators can change these settings by using the **Change Configuration Settings** page (in the Virtual Server Administration pages) for a given virtual server. Site administrators are not allowed to change the security settings for their Web site.

You use the **Set Installation Defaults** page in the Server Administration pages to specify default configuration settings for virtual servers and subwebs on your server.

To specify default settings for all Web sites on a server

1. Click **Start**, point to **Programs**, point to **Administration**, and then click **Microsoft SharePoint Administrator**.
2. On the **Server Administration** page, click **Set installation defaults**.
3. Select the settings you want to use as default settings for Web sites on the server.
4. Click **Submit**.

Creating a New Virtual Server

If you use Microsoft® Windows® 2000 Server, Windows NT® Server 4.0, or UNIX for your operating system, your platform supports multihosting. This means that you can create additional virtual servers to host Web sites. You create the virtual server by using the tools that come with your Web server (either Internet Information Services (IIS) or FrontPage-patched Apache Web server), and then extend the virtual servers with either SharePoint™ Team Services or the FrontPage® Server Extensions 2002.

Creating a new virtual server on Windows 2000 Server

To create a new virtual server on Windows 2000 Server, you use the Internet Services Manager. Before you create the virtual server, make sure you create a folder on your hard

disk (usually under the \Inetpub folder) to store the virtual server's content.

To create a virtual server on Windows 2000 Server

1. Click **Start**, point to **Programs**, point to **Administration Tools**, and then click **Internet Services Manager**.
2. Right-click the server name that you want to add a virtual server to, click **New**, and then click **Web site**.
3. Click **Next**.
4. In the **Description** box, type the description of your virtual server, and then click **Next**.
5. In the **Enter the IP address to use for this Web site** box, select the IP address you want to use.
6. In the **TCP port this web site should use (Default: 80)** box, type the port number to assign to the virtual server.
7. In the **Host Header for this site (Default: None)** box, type the header you want to use (if any), and then click **Next**.
8. In the **Path** box, type or browse to the path on your hard disk where the site content will go.
9. If you do not want to allow anonymous access to your virtual server, clear the **Allow anonymous access to this Web site** check box.
10. Click **Next**.
11. On the **Permissions** panel, select the permissions to use, and then click **Next**.

    If other users are allowed to contribute to the site, you must select at least the **Read**, **Write**, and **Browse** check boxes. If your virtual server allows scripts to be run, you must also select the **Run scripts (such as ASP)** check box. If you want to allow ISAPI applications or CGI scripts to be used on your virtual server, you must also select the **Execute (such as ISAPI applications or CGI)** check box.
12. Click **Finish**.

After you have created the new virtual server, you can extend it by using either the command-line tool or HTML Administration pages.

For more information about creating new Web sites on Windows 2000 Server, see the **Adding Sites** topic in the Help system for Internet Information Services, available from the browser on your server at http://localhost/iisHelp.

Creating a new virtual server on Windows NT Server 4.0 (FrontPage Server Extensions 2002 only)

To create a new virtual server on Windows NT Server 4.0, you use the Internet Services

Manager. Before you create the virtual server, make sure you create a folder on your hard disk (usually under the \Inetpub folder) to store the virtual server's content.

To create a virtual server on Windows NT Server 4.0

1. Click **Start**, point to **Programs**, point to **Windows NT 4.0 Option Pack**, click **Microsoft Internet Information Server**, and then click **Internet Services Manager**.
2. Right-click the server name that you want to add a virtual server to, click **New**, and then click **Web**.
3. In the **Web Site Description** box, type the description of your virtual server, and then click **Next**.
4. In the **Select the IP Address to use for this Web Site** box, select the IP address you want to use.
5. In the **TCP Port this Web Site should use (Default: 80)** box, type the port number to assign to the virtual server.
6. In the **Enter the path for your home directory** box, type or browse to the path on your hard disk where the site content will go.
7. If you do not want to allow anonymous access to your virtual server, clear the **Allow anonymous access to this Web site** check box.
8. Click **Next**.
9. Under **What access permissions do you want to set for this home directory**, select the permissions to use, and then click **Finish**.

   If other users are allowed to contribute to the site, you must select at least the **Allow Read Access**, **Allow Write Access**, and **Allow Directory Browsing** check boxes. If your virtual server allows scripts to be run, you must also select the **Allow Script Access** check box. If you want to allow ISAPI applications or CGI scripts to be used on your virtual server, you must also select the **Allow Execute Access (includes Script Access)** check box.

After you have created the new virtual server, you can extend it by using either the command-line tool or HTML Administration pages.

For more information about creating new Web sites on Windows NT Server 4.0, search for **adding sites** in the Help system for Internet Information Services, available from the browser on your server at http://localhost/iisHelp.

Creating a new virtual server on UNIX (FrontPage Server Extensions 2002 only)

You can also create additional virtual servers on the FrontPage-patched Apache Web server on the UNIX platform. For more information about creating virtual servers on Apache, see the Apache Virtual Host documentation Web site.

Optimizing Web Site Performance

The speed with which a Web server responds to requests from a client depends a lot on the size of the Web site, particularly the number of pages and the other files it contains. You can improve a Web site's response time by tuning it. When you tune a Web site, you set aside a certain amount of memory as a cache for the Web site, based on the number of Web pages. You can tune a Web site for any size, whether you have less than 100 pages or more than 1,000 pages.

In addition to controlling the document cache, when you tune performance, you can specify cache sizes for images and include files. You can also specify how large to make the search index for your Web site.

For Microsoft FrontPage 2000 Server Extensions, you used the Microsoft Management Console (MMC) to adjust performance tuning. For Microsoft's SharePoint™ Team Services and FrontPage Server Extensions 2002, you can use either HTML Administration pages or the command-line tools to tune Web site performance. If you want maximum control over performance tuning, use the command-line properties rather than HTML Administration pages to specify performance settings.

Using the command line to tune Web site performance

To tune performance, you set properties using the **SetProperty** operation with the command-line tools, Owsadm and Owsrmadm. All of the command-line performance properties can be set for either Microsoft® Windows® or UNIX and at all levels (server, virtual server, or subweb). This flexibility in levels is an advantage over HTML Administration pages, which can only be used to tune performance for an entire virtual server. By using the command line, you can also control the document cache separately for read and write operations in Microsoft FrontPage®.

The following table describes the properties you can set to control performance tuning.

Property	Description

CacheMaxDocMeta	Sets the maximum number of documents kept in the cache when FrontPage is doing write operations. This is the maximum number of documents whose property information, such as Web site parameters, you want to keep in active memory. The size of the cache affects how quickly the save and recalculate hyperlinks processes can be completed. In general, you should set this property to be greater than or equal to the number of HTML pages in your Web site that contain FrontPage components. If you have a large number of pages in your Web site, you can choose a lower value to save memory, but this will decrease the performance of the recalculate hyperlinks and save processes. Default value = **4096**.
CacheMinDocMeta	Sets the maximum number of documents kept in the cache when FrontPage is doing read operations. This is the maximum number of documents whose property information, such as Web site parameters, you want to keep in active memory. When an author opens a document after the cache is full, the cache is cleared and starts with the most recently opened document. Default value = **8**.
CacheMaxInclude	Sets the maximum number of included files on a page (that is, files included through the Include Page component) that SharePoint Team Services or FrontPage Server Extensions 2002 will cache while recalculating hyperlinks or saving the page. Increase this value to the highest number of pages that are included in any page in your Web site, if that number is higher than the default value (**16**).
CacheMaxIncludeSize	Sets the maximum document size in kilobytes that SharePoint Team Services or FrontPage Server Extensions 2002 will cache internally. Default values = **32K** on UNIX, **256K** on Windows NT® or Windows 2000.

CacheMaxImage	By default, SharePoint Team Services or FrontPage Server Extensions 2002 set the HEIGHT and WIDTH attributes in all IMG tags in pages saved to a Web site. This improves the appearance of pages when a site visitor downloads them over a slow connection. This property sets the maximum number of images whose HEIGHT and WIDTH attributes the server extensions will cache while recalculating hyperlinks or saving a page. If you set this variable globally or per virtual server, and Web sites on your server frequently contain more than 16 images, you should increase this number. If you set this variable on an individual subweb, do not set it higher than the number of image files in the Web site. Default value = **16**.

TextMemory	If you are using the built-in Wide Area Information Server (WAIS) search engine, setting this property to **0** turns off full-text indexing of the Web site. A non-zero value specifies the number of megabytes of RAM the server extensions are to use during text indexing for hash-tables and other data structures. For Web sites with less than 500 pages, set the value of this property to **1**. For Web sites with 500 to 5000 pages, set it to **2**. For Web sites with more than 5000 pages, set it to **4**. Default value = **1**.

For example, you can restrict the number of files that can be included on a page to 10 files. If you wanted to set this as a limit for all Web sites on a server, you could use the following syntax and set the **CacheMaxInclude** property globally:

```
owsadm.exe –o setproperty –pn CacheMaxInclude –pv 10
```

For more information about setting properties by using the command-line tools, see Setting Configuration Properties.

Using HTML Administration pages to tune Web site performance

You use the **Change Configuration Settings** page in the Virtual Server Administration pages to tune Web site performance. The **Performance Tuning** section on this page lets you tune performance for sites with less than 100 pages, 100-1000 pages, or greater than 1000 pages, or for sites with a custom size. When you select one of the preset sizes, the tuning properties are set for you, but you can always change a specific tuning value, or select **Custom**, and set all of the tuning values yourself.

With HTML Administration pages, you tune Web site performance on a per-virtual server basis. This means that your cache should include enough memory for any subwebs underneath your virtual server.

To tune Web site performance by using HTML Administration pages

1. Click **Start**, point to **Programs**, point to **Administrative Tools**, and then click **Microsoft SharePoint Administrator**.
2. On the **Server Administration** page, next to the virtual server you want to tune, click **Administration**.
3. Click **Change configuration settings**.
4. In the **Performance Tuning** area, in the **Tune for** box, select the number of pages in the Web site.
5. If you do not want to use the default cache sizes, specify the cache sizes to use in the **Tuning properties** area.
6. Click **Submit**.

The following table describes the tuning properties available on the **Change Configuration Settings** page.

Tuning property	Description	Default values
In-memory document cache	Sets the maximum number of documents whose property information, such as Web site parameters, you want to keep in active memory. When an author opens a document after the cache is full, the cache is cleared and starts with the most recently opened document. Specify number of documents. Equivalent to the **CacheMaxDocMeta** command-line property.	<100 pages = **4096** 100-1000 pages = **4096** >1000 pages = **16384**

Include file cache	Sets the maximum number of included files on a page (that is, files included through the **Include Page** component) that SharePoint Team Services or FrontPage Server Extensions 2002 will cache while recalculating hyperlinks or saving the page. Increase this value to the highest number of pages that are included in any page in your site, if that number is higher than the default value (**16**). Specify number of documents. Equivalent to the **CacheMaxInclude** command-line property.	<100 pages = **16** 100-1000 pages = **16** >1000 pages = **16**
Image file cache	By default, SharePoint Team Services or FrontPage Server Extensions 2002 set the HEIGHT and WIDTH attributes in all IMG tags in pages saved to a Web site. This improves the appearance of pages when a site visitor downloads them over a slow connection. This value sets the maximum number of images whose HEIGHT and WIDTH attributes the server extensions will cache while recalculating hyperlinks or saving a page. If you set this variable globally or per virtual server, and Web sites on your server frequently contain more than 16 images, you should increase this number. If you set this variable on an individual subweb, do not set it higher than the number of image files in the Web site. Specify number of images. Equivalent to the **CacheMaxImage** command-line property.	<100 pages = **16** 100-1000 pages = **16** >1000 pages = **16**

Full-text search index size	If you are using the built-in Wide Area Information Server (WAIS) search engine, setting this value to **0** turns off full-text indexing of the Web site. A non-zero value specifies the number of megabytes of RAM the server extensions are to use during text indexing for hash-tables and other data structures. Specify in megabytes (MB). Equivalent to the **TextMemory** command-line property.	<100 pages = **1** 100-1000 pages = **2** >1000 pages = **4**
Max cached document size	Sets the maximum document size in kilobytes that SharePoint Team Services or FrontPage Server Extensions 2002 will cache internally. Specify in kilobytes (KB). Equivalent to the **CacheMaxIncludeSize** command-line property.	<100 pages = **256** 100-1000 pages = **256** >1000 pages = **256**

Managing Roles

Microsoft's SharePoint™ Team Services and Microsoft® FrontPage® Server Extensions 2002 use roles to manage user rights. Each user is a member of at least one role that possesses corresponding rights. FrontPage 2000 Server Extensions supported three levels of user permissions: Browser, Author, and Administrator. SharePoint Team Services adds two roles to these permission levels, Contributor and Advanced Author. In addition, SharePoint Team Services and FrontPage Server Extensions 2002 allow you to edit the rights assigned to a role, create a new role, or delete an unused role.

> **Note** It is possible to add users to a Web site without assigning them to a role. For example, if you are creating new user accounts for the Web site, you can create the user accounts and then assign the users to roles later. You can also remove a member from all roles.

Not only do SharePoint Team Services and FrontPage Server Extensions 2002 offer greater flexibility in assigning roles, they provide user-friendly management tools. Previously, user roles for FrontPage Server Extensions were managed from within the Microsoft FrontPage client, using the **Security** option on the **Tools** menu. Now you can manage roles with either the command-line administration tool or HTML Administration pages.

Defining SharePoint Team Services and FrontPage

Server Extensions 2002 roles

SharePoint Team Services and FrontPage Server Extensions 2002 include the following roles by default:

- **Browser** — Has rights to view pages, view Web document discussions, and read lists.
- **Contributor** — Has Browser rights, plus rights to participate in Web document discussions and subscribe to documents and lists. SharePoint Team Services only. This role is named "collab" on the command line.
- **Author** — Has Contributor rights, plus rights to edit pages and directories, and edit lists.
- **Advanced Author** — Has Author rights, plus rights to define and apply themes and borders, link style sheets, and recalculate a Web site. This role is named "advauthor" on the command line.
- **Administrator** — Has all rights from other roles, plus rights to configure roles, create local machine user accounts, manage source control, create subwebs, manage Web document discussions and subscriptions, manage server health, and manage usage analysis. This role is named "admin" on the command line.

All of these roles are per-Web site in scope. So, the administrators in this list are Web site administrators. To perform some administrative tasks that affect settings for all Web sites and virtual servers on the server computer, you must be both a Web site administrator and an administrator for the server computer (also known as machine administrators).

Note For a complete lists of user rights and to see which are included in each role by default, see the **User Rights** worksheet in the Owscmdln.xls spreadsheet.

Upgrading from FrontPage 2000 roles

When you upgrade from Microsoft FrontPage 2000, your users are automatically sorted into three of the new roles:

- FrontPage 2000 Browsers are placed in the **Browser** role.
- FrontPage 2000 Authors are placed in the **Advanced Author** role.
- FrontPage 2000 Administrators are placed in the **Administrators** role.

You can continue to use the FrontPage 2000 user interface to manage users in these three roles. However, you cannot change the rights assigned to the roles or create new roles using the FrontPage 2000 user interface; those processes must be done through the command-line interface or HTML Administration pages.

Also, in FrontPage 2000, a user could only be a member of a single role. In SharePoint Team Services and FrontPage Server Extensions 2002, a user can be added to multiple roles. Because the old interface only allows you to assign a user to a single role, you might end up with conflicting permissions by using the old interface to manage a user who is a member of

the new roles structure. To use the full functionality of SharePoint Team Services and FrontPage Server Extensions 2002 roles, use the new HTML Administration pages or the command-line interface to manage users for your Web sites.

Customizing rights for roles

You can create a role or customize an existing role to include only the rights you want. For example, if you want only the Advanced Author to be able to edit lists on the site, you can remove the Author Lists right from the Author role.

Some rights, however, depend on other rights. For example, you must be able to edit a page before you can apply a theme. So if you delete the Author Pages right from a role, even though a user in that role might still have the Theme Web right, the user would be unable to open a page and apply a theme.

To prevent such inconsistencies, if a right is deleted from a role, any rights dependent on that right are also deleted, as in the above example. When the Author Pages right is deleted, the Theme Web right is also deleted. In the same way, if you add a right that requires another right, the required right is also added. So, if you grant the Theme Web right to a user, the Author Pages right is granted automatically.

> **Note** For more information about dependencies in user rights, see the **User Rights** worksheet in the Owscmdln.xls spreadsheet.

Security and user rights

User rights are used to give users permission to perform certain actions on a Web site, and to restrict other users from performing those actions. Some rights, however, do not provide complete security. The Theme Web, Border Web, and Link Style Sheet rights allow users to make changes to an entire Web site. Any user with Author Pages rights, however, can perform the same changes on a page-by-page basis in the actual HTML code. Be aware, that if you give a user the Author Pages right (by assigning them to a role that contains the right), you are also giving them the ability to change the theme, border, and style sheet for individual pages in your Web site.

Another right that does not provide strict security is the Manage Usage Analysis right, because any user running a SharePoint Team Services–compatible Web page editor, such as FrontPage 2002, can run usage analysis reports. The Manage Usage Analysis right simply provides security for scheduling the usage analysis processes.

Aside from these non-secure rights, some rights depend on others to perform a complete action. For example, if you have the Manage Subwebs right, you still cannot delete a particular subweb unless you have permissions on that subweb. So, if you are not listed as a user of that subweb, you may not be able to delete it, even though you have the Manage Subwebs right. Similarly, if you do not have the Design Lists right, but you do have the Author Pages right, you can potentially edit the page and break the lists (by deleting the

view files and other files required for the list to work properly).

When you assign rights to users, be sure that you assign the appropriate rights, and do not unintentionally allow users to perform more actions that you want on your Web site. Conversely, be sure that users are not unintentionally restricted from performing the actions they need to perform.

Using the command line to manage roles

There are several command-line operations that you use to manage roles for SharePoint Team Services and FrontPage Server Extensions 2002. You use the **roles** operation to add or delete a role, and the **rolerights** operation to add, delete, or set rights for a role.

Creating a new role

You can create a new role from the command line. To do so, you use the **roles** operation to create the role. For example, to create a new role called Interns for a Web site named InternWeb, you would type the following command:

```
owsadm.exe -o roles –web /internweb –c add
-name Interns
```

This command creates a role called Interns but does not assign any rights to that role. To assign rights, you must use the **rolerights** operation.

Setting rights for a role

To set rights for a role, you use the **rolerights** operation. This operation takes the *command* parameter with the following values: **add**, **del**, and **delall**. You choose the value based on which action you want to perform. For example, if you wanted to give the Interns role the rights to view pages in the Web site, you would type the following command:

```
owsadm.exe -o rolerights -w internweb -n Interns -c add
-r ViewPages
```

Note For the complete list of user rights, with definitions and information about which roles include those rights by default, see the **User Rights** worksheet in the Owscmdln.xls spreadsheet.

You can add multiple rights to a single role by using the **rolerights** operation. For example, if you had not yet assigned any rights to the role and you wanted give the interns the ability to view and edit pages, you would type the following command:

```
owsadm.exe -o rolerights -web /internweb -name Interns
-c add -r viewpages,authorpages
```

If you want to delete a single right from a role, you use the **del** (delete) value with the *command* (*-c*) parameter. For example, to remove the Author Pages right from the Interns role, you would type the following command:

```
owsadm.exe -o rolerights –web /internweb -name Interns
–c del -r authorpages
```

If you want to delete all rights from a role, in preparation for adding a different right, you can use the **delall** value with the *command* (*-c*) parameter. For example, to delete all rights in the Interns role, you would type:

```
owsadm.exe -o rolerights –web /internweb -name Interns
–c delall
```

Deleting an existing role

If you want to delete a role, you use the **roles** operation with the **del** value with the *-c* parameter. For example, when you are done with the Interns role, you can delete it by typing the following command:

```
owsadm.exe -o roles –web /internweb –c del -name Interns
```

Cloning a role

You can also clone a role from the command line. For either the Microsoft Windows® or UNIX operating systems, you can clone both the role and the users at the same time. To clone a role, you use the *clonerights* (*-cr*) parameter. You can also clone the users of that role by using the optional *cloneusers* (*-cu*) parameter. Note that when you use the *cloneusers* parameter, it must be the last parameter in the command. For example, to clone the Interns role and make a new Staff role and duplicate the role's members, you would type:

```
owsadm.exe -o roles –web /internweb -c add -name staff -cr Interns -cu
```

Using HTML Administration pages to manage roles

You can manage roles from the **Site Administration** page for your Web site. To manage roles, you follow the **Manage roles** link on the **Site Administration** page to the **Manage Roles** page. By using this page, you can view a list of roles, change which rights are included in a role, add a new role, or delete a role.

To view the Site Administration page

- If you are a server administrator, on the server computer click **Start**, point to **Programs**, point to **Administrative Tools**, and click **Microsoft SharePoint Administrator**, and then on the **Server Administration** page, click the name of the site you want to manage.

- If you are a site administrator, on your Web site, click **Site Settings**, and then under **Web Administration**, click **Go to Site Administration**.

If you want to add a new role, you use the **Manage Roles** page.

To view a list of roles

- On the **Site Administration** page for your Web site, under **Users and Roles**, click **Manage Roles**.

  The roles available for the Web site are displayed on the **Manage Roles** page.

You can add new roles for use on your site from the **Manage Roles** page.

To add a new role

1. On the **Manage Roles** page, click **Add a role**.
2. In the **Role Name and Description** area, type the name and description for your new role.
3. In the **Rights** area, select the rights you want to include in the new role.
4. Click **Create Role**.

You can create a new role based on an existing role, and even copy the users of the existing role into your new role.

To copy an existing role

1. On the **Manage Roles** page, click the role you want to copy.
2. On the **Edit Role <Rolename>** page, click **Copy Role**.
3. On the **Copy the role <Rolename>** page, in the **Role Name and Description** area, type the name and description for your new role.
4. If you want to copy the users from the existing role into your new role, select the **Copy users from <rolename>** check box.
5. In the **Rights** area, select any additional rights you want the role to contain.
6. Click **Create Role**.

You can also edit an existing role to change the rights assigned to that role.

To edit an existing role

1. On the **Manage Roles** page, click the role you want to change.
2. On the **Edit Role <Rolename>** page, select the rights you want to include and clear any rights that you do not want.
3. Click **Submit**.

If you find that a role is not used, you can delete the role.

To delete an existing role

1. On the **Manage Roles** page, select the check box next to the role you want to delete.
2. Click **Delete selected role(s)**.

Related links

For information about assigning users to roles, see Managing Users.

For more information about permissions, see Managing Web Site Permissions.

For more information about user roles and security, see Windows Security Model and FrontPage Server Extensions Security Under UNIX.

Managing Users

Every Web site has users, and part of your job as administrator is to make sure the users of a Web site have the appropriate permissions to the site. To get permissions to the site, users must be added to the site and assigned to a role.

You can add users to your site by using either the command-line tools or HTML Administration pages for your Web site. On the Microsoft® Windows® platform, when you add a user to your site, you can either add the user with his domain account, or you can automatically create machine accounts for each user. On the UNIX platform, the Web site account is automatically created.

When you create accounts for users, you assign passwords to each user. When the user logs onto the site, he can change his password. If he loses or forgets his password, he cannot look it up. However, you can reset his password by using HTML Administration pages or the command line.

Using the command line to manage users

You can assign users to a role from the command line by using the **roleusers** or **userroles** operations. The **roleusers** operation allows you to add or delete multiple users for a given role. The **userroles** operation allows you to assign a user to one or more roles. These operations both take the *command* (*-c*) parameter with the following values: **set**, **add**, **del**, and **delall**.

With **roleusers**, you can assign multiple users to a role at one time. For example, to add User1, User2, and User3 to the Author role, you would type:

```
owsadm.exe -o roleusers –c add –u user1,user2,user3
–web /subweb -name Author
```

Note When adding multiple users, the user names are separated by commas and no spaces: user1,user2,user3.

With **userroles**, you can assign multiple roles to a single user. For example, if you want to assign a user to a standard role, and also to a custom role you created to perform a specific action, you can do so by using the **userroles** command. To assign the Contributor role ("collab" on the command line) and a custom role named Interns to User1, you would type:

```
owsadm.exe -o userroles –c add –u user1 –web /subweb
-name Collab,Interns
```

Note When adding multiple roles, the role names are separated by commas and no spaces: role1,role2,role3.

You can use either operation to delete a user from a role. For example, to delete User1 from the Author role, you would type either of the following commands:

```
owsadm.exe -o roleusers –c del –u user1 –web /subweb -name Author
```

```
owsadm.exe -o userroles –c del –u user1 –web /subweb -name Author
```

To delete all users from a role, you use the **roleusers** command. For example, to delete all users from the Author role, you would type:

```
owsadm.exe -o roleusers –c delall –web /subweb -name Author
```

To remove a user from all roles he or she is assigned to, you use the **userroles** command. For example, to remove all roles for User1, you would type:

```
owsadm.exe -o userroles –c delall –u user1 –web /subweb
```

Using HTML Administration pages to manage users

You can manage users from the **Site Administration** page for your Web site. To manage users, you follow the **Manage Users** link on the **Site Administration** page to the **Manage Users** page. By using this page, you can view a list of users, check which role a user is assigned to, add new users, delete users, or assign users to roles.

To view the Site Administration page

- If you are a server administrator, on the server computer click **Start**, point to **Programs**, point to **Administrative Tools**, and then click **Microsoft SharePoint Administrator**, and then on the **Server Administration** page, click the name of the site you want to manage.
- If you are a site administrator, on your Web site, click **Site Settings**, and then under **Web Administration**, click **Go to Site Administration**.

If you want to view which roles a user is assigned to, you use the **Manage Users** page.

To view which roles are assigned to a user

- On the **Site Administration** page for your web, under **Users and Roles**, click **Manage Users**.

  The users of the Web site and the roles they are assigned to are displayed on the **Manage Users** page.

If you want to change which role a user is assigned to, you can click the user name, and then select a new role.

To change which role a user is assigned to

1. On the **Manage Users** page, click the user name you want to change.
2. In the **User Role** area, select the roles you want the user to be assigned to.
3. Click **Submit**.

You can add new users to your site from the **Manage Users** page.

To add a new user

1. On the **Manage Users** page, click **Add a user**.
2. On the **Add a User** page, in the **User** area, fill in the information about the user.

   If you are adding a new user, select **Add a new user with the following information**, and then fill in the user name and password, and then confirm the password.

   If you are adding an existing account, select **Add a user or group name (For example, DOMAIN\name)**, and then fill in the user name. Note that this option is only available on the Windows platform. This option is used to either add an existing user with a domain account to a Web site or subweb, or to add an existing user with a local machine account to a subweb.
3. In the **User Role** area, select the roles you want the user to be assigned to.
4. Click **Add User**.

You can also delete users from all roles from the **Manage Users** page. Note that this does not delete the user account, but does remove all rights to the Web site.

To delete a user from all roles

1. On the **Manage Users** page, select the check box next to the user you want to delete.
2. Click **Delete selected user(s) from all roles**.

Note If your site has user account limits, and you want to delete the user account rather than just remove the user from all roles, you can use the **Manage Virtual Server Accounts** page in the Site Administration pages for the virtual server. For more information, see Limiting User Accounts.

Creating local machine accounts for users on the Windows platform

In addition to adding users to a Web site and controlling which roles they are assigned, you can also use the command line and HTML Administration pages to create local machine accounts for users. So, for example, if you want to add a new user to your Web who does not have an account on your network domain, you can do so by using the **users** command or the **Add a User** page. On the UNIX platform, local machine accounts (.htaccess users) are the only type of accounts possible. It is only on the Windows platform that you must choose which type of account to create.

Important Note that creating local machine accounts is the default method for creating accounts on the Windows platform. If you don't want to use local machine accounts, you can turn off local machine accounts by setting the **NoMachineGroups** property to **1**. You must set this property before you extend any virtual servers with Microsoft's SharePoint™ Team Services or Microsoft FrontPage® Server Extensions 2002. For more information about setting properties, see Setting Configuration Properties.

If you are using local machine accounts exclusively on the Windows platform, you can set a property to bypass checks for domain accounts whenever you create a new user or invite a member to your site. To specify that you are using local accounts, rather than domain accounts, set the **LocalNTAccountsOnly** property to **1**. Note that when you set **LocalNTAccountsOnly** to **1**, you cannot use domain accounts at all, even if a particular domain account is valid. Use the following syntax to set **LocalNTAccountsOnly** for a server:

```
owsadm.exe –o setproperty –pn LocalNTAccountsOnly –pv 1
```

Using the command line to create a local machine account

You use the users operation to create a local machine account for a user. The **users** operation allows you to add the account and set the password for the user. It does not allow you to assign the user to a role. The **users** operation takes the following parameters: *command* (**add**, **del**, or **changepassword**), *web*, *username*, and *port* (optional). For example, to add a user account called "localuser" to the virtual server at port 80, you would use the following syntax:

```
owsadm.exe -o users –c add –u localuser –port 80
```

Note You can also assign a password when you create a new user account. To do so, use the *password* parameter with the **user** operation.

After the account has been created, you can then assign the user to a role on your site (either a Web site or subweb). To add the user to your site and assign a role to him or her, you use the **userroles** operation. The **userroles** operation adds, deletes, or deletes all roles for a user. The operation takes the following parameters: *command* (**set**, **add**, **del**, or **delall**), *username*, *web*, *name* (name of role), and *port* (optional). For example, to assign the "localuser" account to the Contributor role on a subweb named UserWeb, you would use the following syntax:

```
owsadm.exe -o userroles –c add –u localuser –web /userweb
–name collab
```

If you want to add multiple users to the same role, you can use the **roleusers** operation. The **roleusers** operation adds, deletes, or deletes users to a particular role. The operation

takes the following parameters: *command* (**set**, **add**, **del**, or **delall**), *username* (single user name or a comma-delimited list of user names), *web*, *name* (name of role), and *port* (optional). For example, to assign the several local users to the Contributor role on a subweb named UserWeb, you would use the following syntax:

```
owsadm.exe –o roleusers –c add –u localuser1,localuser2,localuser3 –web /
userweb –name collab
```

Using HTML Administration pages to create a local machine account

You can add users with local machine accounts from the **Site Administration** page for your virtual server or subweb. To add a user, you follow the **Manage Users** link on the **Site Administration** page to the **Manage Users** page.

To view the virtual server Site Administration page

- If you are a server administrator, on the server computer click **Start**, point to **Programs**, point to **Administrative Tools**, click **Microsoft SharePoint Administrator**, and then on the **Server Administration** page, click the name of the site you want to manage.
- If you are a site administrator, on your Web site, click **Site Settings**, and then under **Web Administration**, click **Go to Site Administration**.

To create a local machine account

1. On the **Manage Users** page, click **Add a user**.
2. On the **Add a User** page, in the **User** area, select **Add a new user with the following information**.

   Note that the **Add user or group name (for example, DOMAIN\name)** option is used to either add an existing user with a domain account to a Web site or subweb, or to add an existing user with a local machine account to a subweb.
3. In the **User name** box, type the user name.
4. In the **Password** box, type the password, and then type it again in the **Confirm password** box.
5. In the **User Role** area, select the roles you want the user to be assigned to.
6. Click **Add User**.

Resetting user passwords

If a user forgets his or her password, you can reset the password by using HTML Administration pages or the command line. You must have Administrator rights to the server computer to reset a user password.

To reset a user's password on the command line, you use the **users** operation with the **changepassword** command. For example, to reset the password for User1 on port 80, you would type:

```
owsadm.exe –o users –c changepassword newpassword –u User1 –p 80
```

To reset a password by using HTML Administration pages, you use the Server Administration pages.

To reset a user password by using HTML Administration pages

1. On your server computer, click **Start**, point to **Programs**, point to **Administrative Tools**, and then click **Microsoft SharePoint Administrator**.
2. Click **Reset user password**.
3. In the **Virtual server** box, select the virtual server that the user belongs to.
4. In the **Web name** box, type the name of the Web site that the user is a member of.
5. In the **User name** box, type the user's account name.
6. In the **New password** box, type a new password for the user.
7. In the **Confirm new password** box, type the new password again.
8. Click **Submit**.

Related links

For information about creating, editing, or deleting roles, see Managing Roles.

For more information about permissions, see Managing Web Site Permissions.

On the Windows platform, you can set quotas and determine how many new user accounts can be created for each virtual server. For more information about setting quotas, see Limiting User Accounts.

For more information about user roles and security, see Windows Security Model and FrontPage Server Extensions Security Under UNIX.

Managing Web Site Permissions

Each Web site and each Web server has different security requirements. An administrator with a single Web site for five users on an intranet, for example, will not need to worry as much about creating a secure Web site as an administrator at an Internet Service Provider (ISP) running several Web servers with hundreds of Web sites and thousands of users. Microsoft's SharePoint™ Team Services and Microsoft FrontPage® Server Extensions 2002 are built to be scalable, and in the area of Web site permissions you can choose to scale them to be as simple or complex as needed.

With SharePoint Team Services and FrontPage Server Extensions 2002, you can choose to:

- Specify which rights are available for a server.

If you do not want to allow certain actions on your server, you can turn off the associated rights.

- Control anonymous access to a Web site.

You can choose whether users can contribute to your site anonymously, and determine what actions anonymous users can perform.

- Create unique permissions for a subweb.

If you manage multiple Web sites and subwebs, you can set separate permissions for each subweb, so that you can have unique users and assign them rights specific to your subweb.

Specifying which rights are available for a server

Server administrators can determine which user rights are available for use on the Web server. As an administrator, if you do not want to allow users of a Web site to perform certain actions, you can disable the associated right on the server. For example, if you do not want users to be able to recalculate a Web site, you can disable the **Recalc Web** right. When you disable a right on your server, it cannot be assigned to any role and, as a result, cannot be granted to any user of the server. Note that if a user already has a right, and you disable the right, the right is also disabled for that user.

Using the command line to specify available rights

To specify available rights on the command line, you set the **GlobalRightsMask** property by using the **rightsmask** operation and the Owsadm or Owsrmadm utilities. The **rightsmask** operation can only be used at the global level and takes the following parameters: *command* (**add**/**set**/**del**/**delall**), *right* (name of right or a comma-separated list of rights), and *port* (optional — only accepts **global** as the value). Any right that you remove from **GlobalRightsMask** property is no longer available to any user on the server.

Note Use the **set** command if you want to reset the rights associated with a role. For example, if you want to change the Contributor role to only allow browsing, you would use the **set** command and specify the **viewpages** right. When you use the **set** command, any rights not specified in the syntax are removed.

The following sample syntax shows how to set the **GlobalRightsMask** property to remove the Theme Web right:

```
owsadm.exe -o rightsmask -c del -r themeweb
```

Note For a complete lists of user rights and to see which are included in each role by default, see the **User Rights** worksheet in the Owscmdln.xls spreadsheet.

Using HTML Administration pages to specify

available rights

You use the **Set List of Available Rights** page in the Server Administration pages to specify which rights are available for roles on a server.

To specify the available rights for a server

1. On your server computer, click **Start**, point to **Programs**, point to **Administrative Tools**, and then click **Microsoft SharePoint Administrator**.
2. On the **Server Administration** page, click **Set list of available rights**.
3. Select or clear the check boxes next to the rights you want to enable or disable.

   You can select all rights by selecting the **Select All** check box. You can clear all rights by clearing the **Select All** check box.
4. Click **Submit**.

Controlling anonymous access to a Web site

If you want users to be able to contribute to your site anonymously, you can configure your site to allow anonymous access. Anonymous access relies on the anonymous user account on your Web server. This account is created and maintained by your Web server (either Internet Information Services or FrontPage-patched Apache Web server), not by SharePoint Team Services and FrontPage Server Extensions 2002. On IIS, the anonymous user account is usually IUSR_*ComputerName*. Apache works differently than IIS. Rather than using a separate anonymous account, when an anonymous user tries to view your site, Apache accesses your data as the user it was configured to run as, resulting in anonymous access. When you turn on anonymous access in SharePoint Team Services and FrontPage Server Extensions 2002, you are enabling that user account for your Web site.

On Microsoft® Windows® operating systems, when you allow anonymous access, you must also assign the anonymous user to a role. For example, you can assign the anonymous user to the Browser role, so that anonymous users can browse pages, but not add to or change information on your Web site. Note that even though the anonymous user can be assigned to any role, even Administrator, the anonymous account is never able to use the Site Administration pages.

If you don't want to allow anonymous access to your Web site, you can turn it off. To control anonymous access, you use HTML Administration pages or the command-line utilities Owsadm or Owsrmadm.

Using the command line to control anonymous access

You can assign a role to the anonymous user by using the **anonrole** operation with the command-line tools. The **anonrole** operation takes the *name* and *web* parameters. For

example, to assign the anonymous user to the Browser role, you would type:

```
owsadm.exe –o anonrole –w webname –n browser
```

Using HTML Administration pages to control anonymous access

You can also control anonymous access by using HTML Administration pages. The **Change Anonymous Access Settings** page lets you turn anonymous access on and select the role to assign to the anonymous user.

To view the Site Administration page

- If you are a server administrator, on the server computer click **Start**, point to **Programs**, point to **Administrative Tools**, and then click **Microsoft SharePoint Administrator**, and then on the **Server Administration** page, click the name of the site you want to manage.
- If you are a site administrator, on your Web site, click **Site Settings**, and then under **Web Administration**, click **Go to Site Administration**.

From the **Site Administration** page, you can manage settings for anonymous access.

To turn anonymous access on or off by using Site Administration pages

1. On the **Site Administration** page for your subweb, in the **Users and Roles** section, click **Change anonymous access settings**.
2. Under **Anonymous Access is**, select **On** or **Off**.
3. On the Windows platform, in the **Assign anonymous users to the following role** box, select a role.
4. Click **Submit**.

Creating unique permissions for a subweb

By default, a subweb inherits all users and permissions from its parent web (the virtual server root web). Any changes to the users on the virtual server are duplicated on the subweb, and any changes to the users on the subweb are duplicated on the virtual server. This works well if you want all users to access all Web sites on a particular virtual server, but if you need different sets of users to use different subwebs on a virtual server, you need unique permissions.

When you choose to have unique permissions, the roles.ini file from the virtual server is copied into your subweb. So, you start with a duplicate of the users and permissions from the virtual server. But you can now add or delete users, or change permissions by using the command line or HTML Administration pages, and the changes will be effective for only your subweb.

You can change the setting for inheriting permissions by using the Site Administration pages

or by using the command-line tools.

Using the command line to create unique permissions

You use the **setperms** operation to toggle whether a subweb uses unique permissions. The **setperms** operation takes the *web* and *inherit* parameters. For example, to set a subweb named Interns to have unique permissions, you would type:

```
owsadm.exe -o setperms –web /Interns -inherit true
```

Using HTML Administration pages to create unique permissions

You can also use HTML Administration pages to set unique permissions for a subweb. Permissions for the subweb are managed from the administration page for the subweb.

To set unique permissions by using the Site Administration pages

1. If you are a server administrator, on the server computer click **Start**, point to **Programs**, point to **Administrative Tools**, click **Microsoft SharePoint Administrator**, and then on the **Server Administration** page, click the name of the site you want to manage.

   If you are a site administrator, on your Web site, click **Site Settings**, and then under **Web Administration**, click **Go to Site Administration**.
2. In the **Subweb** section, click the name of your subweb to view the **Site Administration** page for the subweb.
3. In the **Users and Roles** section, click **Change subweb permissions**.
4. Under **Security permissions**, click **Use unique permissions for this Web site**.
5. Click **Submit**.

If you want to return to using the same permissions as the parent Web site, you can also change back by using HTML Administration pages.

To return to the parent Web's permissions

1. On the Site Administration page for your subweb, in the **Users and Roles** section, click **Change subweb permissions**.
2. Under **Security permissions**, click **Use same permissions as parent Web site**.
3. Click **Submit**.

Related links

For information about creating, editing, or deleting roles, see Managing Roles.

For information about assigning users to roles, see Managing Users.

For more information about user roles and security, see Windows Security Model and FrontPage Server Extensions Security Under UNIX.

Delegating Administration Tasks to Site Owners

If you administer multiple Web sites or virtual servers, you may find that performing tasks such as adding users, checking server health, or running usage analysis for each virtual server or Web site can become burdensome — especially if you manage more than five Web sites. With Microsoft's SharePoint™ Team Services and Microsoft® FrontPage® Server Extensions 2002, you can use roles and rights to delegate such administrative tasks to trusted users and keep high-level administrator tasks (such as extending a new Web site or setting defaults) to yourself.

By default, an administrator can:

- Add new users (per subweb).
- Change user roles (per subweb).
- Manage Web document discussions and Web subscriptions (per virtual server • SharePoint Team Services only).
- Create, delete, or merge subwebs (per virtual server).
- Check server health (per subweb).
- Schedule and run usage analysis reports (per virtual server).

If you want a user, such as an Advanced Author, to be able to perform an administration task, you can either assign them to the Administrator role, or assign them to another role that has the user right needed to perform the task. Note that some of these tasks can only be performed at the virtual server level, as indicated above. Be aware that if you give a user the rights to perform such tasks, the user's actions affect not only her own Web site, but all Web sites on the virtual server.

The simplest way to delegate administrative rights to a user is to make her a member of the Administrator role. If you want to delegate only a few tasks or wish to limit the user's ability to administer the Web site, assign only the necessary user rights to her. To do this, create a custom role with the rights she needs, or edit an existing role to contain those rights.

For example, if you want a user to run usage analysis tasks for the virtual server, create a Usage Guru role, add the Manage Usage Analysis right to that role, and then assign the user to the role.

> **Note** Users can be members of more than one role, so you do not need to remove their old roles before adding them to a new one. Neither do you have to replicate all of their existing rights in the new role. So, Joe User can be a member of both the Advanced

Author and Usage Guru roles at the same time.

The following table lists the tasks you can delegate and the user rights required to perform those tasks.

Task to delegate	User right required
Adding users to the Web (per subweb)	Create Accounts
Changing user roles (per subweb)	Configure Access
Managing Web document discussions (SharePoint Team Services only)	Manage Web Document Discussions
Adding, editing, or removing a list (per Web site)	Manage Lists
Checking server health (per Web site)	Manage Server Health
Managing Web subscriptions (SharePoint Team Services only)	Manage Web Subscriptions
Creating, deleting, or merging subwebs (per Web site)	Manage Subweb
Scheduling and running usage analysis reports (per virtual server)	Manage Usage Analysis

Note For a complete lists of user rights and to see which are included in each role by default, see the **User Rights** worksheet in the Owscmdln.xls spreadsheet.

Related links

For more information about adding users to roles or assigning rights to users, see Managing Users and Managing Roles.

Limiting User Accounts

If you manage multiple Web sites or host sites for other organizations, you need to balance the load on your server computers and ensure an even distribution of resources. One way to achieve this balance and distribution is to control how many users have access to a Web site.

With Microsoft's SharePoint™ Team Services and Microsoft® FrontPage® Server Extensions 2002 on the Microsoft Windows® platform, you can limit the number of users allowed access to a Web site on a per-virtual server basis. This way, you can safely delegate the tasks of creating and deleting accounts to the administrator of each Web site, but still maintain quotas on those actions to ensure that your server stability is not put at risk. SharePoint Team Services and FrontPage Server Extensions 2002 can also track user accounts as they are deleted and remove unused user accounts from the virtual server, freeing the quota for other users. User account limits are applied at the virtual-server level. This means that when

you set a quota, it is per virtual server, not per Web site. In addition, there are safeguards built in to prevent a user from being added to multiple virtual servers on the same server computer. A user can be a member of multiple Web sites on a virtual server, but not of other Web sites on other virtual servers on the same server computer. This makes it easier to control who has access to which sites.

The ability to limit user accounts is available only on the Windows 2000 platforms and is turned off by default. If you want to implement user account limits, you must specify a limit by using either the command-line interface or HTML Administration pages. You must be a machine administrator (a member of the Administrators group of the server computer) to set SharePoint Team Services and FrontPage Server Extensions 2002 to track user accounts.

Storing user account information

To make the quotas possible, user account information is stored in a flat file database for each server computer. When you create a new user account, the user information is added to the database and is marked as belonging to a particular virtual server. When a user is deleted, he or she is removed from this database and will not be able to gain access to any site on that virtual server.

If the quota for the virtual server has been reached, and an administrator tries to add a new user, a message is returned that the quota has been reached and that a user must be deleted before a new user can be added. Also, if an administrator tries to add a user who already has an account on a different virtual server, a message is returned that the user already exists on another virtual server and cannot be added.

Only new local machine accounts are tracked in user account limits. Existing users and domain users do not count against the user account limits. For example, if you have an existing Web site and you upgrade that Web site to SharePoint Team Services or FrontPage Server Extensions 2002, any user accounts that already exist on that Web site are not affected by user account limits. However, if you add a new user after the upgrade, the new user account is counted against the limits.

> **Note** User account limit data is stored in the Owsuser.cnf file in the ..\Documents and Settings\All Users\Application Data\Microsoft\Web Server Extensions\50 folder. If you are using account limits, you must back up this file whenever you back up your collaboration database and your site content. There is no way to repair this file other than restoring it from a backup.

Using the command line to limit user accounts

You use the **accountlimits** operation to control how many accounts can be added to a virtual server. The **accountlimits** operation takes the following parameters.

Parameter	Description
-port	Optional
-multihost	Optional
-limit	Number of local accounts allowed. Set to a number to turn limits on (for example, **30**) or to **unlimited** to turn limits off.

For example, the following command can be used to set port 80 to allow 30 users:

```
owsadm.exe –o accountlimits –p 80 –limit 30
```

And the following command can be used to set port 80 to allow unlimited accounts:

```
owsadm.exe –o accountlimits –p 80 –limit unlimited
```

Using HTML Administration pages to limit user accounts

User account limits are managed from the Virtual Server Administration page.

To set user account limits

1. On the server computer click **Start**, point to **Programs**, point to **Administrative Tools**, and then click **Microsoft SharePoint Administrator**.
2. On the **Server Administration** page, click **Administration** next to the virtual server you want to manage.
3. Under **Administration**, click **Configure user account limits**.
4. In the **Maximum number of user accounts** box, type the number of user accounts you want to allow.
5. Click **Submit**.

If you want to clear your user account limits, and allow site administrators to create as many accounts as they like, you can remove the user account limits by using an option on the same page.

To remove user account limits

1. On the **User Account Limits** page, select the **Unlimited number of user accounts** check box.
2. Click **Submit**.

Deleting users under account limits

When you reach the limit for user accounts, the only way you can add new users is by deleting existing users. You may also need to delete users when they no longer need any

permissions to a site. To delete local machine accounts, you use the **Manage Virtual Server Accounts** page for the root web of the virtual server.

To delete a user account on the virtual server

1. On the server computer click **Start**, point to **Programs**, point to **Administrative Tools**, and then click **Microsoft SharePoint Administrator**, and then on the **Server Administration** page, click the name of the site you want to manage.
2. In the **Users and Roles** section, click **Click here to add or delete accounts**.
3. On the **Manage Virtual Server Accounts** page, select the user account you want to delete.
4. Click **Delete selected user(s)**.

Note that the **Manage Virtual Server Accounts** page is only available when the user account limits are set to a specific number. If user account limits are set to unlimited, the account limit information does not appear on the **Site Administration** page.

Related links

For more information about adding users to roles or assigning rights to users, see Managing Users and Managing Roles.

Inviting Members to a Site

Microsoft's SharePoint™ team Web sites and Microsoft FrontPage®–based Web sites include an invitation feature you can use to send e-mail invitations to members of your site. You use the Send an Invitation Wizard to invite as many users as you like, and to create local user accounts for them on your site. You can even invite a group to your site by using an e-mail group address.

In the Send an Invitation Wizard you can type a message to include in the invitation e-mail message. For example, you can describe your site and what it should be used for, or add a personal message to the default e-mail invitation.

The Send an Invitation Wizard automatically sends e-mail messages to each user, notifying them that they have been invited to your site. Each e-mail message includes a link to the site URL and the password to use to gain access to the site. When users gain access to the site for the first time, they then change the default password to a new password.

> **Note** When you invite a user to your site who already has a domain account, the e-mail message sent to that user does not include the user's password.

You invite single users or a group of users by using their local machine or domain accounts. You must set new users to be automatically assigned to a specific role when they gain access to the site by selecting a role in the **Roles** box. You must be a site administrator to invite users to a site.

To invite members to your site (SharePoint team Web site only)

1. On the navigation bar of your SharePoint team Web site, click **Site Settings**.
2. On the **Site Settings** page, under **Web Administration**, click **Send an invitation**.
3. On the **Enter e-mail addresses** page, type the e-mail addresses of the people you want to invite to your Web site.

   Include each user's full e-mail address (or a group e-mail address), and separate each e-mail address with a return. For example,

   someone@domain.com

   someoneelse@domain.com
4. Click **Next**.

5. On the **Verify accounts** page, look through the list of users and verify the account information.

   Users who do not yet have accounts on the local machine are marked "New." When you finish the wizard, these accounts are created on the local machine. If you want to change anything about the accounts (such as the display name), make the changes on this page.
6. Click **Next**.
7. On the **Personal greeting and role assignment** page, if you want to add a personal message to the standard e-mail message, type a short message to include in the invitation e-mail in the text box.
8. In the **Role** box, select the role to assign users to by default.

   All of the users you invite are assigned to the same role. You can use the **Manage Users** administration page (in the Site Administration pages) to assign users to unique roles later. For more information, see Managing Users.
9. Click **Finish**.

If you are administering a FrontPage-based Web site, you can open the Send an Invitation Wizard from the Site Administration pages.

To invite members to your site (FrontPage-based Web site)

1. On your server computer, click **Start**, point to **Programs**, point to **Administrative Tools**, and then click **Microsoft SharePoint Administrator**.
2. In the list of virtual servers, click the name of the virtual server that hosts the Web site you want to send invitations to.
3. On the **Site Administration** page, under **Users and Roles**, click **Send an invitation**.
4. On the **Enter e-mail addresses** page, type the e-mail addresses of the people you want to invite to your Web site.

Include each user's full e-mail address (or a group e-mail address), and separate each e-mail address with a return. For example,

someone@domain.com

someoneelse@domain.com

5. Click **Next**.

6. On the **Verify accounts** page, look through the list of users and verify the account information.

Users who do not yet have accounts on the local machine are marked "New." When you finish the wizard, these accounts are created on the local machine. If you want to change anything about the accounts (such as the display name), make the changes on this page.

7. Click **Next**.

8. On the **Personal greeting and role assignment** page, if you want to add a personal message to the standard e-mail message, type a short message to include in the invitation e-mail in the text box.

9. In the **Role** box, select the role to assign users to by default.

All of the users you invite are assigned to the same role. You can use the **Manage Users** administration page (in the Site Administration pages) to assign users to unique roles later. For more information, see Managing Users.

10. Click **Finish**.

Inviting groups

If you are on the Microsoft® Windows® platform, you can use the Send an Invitation Wizard to invite groups to your site by using the group's local machine or domain account. The Send an Invitation Wizard does not distinguish between individual and group e-mail names. Whether you type an individual e-mail name, a group name, or a distribution list e-mail name, you get the same result. If the name does not yet have an account on the local machine, an account is created when the wizard is finished, and the account (whether created for an individual or group) is assigned a temporary password to use when first accessing the site.

This process of creating accounts automatically works fine for individual accounts. The individual logs on with his or her account name and temporary password, changes the password to something more secure, then continues working. Automatic account creation also works well for a pre-existing domain group. The individuals that belong to that group already have domain accounts and are able to change the passwords for their own accounts.

However, if you invite a new local group or an e-mail distribution list to your site, a single local account is created for the entire group, with one user name and one password. So, if one member of the group signs on to the site and changes the password, the other members of the group can no longer log on.

To avoid this scenario, you must either create the group ahead of time, and then invite the group, or invite each member of the group individually. Either way, you end up with individual accounts for each user, rather than a shared account that can be accidentally disabled.

Setting command-line properties to control user groups

You can also set the following properties on the command line to help manage groups for invitations.

Property	Description
LocalNTAccountsOnly	Specifies that domain accounts are to be used when sending invitations or adding a user. If you are using local accounts only, set the property value to **1** to bypass the check for a domain account and automatically create local machine accounts when issuing an invitation. Note that setting this value to **1** means that you cannot use any domain accounts, even if a particular domain account is valid.
RestrictAccountVisibility	Specifies that the list of users shows only users that were either created on this virtual server or role members in the current Web site. Set to **1** to restrict the list of users, **0** to show all users.

For more information about setting properties on the command line, see Setting Configuration Properties.

Related links

For more information about adding users to your Web site, see Managing Users.

Managing the Database

In Microsoft's SharePoint™ Team Services, a database stores information about SharePoint team Web site documents (metadata), Web subscriptions and Web document discussions, and other site data. For example, when users discuss or subscribe to a document on your site, their comments are stored in the database.

The database is either Microsoft® SQL Server™ 7.0 (or later), or Microsoft Data Engine (MSDE). You create the database when you install SharePoint Team Services, and that database automatically stores:

- List data

Many special pages in a SharePoint team Web site are lists — announcements, tasks, and contacts, for example. Metadata from these lists is stored, so that you can search for items using date, subject, or author name.

- Document library information

Rather than storing entire documents, document properties are stored in lists, and those properties can be used in searching.

- Web document discussion and Web subscription data

Threaded comments about a document, list, or general topic are stored as discussion items, and all subscription information is stored in the database.

- Usage data

Web site usage data is stored in special read-only lists.

- Security data

Some security information for your site is stored in the database.

Each virtual server on your server computer has only one database. Any Web site on that virtual server shares that database, although discussion information is stored in separate tables for each Web site. You manage the database by using the Virtual Server Administration pages for your server. From these pages you can back up and restore data (MSDE or SQL Server databases), or change the password for the database (MSDE databases only). You can also use the command-line administration tools to view and manage database information.

Note To perform more advanced database administration tasks, you must use the SQL Server tools. For more information, see documentation for Microsoft SQL Server version 7.0 or later.

Changing database settings

Because there is only one database for each virtual server on your Web server, you manage the database settings for all subwebs on a virtual server at one time. You can change database settings by using either the command-line tools or HTML Administration pages. From either interface, you can change the:

- Database user name and password (for MSDE databases only)
- Name of the SQL server used to store database information

The new name is used for any virtual servers that are extended after the name is changed.

Using HTML Administration pages to change

database settings

You can change the database connection settings for your collaboration database by using the **Change Database Connection** page (in the Virtual Server Administration pages). For example, you can change your database connection to use a database on a separate database server. Or you can change to a local database running MSDE.

To change the database connection

1. On your server computer, click **Start**, point to **Programs**, point to **Administrative Tools**, and then click **Microsoft SharePoint Administrator**.
2. On the **Server Administration** page, next to the virtual server that you want to change, click **Administration**.
3. On the **Virtual Server Administration** page, click **Change database connection**.
4. Specify the database settings you want to use, and then click **Submit**.

Using the command line to change database settings

If you want to change the user name, password (MSDE databases only), or server being used for your database, you use the **DatabaseConnection** operation with the Owsadm or Owsrmadm utilities. You can use the **DatabaseConnection** operation to change the following settings.

Parameter	Description
Databaseserver	The name of the server that the database resides on. This is a global value only; you do not need to use the *port* parameter when you set this value. Default is the local machine name.
Databaseuser	The administrator account user name for the database. Can be set either globally or on a per-virtual server basis. Default is sa.
Databasepassword	The password for the database administrator account. Can be set either globally or on a per-virtual server basis. Can only be set for MSDE database. Use the SQL Server administration tools to change the password for a SQL Server database. Default is blank ("").

Note The **DatabaseConnection** operation encrypts the account name and password

before they are stored. It also verifies that the new values are valid and returns an error if they are not.

For example, to change the password for a particular MSDE database, you use the following syntax:

```
owsadm.exe -o databaseconnection -p <port> -du <username>
-dp <password> -c changepassword <new password>
```

Related links

For more information about backing up and restoring the database information, see Backing up and Restoring Data.

For more information about how the database interacts with data on your Web site, see SharePoint Team Services and FrontPage Server Extensions 2002 Architecture.

Backing Up and Restoring Data

Microsoft's SharePoint™ Team Services relies on a Microsoft® SQL Server™ or Microsoft Data Engine (MSDE) database to store user and document information, and Web document discussion and Web subscription data. This database makes features like the document libraries and other lists possible. In addition, SharePoint Team Services relies on the file system to track documents and pages in the site.

Database information should be backed up on a regular basis to maintain a working Web site. The backup information is used to restore corrupted files, recover from a hard-disk crash, or roll back to a previous file version to repair a user mistake. With a Web site like a SharePoint team Web site, where your files and database information must be synchronized, the backup and restore process could get fairly difficult. However, SharePoint Team Services includes tools to back up and restore the database without requiring extensive SQL Server or Microsoft Windows NT® experience.

When you back up the database, the backup file is automatically stored as *webname*.bak in the C:\MSSQL7\BACKUP directory. If you want to keep multiple backup files available, you must rename the files so that they are not overwritten.

You can back up and restore database information by using either HTML Administration pages or the command-line administration tools for SharePoint Team Services.

Note Each virtual server has only one database, even though there may be multiple Web sites on that server. When you back up or restore the database, you do so for every Web site on the virtual server.

Because SharePoint Team Services relies on both the file system and a database to track information about your Web site, you must be sure to keep these systems synchronized. For example, a list in a SharePoint team Web site is a combination of data in the database and

HTML files. If you want to back up a list, you must back up both the data in the database and the HTML files. SharePoint Team Services provides tools to use for backing up and restoring the collaboration database, but you must back up and restore the file system using your usual operating system tools. Be sure to back up the file system whenever you back up the database. Keeping your backups synchronized will help simplify the process if you ever need to restore your Web site.

Using HTML Administration pages to back up and restore data

You can use HTML Administration pages to back up and restore the data in the database, and also to schedule automatic backups for regular intervals. To use these commands, you must have Administrator rights on the Web server computer.

To create a database backup by using HTML Administration pages

1. On the server computer, click **Start**, point to **Programs**, point to **Administrative Tools**, and then click **Microsoft SharePoint Administrator**.
2. On the **Server Administration** page, next to the virtual server that contains the data that you want to back up, click **Administration**.
3. On the **Virtual Server Administration** page, click **Backup and restore database**.
4. On the **Backup and Restore Database** page, in the **Backup path and filename** box, type the path and file name to store the backed up data in.

   For example, C:\MSSQL7\BACKUP*Webname*.bak
5. Click **Backup**.

You can restore the database backup from the same **Database Backup and Restore** page.

To restore a database backup by using HTML Administration pages

1. Open the **Virtual Server Administration** page that contains the data that you want to restore.
2. Click **Backup and restore database**.
3. On the **Backup and Restore Database** page, in the **Restore path and filename** box, type the path and filename to use.

   For example, C:\MSSQL7\BACKUP*Webname*.bak
4. Click **Restore**.

You can also schedule an automatic backup of your database from the **Schedule Database Backup** administration page.

To automatically back up the database

1. On the server computer, click **Start**, point to **Programs**, point to **Administrative Tools**, and then click **Microsoft SharePoint Administrator**.

2. On the **Server Administration** page, next to the virtual server that contains the data that you want to back up, click **Administration**.
3. On the **Virtual Server Administration** page, click **Schedule database backup**.
4. Next to **Automatic database backups are**, select **On**.
5. Under **Automatic database backups occur**, select **Daily**, **Weekly**, or **Monthly**.

6. If you selected **Daily**, in the **Time** box, select the time of day to perform the backup.

– or –

If you selected **Weekly**, in the **Time** box, select the time of day, and then in the **Weekday** box, select the day of the week to perform the backup.

– or –

If you selected **Monthly**, in the **Day** box, select the day of the month, and then in the **Time** box, select the time of the day to perform the backup.

7. Click **Submit**.

Using the command-line tools to back up and restore data

You can use the command line to back up and restore the database either locally, by using Owsadm.exe, or remotely, by using Owsrmadm.exe. To back up the database from the command line, you use the **BackupDB** operation. To restore, you use **RestoreDB**. The **BackupDB** operation takes the optional parameters *–filename*, *-port*, and *–schedule*. To do a simple backup of the database on port 80, you would use the following syntax:

```
Owsadm.exe –o backupdb –p 80 –f path/filename
```

The **RestoreDB** operation takes the optional *–filename* and *–port* parameters. To perform a simple restore of the same database, you would use the following syntax:

```
Owsadm.exe –o restoredb –p 80 –f path/filename
```

You can also back up the database automatically, by creating a scheduled job for the backup process. To schedule automatic backup you use the **BackupDB** operation with the *schedule* parameter, and specify a time/day/month/year to back up the database.

The following example syntax shows how to set the database to be backed up every Monday at 11:59 P.M. for port 80.

```
owsadm –o backupdb –p 80 –schedule "weekly at Mon 23:59"
```

For more information about specifying the times to use for backing up the database, see Scheduling Timed Jobs.

Managing Web Document Discussions and

Subscriptions

Microsoft's SharePoint™ team Web sites include two special collaboration features that allow users to communicate with each other on the World Wide Web:

- Web document discussions

  These are threaded discussions that allow users to collaborate on HTML documents on a server running SharePoint Team Services. Users can add and view discussion items located within documents, or general discussion items located in the discussion pane.

- Web subscriptions

  These allow users to subscribe to an e-mail notification service. When documents on a server running SharePoint Team Services are created or modified, subscribers receive e-mail messages that identify changes.

  Note Web document discussions and Web subscriptions are not available with Microsoft® FrontPage® Server Extensions 2002.

Users can participate in discussions in documents created in SharePoint Team Services–compatible word processing, spreadsheet, and presentation programs, such as Microsoft Word, Microsoft Excel, and Microsoft PowerPoint®, as well as any HTML or Rich Text Format (RTF) file. The threaded discussions are maintained on either the Web server that has SharePoint Team Services installed or a remote Microsoft SQL Server™ computer.

To participate in a discussion, users click the **Discuss** button on the Microsoft Internet Explorer toolbar, or in a SharePoint Team Services–compatible spreadsheet, word processing, or presentation programs; or click the Web Discussions command (**Online Collaboration** submenu on the **Tools** menu). If necessary, they then select the appropriate Web server running SharePoint Team Services to store the discussion items. All user discussion items are displayed in a shared document that users can view either in the original application or in most Web browsers.

Note With SharePoint Team Services, the user can work with discussion servers in two modes. With the first, the user chooses to always try to use the server that a document is hosted on as the discussion server. However, when the user switches to viewing a document on a different discussion server, the user is automatically connected to the new server for discussions. (In this mode, the user can discuss only documents hosted on discussion servers, not generic Web servers.)

Alternatively, the user can switch to a mode where there is a specific server set up for hosting discussions. In this mode, the user must specify the discussion server to use (by using the **Discussion Options** command).

Because discussion items are stored separately from the shared document, users can modify the document without affecting the collaborative discussion. This separation also allows multiple users to create and edit discussion items simultaneously. Users can also add

discussion items to read-only documents.

By using the **Web Discussions** toolbar, users can view, search, or filter discussions by author, or date and time. Inline discussions print as part of the document, while discussions in a discussion pane print on a separate page when the document is printed. Users can view discussions offline, but they cannot add to them offline. Users can participate in discussions about documents that are stored on a server remote from the Web server that contains the SharePoint team Web site.

Editing discussion items

The logon authentication account identifies a user, and depending on the role assigned to a user, the user has different rights for working with discussion items. For example, a user with View Web Document Discussions rights can only view other users' discussion items but cannot contribute to the discussion. Whereas a user with Manage Web Document Discussions rights can not only participate, but can edit or remove other users' discussion items.

With SharePoint Team Services, and with the appropriate user rights, you can perform the following actions on an existing discussion item:

- Reply — Append your discussion item to another user's discussion item.
- Edit — Change your own or another user's discussion item.
- Delete — Delete your own or another user's discussion item.
- Close — Mark a discussion item as closed.

Anyone with discussion rights (from View Web Document Discussions on up) can close discussion items. Resolving discussions allows users to clean up old or annoying discussions from a document. When discussions are marked closed, they are not deleted from the database, and therefore there is no risk of losing important information. If the discussion turns out to be important, it can be restored.

Modifying documents that contain discussion threads

Because discussion items are not stored in the document, users can modify the document independently from any discussions they create.

The following table explains how a modification to a document can affect threaded discussions in a document.

When a document is modified in this way	The discussion data is affected this way
The change is made in an area without a discussion item attached.	Inline and general discussions are not affected.

The change is made in an area with a discussion item attached.	Inline discussions attached to modified text are deleted. General discussions are not affected.
The entire document is moved, renamed, or deleted.	All inline and general discussions are lost.

Configuring and managing Web document discussions

You can manage Web document discussions for one virtual server or all virtual servers on a Web server by using the command-line tools or HTML Administration pages. If you use HTML Administration pages, you can:

- Turn Web document discussions on or off.
- Delete Web document discussion items.
- Edit Web document discussion items.
- View Web document discussion settings.
- Limit Web document discussions to documents local to the site.
- Set aging policy for Web document discussions.

If you are using the command-line tools, you can:

- Turn Web document discussions on or off.
- Limit Web document discussions to documents local to the site.
- Set expiration for Web document discussions.

Note You can administer Web document discussions on a per-virtual server basis only, not per subweb. This is because Web document discussion data is stored in the SharePoint Team Services database, and there is only one database for each virtual server.

Using HTML Administration pages to manage Web document discussions

You can use HTML Administration pages to manage Web document discussions. With HTML Administration pages, you get an easier-to-use interface, with all of the choices listed on the form. Select the options you want, click **Submit**, and you are done. To manage Web document discussions, you use commands on the **Site Administration** page.

To view the Site Administration page

- If you are a server administrator, on the server computer click **Start**, point to **Programs**, point to **Administrative Tools**, and then click **Microsoft SharePoint Administrator**, and then on the **Server Administration** page, click the name of the site

you want to manage.

- If you are a site administrator, on your Web site, click **Site Settings**, and then under **Web Administration**, click **Go to Site Administration**.

To turn discussions on or off for a virtual server

1. On the Site Administration page for the virtual server, under **Web Discussions and Subscriptions**, click **Change Web discussion settings**.
2. In the **Enable Web Discussions** area, next to **Web Discussions are**, click **On** or **Off**.
3. Click **Submit**.

If you want discussion items to automatically expire and be deleted from the database, you can change the settings on the **Change Web Discussion Settings** page.

To set an expiration time for discussion items

1. On the **Site Administration** page for the virtual server, under **Web Discussions and Subscriptions**, click **Change Web discussion settings**.
2. In the **Web Discussions Settings** area, select the **Automatically delete stored discussions after ___ day(s)** check box.
3. In the **Automatically delete stored discussions after ___ day(s)** box, type the number of days to store Web discussion items before deleting them.
4. Click **Submit**.

You can also specify whether to allow Web discussions anywhere on the Internet, or only on the current server, by selecting options on the **Change Web Discussion Settings** page.

To allow Web discussions on the current server only

1. On the **Site Administration** page for the virtual server, under **Web Discussions and Subscriptions**, click **Change Web discussion settings**.
2. In the **Web Discussion Settings** area, under **Allow Web Discussions** on, select the **Documents located on this server only** check box.
3. Click **Submit**.

If you want to use the default Web discussion settings for your server, rather than the settings currently used for your virtual server, you can revert to the default settings by clicking **Restore Defaults** on the **Change Web Discussion Settings** page.

You can also view and delete discussion threads on your discussion server. To manage discussion threads, you use the **Manage Web Discussions** page in the Site Administration pages.

To manage web discussion threads

1. On the **Site Administration** page for the virtual server, under **Web Discussions**

and Subscriptions, click **Manage Web discussions**, and then do any of the following:

- To see all discussion threads on your server, click **All Web document discussions**, and then click **Update**.
- To see discussion threads filtered by a particular URL, click **Web document discussions filtered by URL**, type a URL in the box, and then click **Update**.
- To view a particular discussion thread, click the URL for the thread.
- To delete a particular discussion thread, select the check box next to the thread, and then click **Delete**.
- To delete all discussion threads on your server, click **Delete all discussions**.

Using the command line to manage discussions

You can manage discussions from the command line by using the **GetProperty**, **SetProperty**, and **DeleteProperty** operations with Owsadm.exe or Owsrmadm.exe. You can set the following properties to manage how discussions work.

Property	Description
DiscussionsExpiry	Specifies the length of time (in days) before discussions expire and are deleted from the database. Default is **0** (discussions never expire).
EnableDiscussions	Specifies whether discussions are turned on or off (either at the virtual server level or globally). Default is **1** (discussions on).
LocalDiscussionsOnly	Specifies whether discussions are restricted to the current server. Default is **0** (discussions not restricted).

The following example shows the syntax to use to set an aging policy for discussions, so that discussion items are deleted after 60 days:

```
owsadm.exe -o setproperty -p <port> -pn DiscussionsExpiry -pv 60
```

SharePoint Team Services stores documents and Web document discussion items separately — documents are stored in the file system, and discussion items are stored in the database. Because the discussion items are separate, users can use your SharePoint Team Services discussion server to discuss any document (either HTML or binary format) on any Web or file server and store their discussion data in your database.

If you want to restrict discussions to only work on documents on your Web server running SharePoint Team Services (thus possibly saving space in your database), you can use the **LocalDiscussionsOnly** property to do so. By default, discussions are not restricted. The following example sets the property to restrict discussions to the local server only.

```
owsadm.exe -o setproperty -p <port> -pn LocalDiscussionsOnly
-pv true
```

Configuring and managing subscriptions

You can manage subscriptions for one virtual server or all virtual servers on a Web server. You can use the command-line tools or HTML Administration pages to administer discussion settings. If you use HTML Administration pages, you can:

- View subscription settings.
- Turn subscriptions on or off.
- Delete subscriptions.
- Edit subscriptions.
- Change the e-mail address that users reply to regarding subscription notifications.
- Set timer settings for subscriptions.

If you are using the command-line tools, you can:

- Turn subscriptions on or off.
- Turn folder subscriptions on or off.
- Change the SMTP mail server used for subscriptions.
- Change the e-mail address that sends subscription notifications.
- Change the e-mail address that users reply to regarding subscription notifications.

Note You can administer subscriptions on a per-virtual server basis only, not per-Web site. This is because subscription data is stored in the SharePoint Team Services database and there is only one database for each virtual server.

Using HTML Administration pages to manage subscriptions

You can use HTML Administration pages to manage subscriptions. With HTML Administration pages, you get an easier-to-use interface, with all of the choices listed on the form. Select the options you want, click **Submit**, and you are done. The Web subscription options are on the Site Administration page.

To view the Site Administration page

- If you are a server administrator, on the server computer click **Start**, point to **Programs**, point to **Administrative Tools**, and then click **Microsoft SharePoint Administrator**, and then on the **Server Administration** page, click the name of the site you want to manage.
- If you are a site administrator, on your Web site, click **Site Settings**, and then under

Web Administration, click **Go to Site Administration**.

To turn subscriptions on or off for a virtual server

1. On the **Site Administration** page for the virtual server, under **Web Discussions and Subscriptions**, click **Change Web subscription settings**.
2. In the **Enable Web Subscriptions** area, next to **Web Subscriptions are**, click **On** or **Off**.
3. Click **Submit**.

You can determine when subscription notifications are sent by default. For example, you could set the default weekly notifications to be sent out at 5 P.M. every day. To specify these settings, you use the **Change Web Subscription Settings** page.

To set the notification time for subscriptions

1. On the **Site Administration** page for the virtual server, under **Web Discussions and Subscriptions**, click **Change Web subscription settings**.
2. In the **Notification Recurrence Settings** area, in the **Immediate notifications every ___ minutes** box, specify how quickly to send out immediate notifications.
3. In the **Daily notifications at** box, specify the time to send out daily notifications.
4. In the **Weekly notifications** box, specify the day of the week and the time to send out weekly notifications.
5. Click **Submit**.

Also, you can specify the from and reply-to addresses to use for the subscription notifications.

To specify the from and reply-to addresses for Web subscriptions

1. On the **Site Administration** page for the virtual server, under **Web Discussions and Subscriptions**, click **Change Web subscription settings**.
2. In the **Mail Settings** area, in the **From address** box, specify an e-mail address to use as the From address in subscription notifications.
3. In the **Reply-to address** box, specify an e-mail address that users should reply to if they have any questions.

   Be sure that the address you use for the reply-to address is monitored.
4. Click **Submit**.

If you want to use the default Web subscription settings for your server, rather than the settings currently used for your virtual server, you can revert to the default settings by clicking **Restore Defaults** on the **Change Web Subscription Settings** page.

You can also view and delete subscriptions on your server. To manage subscriptions, you use the **Manage Web Subscriptions** page in the Site Administration pages.

To manage Web subscriptions

5. On the **Site Administration** page for the virtual server, under **Web Discussions**

and Subscriptions, click **Manage Web subscriptions**, and then do any of the following:

- To see all subscriptions on your server, click **All Web Subscriptions**, and then click **Update**.
- To filter the list of subscriptions, click **Web subscriptions Where**, select a criteria, and in the equals (=) box type the value to filter by, and then click **Update**.
- To delete a particular subscription, select the check box next to the subscription, and then click **Delete**.

Using the command line to manage subscriptions

You can manage subscriptions from the command line by using the **GetProperty**, **SetProperty**, and **DeleteProperty** operations with Owsadm.exe or Owsrmadm.exe. You can set the following properties to manage how subscriptions work.

Property	Description
EnableNotifications	Specifies whether subscriptions are turned on or off (either at the virtual server level or globally). Default is **1** (subscriptions on).
FolderSubscriptions	Specifies whether users can subscribe to folders as well as documents. Default is **1** (subscriptions are allowed for folders).
SMTPHost	Specifies the IP address of the SMTP server to use in sending subscription notifications.
SendMailCommand	Sets the name of the program to which e-mail should be piped. Typically this is sendmail, but it could be any program.
MailReplyTo	Specifies the e-mail address to use when replying to subscription notifications messages. The default for SMTP is user@host, where user is the current user account and host is the current host name.
MailSender	Sets the user name to use as the From account when sending subscription notifications. The default for SMTP is user@host, where user is the current user account and host is the current host name.

The following example shows the syntax to use to turn off folder subscriptions:

```
owsadm.exe -o setproperty -p <port> -pn FolderSubscriptions
-pv false
```

The following example shows the syntax to use to set the **MailReplyTo** property to send

messages to a particular e-mail address (for example, someone@microsoft.com):

```
owsadm.exe -o setproperty -p <port> -pn MailReplyTo
-pv <e-mail address>
```

Related links

For more information about using the command-line tools or setting properties on the command line, see Command-line Administration.

For more information about using HTML Administration pages, see HTML Administration.

Configuring Support for Database Integration with a Web Site

With Microsoft's SharePoint™ Team Services or Microsoft® FrontPage Server Extensions 2002, Web authors can integrate a database with their Web site to incorporate live data into their Web pages. The database used for this feature is separate from any database already used by your Web server to support collaboration features such as Web document discussions or document libraries.

Database connectivity

Web authors can incorporate data from any open database connectivity (ODBC)–compliant database. The database can reside either on the Web server or on a remote database server. The SharePoint Team Services-compatible Web page editor, Microsoft FrontPage® 2002, provides drivers for:

- Tab- or comma-delimited text files
- Microsoft Excel spreadsheets
- Microsoft Access files
- the Microsoft Visual FoxPro® database development system
- Inprise dBASE files
- Corel Paradox files

Drivers for server-based databases include those for Microsoft SQL Server™ and Oracle.

All database connection information is stored in a file called Global.asa on the Web author's computer. If the database resides outside the author's FrontPage-based Web site, a data source name (DSN) is also required to connect to the database.

Database security

Using SharePoint Team Services or FrontPage Server Extensions, you can ensure that only users with administrative or authoring privileges for a Web site will be able to gain access to

databases in that Web site.

The recommended location for file-based databases is in the fpdb folder in the FrontPage-based Web site. With SharePoint Team Services or FrontPage Server Extensions installed on the Web server, FrontPage automatically marks this folder as not browsable, scriptable, or executable. By default, when a new database is created, a Microsoft Access database is placed in the fpdb folder. When Web authors import an existing database to a Web site, the fpdb folder is created, if it does not already exist, and the file is uploaded to the fpdb folder or to a folder specified by the user. If the user chooses not to use the fpdb folder, the file is placed at the root of the Web site.

If the user places the database in a folder other than fpdb, the Component Errors report in FrontPage 2002 recommends moving the database to the fpdb folder for security reasons.

No additional database security is provided, beyond the security settings that already exist within the database. If a Web author creates a database, he or she must also configure the access restrictions within the database application and apply them to the database file. If access restrictions are not set within the database, a user with authoring or administrative rights to the Web site might be able to access and change the contents of the database.

If you are administering a SharePoint team Web site or FrontPage-based Web site, you can take the following steps to make databases in users' Web sites more secure:

- In FrontPage 2002, in Reports view, check to see if any databases should be moved to the fpdb folder, and move them.
- Use the database's built-in security mechanisms to restrict who can update the database content.
- Check to see what database privileges are available to the account that Web authors are using. Generally, accounts should not need privileges beyond SELECT and UPDATE, which are used by FrontPage.

Setting configuration properties to support database features

SharePoint Team Services and FrontPage Server Extensions use several properties that you can set to configure how your Web server supports database features for a Web site. Because FrontPage's database integration relies on Active Server Pages (ASP pages), Web authors must be able to save ASP pages to the server. The configuration properties that affect database support are **AllowExecutableScripts**, **ListSystemDSNs**, **NoExecutableCGIUpload**, and **NoMarkScriptable**.

Toolbox For a complete list of the properties you can set from the command line, see the Properties worksheet in the Owscmdln.xls spreadsheet.

AllowExecutableScripts

When **AllowExecutableScripts** is turned on, Web site authors can execute programs and scripts, such as CGI scripts, ISAPI extensions, and ASP pages. However, for security reasons **AllowExecutableScripts** is turned off by default when you first install SharePoint Team Services. You must set **AllowExecutableScripts** to a non-zero value, either globally or for each virtual server where you want to allow Web authors to run scripts.

ListSystemDSNs

FrontPage 2002 lets Web authors list all the data source names (DSNs) on a server. This is a potential security hole because it exposes a list of resources. In addition, it is unlikely that the DSNs are password protected, because until SharePoint Team Services or FrontPage Server Extensions were installed on the server, the DSNs were not accessible from a Web page.

You can hide system DSNs by turning the **ListSystemDSNs** configuration property off, either globally or for each virtual server. This setting defaults to true when you first install SharePoint Team Services or FrontPage Server Extensions.

NoExecutableCGIUpload

When **NoExecutableCGIUpload** is turned on, Web authors cannot upload files to folders whose scriptable (or executable) bit is set. **NoExecutableCGIUpload** is turned on by default when you install SharePoint Team Services or FrontPage Server Extensions, which means that Web authors cannot upload ASP pages to servers, such as Internet Information Services 3.0, that do not have separate scriptable or executable bits. You can turn the **NoExecutableCGIUpload** configuration property off, either globally or for each virtual server where you want to allow Web authors to be able to upload executable files.

NoMarkScriptable

When you install SharePoint Team Services or FrontPage Server Extensions, **NoMarkScriptable** property is turned off, which means that Web authors can change the "scriptable" attribute on a folder from FrontPage 2002. If you turn **NoMarkScriptable** on, either globally or for virtual servers, you must provide some scriptable folders for Web authors to be able to use the FrontPage database features and other ASP-based pages.

You can also turn **NoMarkScriptable** on or off for subwebs on a server. If you want to selectively turn database support on for customers who want it, but disallow database support for all other customers, you would turn off the **NoMarkScriptable** configuration property for the server, but turn it on for selected Web sites.

Related link

For more information about how to configure SharePoint Team Services by using properties, see Setting Configuration Properties.

Monitoring Server Health

Web sites, especially Microsoft's SharePoint™ team Web sites, undergo many changes each day. This high level of change creates potential server problems. SharePoint Team Services and Microsoft® FrontPage® Server Extensions 2002 include functionality that helps you detect and repair common server problems when they arise.

When you check your server's health, you can perform several functions either individually or all at once. You can also choose whether to be notified of problems, or to have the problems automatically corrected. Server health checks are performed for an entire virtual server at one time. You must be an administrator of the virtual server, or have the **Manage Server Health** right to run a server health check.

During a server health check, you can choose to:

- Synchronize the database (SharePoint Team Services only).

  This check ensures that database information matches that of the SharePoint team Web site file system. If content is added to a Web site when the database is offline, the metadata for the content may not be added to the database. As a result, the database becomes unsynchronized with your Web site's content — a problem that can occur in other ways, as well. Because many SharePoint Team Services and FrontPage Server Extensions 2002 operations rely on data in the database, you want the database to accurately reflect your Web site content. You can synchronize the database with the metadata during a server health check.

- Verify existence of Web sites.

  This check verifies whether the subwebs on a root web exist in the file structure. It looks through the services.cnf file in the root web to find which subwebs should exist, and then checks the file system to see if they do. If a directory or subweb does not exist, and you specify that you want to fix this problem, the services.cnf file is updated to reflect the fact that the subweb does not exist or that the folder is not a subweb. This process is repeated for each subweb of the root web.

  Note that this is not always a safe operation to repair. For example, if you have a subweb that is stored on a network file share, rather than locally, and the network is temporarily unavailable, this check will report that the subweb does not exist. If you then repair the problem, the subweb will be deleted from the root web's list.

- Check roles configuration.

  Ensures that user role settings can be enforced. This check detects and repairs possible problems with roles, including the following: a user record has an invalid format, the user has a record in the list of users, but no matching user account, the user record refers to a non-existent role, there is no role for the anonymous user, and so on.

- Reapply file system security.

This option looks at the user and roles permissions you have created for your Web site, and then applies those settings to the file system. There is no detect portion to this server health check, only the repair process.

- Tighten security.

This check ensures that all the necessary Web site files and directories are present, and that only users with the proper permissions have access to them.

- Check anonymous access.

FrontPage 2000 Server Extensions did not support anonymous authoring of a Web site. With SharePoint Team Services and FrontPage Server Extensions 2002, you can use anonymous authoring of your site if you want. This option checks the anonymous user access rights for your Web sites and all subwebs to ensure that anonymous users have not been accidentally given more rights than are allowed by the role assigned to anonymous users.

Server health checks can be performed from either HTML Administration pages or the command line.

Using the command line to check server health

To perform a basic server health check from the command line, you use the **check** operation. This operation runs the server health check, and then displays a list of problems found and notes about which problems were fixed, if any.

The following example shows the syntax to use for running a server health check:

```
owsadm.exe –o check
```

You can also set the server health check to be performed at regular intervals by using the *schedule* parameter (on the Microsoft Windows® platform, or by using the cron facility on the UNIX platform). For more information about scheduling server health checks, see Scheduling Timed Jobs.

The **check** operation runs the server health check; it does not allow you to determine which actions are performed during the check. To control the actions performed during the check, you use the **setproperty** operation and specify the following properties:

- **RepairDB** (SharePoint Team Services only — corresponds to **Synchronize the database** option in HTML Administration pages)
- **ValidateServicesCnf** (corresponds to **Verify existence of webs** option in HTML Administration pages)
- **CheckRolesConfig** (corresponds to **Check roles configuration** option in HTML Administration pages)
- **CheckUserAccess** (corresponds to **Reapply file system security** option in HTML

Administration pages)

- **TightenSecurity** (corresponds to **Tighten security** option in HTML Administration pages)
- **RemoveAnonymousAuthoring** (corresponds to the **Check anonymous access** option in HTML Administration pages)

All of these properties accept the following values: **check** and **fix**. For example, if you want to include the Repair Database action in your server health check, and want to have the database checked and any problems repaired, you would set the **RepairDB** property to **fix**, as in the following example syntax:

```
owsadm.exe –o setproperty –pn RepairDB –pv fix -p 80
```

After you set this property, whenever you run the server health check by using the **check** operation, the database is checked and repaired. The default setting for all of these properties is **null**, which means that these properties are ignored by the check operation until you specify either **check** or **fix**.

Using HTML Administration pages to check server health

You can use HTML Administration pages to specify which actions to perform during a server health check and to run the check itself. You use the **Check Server Health** page (in the Site Administration pages) to specify settings and to run the server health check.

To view the Site Administration page

- If you are a server administrator, on the server computer click **Start**, point to **Programs**, point to **Administrative Tools**, click **Microsoft SharePoint Administrator**, and then on the **Server Administration** page, click the name of the site you want to manage.
- If you are a site administrator, on your Web site click **Site Settings**, and then under **Web Administration**, click **Go to Site Administration**.

To run a server health check

1. Open the **Site Administration** page for the virtual server you want to check.
2. In the **Server Health** area, click **Check server health**.
3. In the **Detect and Repair** area, select the **Detect** and **Repair** check boxes for any server health actions you want to perform.

   If you only want to check for problems, but not fix them, do not select the **Repair** check box for that action.
4. Click **OK**.

Clicking **OK** accepts your settings and performs the server health check. After the server

health check runs, you can view the results from the **Server Health Report** page. If you want to run additional health checks, you can do so from the **Server Health Report** page. Just select the **Fix this problem** check box for any actions you want to perform, and then click **Submit**.

On the Windows platform, you can enable or disable server health checks for your site, and schedule server health checks to be performed automatically. To do so, you use the **Change Server Health Settings** page.

To specify server health settings on the Windows platform

1. Open the **Site Administration** page for the virtual server you want to check.
2. In the **Server Health** area, click **Change server health settings**.
3. To enable or disable server health, next to **Server health is**, select **On** or **Off**.
4. To schedule automatic server health checks, in the **Recurrence Settings** area, select **Daily**, **Weekly**, or **Monthly**.

   If you selected **Daily**, select the time of day to perform the check. If you selected **Weekly**, select the time of day and the day of the week to perform the check. If you selected **Monthly**, select the day of the month and time of day to perform the check.
5. Click **Submit**.

If you want to use the default server health settings, rather than the settings currently used for your virtual server, you can revert to the default settings by clicking **Restore Defaults** on the **Change Server Health Settings** page.

Managing Source Control

Microsoft's SharePoint™ Team Services and Microsoft® FrontPage Server Extensions 2002 provide built-in source control (on Microsoft Windows® or UNIX platforms) and can also work with Microsoft Visual SourceSafe™ for Windows version 5.0 or later (Windows platform only). With either source-control method enabled, authors can use commands in a SharePoint Team Services–compatible Web page editor, such as Microsoft FrontPage® 2002, to check pages and other files in and out of a SharePoint team Web site or FrontPage-based Web site. With Visual SourceSafe (VSS) source control enabled, authors benefit from advanced functionality such as tracking and storing changes to each file, reviewing a file's history, and returning to earlier versions of a file.

With either type of source control enabled, SharePoint Team Services and FrontPage Server Extensions 2002 perform background check-outs and check-ins after some operations are performed in a SharePoint Team Services–compatible Web page editor. For example, when a theme is applied to a Web site that has source control enabled, each page that is not in use is checked out in the background, the new theme is applied, and the page is checked in. By doing this, SharePoint Team Services and FrontPage Server Extensions 2002 protect the consistency of a Web site and help prevent it from containing errors such as broken

hyperlinks.

The following table describes how SharePoint Team Services and FrontPage Server Extensions 2002 implement source control for common page and Web site operations.

When an author...	SharePoint Team Services or FrontPage Server Extensions 2002...	Unless...
Edits and saves a page that has not been checked out	Checks the page out, updates it, and checks it back into the Web site and the VSS project (if VSS control is enabled).	
Moves a file	Moves the file in the Web site and the VSS project (if VSS control is enabled). Then, checks out every page in the Web site that contains a hyperlink to the moved file, updates all hyperlinks to it, and checks every page back in.	A page with a hyperlink to the moved file is checked out. In this case, the author cannot move the file.
Renames a file	Renames the file in the Web site and the VSS project (if VSS control is enabled). Then, prompts the author to check out every page in the Web site that contains a hyperlink to the moved file, updates all hyperlinks to it, and checks every page back in.	A page with a hyperlink to the renamed file is already checked out. In this case, the author cannot rename the file.
Deletes a file	Deletes the file in the Web site and the VSS project (if VSS control is enabled).	The file is checked out. In this case, the author cannot delete the file.
Imports a file	Checks the file into the Web site and the VSS project (if the file is new, and VSS control is enabled). If the file already exists in the Web site, the file is checked out, updated, and checked back into the Web site and the VSS control project.	

Creates a new page	Checks the page into the Web site and the VSS project (if VSS control is enabled).	
Edits and saves a page or graphic that is included in other pages	Tries to check out every page in the Web site that includes the page or graphic, updates all pages, and checks every page back into the Web site and the VSS project (if VSS control is enabled).	A page that includes the page or graphic is already checked out. In this case, the page is not updated. To update all pages, have all authors check in all pages, then use the **Recalculate Hyperlinks** command (**Tools** menu in FrontPage).
Applies a theme to a Web site	Tries to check out every page in the Web site that uses the default Web site theme, updates all pages, and checks every page back into the Web site and the VSS project (if VSS control is enabled).	A page that uses the default theme is already checked out. In this case, the theme is not applied to the page. To update all pages, have all authors check in all pages, then reapply the theme to the Web site.

Edits and saves a shared border	Tries to check out every page in the Web site that contains the shared border, updates all pages, and checks every page back into the Web site and the VSS project (if VSS control is enabled).	A page that uses the shared border is already checked out. In this case, the shared border is not updated. To update all pages, have all authors check in all pages, then remove and reapply the shared borders to the Web site using the **Shared Borders** command (**Format** menu in FrontPage).

Enabling and configuring Microsoft Visual SourceSafe

Integration with Microsoft Visual SourceSafe 5.0 or later is available on Microsoft

Windows 2000 Server, Microsoft Windows 2000 Professional, and Windows NT® Server 4.0 or later. Install Visual SourceSafe to an NTFS file system–formatted partition.

Enabling SourceSafe integration

To use Visual SourceSafe with a SharePoint team Web site or FrontPage-based Web site, you must choose **Enable SourceSafe Integration** during Visual SourceSafe Setup.

To enable SourceSafe integration

1. Run the Visual SourceSafe installation program, Setup.exe, from the Visual SourceSafe CD.
2. Select the **Custom** installation option.
3. In the Visual SourceSafe **Custom** dialog box, select the **Enable SourceSafe Integration** check box.
4. Click **Continue**.

Configuring Visual SourceSafe for remote use

You can store your Visual SourceSafe database on the same computer as the Web server (a local database) or on a different computer (a remote database).

> **Note** Although it is possible to configure the Visual Source Safe database on a separate computer from the Web server, this is not considered a common scenario and may not work in all circumstances.

If the database is on the same computer as the Web server but is accessed using UNC paths, the database is considered to be remote. For example, the following database is remote because the Srcsafe.ini file on the Internet Information Services computer has one of the following lines:

```
Data_Path = X:\vssdata
```

```
Data_Path = \\IIS-Machine\vssdata
```

You can also view the registry to find out whether the database is accessed with UNC or mapped paths.

You should note which configuration you have, because if you are using a remote database, you must to complete a few extra steps below.

About the Visual SourceSafe INI file

SharePoint Team Services or FrontPage Server Extensions 2002 must find a Srcsafe.ini file to perform Visual SourceSafe operations through OLE automation. Visual SourceSafe 5.0 usually uses the registry key HKEY_LOCAL_MACHINE\Software\Microsoft\SourceSafe \SCCServerPath. While this key points to the Ssscc.dll in the VSS\Win32 directory,

SharePoint Team Services or FrontPage Server Extensions 2002 uses the Srcsafe.ini file in the VSS directory. Visual SourceSafe 6.0 normally uses the Srcsafe.ini file in the Web server's installation of Visual SourceSafe.

Setting Visual SourceSafe directory permissions

Before your users can use Visual SourceSafe, you must set directory permissions on the Visual SourceSafe directory structure and files. Generally, you assign **Change** (Windows NT 4.0) or **Modify** (Windows 2000) permissions to all Visual SourceSafe logon accounts for all files and subdirectories under the Visual SourceSafe server installation directory (including the Visual SourceSafe data directory if it is in a different location). You should also grant Administrators and System accounts **Full Control** to the entire Visual SourceSafe directory hierarchy. Although tighter file restrictions are possible, full Visual SourceSafe functionality can be jeopardized by tighter restrictions.

You configure the security and sharing permissions to the Visual SourceSafe directories on the computer that contains the Visual SourceSafe database. When you are assigning permissions, be sure to grant at least read permissions to all Visual SourceSafe logon accounts for the following files:

- Srcsafe.ini
- Um.dat
- Users.txt

Be sure to also grant at least read\write permissions to all Visual SourceSafe logon accounts for the following files or directories:

- Vss\Users\<logon account directory>\
- Vss\Data\
- Vss\Temp\
- Rights.dat
- Status.dat

The procedures for assigning permissions are different for Windows 2000 and Windows NT 4.0. Use the procedure for the operating system you are using.

To set Visual SourceSafe directory permissions for Windows 2000

1. In Windows Explorer, right-click the directory or share, click **Properties** on the shortcut menu, click the **Security** tab (**Sharing** tab for a shared folder).
2. Select the group name that you want to set permissions for, and then in the **Permissions** list, next to **Modify**, click **Allow**.

To set Visual SourceSafe directory permissions for Windows NT 4.0

3. In Windows Explorer, right-click the directory or share, click **Properties** on the shortcut menu, click the **Security** tab (**Sharing** tab for a shared folder), and then click

Permissions.

4. Select the group name that you want to set permissions for, and then in the Type of Access list, select **Change**.

Setting permissions for a remote Visual SourceSafe database

If you are using a remote Visual SourceSafe database, you must also set the WWW Service properties on the Web server computer. Also, if you are allowing Anonymous access to your server, you must also set Anonymous account permissions for your remote database.

To set the WWW Service properties

1. In the Internet Service Manager, open the **Internet Information Server** folder, right-click the Web server, and then click **Properties** on the shortcut menu.
2. In **Master Properties**, click **WWW Service**, and then click **Edit**.
3. Click the **Directory Security** tab, and then under **Anonymous Access and Authentication Control**, click **Edit**.

4. For Windows 2000, select the **Basic authentication (password is sent in clear text)** check box and clear the **Integrated Windows Authentication** check box.

   For Windows NT 4.0, select the **Basic (Clear Text)** check box and clear the **Windows NT Challenge/Response** check box.

   Because you have turned off Windows NT Challenge/Response authentication (Windows NT 4.0) or Integrated Windows Authentication (Windows 2000), users must now type their user name and domain password when they open a Web site. Use the format DomainName\UserName for the user name when working with multiple domains.
5. If you want to allow anonymous access to your server, select the **Anonymous access** check box (for Windows 2000) or the **Allow Anonymous** check box (for Windows NT 4.0).

If you are using an anonymous account, you must also set Anonymous access for the computer with the Visual SourceSafe database. The procedures for setting Anonymous access are different for Windows 2000 and Windows NT 4.0. Use the procedure for the operating system you are using.

To set Anonymous access for Windows NT 4.0

1. In User Manager For Domains, on the **User** menu, click **New User** to add the Web server's Anonymous account. For example, if the Anonymous account on the Web server is IUSR_WEBSRV, add that user as a local account.
2. Give the account the same password as it has on the Web server.
3. Click **OK**.
4. Under **Username**, click the Anonymous account.

5. On the **Policies** menu, click **User Rights**.
6. From the **Right** list, click **Log on locally**.

Make sure that the Anonymous account is listed in the **Grant To** box individually or as a member of one of the groups, and then click **OK**.

To set Anonymous access for Windows 2000

1. In Computer Management, click the plus sign (+) next to **Local Users and Groups**.
2. Click **Users**.
3. On the **Action** menu, click **New User** to add the Web server's Anonymous account.

For example, if the Anonymous account on the Web server is IUSR_WEBSRV, add that user as a local account.

4. Give the account the same password as it has on the Web server.
5. Click **Create**.
6. In the list of user names, right-click the Anonymous account, and then click **Properties**.
7. Click the **Dial-in** tab.
8. Under **Remote Access Permission (Dial-in or VPN)**, select **Allow access**.

Note You can also set remote access by using a remote access policy. For more information, type **remote access policies** in the index for the Microsoft Windows 2000 Help system.

9. Click **OK**.

Configuring user accounts for use in Visual SourceSafe projects

Microsoft Visual SourceSafe includes a Visual SourceSafe Administrator utility that you use to set up and configure accounts for use in VSS projects. For Visual SourceSafe 5.0 and later, the Visual SourceSafe Administrator is available on the **Start** menu.

To enable Visual SourceSafe integration with SharePoint Team Services or FrontPage Server Extensions 2002, you must add the following user accounts by using the Visual SourceSafe Administrator.

Users and groups	VSS permissions
All administrator, author, and content manager groups and users for SharePoint Team Services or FrontPage Server Extensions 2002	read-write

The Internet guest account	read

You can add both individual user and group accounts to Visual SourceSafe.

To add a user or group account in the Visual SourceSafe Administrator

1. Open the Visual SourceSafe Administrator.
2. On the **Users** menu, click **Add User**.
3. In the **Add User** dialog box, in the **User name** box, type the user or group name and leave the **Password** field blank.
4. Click **OK**.

After you have added the users and groups, you must configure the accounts to automatically log in.

To configure the Visual SourceSafe accounts

1. In the Visual SourceSafe Administrator, add the SharePoint Team Services or FrontPage Server Extensions 2002 users (without passwords).
2. On the **Tools** menu, click **Options**, click the **General** tab, and make sure that **Use network name for automatic user log in** is selected.

Configuring the Anonymous account

If you use an Anonymous account, you must configure it on the Web server computer. Before you configure the Anonymous account, check that the Anonymous account has the same password in both Internet Services Manager and Computer Management (Windows 2000) or User Manager For Domains (Windows NT 4.0). You might have to reenter the password in both places.

To set or change the Anonymous account password in the IIS Service Manager

1. In Internet Services Manager, open the **Internet Information Server** folder, right-click the Web server, and then click **Properties** on the shortcut menu.
2. In **Master Properties**, click **WWW Service**, and then click **Edit**.
3. Click the **Directory Security** tab, and then under **Anonymous Access and Authentication Control**, click **Edit**.
4. Under **Allow Anonymous Access**, click **Edit**.
5. Change the password for the Anonymous account as needed, and then close all remaining dialog boxes.

   Note the name of the Anonymous account, because you will need to identify it in Computer Management (Windows 2000) or User Manager For Domains (Windows NT 4.0).
6. Open **Control Panel**, and then double-click **Services**.
7. Click **World Wide Web Publishing Service**, click **Stop**, click **Yes** to confirm, and

then click **Start**.

After you set or change the password in Internet Services Manager, you must also set or change the password in Computer Management (Windows 2000) or User Manager For Domains (Windows NT 4.0).

Note Passwords appear as 14 characters, regardless of the actual length of your password.

The procedures for setting and changing passwords are different for Windows 2000 and Windows NT 4.0. Use the procedure for the operating system you are using.

To set or change the Anonymous account password in the Computer Management (Windows 2000)

1. In Computer Management, under **Local Users and Groups**, click **Users**.
2. In the list of users, right-click the Anonymous account, and then click **Properties**.
3. Change the password for the Anonymous account as needed, and then click **OK**.

To set or change the Anonymous account password in the User Manager For Domains (Windows NT 4.0)

1. In User Manager For Domains, under **Username**, click the Anonymous account (if you are uncertain which account this is, see the previous procedure).
2. On the **User** menu, click **Properties**.
3. Change the password for the Anonymous account as needed, and then click **OK**.

After the password is set, you can configure the Anonymous account to have logon rights and to log on automatically. The procedures for configuring the logon rights for the Anonymous account are different for Windows 2000 and Windows NT 4.0. Use the procedure for the operating system you are using.

To configure logon rights for the Anonymous account on Windows NT 4.0

1. In User Manager For Domains, under **Username**, click the Anonymous account.
2. On the **Policies** menu, click **User Rights**.
3. From the **Right** list, click **Log on locally**.

   Make sure that the Anonymous account is listed in the **Grant To** box individually or as a member of one of the groups, and then click **OK**.
4. With the Anonymous account still selected, on the **User** menu, click **Properties**.
5. Clear the **User Must Change Password at Next Logon** and **Account Disabled** check boxes, and then click **OK**.

To configure logon rights for the Anonymous account on Windows 2000

1. In Computer Management, under **Local Users and Groups**, click **Users**.

2. In the list of users, right-click the Anonymous account, and then click **Properties**.
3. Click the Dial-in tab.
4. Under **Remote Access Permission (Dial-in or VPN)**, select **Allow access**.

Note You can also set remote access by using a remote access policy. For more information, type **remote access policies** in the index for the Windows 2000 Help system.

5. Click the **General** tab.
6. Clear the **User must change password at next logon** and **Account is disabled** check boxes, and then click **OK**.

After you have configured the logon rights for the Anonymous account, you add the Anonymous user to Visual SourceSafe and configure the permissions for the Anonymous account in the same way that you configured the other users.

To add and configure the Anonymous account in the Visual SourceSafe Administrator

1. Open the Visual SourceSafe Administrator.
2. On the **Users** menu, click **Add User**.
3. In the **Add User** dialog box, in the **User name** box, type the Anonymous account user name, and leave the **Password** field blank.
4. Click **OK**.
5. On the **Tools** menu, click **Options**, click the **General** tab, and make sure that **Use network name for automatic user log in** is selected.

Synchronizing a SharePoint team Web site or FrontPage-based Web site with a Visual SourceSafe project

After you have installed Microsoft Visual SourceSafe 5.0 or later and established the user and group accounts that will have permissions to use the Visual SourceSafe project, you must synchronize the Visual SourceSafe project with your SharePoint team Web site or FrontPage-based Web site. To synchronize your SourceSafe project, you use the **Configure Version Control** page in the Site Administration pages.

To view the Site Administration page

- If you are a server administrator, on the server computer click **Start**, point to **Programs**, point to **Administrative Tools**, click **Microsoft SharePoint Administrator**, and then on the **Server Administration** page, click the name of the site you want to manage.

- If you are a site administrator, on your Web site, click **Site Settings**, and then under **Web Administration**, click **Go to Site Administration**.

To synchronize a Visual SourceSafe project

1. On the **Site Administration** page, click **Configure version control**.
2. On the **Configure Version Control** page, select **Use external version control (Visual SourceSafe only)**.
3. In the text box, type the name of the Visual SourceSafe project you want to use. Type **$/** followed by a project name, as in **$/rootweb**.
4. Click **Submit**.

If the project does not already exist in Visual SourceSafe, it is created, and all files in the Web site are checked in. If the project already exists, the SharePoint team Web site or FrontPage-based Web site and the project are merged. Any file in the SharePoint team Web site or FrontPage-based Web site that is in an ambiguous state (such as a file that has the same name as a file in the Visual SourceSafe project but that was never in the Visual SourceSafe database) remains checked out.

Once a SharePoint team Web site or FrontPage-based Web site is synchronized with a Visual SourceSafe project, all files in the Visual SourceSafe project will be displayed in your Web page editor with a green dot (to indicate that the file is checked in) or a red checkmark (to indicate that the file is checked out).

Enabling built-in source control

You can enable the source control built into SharePoint Team Services or FrontPage Server Extensions 2002 on Microsoft Windows 2000 Server or Professional, or Microsoft Windows NT Server 4.0, or on FrontPage Server Extensions 2002 on UNIX with FrontPage-patched Apache Web server. To enable built-in source control, you use the **Configure Version Control** page in the Site Administration pages.

To view the Site Administration page

- If you are a server administrator, on the server computer click **Start**, point to **Programs**, point to **Administrative Tools**, click **Microsoft SharePoint Administrator**, and then on the **Server Administration** page, click the name of the site you want to manage.
- If you are a site administrator, on your Web site, click **Site Settings**, and then under **Web Administration**, click **Go to Site Administration**.

To enable built-in source control

1. On the **Site Administration** page, click **Configure version control**.
2. On the **Configure Version Control** page, select **Use built-in version control**.
3. Click **Submit**.

Analyzing Web Site Usage

If you want to know what kind of impact your Web site has, you need to track how many users visit your site, the type and number of hits your site receives, and other site-usage information. Microsoft's SharePoint™ Team Services and Microsoft® FrontPage Server Extensions 2002 include features that analyze usage of your site. Summary and detailed usage reports supply information such as:

- Total hits on a site
- Total unique hits on a site
- Top page by hits
- Most popular browsers
- Top referring site
- Hits per page

The usage reports rely on a plain-text log file generated by the Web server that the Web site is hosted on (either Internet Information Services or FrontPage-patched Apache Web server), which keeps a distilled record of every transaction on your Web site. When you create a usage report in a SharePoint Team Services–compatible Web page editor, such as Microsoft FrontPage® 2002, the data from the log file is arranged into a readable format.

Note that usage data is collected for an entire virtual server at a time, even if you have separate Web sites on a virtual server.

Although you view these reports from within a SharePoint Team Services–compatible Web page editor, you administer the settings for processing the usage log with SharePoint Team Services and FrontPage Server Extensions 2002 commands. Using HTML Administration pages and command-line tools, you can control:

- Whether the usage data is generated.

  If you do not want to use the usage analysis features, you can turn off the usage analysis log to conserve hard-disk space (although the log files themselves are not large). If you decide that you do want to use these features, you can turn the log processing on again. Note that each time the log file is processed, the data is appended to the original log. If you want to conserve disk space, you should turn off usage analysis before it has been run the first time. This option is available only on the Microsoft Windows® platform.

- When the usage log is processed.

  You can schedule the usage log to be generated at a convenient downtime for your Web site (by using the command line or HTML Administration pages on Windows or the cron facility on UNIX). If internal employees primarily use the Web site, for example, you might schedule the log to be processed at night, when demand on the site is lower than during working hours. You can schedule the log to be processed on a daily, weekly, or monthly basis, depending on how closely you want to track your site's usage. If you only run usage

reports once a month, you can process the log once a month. If you want to know how your site is being used every day, you can set the log to be processed daily.

- How long usage data is stored.

You can store usage data indefinitely. However, unless you are running long-term usage studies, you probably do not need to store that much data. You can specify that usage analysis data be automatically deleted after a certain number of months.

- Whether to use 24-hour increments in the log file.

If you use this option, data is accumulated for full days only. For example, if you process the usage log at 12:35 A.M. on Friday, the data is processed through midnight on Thursday. If you turn off this option, all hits are processed right up to the time set for processing. In this case, usage processing contains all hits up until 12:35 A.M. on Friday. The data in the log file determines the data used for the usage reports. Choose the 24-hour increments option if you prefer a cleaner usage report, with data for full days only.

Analyzing Web site usage on Windows

On the Windows platform, you have the choice of using HTML Administration pages or the command-line tools to administer usage analysis settings. On the Windows platform, the usage analysis processes rely on the SharePoint Timer Service for scheduling, and scheduling the Timer Service can be done with either HTML Administration pages or the command-line tools. For more information about the SharePoint Timer Service, see Scheduling Timed Jobs.

Using HTML Administration pages to administer usage settings

You can control the usage analysis log settings by using HTML Administration pages. By using the Site Administration pages, you can turn usage analysis on or off, set the schedule when to process the usage log, and specify when to delete stored usage data.

To view the Site Administration page

- If you are a server administrator, on the server computer click **Start**, point to **Programs**, point to **Administrative Tools**, and then click **Microsoft SharePoint Administrator**, and then on the **Server Administration** page, click the name of the site you want to manage.
- If you are a site administrator, on your Web site, click **Site Settings**, and then under **Web Administration**, click **Go to Site Administration**.

Usage analysis is enabled by default. If you do not want to use the usage analysis features for your Web site, you can disable the usage analysis logging process.

To turn off usage analysis log processing

1. Go to the **Site Administration** page for the virtual server you want to change.

2. Under **Configure Usage Analysis Settings**, click **Change usage analysis settings**.
3. In the **Usage Analysis** section, select **Off**.

   If you later decide that you do want usage analysis, you can turn the log processing on again by selecting **On** in the **Usage Analysis** section.
4. Click **Submit**.

If you choose to use the usage analysis features, you can control how often and when the usage log is processed. You can set the log to be generated on a daily, weekly, or monthly basis, and then set the time of day to process the log.

To specify when to process the usage analysis log

1. Go to the **Site Administration** page for the virtual server you want to change.
2. Under **Configure Usage Analysis Settings**, click **Change usage analysis settings**.
3. In the **Recurrence settings** area, do one of the following:
   - To set a daily recurrence to run the log, select **Daily**, then select the time.
   - Or, to set a weekly recurrence, select **Weekly**, then select time and day of the week.
   - Or, to set a monthly recurrence, select **Monthly**, then select a day of the month and a time.
4. Click **Submit**.

You can store usage data indefinitely, but that can lead to a cluttered usage report. Because you most likely will not need old usage data, you can set the stored data to be deleted automatically. You specify how long to keep usage data in monthly increments (specify **0** to keep data only for the current month, **1** to keep data from the previous month, and so on). By default, usage data is stored for 12 months.

To set expiration for stored usage data

1. Go to the **Site Administration** page for the virtual server you want to change.
2. Under **Configure Usage Analysis Settings**, click **Change usage analysis settings**.
3. In the **Additional Usage Analysis Settings** area, select the **Automatically delete stored usage data after ___ months** check box.
4. In the **months** box, type the number of months you want to store usage analysis data.
5. Click **Submit**.

When you process the usage log, you can specify whether to use full-day increments, or continue collecting data up until the time for log processing.

To track usage data by full days only

1. Go to the **Site Administration** page for the virtual server you want to change.
2. Under **Configure Usage Analysis Settings**, click **Change usage analysis settings**.

3. In the **Additional Usage Analysis Settings** section, select the **Process log file data for full days only** check box.
4. Click **Submit**.

You can also specify an administrator e-mail name so that you can be notified when the automatic usage analysis processes have been run.

To specify an administrator e-mail name

1. Go to the **Site Administration** page for the virtual server you want to change.
2. Under **Configure Usage Analysis Settings**, click **Change usage analysis settings**.
3. In the **Additional Usage Analysis Settings** section, in the **Administrator E-mail Address(es)** box, type the full e-mail address for the administrator to notify.

   If you want to include multiple e-mail addresses, separate each e-mail address with a comma. For example, "someone@microsoft.com,someone@microsoft.com".
4. Click **Submit**.

If you no longer want separate usage analysis settings for your Web site, you can reset the usage analysis settings to match the global usage analysis settings for your server. When you restore the defaults for usage analysis, the per-Web site or per-virtual server settings are deleted, and the global settings are applied to your Web site. For example, if the global usage analysis settings specified that usage data is kept for one month, and the log file data is processed in 24-hour increments, when you click **Restore Defaults**, these settings are applied to your Web site.

To restore the default usage analysis settings

1. Open the **Configure Usage Analysis** administration page.
2. Click **Restore Defaults**.

Using the command line to administer usage settings

You can also use the command-line tools, Owsadm and Owsrmadm, to control the usage analysis settings. To change the usage analysis settings, you use the **usage** operation.

Scheduling usage log processing

The **usage** operation takes the *schedule* parameter and controls all of the usage analysis settings. Usage analysis is enabled by default, and the log is set to process every Sunday morning. To turn usage analysis off, you set the *schedule* parameter to **off**. The following example syntax shows how to turn off usage analysis for port 80:

```
owsadm.exe –o usage –schedule off –p 80
```

To turn usage analysis on and set the usage log file to be processed at a particular time, you use the **usage** operation and the *schedule* parameter set to a particular time. The following example syntax shows how to set the usage analysis job to be run every Monday at 11:59 P. M. for port 80.

```
owsadm.exe –o usage –schedule "weekly at Mon 23:59" –p 80
```

You can schedule the usage log to run daily, weekly, or monthly simply by specifying a day, a week, or a month in the *schedule* parameter. To set the log to be processed daily, use the format "daily at <time>" in the *schedule* parameter, for example "daily at 23:59". To specify monthly processing, use the format "monthly at <day> <time>". For example, use "monthly at 12 23:59" to process the log on the twelfth day of every month at 11:59 P.M. For more information about the values to use with the *schedule* parameter, see Scheduling Timed Jobs.

If you want to start the usage log to process now rather than at the next scheduled time, you can use the **usage** operation without the *schedule* parameter. Using the operation this way gives you an instant build of the usage analysis log.

Setting other usage analysis properties

To set the expiration time for usage data, you use the **usageanalysislogexpiry** property with the **setproperty** operation. For example, to set the usage analysis log to expire in six months, you would use the following syntax:

```
owsadm.exe –o setproperty –pn usageanalysislogexpiry –pv 6
```

To set the usage log to store information for full days only, you use the **usagefulldaysonly** property with the values **true** or **false**.

For more information about setting properties by using the command-line tools, see Setting Configuration Properties.

Choosing a log format

There are several standard log formats in use on the Windows and UNIX platforms. For Windows, it is recommended that you use the W3C Extended Log File Format for best results with usage analysis. You can, if you prefer, use the NCSA common log format or the IIS format, but less data is accumulated when you use these formats. On the Windows platform, usage analysis uses the following logging properties: Date, Time, Client IP Address, User Name, Method, URI Stem, Http Status, Bytes Sent, User Agent, and Referrer. The Method, URI Stem, and Http Status properties are required in order for usage analysis to run properly. The Date and Time properties are also required if they cannot be determined from the log file name.

For UNIX platforms, the preferred format is the NCSA's extended/combined log format. This format logs all of the information that the usage reports need. If you prefer to use a different log format for UNIX, be sure that the log format follows these guidelines:

- Each directive (such as %t or %u) must be separated by a comma, a space, or some other standard separator.
- Usage analysis processing uses the following directives for UNIX: Date/Time: %t; Client IP Address: %a; User Name: %u; Method/URI Stem: %r; Http Status: %s; Bytes Sent: %b; User Agent: %{User-agent}I; Referer: %{Referer}. The Http Status and URI Stem directives are required. It is recommended that you include the %t directive if the date and time cannot be determined from the log file name.
- If you do not want to use the %r directive, then you must include the %U (file name) directive.
- The Date/Time directive must appear as %t. Usage analysis does not recognize the %{format}t directive.
- You can log data from multiple virtual servers in a single file if the %v (full host name) directive is included in the log format.

Changing logging settings on Windows

Usage reports are generated from log files created by the Web server your Web site is hosted on (either Internet Information Services or FrontPage-patched Apache Web server). The usage data you have to work with depends on the logging settings specified for your Web server. If you want the full capabilities of the usage analysis features in SharePoint Team Services and FrontPage Server Extensions 2002, you must make sure that the logging settings for your Web server are configured to produce the data you want.

For IIS 5.0, the default settings generate most of the usage analysis data you need (with the exception of referral data). For IIS 4.0, you must change the logging settings to generate the complete usage analysis data.

To change the usage logging settings for IIS 4.0

1. On your server computer, click **Start**, point to **Programs**, point to **Windows NT 4.0 Option Pack**, point to **Microsoft Internet Information Server**, and then click **Internet Services Manager**.
2. Click the plus sign (+) next to **Internet Information Server**.
3. Right-click your Web server name, and then select **Properties**.
4. In the **Master Properties** box, select **WWW Service**, and then click **Edit**.
5. On the **Web Site** tab, in the **Active log format** box, select **W3C Extended Log File Format**, and then click **Properties**.
6. In the Extended Logging Properties box, on the **Extended Properties** tab, select the following check boxes: **Date**, **Time**, **Client IP Address**, **User Name**, **Method**, **URI Stem**, **Http Status**, **Bytes Sent**, **User Agent**, **Referrer**.

Note that the **Method**, **URI Stem**, and **Http Status** check boxes must be selected (and

Date and **Time** are recommended) in order for usage analysis to run properly. Also note that you can select as many logging check boxes as you like, but usage analysis only uses data from the check boxes listed in this step.

7. Click **OK** to close the **Extended Logging Properties** box.
8. Click **OK** to close the **WWW Service Master Properties** box.
9. Click **OK** to close the **Properties** box.

By default, all of the data you need for usage reports is collected by IIS 5.0, with the exception of referral data. If you want to know which site referred a user to your site, you must add the referral data to your log files for IIS 5.0.

To add referral data to IIS 5.0 log files

1. Click **Start**, point to **Programs**, point to **Administrative Tools**, and then click **Internet Services Manager**.
2. Right-click your Web server name, and then select **Properties**.
3. In the **Master Properties** box, select **WWW Service**, and then click **Edit**.
4. On the **Web Site** tab, in the **Active log format** box, select **W3C Extended Log File Format**, and then click **Properties**.
5. In the **Extended Logging Properties** box, on the **Extended Properties** tab, select the **Referer** check box.

   Note that the **Method**, **URI Stem**, and **Http Status** check boxes must also be selected for usage analysis to run properly.
6. Click **OK** to close the **Extended Logging Properties** box.
7. Click **OK** to close the **WWW Service Master Properties** box.
8. Click **OK** to close the **Properties** box.

Analyzing Web site usage on UNIX

On the UNIX platform, you can use HTML Administration pages to set the expiration for stored usage data or to specify whether to track usage data by full days only. If you have a mail server available, you can also specify an e-mail name to receive notification that the usage log processing has been run. To schedule usage analysis log processing, however, you must use the **cron** facility.

Using HTML Administration pages to administer usage settings on UNIX

You can set stored usage data to be deleted automatically by specifying how long to keep the data in monthly increments (specify **0** to keep data only for the current month, **1** to keep data from the previous month, and so on). By default, usage data is stored for 12 months.

To set expiration for stored usage data

1. Go to the **Site Administration** page for the virtual server you want to change.
2. Under **Configure Usage Analysis Settings**, click **Change usage analysis settings**.
3. In the **Additional Usage Analysis Settings** area, select the **Automatically delete stored usage data after ___ months** check box.
4. In the **months** box, type the number of months you want to store usage analysis data.
5. Click **Submit**.

When you process the usage log, you can specify whether to use full-day increments, or continue collecting data up until the time for log processing.

To track usage data by full days only

1. Go to the **Site Administration** page for the virtual server you want to change.
2. Under **Configure Usage Analysis Settings**, click **Change usage analysis settings**.
3. In the **Additional Usage Analysis Settings** section, select the **Process log file data for full days only** check box.
4. Click **Submit**.

You can also specify an administrator e-mail name so that you can be notified when the automatic usage analysis processes have been run.

To specify an administrator e-mail name

1. Go to the **Site Administration** page for the virtual server you want to change.
2. Under **Configure Usage Analysis Settings**, click **Change usage analysis settings**.
3. In the **Additional Usage Analysis Settings** section, in the **Administrator E-mail Address(es)** box, type the full e-mail address for the administrator to notify.

   If you want to include multiple e-mail addresses, separate each e-mail address with a comma. For example, "someone@microsoft.com,someone@microsoft.com".
4. Click **Submit**.

If you no longer want separate usage analysis settings for your Web site, you can reset the usage analysis settings to match the global usage analysis settings for your server. When you restore the defaults for usage analysis, the per-Web site or per-virtual server settings are deleted, and the global settings are applied to your Web site. For example, if the global usage analysis settings specified that usage data is kept for one month, and the log file data is processed in 24-hour increments, when you click **Restore Defaults**, these settings are applied to your Web site.

To restore the default usage analysis settings

1. Open the **Configure Usage Analysis** administration page.
2. Click **Restore Defaults**.

Using the command line to administer usage settings on UNIX

On the UNIX platform, you must use the command-line tools, Owsadm and Owsrmadm, to administer usage analysis settings. This is because on UNIX, the usage analysis processes are scheduled by using the **cron** facility. The **cron** facility is available only from the command line, so HTML Administration pages cannot be used for scheduling usage analysis tasks on UNIX. For more information about using **cron** and **crontab**, see the manual (man) pages for your UNIX system.

Running usage log processing on UNIX

If you want regular and consistent usage analysis reports, you must process the FrontPage-patched Apache log files on a regular basis. To process the log files, you use the **usage** operation with Owsadm or Owsrmadm. You can use the same parameters as you use on the Windows platform to run the log processing. To schedule the log to be processed automatically, however, you use the **cron** facility rather than the **usage** operation with the *schedule* parameter. For example, to run usage processing for all virtual servers on the current server, you would use the following syntax with the **cron** facility:

```
/usr/local/frontpage/version5.0/bin/owsadm.exe –o usage [–p <port> -m
<hostname>]
```

Running remote usage processing on the UNIX platform

On the UNIX platform, you have the ability to run usage analysis processing from a remote computer. You can set up a computer to read the content from the Web server and do the usage analysis, without running the usage analysis process on the actual Web server. When you do usage processing remotely, you do not have to worry about interrupting or slowing service on your main Web server while the usage analysis is running.

For example, say you have a Web server named ServerA. It hosts the content for your site and performs all of the FrontPage Server Extensions 2002 processes.

1. Using network file system (NFS), you create a share on ServerA that has all of the Web site content in it.
2. You then create another share that has the server root in it.
3. On ComputerB, install FrontPage Server Extensions 2002 in exactly the same position in its file system (so the two systems mirror each other — for example, all content on both computers is in \usr\www\htdocs\webcontent).
4. Copy the **we*.cnf** files from \usr\local\FrontPage\version5.0 from ServerA to ComputerB.

5. Run the usage operation by running `owsadm.exe –o usage` on ComputerB.

At the end of this process, the configuration files copied from ServerA are read on ComputerB, the usage analysis processing is performed, and all of the data is written back into the content that was shared out from ServerA.

Specifying log file locations

You must specify where the Web server log files are stored when you run usage processing on the UNIX platform. You use the **LogFileLocation**, **LogRollover**, and **LogFileFormat** properties to specify this information. The following table describes these properties.

Property name	Description
LogFileLocation	Specifies the full path to the log file. Optionally, can include wildcards for server log files with names based on the date and time that they were created. Use in combination with the **LogRollover** property to set how often to refresh the data.
LogRollover	Specifies the frequency and time at which the log files are rolled over. Works in conjunction with the substitution patterns in the **LogFileLocation** value.
LogFileFormat	If not specified, FrontPage Server Extensions read this value from the httpd.conf file for the server. The property's value is any valid value for the Apache "LogFormat" directive. The value should not be quoted, but may contain embedded quotes following the standard Apache convention.

For example, to use log files created every week on Sunday at midnight, the **LogFileLocation** value would be **/usr/local/www/logs/access_log.%m-%d-%Y**, and the **LogRollover** value would be **weekly at Sun 00:00**. The **LogRollover** value must be a valid SharePoint Timer Service value, such as **"Mon at 03:00"** or **"Weekly at Mon 1:21"**. For more information about the values you can use with the Timer Service, see Scheduling Timed Jobs.

Toolbox For a complete list of the properties you can set from the command line, see the **Properties** worksheet in the Owscmdln.xls spreadsheet.

To get an actual path to a log file, you combine the **LogFileLocation** string with a time and date, which is derived from the **LogRollover** setting. The following table lists the format specifiers used in the **LogFileLocation** path:

Format Specifier	Description

%A[WEEKDAYNAMES]	Arbitrary weekday names. For example, %A[SU,MO,TU,WE,TH,FR,SA] or %A[Sun,Mon,Tue,Wed,Thu,Fri,Sat]
%a	Abbreviated weekday name (Sun,Mon,Tue,Wed,Thu,Fri,Sat). Equivalent to %A[Sun,Mon,Tue,Wed,Thu,Fri,Sat].
%B[MONTHNAMES]	Arbitrary month names. For example, %B[January,February,March,April,May,June,July,August,September,October,November,December] or %B[Jan,Feb,Mar,Apr,May,Jun,Jul,Aug,Sep,Oct,Nov,Dec]
%b	Abbreviated month name (Jan,Feb,Mar,Apr,May,Jun,Jul,Aug,Sep,Oct,Nov,Dec). Equivalent to %B[Jan,Feb,Mar,Apr,May,Jun,Jul,Aug,Sep,Oct,Nov,Dec].
%d	Day of month as decimal number (01-31)
%H	Hour in 24-hour format (00-23)
%j	Day of year as decimal number (001-365)
%m	Month as decimal number (01-12)
%n	Decimal sequence number (for sequentially-numbered log files)
%U	Week of year as decimal number, with Sunday as the first day of the week (00-53)
%v	Leaf host name (for example, "user")
%V	Full host name (for example, "user.isp.com")
%w	Weekday as decimal number (0-6; Sunday is 0)
%W	Week of year as decimal number, with Monday as the first day of the week (00-53)
%x	Week of month as decimal number, with Sunday as the first day of the week (00-53) IIS Servers only.
%y	Year without century as decimal number (00-99)
%Y	Year with century as decimal number
%%	Percent sign

Note Any of the format specifiers with a numeric value can be preceded by an integer (for example, "%2n") to specify the exact number of digits to use. Otherwise, for all

specifiers except %y and %Y, the minimum number of digits will be used.

The following format specifiers are not supported: hour in 12-hour format (%I), AM/PM indicator (%p), minute as decimal number (%M), seconds (%S), long-format day names (%A), and long-format month names (%B).

To specify the **LogFileLocation**, you include both the path and the format specifiers to use. For example, the following string gives you a log file path to files named with the month and day of the month in decimal format:

```
/usr/local/www/vhost1/host%m%d.log
```

As another example, if you wanted to rotate your log files at 3:00 A.M. every Saturday, you could copy the log files to a file with the absolute path /usr/local/www/t1234/logs/MMDD.log, where MMDD is the month and day of the month when the log files are copied. You would then set the **LogRollover** property to **"weekly at Sat 03:00"** and set the **LogFileLocation** property to **/usr/local/www/t1234/logs/%2m%2d.log**.

Specifying a domain name for usage log processing

Usage processing uses the domain name for your Web site to determine how many visits your site has received. With some Web server configurations, the domain name cannot be determined accurately. In this case, you can use the **UsageServerUrls** property to specify the domain name or names for your Web site. You must specify the domain name for your Web site if your Web site is configured in any of the following ways:

- Your Web site accepts requests for multiple domain names (for example, http://www.mycompany.com and http://mycompany.com)
- Your Web site is set up for multihosting through hardware, rather than software (you only specified the IP address in the server configuration).
- No domain name is specified in the configuration file and the Web server domain name is different from the machine name.

Set the **UsageServerUrls** property to a comma-separated list of all the domain names that can be used to gain access to your Web site. Include the http:// prefix at the beginning of each entry in the list. For example, in the first scenario, you would specify **http://www.mycompany.com,http://mycompany.com** as the value for the **UsageServerUrls** property. If your Web site uses Secure Sockets Layer (SSL), you must also include the https:// prefix in the list.

The following example syntax sets the **UsageServerUrls** property to recognize both www.mycompany.com and mycompany.com:

```
owsadm.exe –o setproperty –pn usageserverurls –pv "http://www.mycompany.com,
http://mycompany.com" –p 80
```

Related links

For more information about the SharePoint Timer Service, see Scheduling Timed Jobs.

For more information about setting the usage analysis properties on the command line, see Setting Configuration Properties.

Scheduling Timed Jobs

Several features of Microsoft's SharePoint™ Team Services and Microsoft® FrontPage® Server Extensions 2002 rely on scheduled background processes. For example, to be able to review usage analysis information, the information must be gathered, preferably when your site is not being heavily used. With SharePoint Team Services and FrontPage Server Extensions 2002, you can schedule the following operations to occur automatically at specific times:

- Running usage analysis
- Checking server health
- Backing up the database

Scheduled times apply to all Web sites on a particular virtual server. You cannot schedule usage analysis to run at 9 A.M. on one Web site and at 10 A.M. on another Web site within the same virtual server. You can schedule jobs for all virtual servers on a server computer, but any existing virtual server settings take precedence. For example, if you have a virtual server (Vserver1) set to run a job at midnight every night, and you set all virtual servers to run the job at 11 P.M. instead, the job is still run at midnight for Vserver1.

On the Windows platform, the SharePoint Timer Service — a background utility included with SharePoint Team Services and FrontPage Server Extensions 2002 — handles scheduled jobs. To set times for the SharePoint Timer Service, you use commands in the command-line tool or HTML Administration pages. On UNIX, you use the built-in UNIX cron facility to schedule jobs.

About the calendar

The SharePoint Timer Service relies on the Gregorian calendar for scheduling. For every job you schedule, you must specify a beginning time for that job based on a 24-hour clock. You specify the time in local time versus an offset from Universal Coordinate Time (UCT), and the time is stored in that format as well. Any other scheduling specifications (such as day of the week) are optional. If none are given, then the job is run at the given time every day.

The dates used by the SharePoint Timer Service are not stored in context. This means that you cannot schedule jobs to run every X days/weeks/months/years, where X is greater than 1. So, while you can schedule jobs to run every day, every week, every month, or every year, you cannot schedule a process for every two days, and so on. Neither can you schedule jobs for relative days in a month, such as the third Monday of every month.

You can, however, specify jobs to recur every <x> minutes, hourly at <minute>, daily at <time>, weekly at <day of week> <time>, monthly at <day of month> <time>, and yearly at <month> <day> <time>. You specify times using the HH:MM format. Specify days of the week by using the following abbreviations: Mon, Tue, Wed, Thu, Fri, Sat, Sun. Specify months by using the following abbreviations: Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec.

> **Note** The days of the week and names of the months must be typed in their English abbreviated form (for example, Mon, Tue, Wed and Jan, Feb, Mar). The SharePoint Timer Service does not recognize the names of the months or days of the week in other languages.

For example, you can set a process to run at the following times using the following syntax:

- Every 10 minutes
- Hourly at 10
- Daily at 22:30
- Weekly at Mon 1:21
- Monthly at 1 2:00
- Yearly at Jan 4 12:31

Scheduling jobs under Windows

You use commands in the command-line tool or HTML Administration pages to schedule jobs on the Microsoft Windows NT® platform. The commands work like any other command in the command-line tool or HTML Administration pages. On the command line, you type the appropriate operation and include the scheduling information. On HTML Administration pages, you use the administration page for the job you want to schedule, and then select the time and date that you want to run the job.

Using the command line to schedule jobs

You schedule jobs from the command line by using the *–schedule* parameter with the operation you want to schedule. The following operations take the *schedule* parameter:

Operation	Definition
Usage	Runs usage analysis reports. Specify a time/day/month/year to run the usage analysis reports.

Check	Checks for problems with SharePoint Team Services and FrontPage Server Extensions 2002 and fixes them. Specify a time/day/month/year to run server health checks.
Backupdb	Backs up the data in the database. Specify a time/day/month/year to back up the database. SharePoint Team Services only.

The following example syntax shows how to set the usage analysis job to be run every Monday at 11:59 P.M. for port 80.

```
owsadm.exe –o usage –schedule "weekly at Mon 23:59" –p 80
```

Using HTML Administration pages to schedule jobs

You can also schedule jobs by using HTML Administration pages. To do so, you go to the page that contains the settings for the job you want to schedule, and then select the day, date, month, year, and time that you want the jobs performed. For example, you would use the following procedure to set SharePoint Team Services weekly subscription notifications to be sent out at noon on Mondays. The Web subscriptions settings are managed from the Site Administration pages.

To view the Site Administration page

- If you are a server administrator, on the server computer click **Start**, point to **Programs**, point to **Administrative Tools**, and then click **Microsoft SharePoint Administrator**, and then on the **Server Administration** page, click the name of the site you want to manage.
- If you are a site administrator, on your Web site, click **Site Settings**, and then under **Web Administration**, click **Go to Site Administration**.

To change the time for weekly subscription notifications

1. On the **Site Administration** page for the virtual server, under **Web Discussions and Subscriptions**, click **Change Web subscription settings**.
2. In the **Notification Recurrence Settings** area, in the **Weekly notifications** box, specify the day of the week and the time to send out weekly notifications.
3. Click **Submit**.

Scheduling jobs under UNIX

You use the built-in UNIX **cron** and **crontab** facilities to schedule jobs on the UNIX platform. To run a scheduled job for FrontPage Server Extensions 2002, you simply add the job to the list for **cron** by using **crontab**. For example, if you want to back up the database every week, you would add the Owsadm command-line syntax (`owsadm.exe –o backupdb –p 80`) to the **crontab** file, and then set the schedule using the standard **crontab** syntax. For UNIX,

any FrontPage Server Extensions 2002 command can be a scheduled job. For more information about using **cron** and **crontab**, see the manual (man) pages for your UNIX system.

Related links

For more information about running particular jobs, see the following topics: Analyzing Web Site Usage, Managing Web Document Discussions and Subscriptions, Backing Up and Restoring Data, and Monitoring Server Health

Searching SharePoint Team Web Sites and FrontPage-based Web Sites

Microsoft's SharePoint™ Team Services and Microsoft® FrontPage® Server Extensions 2002 include full-text search functionality so that your users can find documents on your Web sites. The search functionality is implemented differently depending on the platform you are using. On the Microsoft Windows® platform, Index Server provides searching; on the UNIX platform, the built-in Wide Area Information Server (WAIS) search engine provides searching.

About Index Server

Index Server is a content-indexing and search component included with Microsoft Internet Information Server (IIS) 3.0 and later. When SharePoint Team Services or FrontPage Server Extensions 2002 are installed on the Windows platform, the search features use the document index and search component of Index Server. The Index Server documentation provides a full discussion on such features as creating custom searches, and provides more information on installing Index Server, setting up the search catalog, associating it with a virtual server, and setting administration options.

> **Note** The search features in SharePoint Team Services and FrontPage Server Extensions 2002 are turned on automatically when you install SharePoint Team Services. There are no administrative controls within SharePoint Team Services or FrontPage Server Extensions 2002 to disable searching. You can turn off searching by turning off Index Server, but the search controls in the SharePoint team Web site user interface are not disabled.

Index Server uses catalogs to index the content of your Web site. If you are running Windows 2000 Server, you have a dedicated catalog for SharePoint Team Services. If you are running Windows NT® 4 Server or Workstation (FrontPage Server Extensions 2002 only), or Windows 2000 Professional, Web site content is indexed in the default catalog for your Web server.

Index Server catalogs on Windows 2000 Server

On Windows 2000 Server, SharePoint team Web sites and FrontPage-based Web sites are indexed in a dedicated search catalog used only by SharePoint Team Services and FrontPage Server Extensions 2002. This catalog is created automatically when SharePoint Team Services or FrontPage Server Extensions 2002 is installed and appears as a virtual server called OWS_Search_Proxy on the root web of your Web server. When you add virtual servers to your Web server and extend them with SharePoint Team Services or FrontPage Server Extensions 2002, each virtual server is automatically added to the catalog as a virtual directory. For example, if you have Web server X with virtual servers a, b, and c, you will have virtual directories named OWS_VDIR_A, OWS_VDIR_B, and so on within the OWS_Search_Proxy virtual server. These virtual directories point back to the content on the individual virtual servers. Segmenting the catalog into virtual directories allows you to search within a single virtual server, rather than across all virtual servers on the server computer.

Because the catalog is owned by SharePoint Team Services or FrontPage Server Extensions 2002, it is updated automatically when you uninstall SharePoint Team Services or FrontPage Server Extensions 2002. If you uninstall from a single virtual server, the virtual directory for that virtual server is automatically removed from the OWS_Search_Proxy virtual server. If you uninstall completely, the OWS_Search_Proxy virtual server is removed as well.

Index Server catalogs on Windows NT 4 or Windows 2000 Professional

On Windows NT 4 Server or Workstation (FrontPage Server Extensions 2002 only), or Windows 2000 Professional, all of your Web sites are indexed in the default catalog for Index Server. This affects how searching is implemented in a couple of ways:

- Because all of your virtual servers are indexed in the default catalog, you cannot limit a search to a single virtual server. If your Web server has multiple virtual servers and you are using the default Index Server catalog, the users see results from all of the virtual servers at once.
- The default catalog is not automatically updated when you add a new virtual server. If you add a new virtual server, you must also add it to the default Index Server catalog using the Index Server tools in the Microsoft Management Console (MMC).

If you prefer, however, you can use a catalog other than the default catalog for your SharePoint Team Services or FrontPage Server Extensions 2002 virtual servers. To use a different catalog, you must first use the Index Server management tools to create the catalog, and then edit the `catalog: web` line in the Search.IDQ file to point to your new catalog. For more information about working with search catalogs, see the Index Server documentation.

About WAIS

On the UNIX platform, the FrontPage Server Extensions 2002 use the WAIS search engine.

With the WAIS engine, a full-text index is created in the _vti_txt directory under the Web's top-level directory. The index is updated whenever a page is saved or whenever the Web is recalculated. New content is automatically added to the index as pages are saved. However, deleted content is not removed from the index until the Web site is recalculated, using the **Recalculate Hyperlinks** command (**Tools** menu in FrontPage).

Files and Permissions on Internet Information Service

This appendix lists the detailed, minimum file permission settings that must be in place for Microsoft's SharePoint™ Team Services and Microsoft® FrontPage Server Extensions 2002 to perform as designed. The extensions only edit ACLs, they never affect the permissions of any accounts not listed below.

File permissions assigned by the server health features

The server health commands included with SharePoint Team Services and FrontPage Server Extensions 2002 correct problems in NTFS permissions. The following list of files shows the minimum permissions required for SharePoint Team Services and FrontPage Server Extensions 2002, as set by this command.

Windows NT directory	User	Permissions
\WINNT\system32\infoadmn.dll	INTERACTIVE NETWORK	Read (RX)(RX) Read (RX)(RX)
\WINNT\system32\mfc42.dll	INTERACTIVE NETWORK	Read (RX)(RX) Read (RX)(RX)
\WINNT\system32\msvcirt.dll	INTERACTIVE NETWORK	Read (RX)(RX) Read (RX)(RX)
\WINNT\system32\msvcrt.dll	INTERACTIVE NETWORK	Read (RX)(RX) Read (RX)(RX)
\WINNT\system32\netapi32.dll	INTERACTIVE NETWORK	Read (RX)(RX) Read (RX)(RX)
\WINNT\system32\netrap.dll	INTERACTIVE NETWORK	Read (RX)(RX) Read (RX)(RX)
\WINNT\system32\rpcltc1.dll	INTERACTIVE NETWORK	Read (RX)(RX) Read (RX)(RX)

\WINNT\system32\wsock32.dll	INTERACTIVE NETWORK	Read (RX)(RX) Read (RX)(RX)

SharePoint Team Services and FrontPage Server Extensions 2002 installation directory

The SharePoint Team Services and FrontPage Server Extensions 2002 installation directory is C:\Program Files\Common Files\Microsoft Shared\Web Server Extensions by default.

Installation directory	User	Permissions
\Program Files\Common Files \Microsoft Shared\Web Server Extensions \40\servsupp\	INTERACTIVE NETWORK	Read (RX)(RX) Read (RX)(RX)
\Program Files\Common Files \Microsoft Shared\Web Server Extensions\40\servsupp\fp30msft.dll	INTERACTIVE NETWORK	Read (RX) Read (RX)
\Program Files\Common Files \Microsoft Shared\Web Server Extensions \40\servsupp\servers.cnf	INTERACTIVE NETWORK	Special Access (R) Special Access (R)
\Program Files\Common Files \Microsoft Shared\Web Server Extensions \40\bin\	INTERACTIVE NETWORK	List (RX)(Not Specified) List (RX)(Not Specified)
\Program Files\Common Files \Microsoft Shared\Web Server Extensions \40\bin\fp4*utl.dll	INTERACTIVE NETWORK	Read (RX) Read (RX)
\Program Files\Common Files \Microsoft Shared\Web Server Extensions \40\servsupp\fp4*txt.dll	INTERACTIVE NETWORK	Read (RX) Read (RX)
\Program Files\Common Files \Microsoft Shared\Web Server Extensions \40\servsupp\fp4*wel.dll	INTERACTIVE NETWORK	Read (RX) Read (RX)

\Program Files\Common Files \Microsoft Shared\Web Server Extensions \40\bin\fp4*vss.dll	INTERACTIVE NETWORK	Read (RX) Read (RX)

\Program Files\Common Files \Microsoft Shared\Web Server Extensions \40\bin\1033\fpext*.msg Note that these files are only present for multilanguage support, and are not normally present in an English installation of the FrontPage Server Extensions.	INTERACTIVE NETWORK	Read (RX) Read (RX)
\Program Files\Common Files \Microsoft Shared\Web Server Extensions \40\isapi\	INTERACTIVE NETWORK	Read (RX)(RX) Read (RX)(RX)
\Program Files\Common Files \Microsoft Shared\Web Server Extensions \40\isapi\shtml.dll	INTERACTIVE NETWORK	Read (RX) Read (RX)
\Program Files\Common Files \Microsoft Shared\Web Server Extensions \40\isapi_vti_adm\	INTERACTIVE NETWORK	Read (RX)(RX) Read (RX)(RX)
\Program Files\Common Files \Microsoft Shared\Web Server Extensions \40\isapi_vti_adm \admin.dll	INTERACTIVE NETWORK	Read (RX) Read (RX)
\Program Files\Common Files \Microsoft Shared\Web Server Extensions \40\isapi_vti_aut\	INTERACTIVE NETWORK	Read (RX)(RX) Read (RX)(RX)
\Program Files\Common Files \Microsoft Shared\Web Server Extensions \40\isapi_vti_aut \author.dll	INTERACTIVE NETWORK	Read (RX) Read (RX)

Web content area

This listing is for a Web content area of \inetpub\wwwroot.

Web content area	User	Permissions
\inetpub Note that all directories enclosing the content root will grant LIST permissions to these accounts.	INTERACTIVE NETWORK	List (RX)(Not Specified) List (RX)(Not Specified)
\inetpub\wwwroot_vti_pvt	INTERACTIVE NETWORK	Special Access (RWXD)(RWD) Special Access (RWXD)(RWD)
\inetpub\wwwroot_vti_pvt\botinfs.cnf	INTERACTIVE NETWORK	Special Access (R) Special Access (R)
\inetpub\wwwroot_vti_pvt\bots.cnf	INTERACTIVE NETWORK	Special Access (R) Special Access (R)
\inetpub\wwwroot_vti_pvt\services.cnf	INTERACTIVE NETWORK	Special Access (R) Special Access (R)

Visual SourceSafe files

This listing is for files used for source control.

SourceSafe file	User	Permissions
\VSS\win32\ssapi.dll Note that this file's security settings are modified only if Visual SourceSafe 5.0 or later is installed.	INTERACTIVE NETWORK	(RX) (RX)
\VSS\win32\ssxx.dll Note that this file's security settings are modified only if Visual SourceSafe 5.0 or later is installed. The *xx* value is the country code, and ssus.dll is the default if no other country code is present.	INTERACTIVE NETWORK	(RX) (RX)

Additional file permissions with Microsoft FrontPage

The following is a list of additional file permissions assigned when the Microsoft FrontPage® 2002 client is installed. This list assumes that the built-in Microsoft Windows NT® groups "Administrators" and "SYSTEM" already have full control over the entire drive, and that the IUSR_<hostname> account is granted READ access to the Web content before FrontPage is installed.

FrontPage will assume any account with READ access to the Web content will need continued access after installation. Such accounts will become end users of the Web content. IUSR_<hostname> is only granted access in the list below if it had access to the files at installation time. You can substitute "all user accounts with read access to the Web content" in place of IUSR_<hostname>. Regardless of what level access these accounts were assigned prior to installation, they will be normalized to the access levels described below by FrontPage Server Extensions Setup.

FrontPage will assign "Administrators" and "SYSTEM" full control everywhere.

The installing account is explicitly given Admin rights throughout the content area even though the installer is already an admin. You must be a Windows NT Administrator to successfully run FrontPage Server Extensions administration tools such as Fpsrvadm.

Web Content Area	User	Permissions
\inetpub\wwwroot\	IUSR_<host_name> The Installing Account or the Windows Administrators group	Special Access (RWXD) (RWD) Special Access (RWXD) (RWD)
All Browseable Content	IUSR_<host_name> The Installing Account or the Windows Administrators group	Special Access (RX)(R) Special Access (RWPXD) (RWPD)
\inetpub\wwwroot_vti_log\	IUSR_<host_name> The Installing Account or the Windows Administrators group	Special Access (RWXD) (RWD) Special Access (RWPXD) (RWPD)
\inetpub\wwwroot_vti_pvt\	The Installing Account or the Windows Administrators group INTERACTIVE NETWORK	Special Access (RWPXD) (RWPD) Special Access (RWD) Special Access (RWD)

\inetpub\wwwroot_vti_pvt\access.cnf	IUSR_<host_name> The Installing Account or the Windows Administrators group	Special Access (RWD) Special Access (RWD)
\inetpub\wwwroot_vti_pvt\doctodep.btr	IUSR_<host_name> The Installing Account or the Windows Administrators group	Special Access (RWD) Special Access (RWD)

\inetpub\wwwroot_vti_pvt\deptodoc.btr	IUSR_<host_name> The Installing Account or the Windows Administrators group	Special Access (RWD) Special Access (RWD)
\inetpub\wwwroot_vti_pvt\linkinfo.cnf	IUSR_<host_name> The Installing Account or the Windows Administrators group INTERACTIVE NETWORK	Special Access (RWD) Special Access (RWD) Special Access (RWD) Special Access (RWD)
\inetpub\wwwroot_vti_pvt\service.cnf	IUSR_<host_name> The Installing Account or the Windows Administrators group INTERACTIVE NETWORK	Special Access (RWD) Special Access (RWD) Special Access (RWD) Special Access (RWD)
\inetpub\wwwroot_vti_pvt\structure.cnf	IUSR_<host_name> The Installing Account or the Windows Administrators group INTERACTIVE NETWORK	Special Access (RWD) Special Access (RWD) Special Access (R) Special Access (R)

\inetpub\wwwroot_vti_pvt\svcacl.cnf	IUSR_<host_name> The Installing Account or the Windows Administrators group INTERACTIVE NETWORK	Special Access (RWD) Special Access (RWD) Special Access (RWD) Special Access (RWD)
\inetpub\wwwroot_vti_pvt\uniqperm.cnf	IUSR_<host_name> The Installing Account INTERACTIVE NETWORK	Special Access (RWD) Special Access (RWD) Special Access (RWD) Special Access (RWD)
\inetpub\wwwroot_vti_txt\	IUSR_<host_name> The Installing Account INTERACTIVE NETWORK	Special Access (RWXD) (RWD) Special Access (RWXD) (RWD) Special Access (RWXD) (Not Specified) Special Access (RWXD) (Not Specified)
\inetpub\wwwroot_vti_bin\	IUSR_<host_name> The Installing Account or the Windows Administrators group INTERACTIVE NETWORK	Read (RX)(RX) Read (RX)(RX) List (RX) (Not Specified) List (RX) (Not Specified)
\inetpub\wwwroot_vti_cnf\	IUSR_<host_name> The Installing Account	Special Access (RX) (R) Special Access (RWPXD) (RWPD)
\inetpub\wwwroot_private\	IUSR_<host_name> The Installing Account or the Windows Administrators group	Special Access (RX) (R) Special Access (RWPXD) (RWPD)

Files and Permissions on Apache

This appendix lists the permissions settings for the directories and files that contain the Microsoft FrontPage Server Extensions 2002. The initial installation of the FrontPage Server Extensions will install the files and directories listed in this appendix. As virtual servers are added, the appropriate files will be copied from these directories. This appendix reflects the file and folder permissions that are set by the fp_install.sh script.

In the /usr/local/frontpage/version5.0/ folder:

Permissions	User and Group owner	File or subdirectory	Description
-r — r — r—	bin bin	LICENSE.TXT	Software license agreement
dr-xr-xr-r	bin bin	admin	HTML Administration forms directory
drwxr-xr-x	bin bin	apache-fp	FrontPage Apache patch directory
dr-xr-xr-x	bin bin	bin	Administration tools directory
dr-xr-xr-x	bin bin	exes	Server extensions CGI implementation directory
-rw-r — r—	root other	filelist.txt	List of files
-rwxrw-r—	root other	fp_chown.sh	Server extensions script
-r-xr-xr-x	bin bin	fp_install.sh	Server extensions installation script
-rw-rw-r—	bin bin	frontpage.cnf	Server extensions configuration file
-rw-rw-r—	bin bin	frontpage.cnf.orig	Server extensions configuration file
dr-xr-xr-x	bin bin	nls	Natural Language Support files directory
-r — r — r—	bin bin	readme.htm	Server extensions informational file
-r-xr-xr-x	bin bin	set_default_perms.sh	Resets permissions on server extensions files
dr-xr-xr-x	bin bin	_vti_bin	Server Extensions CGI directory

In the /usr/local/frontpage/version5.0/admin folder:

Permissions	User and Group owner	File or subdirectory	Description

dr-xr-xr-x	bin bin	1033	Server Extensions administration pages folder

In the /usr/local/frontpage/version5.0/admin/1033 folder:

Permissions	User and Group owner	File or subdirectory	Description
-r — r — r—	bin bin	*.htm	Server Extensions administration pages
-r — r — r—	bin bin	*.css	Style sheet for administration pages

In the /usr/local/frontpage/version5.0/apache-fp folder:

Permissions	User and Group owner	File or subdirectory	Description
drwxr-xr-x	bin bin	_vti_bin	FrontPage Apache patch stub program directory
-rw-r — r—	bin bin	fp-patch-apache_1.3.12	FrontPage Apache patch source code
-rw-r — r—	root bin	fpexe.c	fpexe program source code
-rwxr-xr-x	root bin	httpd	Prebuilt patched Apache daemons
-rw — — — -	root bin	suidkey	Custom key file created by change_server.sh

In the /usr/local/frontpage/version5.0/apache-fp/_vti_bin folder:

Permissions	User and Group owner	File or subdirectory	Description
-rwsr-xr-x	root bin	fpexe	FrontPage Apache patch stub program

In the /usr/local/frontpage/version5.0/bin folder:

Permissions	User and Group owner	File or subdirectory	Description
-r — r — r—	bin bin	_vti_inf.html	FrontPage information file
-r-xr-xr-x	bin bin	owsadm.exe	FrontPage Server Extensions administration utility
-r — r — r—	bin bin	postinfo.html	WebPost information file

In the /usr/local/frontpage/version5.0/exes folder:

Permissions	User and Group owner	File or subdirectory	Description
dr-xr-xr-x	bin bin	_vti_bin	FrontPage Server Extensions program directory

In the /usr/local/frontpage/version5.0/exes/_vti_bin folder:

Permissions	User and Group owner	File or subdirectory	Description
dr-xr-xr-x	bin bin	_vti_adm	Administration program directory
dr-xr-xr-x	bin bin	_vti_aut	Authoring program directory
-r-xr-xr-x	bin bin	fpcount.exe	Hit counter component runtime
-r-xr-xr-x	bin bin	shtml.exe	Server extensions browse-time program

In the /usr/local/frontpage/version5.0/exes/_vti_bin/_vti_adm folder:

Permissions	User and Group owner	File or subdirectory	Description
-r-xr-xr-x	bin bin	admin.exe	Server extensions administration program

-r-xr-xr-x	bin bin	fpadmcgi.exe	Server extensions administration program
dr-xr-xr-x	bin bin	Images	Administration pages images folder

In the /usr/local/frontpage/version5.0/exes/_vti_bin/_vti_adm/images folder:

Permissions	User and Group owner	File or subdirectory	Description
-r — r — r—	bin bin	*.gif	Administration pages images

In the /usr/local/frontpage/version5.0/exes/_vti_bin/_vti_aut folder:

Permissions	User and Group owner	File or subdirectory	Description
-r-xr-xr-x	bin bin	author.exe	Server extensions authoring program

In the /usr/local/frontpage/version5.0/nls folder:

Permissions	User and Group owner	File or subdirectory	Description
-r — r — r—	bin bin	*.nls	Code page description files

Content area file and directory permissions

The file list below shows the locations and permissions for the FrontPage Server Extensions 2002 on a UNIX server when fpsrvadm.exe is run with the recommended umask of 022 by the administrator "kelly", a member of group "Volcano". The permissions do not allow group write access. They are 644 (rw-r — r —) for files and 755 (rwxr-xr-x) for directories. To generate any other set of permissions, set your umask accordingly before running fpsrvadm.exe.

Permissions	Administrator	Group	Location

drwx — —	Kelly	Volcano	_private
drwxrwxr-x	Kelly	Volcano	_vti_bin
drwxrwxr-x	Kelly	Volcano	_vti_cnf
drwxrwxr-x	Kelly	Volcano	_vti_log
drwxrwxr-x	Kelly	Volcano	_vti_pvt
drwxrwxr-x	Kelly	Volcano	_vti_txt
drwxrwxr-x	Kelly	Volcano	images
-rw-r — r—	Kelly	Volcano	postinfo.html

In the _vti_bin folder:

Permissions	Administrator	Group	Location
drwxrwxr-x	Kelly	Volcano	_vti_adm
drwxrwxr-x	Kelly	Volcano	_vti_aut

In the _vti_pvt folder:

Permissions	Administrator	Group	Location
-rw-r — r—	Kelly	Volcano	.roles
-rw-rw-r—	Kelly	Volcano	access.cnf
-rw-rw-r—	Kelly	Volcano	botinfs.cnf
-rw-rw-r—	Kelly	Volcano	bots.cnf
-rw-rw-r—	Kelly	Volcano	deptodoc.btr
-rw-rw-r—	Kelly	Volcano	doctodep.btr
-rw-rw-rw-	Kelly	Volcano	frontpg.lck
-rw-rw-r—	Kelly	Volcano	service.cnf
-rw-rw-r—	Kelly	Volcano	service.grp
-rw-rw-r—	Kelly	Volcano	service.lck
-rw-rw-r—	Kelly	Volcano	service.pwd
-rw-rw-r—	Kelly	Volcano	services.cnf

-rw-rw-r—	Kelly	Volcano	svcacl.cnf
-rw-rw-r—	Kelly	Volcano	writeto.cnf

For maximum security, files created in Microsoft FrontPage® are created with a umask of 022. To allow files to be created with a custom umask, edit and recompile the stub executables, shtml.exe, author.exe, admin.exe, and fpcount.exe. The source for these files is in /usr/local/frontpage/version5.0/srcs/fpsuid.c. To create a custom umask:

1. Edit fpsuid.c, changing the umask as desired.
2. Compile the file once for each of the stub executables.
3. Replace the stub executables with the new ones.

Special Directories and Storage Locations

This appendix lists folders and files that are added to each root web or subweb when a SharePoint™ team Web site or a Microsoft FrontPage®-extended Web site is created.

SharePoint team Web site files and folders

Folder	Files	Purpose
	Default.htm	Home page for the SharePoint team Web site.
	Postinfo.html	Allows users to post files to a team Web site.
	_vti_inf.html	Contains information for communication between client applications and the server running SharePoint Team Services.
_layouts	*.htm	Page layouts for the team Web site.
_layouts	Search.htx, search.idq, query.iqy	Query files to support search on the team Web site.
_layouts_vti_cnf	*.htm	Metadata for the *.htm files in the parent folder.
_layouts_vti_cnf	Search.htx, search.idq, query.iqy	Metadata for the query files in the parent folder.
_layouts\images	*.gif	Images used in page layouts.
_layouts\images_vti_cnf	*.gif	Metadata for the images in the parent folder.

_layouts\styles	*css	Style sheets for the team Web site.
_layouts\styles_vti_cnf	*css	Metadata for the style sheets in the parent folder.
_vti_cnf	Default.htm	Metadata about the home page for the team Web site.
_vti_pvt	*.btr	Dependency database for the Web site.
_vti_pvt	*.cnf	Files that store metadata about the team Web site structure.
_vti_pvt	*.lck	Lock files to keep web resources from being accessed simultaneously.
_vti_script	Search.idq	Query file to support search on the team Web site.
Lists\Announcements		Pages and forms used to create the Announcements list.
Lists\Announcements _vti_cnf		Metadata for the *.htm files in the parent folder.
Lists\Contacts		Pages and forms used to create the Contacts list.
Lists\Contacts _vti_cnf		Metadata for the *.htm files in the parent folder.
Lists\Events		Pages and forms used to create the Events list.
Lists\Events_vti_cnf		Metadata for the *.htm files in the parent folder.
Lists\General Discussion		Pages and forms used to create the General Discussion list.
Lists\General Discussion _vti_cnf		Metadata for the *.htm files in the parent folder.
Lists\Links		Pages and forms used to create the Links list.
Lists\Links_vti_cnf		Metadata for the *.htm files in the parent folder.
Lists\Tasks	*.htm	Pages and forms used to create the Tasks list.
Lists\Tasks	Task.ics	A template that is used to export tasks from a SharePoint team Web site to a SharePoint Team Services–compatible calendar/address book, such as Microsoft Outlook® 2002.
Lists\Tasks_vti_cnf		Metadata for the *.htm files in the parent folder.

Lists\Tasks_vti_cnf	Task.ics	Metadata for the Task.ics file in the parent folder.
Shared Documents _vti_cnf	Fp_folder_info	Metadata for the Shared Documents folder. Contains information such as whether the folder contains a list, whether anonymous users can upload documents, and so on.
Shared Documents \Forms	*.htm	Pages and forms used to create the Shared Documents list.
Shared Documents \Forms_vti_cnf	*.htm	Metadata for the *.htm files in the parent folder.

FrontPage-extended Web site files and folders

Folder	Files	Purpose
	Postinfo.html	Allows users to post files to a team Web site.
	_vti_inf.html	Contains information for communication between client applications and the server running SharePoint Team Services.
_borders	Top.htm, Left.htm, Right.htm, Bottom.htm	Shared border definition files that are included in pages that use shared borders.
_derived	*.htx, *.gif	Files that are dynamically generated by FrontPage and that can be regenerated at any time, such as *.htx files created by the Search component and composite text on GIF images.
_fpclass	*.class	Compiled Java class files that implement some FrontPage-based features.
_overlay	*.gif	Overlay files for text on GIF and navigation buttons. These files are transparent GIFs that are overlaid on top of button or banner images to build a complete text on GIF or navigation button composite image.

_private		This folder is created in each FrontPage-extended Web site. It is intended to contain user files that are not viewable by a site visitor.

_themes	*.gif, *.inf, *.css, *.utf8, *.jpg files for each theme	Themes storage. Each theme that is applied to a Web site is uploaded into a uniquely named subfolder of the _themes directory.
_vti_bot		Used for custom FrontPage-based components (WebBots) created using the FrontPage SDK. This folder contains the implementations of custom components. Custom components installed in these directories are optionally downloaded to the author's machine when that component is inserted by the author.
_vti_cnf		For each HTML page and graphics file in a FrontPage-extended Web site, there is a configuration file of the same name. Each configuration page contains a set of name-value pairs, identifying such things as the last author to edit a page or the editor associated with a file.
_vti_log		Contains log files with information about the FrontPage-extended Web site.
_vti_map		Contains image map files.
_vti_pvt		Contains settings for the FrontPage-extended Web site
	_x_todo.htm	Contains the current settings of the To Do list.
	_x_todoh.htm	Contains the changes to the To Do list.
	Access.cnf	Contains HTTP server-specific access control information.

	Bots.cnf	Used for custom FrontPage-based components (WebBots) created using the FrontPage SDK.
	Botinfs.cnf	Custom FrontPage-based components (WebBots) information listing.
	Deptodoc.btr	Dependency database for the web.
	Doctodep.btr	Dependency database for the web.

	Frontpg.lck	Root web only. Lock file to keep Web server resources from being accessed simultaneously. Should be a zero-byte file at idle time. With IIS servers, this file is located in C:\Program Files\Microsoft FrontPage\50\Temp, not in the Web site content area's _vti_pvt directory.
	Linkinfo.cnf	Back links for URLs not in the current web.
	Service.cnf	Contains meta-information about the web.
	Service.pwd	Contains the encrypted password files. Apache only.
	Services.cnf	Contains the list of subwebs.
	Services.org	File used as the basis for Services.cnf when reinstalling the server extensions, so that subwebs are not created as subdirectories of the root web. (This file only exists after an uninstall.)
	Structure.cnf	Contains FrontPage-extended web navigational information storage.
	Svcacl.cnf	File used to store whether subwebs have unique permissions settings and any IP address restrictions, so that when reinstalling the server extensions the correct permissions settings are reapplied to any subwebs.

	Uniqueperm.cnf	The presence of this file indicates that the subweb has unique permissions settings, as opposed to permissions that it inherited from the root web. Present only in subwebs. Used on IIS only.
	Writeto.cnf	Back links for files that can be written to by users of the web, such as Save Results form handler result files. Files that can be written to by users of the web have a looser security setting than regular web content.
_vti_script	*.idq	Query files created by the FrontPage Search component when using Index Server on IIS.
_vti_txt		This folder contains text indexes for the WAIS search engine only. It is not used by Index Server on IIS.
fpdb		When FrontPage creates a Microsoft Access database as part of the Save to Database form handler, it stores it in this folder. Also, the sample database included in FrontPage is stored in this folder.
images		Contains image files used in the FrontPage-extended Web site.

This is a preliminary document and may be changed substantially prior to final commercial release. This document is provided for informational purposes only and Microsoft makes no warranties, either express or implied, in this document. Information in this document, including URL and other Internet Web site references, is subject to change without notice. The entire risk of the use or the results of the use of this document remains with the user. Unless otherwise noted, the example companies, organizations, products, people and events depicted herein are fictitious and no association with any real company, organization, product, person or event is intended or should be inferred. 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.

Unpublished work. © 2000 Microsoft Corporation. All rights reserved.

Microsoft, MS-DOS, Windows, Windows NT, Excel, FrontPage, Internet Explorer, Office, and Outlook 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.