

Migrating Apache Web Sites to IIS 6.0



Migration is the process of installing a Microsoft® Windows® Server 2003 operating system and Internet Information Services (IIS) 6.0 on a server and then moving or reinstalling existing Apache Web sites and applications to that server. Migrating Apache Web sites and applications to IIS 6.0 allows you to take advantage of the improved security, performance, and management provided by Windows Server 2003 and IIS 6.0. The migration process provides minimal outage of service to the users accessing the Web sites and applications, while ensuring that the existing configuration and content of the Web sites and applications are maintained.

In This Chapter

Overview of Migrating Apache Web Sites to IIS 6.0	254
Preparing for Migration	258
Deploying the Target Server	267
Performing the Migration	270
Migrating Apache-Specific Extensions	278
Configuring IIS 6.0.....	286
Enabling Client Access After Migration.....	292
Additional Resources	293

Related Information

- For information about upgrading IIS Web servers, see “Upgrading an IIS Server to IIS 6.0” in this book.
- For information about migrating Web sites on earlier versions of IIS to IIS 6.0, see “Migrating IIS Web Sites to IIS 6.0” in this book.

Overview of Migrating Apache Web Sites to IIS 6.0

The Apache Web site migration process is completed with the *Apache to IIS 6.0 Migration Tool*. This tool is a set of configurable modules to migrate Apache Web sites — including FrontPage® Server Extensions from Microsoft, .htaccess files, and user directories — to servers running IIS 6.0 on Microsoft Windows® Server 2003, Standard Edition; Windows® Server 2003, Enterprise Edition; Windows® Server 2003, Datacenter Edition; and Windows® Server 2003, Web Edition. The tool is a Perl-based command-line utility that migrates both Web content and configuration settings.

The Apache to IIS 6.0 Migration Tool supports Apache versions 1.3.0–1.3.22, as well as the following Linux distributions:

- Red Hat software (Linux versions 6.0, 6.2, 7.0, 7.1, and 7.2)
- Mandrake Linux (versions 8.0, 8.1, and 8.2)
- SuSE (versions 7.3 and 8.0)

You can use the Apache to IIS 6.0 Migration Tool to migrate site content and configuration settings from a *source server* (a server running a supported version of Linux and Apache) to a *target server* (a server running a Windows Server 2003 operating system and IIS 6.0). You can perform the migration from the source server, the target server, or on an *intermediate computer*. The intermediate computer must have File Transfer Protocol (FTP) access to the source and target servers.

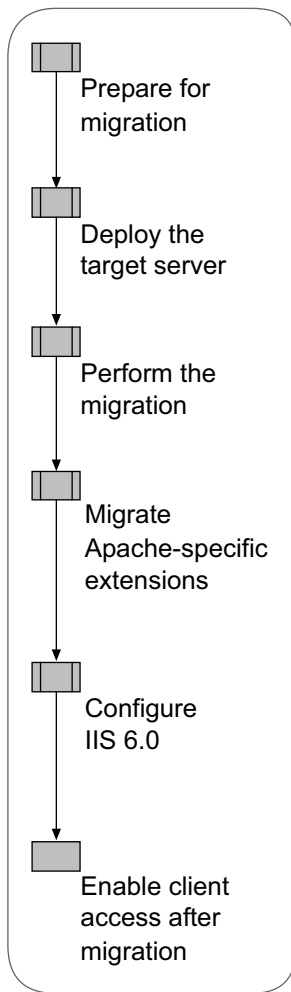
The Apache to IIS 6.0 Migration Tool migrates only the Apache Web site content and configuration. If the Apache Web site is made up of only static content, such as Hypertext Markup Language (HTML) pages, the migration can most likely be completed with a few simple steps. However, Apache Web sites containing dynamic content, such as PHP: Hypertext Preprocessor (PHP) or Common Gateway Interface (CGI) scripts, might need modifications to the code in the dynamic content after migration. In addition, any existing provisioning scripts or setup programs for the existing Apache Web sites typically require modifications after the migration process is complete. Because each organization has different provisioning scripts and setup programs, you need to review and make modifications based on input from your developers, and then test the Web site after completing the modifications. For more information about possible necessary modifications to dynamic content, see “Determining Web Site Compatibility with IIS 6.0” later in this chapter.

For more information about the Apache to IIS 6.0 Migration Tool, including how to install it, see “Preparing for Migration” later in this chapter. For more information about the additional migration tasks that you must perform after running the Apache to IIS 6.0 Migration Tool, see “Migrating Apache-Specific Extensions” later in this chapter.

Process for Migrating Apache Web Sites to IIS 6.0

In addition to migrating the existing Web site content, the migration process automatically migrates the Web site configuration including FrontPage Server Extensions. Before performing the migration of the Web site, you need to evaluate the compatibility of the Apache Web sites with the software installed on your existing server, including dynamic content, database connections, and *external modules*. External modules are software components that are not shipped as a part of the Linux operating system.

If your Apache Web sites contain dynamic content or you want to further secure the Web sites, you can use the Apache to IIS 6.0 Migration Tool to customize the configuration of IIS 6.0 and migrate the dynamic content. Figure 7.1 illustrates the process for migrating existing Apache Web sites to IIS 6.0.

Figure 7.1 Migrating Apache Web Sites to IIS 6.0**Tip**

To upgrade a Web farm, use the process described in this chapter to upgrade each server in the Web farm.

Depending on your familiarity with Linux, Windows Server 2003, Apache, IIS 6.0, and the migration process, you might require less guidance through the migration process. The following quick-start guide provides a detailed overview of the Apache to IIS migration process described in this chapter. You can use this guide to help identify the steps of the migration process where you need additional information and skip the information with which you are already familiar. In addition, all of the procedures required to complete the migration process are documented in “IIS Deployment Procedures” in this book.

Prepare for Migration

1. Determine Web site compatibility with IIS 6.0.
2. Determine application compatibility with worker process isolation mode by evaluating the differences between application isolation modes.
3. Identify the role of the Apache to IIS 6.0 Migration Tool in the migration process.
4. Select one of the following Apache to IIS 6.0 Migration Tool installation options:
 - The source server installation option
 - The target server installation option
 - The intermediate computer installation option

Deploy the Target Server

1. Install Windows Server 2003.
2. Install IIS 6.0.
3. Configure the FTP service.

Perform the Migration

1. Install the Apache to IIS 6.0 Migration Tool:
 - Install the migration tool on computers running Linux.
 - Install the migration tool on computers running Windows.
 - Configure the target server for migration.
2. Verify that clients are not accessing the Web sites.
3. Migrate the Web site content.
4. Migrate the Web site configuration.
5. Recover from an interruption in the migration process:
 - Determine the cause of and resolve any errors.
 - Restart the migration tool in recovery mode.

Migrate Apache-Specific Extensions

1. Migrate dynamic content.
2. Migrate database content and connectivity:
 - Migrate the database content.
 - Migrate the database connectivity.
3. Migrate external modules.

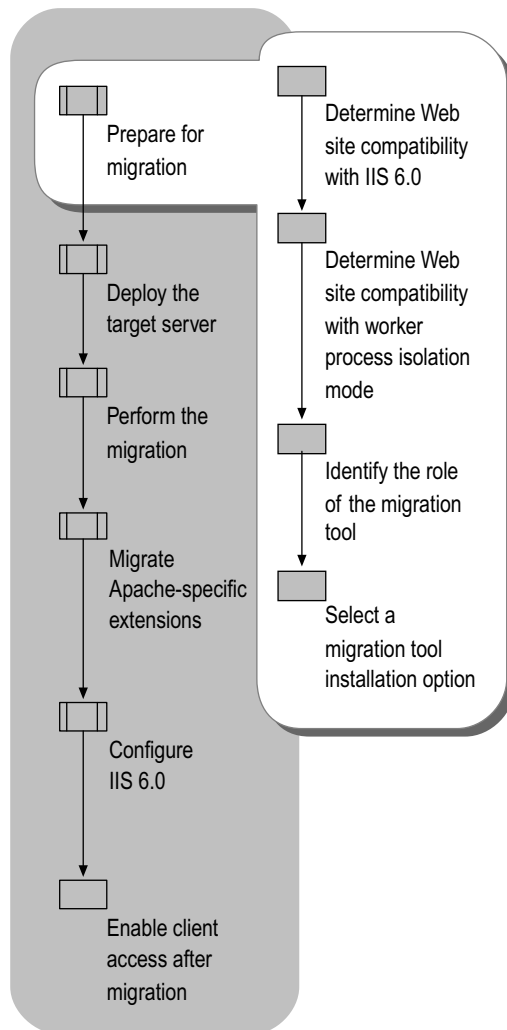
Configure IIS 6.0 After Migration

1. Configure Web service extensions.
2. Configure Multipurpose Internet Mail Extensions (MIME) types.
3. Configure Web site properties.
4. Configure server certificates for Secure Sockets Layer (SSL).
5. Back up the target server.

Enable Client Access After Migration

Preparing for Migration

Before initiating the migration, ensure that existing Apache Web sites are compatible with IIS 6.0 and Windows Server 2003. In addition, determine if the existing Web sites are compatible with worker process isolation mode in IIS 6.0. Then, you must identify which steps in the migration process are automated by the Apache to IIS 6.0 Migration Tool and which steps you need to complete after running the tool. Finally, you must decide which option to use to install the tool.

Figure 7.2 Preparing for Migration of Apache Web Sites to IIS 6.0

Determining Web Site Compatibility with IIS 6.0

Before you migrate to IIS, you must determine the compatibility of the Apache Web site with IIS 6.0. Because of architectural differences, Web sites hosted on Apache have different dependencies than Web sites hosted on IIS. For example, some PHP applications running on Apache use MySQL to connect to databases, whereas IIS Web sites use Microsoft ActiveX® Data Objects (ADO) or Open Database Connectivity (ODBC) to connect to databases. After you migrate an Apache Web site that uses MySQL for database connectivity, you must modify the Web site content to use ADO database connectivity.

Table 7.1 lists the Apache Web site components that you must consider before migrating the Apache Web site to IIS 6.0, along with brief descriptions of the compatibility issues, if any, that can result after migration.

Table 7.1 Apache Web Site Component Migration Compatibility Issues

Web Site Component	Description	Compatibility Issues
Static content	Static content contains no executable code. Examples of static content include HTML, .jpg, and .gif files.	<p>Static content should migrate to IIS 6.0 without any modifications. However, IIS 6.0 MIME types might need to be configured to enable static content file types that are not well known.</p> <p>For more information about MIME types, see “Configuring MIME Types” later in this chapter.</p>
Dynamic content	Dynamic content contains executable code. Examples of dynamic content include CGI and PHP files.	<p>To run Apache dynamic content on IIS, you must do one of the following:</p> <ul style="list-style-type: none"> ▪ Run the executable code on an appropriate environment that is compatible with IIS 6.0. ▪ Convert the executable code to an environment that is supported by IIS 6.0. <p>For more information, see “Migrating Apache-Specific Extensions” later in this chapter.</p>
Database connectivity	Database connectivity provides a standardized method of connecting to various database systems. MySQL is the most common database connectivity standard used in Apache Web sites.	<p>To provide database connectivity on IIS 6.0, you must do one of the following:</p> <ul style="list-style-type: none"> ▪ Obtain a version of MySQL that is compatible with IIS 6.0. ▪ Convert the executable code so that it uses ADO. ADO is the database connectivity standard supported by IIS 6.0. <p>For more information, see “Migrating Apache-Specific Extensions” later in this chapter.</p>
External modules	External modules provide extended features to Apache Web sites. These components include Cold Fusion and Wpoison.	<p>To use the features provided by these external modules, you must do one of the following:</p> <ul style="list-style-type: none"> ▪ Obtain a version of the external module that is compatible with IIS 6.0. ▪ Convert the external module to IIS 6.0. For example, Wpoison is written in CGI and can be converted to run on IIS 6.0. <p>For more information, see “Migrating Apache-Specific Extensions” later in this chapter.</p>

Determining Web Site Compatibility with Worker Process Isolation Mode

IIS 6.0 can run in one of two distinct modes of operation, which are called application isolation modes.

Application isolation is the separation of applications by process boundaries that prevent the applications from affecting one another, and is configured differently for each of the two IIS isolation modes: IIS 5.0 isolation mode and worker process isolation mode.

Worker process isolation mode takes advantage of the redesigned architecture for IIS 6.0. This isolation mode runs all application code in an isolated environment; however, unlike earlier versions of IIS, IIS 6.0 provides isolation without a performance penalty because there are no added process hops. Worker process isolation mode is compatible with most existing Web sites and applications. Whenever possible, run IIS 6.0 in worker process isolation mode to take advantage of the enhanced performance and security in IIS 6.0. When IIS 6.0 is configured to use worker process isolation mode, the metabase property **IIs5IsolationModeEnabled** contains a value of **False**.

IIS 5.0 isolation mode provides compatibility for applications that depend upon the process behavior and memory model of IIS 5.0. Run IIS in this mode only when a Web site or application has compatibility issues when running in worker process isolation mode and you cannot resolve the problem. When IIS 6.0 is configured to use IIS 5.0 isolation mode, the metabase property **IIs5IsolationModeEnabled** contains a value of **True**.



Important

IIS 6.0 cannot run both application isolation modes simultaneously on the same server. Therefore, on a single server running IIS 6.0, you cannot run some Web applications in worker process isolation mode and others in IIS 5.0 isolation mode. If you have applications that require separate modes, you must run them on separate servers.

After you install the target server, IIS is configured to use worker process isolation mode by default. *The only reason for configuring IIS to use IIS 5.0 isolation mode is if there are Apache extensions that run only in IIS 5.0 isolation mode.* For example, if the preprocessor for the dynamic content in an Apache Web site is incompatible with worker process isolation mode, you must configure IIS to use IIS 5.0 isolation mode. Determine Web site compatibility, in particular Apache-specific extensions to the Web sites, with IIS in your lab before migrating your existing Apache Web sites. In most cases, IIS hosts the Web sites in worker process isolation mode without any problems.

For more information about how to migrate Apache-specific extensions and how to determine compatibility of these extensions with worker process isolation mode, see “Migrating Apache Specific Extensions” later in this chapter.

**Note**

Identifying a complete list of potential incompatibilities that Web sites can experience with worker process isolation mode is beyond the scope of this book. Even after following the guidelines in this chapter, you need to verify in your lab whether your Web sites are compatible with worker process isolation mode.

Identifying the Role of the Migration Tool

Before you migrate Apache Web sites to IIS 6.0, you must know which steps of the migration process are automated by the Apache to IIS 6.0 Migration Tool and which steps you must complete manually. This will allow you to have the appropriate tools and resources available when you are ready to start the migration. In addition, you need to determine the configuration and placement of your Web site after running the tool. The automation provided by the Apache to IIS 6.0 Migration Tool is the same for static and dynamic content. However, depending on the content in your Apache Web site, further configuration might be required after running the tool.

The following steps in the Apache Web site migration process are automated by the Apache to IIS 6.0 Migration Tool:

Copying of Web site content. The migration tool creates a copy of the files and folders containing Web site content. The copied files are converted from standard UNIX text file format (end-of-line denoted by carriage return) on the server running Apache to standard Windows text file format (end-of-line denoted by carriage return and line feed characters) on the server running IIS. The Web site content is placed on the server running IIS in a location selected while running the tool.

Translation of Web site configuration. The configuration of the Apache Web server and individual Web sites is translated to the corresponding IIS Web server and Web site configuration settings in the IIS metabase.

Migration of ~user directories. ~user directories are private virtual Web directories that contain individual user Web sites. The tool migrates the contents of these directories, creates corresponding users, and converts permissions found in these virtual Web directories.

Translation of .htaccess permissions. The htaccess file sets permissions for access for the virtual Web directories on an Apache Web server. The permissions in this file are translated by the tool into the corresponding directory permissions in the NTFS file system.

Migration of MIME types and port numbers. MIME types define the types of *static* files that are served by the Web server. The tool automatically determines the MIME types defined on the Web server running Apache and then creates the associations of MIME types to extensions on the Web server running IIS.

Migration of FrontPage Server Extensions. The tool detects the presence of FrontPage 2000 Server Extensions and FrontPage 2002 Server Extensions on the Apache Web server, enumerates the Web sites and subweb sites, enumerates the FrontPage Server Extensions settings, users, and roles, creates the Web sites and subweb sites, and recalculates the links for each Web site and subweb site.

Creation of status file. The tool creates a file that contains the status of the migration steps completed by the tool. The status file can be used for confirming that the migration completed successfully or for troubleshooting any problems encountered while performing the migration with the tool.

The following steps in the Apache Web site migration process need to be completed after running the Apache to IIS 6.0 Migration Tool:

Convert code within dynamic content. Dynamic content, such as PHP or CGI, on Apache needs to be modified to run on IIS 6.0.

Modify database connectivity within dynamic content. For the Web sites and applications that have database connectivity, the content needs to be modified to use database connectivity provided by IIS 6.0 and Windows Server 2003.

Migrate external modules used by Web sites. After migration, you need to migrate external modules that are used by the Web sites and applications.

Add Web service extensions used by the Web sites. Any dynamic content types that are not automatically migrated by the tool must be added as Web service extensions. These new Web service extensions might be necessary for any modifications or conversions of dynamic code.

Configure advanced Web site properties. After tool completes its migration tasks, the Web sites are configured comparably to how they were on Apache. You might want to configure the Web site to take advantages of the advanced security and availability features in IIS 6.0.

Obtain server certificates for SSL. For each Web site on the Apache Web server that uses SSL, you must either renew the existing server certificates in a format that is compatible with IIS 6.0 or you must request new server certificates in a format that is compatible with IIS 6.0. This is because the server certificates on the Apache Web server are in a format that is incompatible with IIS. Once you have the server certificates, you must import the certificates into IIS and configure your Web sites to use the appropriate certificates.

Selecting a Migration Tool Installation Option

The Apache to IIS 6.0 Migration Tool supports four options you can choose from when determining how you want to migrate Apache Web sites to IIS. Your familiarity with the Linux or Windows operating systems and the access that you have to the Web servers involved in the migration will determine the process you should follow to install and then run the tool. In some cases, you might not have unrestricted access to the Web servers because another group manages the Apache servers or the Web servers are behind firewalls that prevent FTP traffic.

To install and run the Apache to IIS 6.0 Migration Tool for any of the installation options, you must be able to:

- Log on to the source server with an account that has root-level or super-user access.
- Log on to the target server with an account that is a member of the local Administrators group.

In addition, when you select the intermediate computer installation option, you must be able to:

- Log on to the intermediate computer running Linux with an account that has root-level or super-user access.
- Log on to the intermediate computer running Windows with an account that is a member of the local Administrators group.

Table 7.2 lists and describes the installation options for the Apache to IIS 6.0 Migration Tool. Read this table to select the installation option that best fits your needs.

Table 7.2 Apache to IIS 6.0 Migration Tool Installation Options

Installation Option	Installation Details
The source server (See Figure 7.3)	<ul style="list-style-type: none"> ▪ Install and run the Apache to IIS 6.0 Migration Tool on your Linux-based source server (you do not need to physically access the Windows-based target server). ▪ Install Perl on both the Linux-based source server and the Windows-based target server. ▪ Enable the FTP service on the Windows-based target server only.
The target server (See Figure 7.4)	<ul style="list-style-type: none"> ▪ Install and run the migration Apache to IIS 6.0 Migration Tool on the Windows-based target server (you do not need to physically access the Linux-based source server). ▪ Install Perl on the Windows-based target server only. ▪ Enable the FTP service on the Linux-based source server only.
An intermediate computer running Microsoft Windows 2000 or later (See Figure 7.5)	<ul style="list-style-type: none"> ▪ Install and run the Apache to IIS 6.0 Migration Tool on a Windows-based intermediate computer (you do not need to physically access either the Linux-based source server or the Windows-based target server). ▪ Install Perl on both the Windows-based intermediate computer and the Windows-based target server. ▪ Enable the FTP service on both the Windows-based target server and the Linux-based source server.
An intermediate computer running a supported version of Linux (See Figure 7.5)	<ul style="list-style-type: none"> ▪ Install and run the Apache to IIS 6.0 Migration Tool on a Linux-based intermediate computer (you do not need to physically access either the Linux-based source server or the Windows-based target server). ▪ Install Perl on both the Linux-based intermediate computer and the Windows-based target server. ▪ Enable the FTP service on both the Windows-based target server and the Linux-based source server.

The Apache to IIS 6.0 Migration Tool must be installed on the source server, target server, or intermediate computer. The high-level process for installing the tool includes the following:

1. Copying the compressed distribution files (.tar for Linux or .zip for Windows) to the appropriate computer based on the installation option selected.
2. Expanding the compressed distribution files.
3. Installing the Perl interpreter, if it is not already installed.

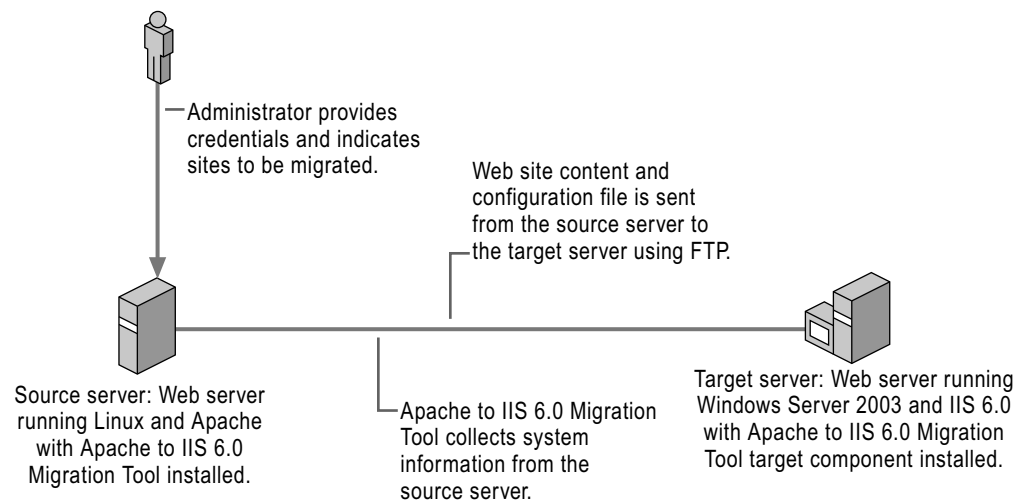
For more detailed information about the installation option that you have selected for the Apache to IIS 6.0 Migration Tool, see “Installing the Migration Tool” later in this chapter.

Source Server Installation Option

When you select the source server installation option, you install and run the Apache to IIS 6.0 Migration Tool on the source server, which runs Apache. In addition, you must install and configure the FTP service on the target server.

Figure 7.3 illustrates the migration process for the source server installation option.

Figure 7.3 Migrating from Apache to IIS 6.0 with the Source Server Installation Option



The process for migrating Apache Web sites when you select the source server installation option is as follows:

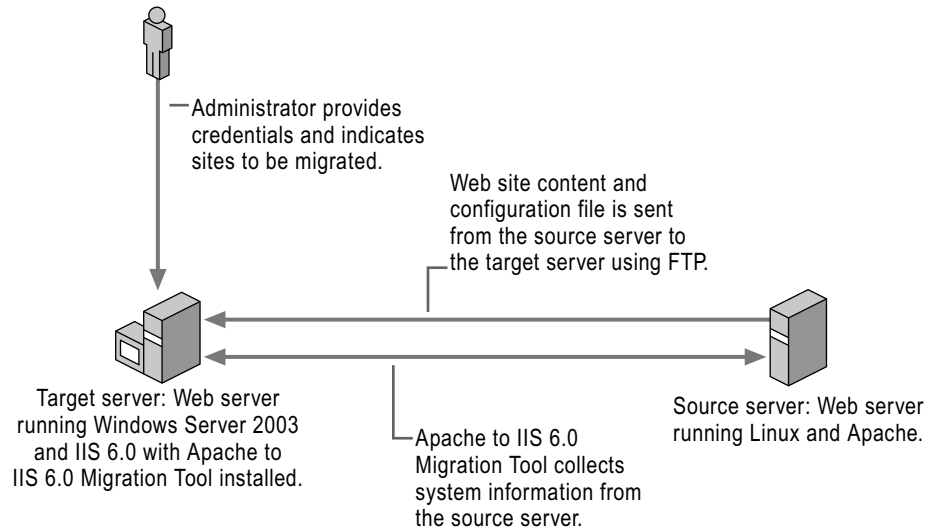
1. The Apache to IIS 6.0 Migration Tool prompts you for the appropriate user credentials for the source server and target servers.
2. The tool then transfers the Apache-based Web site content and the configuration information from the source server to the target server by way of FTP.
3. The tool applies the Web site configuration settings to the IIS Web sites on the target server.

Target Server Installation Option

When you select the target server installation option, you install and run the Apache to IIS 6.0 Migration Tool on the target server, which runs Windows Server 2003 and IIS 6.0. In addition, you must install and configure the FTP service on the source server.

Figure 7.4 illustrates the migration process for the target server installation option .

Figure 7.4 Migrating from Apache to IIS 6.0 with the Target Server Installation Option



The process for migrating Apache Web sites when you select the target server installation option is as follows:

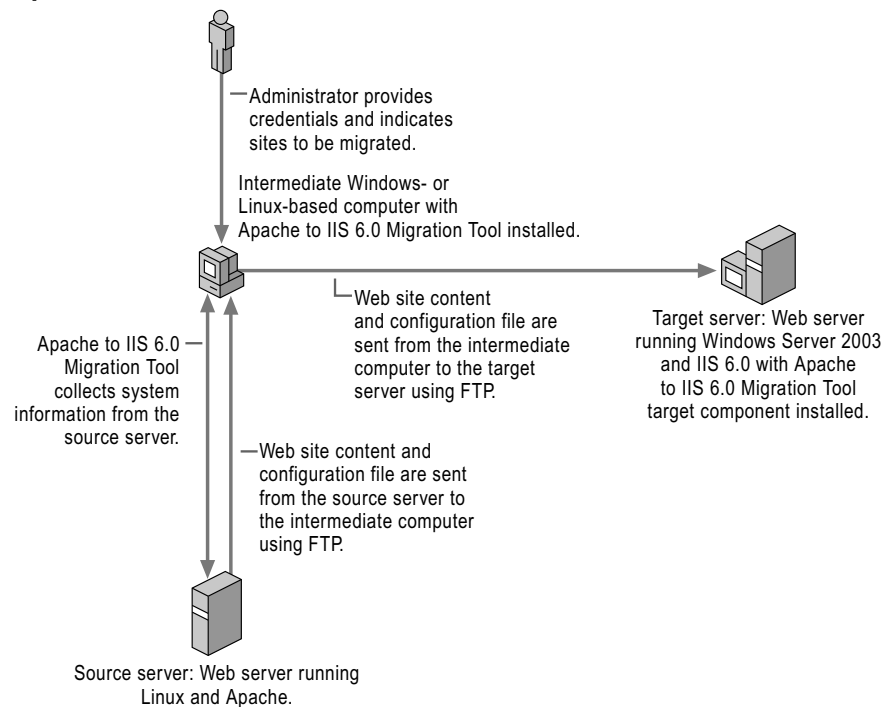
1. The Apache to IIS 6.0 Migration Tool prompts you for the appropriate user credentials for the source and target servers.
2. The tool then transfers the Apache-based Web site content and the configuration information from the source server to the target server by way of FTP.
3. The tool applies the Web site configuration settings to the IIS 6.0 Web sites on the target server.

Intermediate Computer Installation Option

When you select the intermediate computer installation option, you install and run the Apache to IIS 6.0 Migration Tool on the intermediate computer, which runs either Windows or Linux. In addition, you must install and configure the FTP service on the source and target servers.

Figure 7.5 illustrates the migration process for the intermediate computer installation option.

Figure 7.5 Migrating from Apache to IIS 6.0 with the Intermediate Computer Installation Option

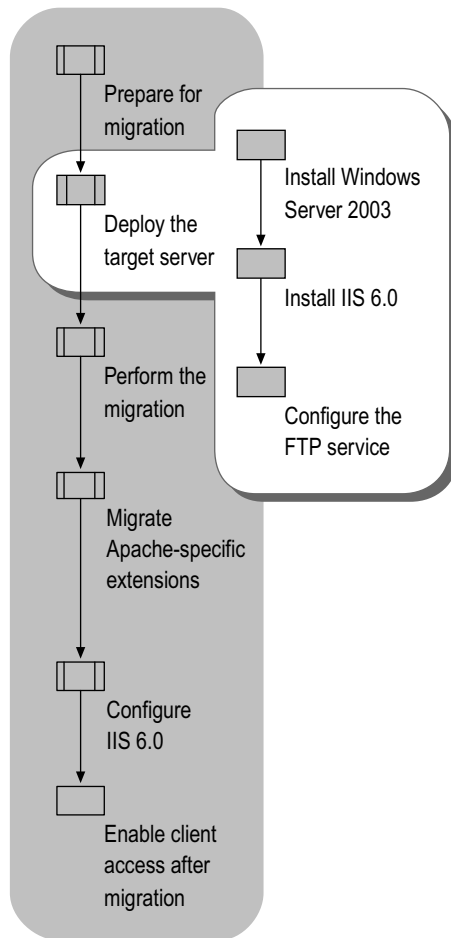


The process for migrating Apache Web sites when you select the intermediate computer installation option is as follows:

1. The Apache to IIS 6.0 Migration Tool prompts you for the appropriate user credentials for the source and target servers.
2. The tool transfers the Apache-based Web site content and the configuration information from the source server to the intermediate computer by way of FTP.
3. The tool applies the site configuration settings to the IIS Web sites on the target server.

Deploying the Target Server

The Web server running IIS 6.0 must be installed and configured before you can migrate the Apache Web sites. In addition to installing Windows Server 2003, you must install and configure IIS 6.0 and, based on the installation option you selected, install and configure the FTP service on the target server. The tool uses FTP to transfer Web site content between the source and target servers.

Figure 7.6 Deploying the Target Server

If the target server is already installed and configured, you can proceed to “Performing the Migration” later in this chapter.

Installing Windows Server 2003

The primary concern when installing Windows Server 2003 is to ensure that the security of the target server is maintained. When you install Windows Server 2003 as a dedicated Web server, the default components and services are configured to provide the lowest possible attack surface. You can further reduce the attack surface of the target server by enabling only the essential Windows Server 2003 components and services.

The migration process presented here assumes that you install Windows Server 2003 with the default options. If you use other methods for installing and configuring Windows Server 2003, such as unattended setup, the default configuration settings might be different.

**Note**

When you complete the installation of Windows Server 2003, Manage Your Server automatically starts. The migration process assumes that you close Manage Your Server, and then further configure the Web server in **Add or Remove Programs** in Control Panel.

For more information about the essential components and services to enable in Windows Server 2003, see “Enabling Only Essential Windows Server 2003 Components and Services” in “Securing Web Sites and Applications” in this book.

Installing IIS 6.0

Because IIS 6.0 is not installed during the default installation of Windows Server 2003, the next step in deploying the target server is to install and configure IIS 6.0. The migration process presented here assumes that you install IIS 6.0 with the default options in **Add or Remove Programs** in Control Panel. If you use other methods for installing and configuring Windows Server 2003, such as Manage Your Server, the default configuration settings might be different.

Install and configure IIS 6.0 by completing the following steps:

1. Install IIS 6.0 with only the essential components and services.

As with installing Windows Server 2003, the primary concern when installing and configuring IIS 6.0 is to ensure that the security of the target server is maintained. Enabling unnecessary components and services increases the attack surface of the target server. You can help ensure the target server is secure by enabling only the necessary components and services in IIS 6.0.

For more information about how to install IIS 6.0, see “Install IIS 6.0” in “IIS Deployment Procedures” in this book. For more information about the necessary IIS 6.0 components and services, see “Enabling Only Essential IIS Components and Services” in “Securing Web Sites and Applications” in this book.

2. In addition to the necessary components and services recommendations in step 1, enable the File Transfer Protocol (FTP) Service.

The FTP service migrates content to the target server when you install and run the Apache to IIS 6.0 Migration Tool on the source server or on an intermediate computer.

For more information about how to enable the FTP service, see “Configure IIS Components and Services” in “IIS Deployment Procedures” in this book.

3. If the source server has Web sites that are FrontPage extended, enable FrontPage 2002 Server Extensions on the target server.

For more information about how to enable FrontPage Server Extensions, see “Configure a Web Site to be FrontPage Extended” in “IIS Deployment Procedures” in this book.

Configuring the FTP Service

The Apache to IIS 6.0 Migration Tool requires the FTP service to migrate Apache Web sites to the target server. The installation option you previously selected will determine on which computers you need to enable the FTP service. Table 7.3 lists the FTP service requirements for each installation option.

Table 7.3 Methods for Installing and Configuring the FTP Service

If You Install and Run the Migration Tool On	Install and Configure FTP Service On
The source server	The target server
The target server.	The source server
An intermediate computer	Both the target and source server

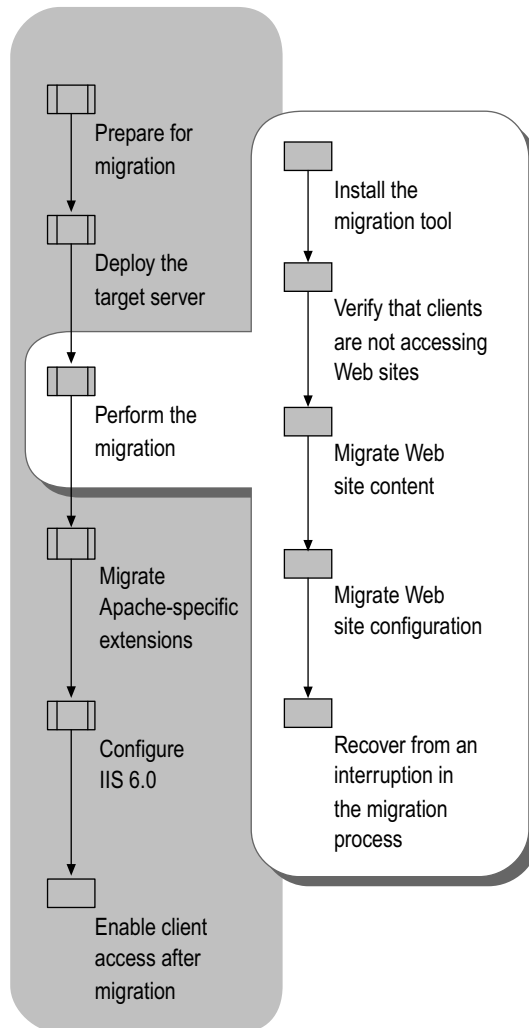
Configure the FTP service on the target server After you install the FTP service on the target server, no additional configuration is required. The Apache to IIS 6.0 Migration Tool works with the FTP service in the default configuration. Later in the deployment process, the tool creates a default FTP site to be used for Web site migration.

Configure the FTP service on the source server Configuring the FTP service on the source server requires installing and configuring the FTP package from the Linux distribution disc. After the FTP service is installed, you must configure it to allow access to the Web sites.

For more information about how to install and configure the FTP service on the source computer, see the FTP documentation that accompanies the version of Linux that is installed on the source server.

Performing the Migration

In the next step of the migration process, the Apache to IIS 6.0 Migration Tool copies the Web site content from the source server to the target server. Then, the tool transfers the Apache Web site settings to the corresponding IIS 6.0 metabase properties.

Figure 7.7 Performing the Migration from Apache to IIS 6.0

Installing the Migration Tool

Now you are ready to install the Apache to IIS 6.0 Migration Tool based on the method you selected earlier in the “Selecting a Migration Tool Installation Option” section of this chapter. You can install the tool on one of the following:

- Source server, which is running Linux
- Target server, which is running Windows Server 2003
- Intermediate computer, which can be running either Linux or Windows

The installation process is dependent on the operating system. So, for example, you can use the same procedure for installing the Apache to IIS 6.0 Migration Tool on the source server or for an intermediate computer running Linux.

Install the Apache to IIS 6.0 Migration Tool by completing the following steps:

1. Install the tool on a computer running Linux. Or, install the tool on a computer running Windows.
2. Configure the target server for migration.

Installing the Migration Tool on Computers Running Linux

The Apache to IIS 6.0 Migration Tool installation process on the source server and on an intermediate computer running Linux is identical. The versions of Linux supported by the tool are listed in “Overview of Migrating Apache Web Sites” earlier in this chapter.

Install the Apache to IIS 6.0 Migration Tool on the computer running Linux by completing the following steps:

1. Install the *Windows Server 2003 Deployment Kit* companion CD or the *Internet Information Services (IIS) 6.0 Resource Kit* companion CD on a computer other than the target server.
2. Install Perl on the computer running Linux.

Because the Apache to IIS 6.0 Migration Tool is written in Perl, you must install a Perl interpreter before you can run the tool. Install Perl on the computer running Linux and verify that the Perl interpreter is configured properly before installing the tool. To obtain a free copy of Perl, see the ActivePerl link on the Web Resources page at <http://www.microsoft.com/windows/reskits/webresources>.

3. Create a folder to contain the Apache to IIS 6.0 Migration Tool on the computer running Linux.
4. Change to the folder created in Step 3 so that the folder is the current working folder.
5. Copy **AIISMT_linux.tar.gz** from the *systemroot*\Program Files\AIISMT folder on the computer in Step 1 to the folder created in Step 3.
6. At a Linux command line type **tar -zxvf AIISMT_linux.tar.gz**.

The compressed files are extracted to the folder created in step 3.

Installing the Migration Tool on Computers Running Windows

The Apache to IIS 6.0 Migration Tool installation process on the target server and on an intermediate computer running Windows is identical. The target server must be running Windows Server 2003. The intermediate computer can be running Microsoft Windows 2000 Professional, Microsoft Windows 2000 Server, Windows 2000 Advanced Server, or Microsoft Windows Server 2003.

Install the Apache to IIS 6.0 Migration Tool on the computer running Windows by completing the following steps:

1. Install the *Windows Server 2003 Deployment Kit* companion CD or the *Internet Information Services (IIS) 6.0 Resource Kit* companion CD on a computer other than the target server.
2. Install Perl on the computer running Windows.

Because the Apache to IIS 6.0 Migration Tool is written in Perl, you must install a Perl interpreter before you can run the tool. Install Perl on the computer running Windows and verify that the Perl interpreter is configured properly before installing the tool. To obtain a free copy of Perl, see the ActivePerl link on the Web Resources page at <http://www.microsoft.com/windows/reskits/webresources>.

3. Copy the **AIISMT_windows.exe** from the *systemroot*\Program Files\AIISMT folder on the computer in step 1 to the desktop of the target server.
4. Double-click **AIISMT_windows.exe**.

The **WinZip Self-Extractor** dialog box appears.

5. In the **Unzip to folder** box, type the path of the folder where you want to extract the tool or click **Browse** to browse to the folder.

6. Click **Unzip**.

The WinZip Self-Extractor extracts the Apache to IIS 6.0 Migration Tool into the folder selected in step 5.

Configuring the Target Server for Migration

You must configure the target server before you can perform the migration process. The configuration process creates virtual directories in the FTP root mapped to the physical drives of the server.

Configure the target server for migration by completing the following steps:

1. At the command line, type **cd *installation_path***, where *installation_path* is the fully qualified path where you installed the Apache to IIS 6.0 Migration Tool, and then press ENTER.

This changes the default folder to the folder where you installed the tool. For more information about how the Apache to IIS 6.0 Migration Tool was installed, see “Installing the Migration Tool on Computers Running Windows” earlier in this chapter.

2. At the command line, type **aiismt_setup**, and then press ENTER.

3. When prompted to create a virtual directory, type **y** to create a virtual directory on the disk volume you want to use for migration.

You must select at least one disk volume and create a virtual directory.



Important

You cannot create a virtual directory on a drive formatted with the FAT file system because access rights to directories on that drive cannot be set.

The `aiismt_setup` program creates the virtual directories in the FTP site root and configures FTP to listen on TCP port 21.

Verifying that Clients Are Not Accessing Web Sites

The Apache to IIS 6.0 Migration Tool requires only read access to the Web site content and configuration settings on the source server; therefore, the source server can remain online in your production environment. However, you might need to remove the source server from your production network and move it to a private network segment that has direct network connectivity to the target server when:

- The amount of files and configuration information being copied across the network can generate a high volume of traffic and slow the production network.
- Firewalls that reside between the source and target servers prevent the migration tool from performing the migration. This is most commonly caused because the migration tool uses FTP for communicating with the source and target servers, and the FTP protocol is blocked by the firewalls.
- Security-related configuration settings on the source server need to be changed to allow the migration tool to work. This can include enabling FTP on production Web servers that are directly connected to the Internet.

Migrating Web Site Content

The migration of your Apache Web site involves migrating both the Web site content and its configuration. Based on the installation option you selected, the Apache to IIS 6.0 Migration Tool prompts you for the information required to perform the migration of the Web site content.

To run the tool, do one of the following, depending on the operating system on the computer:

On Linux-based computers From the Linux shell, change the current folder to the folder where you installed the tool, and then type `perl aiismt_main.pl`.

On Windows-based computers: At the command line, change the current folder to the folder where you installed the tool, and then type `aiismt`.

After the Apache to IIS 6.0 Migration Tool starts, it prompts you for information. Table 7.4 lists the information that the migration tool requires to migrate the Web site content.

Table 7.4 Information Required by Migration Tool During Web Site Content Migration

Information Required	Do the Following
Session name	<p>Type a:</p> <ul style="list-style-type: none"> • Unique session name to indicate that the tool should begin a new migration process. • Previous session name if you want to resume a previous migration. Supplying a previous session name causes the tool to run in recovery mode.
Source server IP address	<p>If you run the tool on the:</p> <ul style="list-style-type: none"> • Target server or on an intermediate computer, type the IP address of the source server. • Source server, press ENTER to see the default IP address displayed. If the tool cannot detect the source server's IP address, it will prompt you to confirm that the source server is running on a supported operating system.
Root user password	<p>If you run the Apache to IIS 6.0 Migration Tool on the:</p> <ul style="list-style-type: none"> • Target server or on an intermediate computer, type the root user password to gain access to the Linux-based source server's system files and FTP service. • Source server, you do not need to type the root user password.
Target server IP address	<p>If you run the Apache to IIS 6.0 Migration Tool on the:</p> <ul style="list-style-type: none"> • Source server or on an intermediate computer, type the IP address of the source server. • Target server, type the IP address of the target server or press ENTER to use the default (LocalHost) IP address.
User name and password for the target server	<p>If you run the Apache to IIS 6.0 Migration Tool on the:</p> <ul style="list-style-type: none"> • Source server or on an intermediate computer, type a user name that is a member of the local Administrator's group on the target server, and the associated password. • Target server, you do not need to type a user name and password.
Configuration file path	<p>Type the complete path to the <i>httpd.conf</i> file, which is a file that contains configuration and settings for Apache on the source server. You can also press ENTER to see the default path (/Etc/Httpd/Conf/Httpd.conf).</p>

(continued)

Table 7.4 Information Required by Migration Tool During Web Site Content Migration (continued)

Information Required	Do the Following
Web site directory on target server	Type the path of the directory on the target server where you want the Web site content to be migrated. You must supply a path for each Web site migrated. When a Web site has SSL certificates associated with it, the Apache to IIS 6.0 Migration Tool prompts you to indicate whether you want the certificates to be migrated as well.
Listening IP address and port number	When you want to change the listening IP address and port number, type the new values. The new IP address and port numbers are applied to all the sites you selected for migration.
Web site IP address	When you want to change the IP address at the site level, type the new IP address, or type "" to use all unassigned IP addresses.

**Note**

If you run the Apache to IIS 6.0 Migration Tool on the source server, the target server credentials are authenticated. If you run the tool on the target server, the source server credentials are authenticated.

For more information about how to migrate Web sites using the Apache to IIS 6.0 Migration Tool, see the “Apache to IIS 6.0 Migration Tool User Guide” on the *Windows Server 2003 Deployment Kit* companion CD or the *Internet Information Services (IIS) 6.0 Resource Kit* companion CD.

If the migration tool is interrupted or fails at any point during the process, you can run it in recovery mode. For more information about the recovery process, see “Recovering from an Interruption in the Migration Process” later in this chapter.

Migrating Web Site Configuration

After migrating the Web site content, you must migrate the Web site configuration. Before you continue in the migration process, check the Apache to IIS 6.0 Migration Tool status file, `Aiismt_Status.html` in the session directory, to ensure the migration has completed successfully up to this point.

To start Web site configuration migration on the target server, at the command line, change the current folder to the folder where you installed the Apache to IIS 6.0 Migration Tool, and then type **aiismt**.

During the migration of the Web site configuration the tool, prompts you to supply the session name. The files with the configuration settings are located in a folder under the FTP root. The folder uses the name of the current migration session, which you specified when you migrated the Web site content in “Migrating Web Site Content” earlier in this chapter.

During the Web site configuration process on the target server, the Apache to IIS 6.0 Migration Tool does the following:

- Migrates Apache Web site directives and maps them to corresponding IIS configuration settings, including security settings.
- Migrates the FrontPage Server Extensions settings, users, and roles if any of the Apache Web sites are FrontPage extended.
- Sets the security settings on the sites by assigning user rights to the directories that contain site-related content.
- Verifies that settings have been correctly applied to the Web sites on the target server when the Web site configuration migration is complete.

Recovering from an Interruption in the Migration Process

If you received no error messages while migrating the Web site content and configuration, you can proceed to one of the following sections in this chapter:

- “Migrating Apache-Specific Extensions” if any of your Web sites contain dynamic content.
- “Configuring IIS 6.0 After Migration” if all of your Web sites contain static content.

If you received an error message while running the Apache to IIS 6.0 Migration Tool, do the following:

1. Determine the cause of the error and resolve the problem.
2. Restart the Apache to IIS 6.0 Migration Tool in recovery mode.

In addition to resolving migration errors, you can use the tool in recovery mode to:

- Perform another migration if the majority of the settings are the same.
For example, you might run the tool to migrate a specific Web site on a Web server that hosts multiple Web sites. You can migrate subsequent Web sites by running the tool in recovery mode and change only the name of the Web site to be migrated. The remainder of the settings remains unchanged.
- Perform the actual migration of Web site content and configuration after a trial migration.
You can perform a *trial migration* of the Web site and content by supplying all information except the Web sites to migrate. You can use this method to validate the majority of the migration settings. You can subsequently run the tool in recovery mode.
- Provide a scripted environment for automating the Apache to IIS 6.0 Migration Tool migration process.

The tool creates an answer file that is used by the recovery process, which you can edit to automate the tool. The answer file has the naming convention of `AIISMT_ session_name_Recovery.txt`, where *session_name* is the session name for the migration.

Determining Cause of and Resolving Errors

The Apache to IIS 6.0 Migration Tool automatically creates a status file for each migration session in the session folder, located in the FTP root on the target server. The name of the session folder is identical to the migration tool session name. The status file, `Aiismt_Status.html`, contains information about every phase of the migration process.

If an interruption occurs while the tool is running:

1. Refer to the status file to determine the cause of the failure.
2. Resolve any errors before proceeding with the upgrade process.

Restarting the Migration Tool in Recovery Mode

If the Apache to IIS 6.0 Migration Tool is interrupted or fails at any point during the process, you can run it again in recovery mode. Recovery mode allows you to pick up the migration process from the point at which the tool was interrupted, rather than starting over from the beginning. Any of the following events can cause a failure or interruption in the migration tool:

- System failure.
- Network connection terminated.
- FTP access denied.

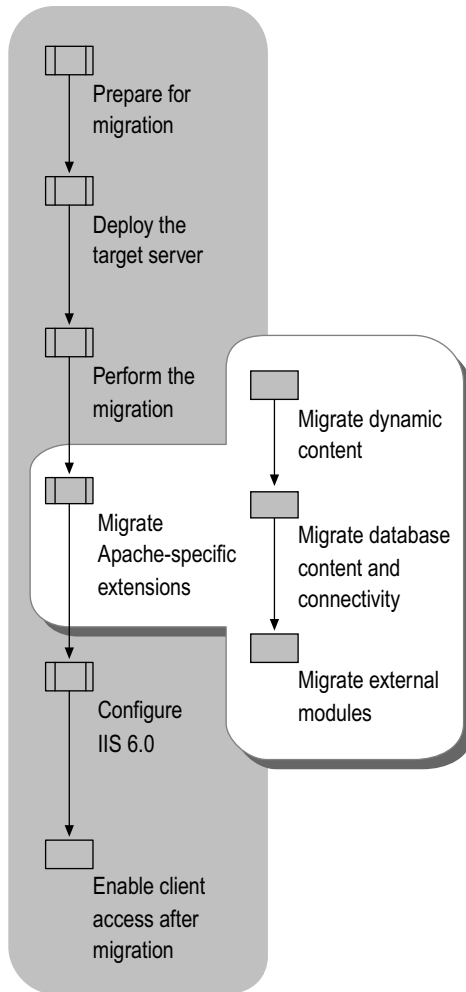
You can restart the migration tool in recovery mode when:

- An interruption occurs during data entry.
If the tool terminates as you are entering information about the migration (such as IP addresses or file paths) you can restart it in recovery mode. The migration tool validates your user credentials, and the migration process proceeds from the point where the tool was interrupted. You do not need to retype the information you have already entered.
- An interruption occurs during file transfer.
If the migration tool terminates during the process of transferring files from the source server to the target server, you can restart it in recovery mode. The tool resumes migration with the file that was being transferred during the interruption.

Migrating Apache-Specific Extensions

If the Apache Web site contains dynamic content, requires database connectivity, or includes external modules, you must separately migrate these Apache-specific extensions to complete the migration process.

Figure 7.8 illustrates the process for migrating Apache-specific extensions to IIS 6.0.

Figure 7.8 Migrating Apache-Specific Extensions to IIS 6.0

The Apache-specific extensions on Linux commonly include MySQL and PHP. The combination of Linux, Apache, MySQL, and PHP are often referred to as LAMP. LAMP is the application platform used for Linux systems. The migration of the Apache Web sites must include the migration of LAMP.

For more information about migrating Apache Web sites to IIS, see the Migration of Linux-Apache-MySQL-PHP Platform to Windows 2000 link on the Web Resources page at <http://www.microsoft.com/windows/reskits/webresources>. Although the white paper refers to IIS 5.0 and Windows 2000 Server, the processes and methods are applicable for IIS 6.0 and Windows Server 2003.



Important

Test the migration of Apache-specific extensions in your lab environment before starting the Web site migration.

The Apache-specific extensions that you might need to migrate include:

- Dynamic content.
- Database content and connectivity.
- External modules.

Migrating Dynamic Content

The Apache to IIS 6.0 Migration Tool transfers the files that make up the dynamic content to the target server during the Web site content migration. However, the dynamic content typically cannot run on IIS and requires further steps. Table 7.5 describes the advantages and disadvantages of the methods for migrating dynamic content.

Table 7.5 Comparison of Methods for Migrating Dynamic Content

Migration Method	Advantages	Disadvantages
Run dynamic content on a preprocessor that ships with IIS and Windows Server 2003.	<ul style="list-style-type: none"> • Requires minimal migration effort, depending on the extent of changes to the dynamic content. • Requires no purchase of additional preprocessor. 	<ul style="list-style-type: none"> • Might not run as efficiently as dynamic content written in ASP or ASP.NET.
Run dynamic content on a preprocessor external to IIS and Windows Sever 2003.	<ul style="list-style-type: none"> • Requires minimal migration effort, depending on the extent of changes to the dynamic content. 	<ul style="list-style-type: none"> • Might not run as efficiently as dynamic content written in ASP or ASP.NET. • Requires purchase of an additional preprocessor.
Rewrite the dynamic content in ASP or ASP.NET.	<ul style="list-style-type: none"> • Runs the dynamic content more efficiently than the other methods. • Requires no purchase of additional preprocessor. 	<ul style="list-style-type: none"> • Requires the most migration effort because the dynamic content must be completely rewritten.



Important

When you run dynamic content written in preprocessors, you need to enable the *Web service extensions* (request handlers, such as Perl and CGI, that extend IIS functionality beyond serving static pages) for the preprocessor. For more information, see “Configuring Web Service Extensions” later in this chapter.

Run Dynamic Content on a Preprocessor that Ships with IIS

When you select this method for migrating the dynamic content, you must have the source code to the dynamic content and you must use a preprocessor that ships with IIS 6.0 and Windows Server 2003, such as CGI or Perl. You might have to adjust the code to accommodate any differences between preprocessors on Apache and IIS and between Linux and Windows.

Ensure that the version of the preprocessor supported on IIS 6.0 and Windows Server 2003 is compatible with the version of the code in the dynamic content. You most likely need to make modifications to the code in the dynamic content to compensate for version and platform differences. For example, CGI scripts migrated from Apache Web sites typically require modification to run on IIS 6.0 and Windows Server 2003.

Run Dynamic Content on a Preprocessor External to IIS

This method for migrating the external module assumes that you have the source code to the dynamic content and you can purchase or obtain a version of the preprocessor that runs with IIS, such as PHP. You might have to adjust the code to accommodate any differences between preprocessors on Apache and IIS 6.0.

For example, if the dynamic content is written in PHP, you must obtain a version of a PHP preprocessor for IIS to install and configure the PHP preprocessor on IIS 6.0. You must then adjust the PHP code to accommodate any differences between the PHP preprocessors on Apache and IIS 6.0.

Rewrite the Dynamic Content in ASP or ASP.NET

When you select this method for migrating the dynamic content, you need to evaluate the function of the original dynamic content, draft a functional specification for the dynamic content, and then develop a version of the dynamic content in ASP or ASP.NET.

For more information about selecting either ASP or ASP.NET, see the Active Server Pages and ASP.NET links on the Web Resources page at <http://www.microsoft.com/windows/reskits/webresources>.

Migrating Database Content and Connectivity

Database connectivity provides a standardized method of connecting to various database systems. Although the Apache to IIS 6.0 Migration Tool migrated the Web site content and configuration, any database content and database connectivity must be migrated separately. Many Apache Web sites and applications use a database server, such as MySQL, the most common database connectivity standard used in Apache Web sites. Dynamic and static content can make requests to databases through MySQL.

For more information about how to migrate database content and connectivity to IIS, see the Migration of Linux-Apache-MySQL-PHP Platform to Windows 2000 link on the Web Resources page at <http://www.microsoft.com/windows/reskits/webresources>. Although the white paper refers to IIS 5.0 and Windows 2000 Server, the processes and methods are applicable for IIS 6.0 and Windows Server 2003.

Migrating the Database Content

The Apache to IIS 6.0 Migration Tool does not transfer the database content to the target computer. You must migrate the database content by other methods, whose advantages and disadvantages are described in Table 7.6.

Table 7.6 Comparison of Methods for Migrating Database Content

Migration Method	Advantages	Disadvantages
Obtain a copy of the database that is compatible with IIS 6.0 and Windows Server 2003.	<ul style="list-style-type: none"> Requires minimal migration effort. Administrators are familiar with administering the database engine. 	<ul style="list-style-type: none"> Might not run as efficiently as Microsoft Data Engine (MSDE) or Microsoft SQL Server™ 2003. Requires purchase of an additional preprocessor.
Convert the database to MSDE or SQL Server 2000.	<ul style="list-style-type: none"> Might run more efficiently than the original database engine on IIS 6.0 and Windows Server 2003. Requires no purchase of additional database when using MSDE. 	<ul style="list-style-type: none"> Requires more extensive conversion. Administrators might not be familiar with administering MSDE or SQL Server 2000.

Obtain a Copy of the Database Engine that is Compatible with IIS

Most database engines on Linux have compatible versions that are designed for Windows. The easiest method of converting the database is to run the same database engine because little or no modification to the dynamic content is required. For more information, see the Web site or product documentation for the database engine used by the source server to determine if there is a version for Windows.

Convert the Database to MSDE or SQL Server 2000 Microsoft SQL Server supports the importing of data stored in other databases into SQL Server 2000. For example, you can import MySQL data directly into SQL Server 2000. You can use ODBC connectivity between MySQL and MSDE or SQL Server 2000 to import data as well. Client applications, such as Microsoft Access XP, can be used to import data through ODBC to MSDE or SQL Server 2000.

For more information about importing data into SQL Server 2000, see the SQL Server link on the Web Resources page at <http://www.microsoft.com/windows/reskits/webresources>. You can use the same methods for migrating data into MSDE as SQL Server 2000 because MSDE is based on SQL Server 2000.

Migrating the Database Connectivity

You need to ensure the dynamic content migrated to the target computer by the Apache to IIS 6.0 Migration Tool can use a database connectivity supported on IIS 6.0 and Windows Server 2003. Table 7.7 describes the advantages and disadvantages of the methods for migrating database connectivity.

Table 7.7 Comparison of Methods for Migrating Database Connectivity

Migration Method	Advantages	Disadvantages
Obtain a version of the database that runs on IIS 6.0 and Windows Server 2003.	<ul style="list-style-type: none"> Requires minimal migration effort. 	<ul style="list-style-type: none"> Requires purchase of the database that is compatible with IIS 6.0 and Windows Server 2003.
Convert the dynamic content to use ADO or ODBC.	<ul style="list-style-type: none"> Requires no purchase of additional database. 	<ul style="list-style-type: none"> Requires more extensive conversion.

Obtain a Version of the Database Connectivity that Runs on IIS 6.0

The database connectivity for dynamic content is implemented on Apache as libraries that are called by the dynamic content. When you install a version of the database connectivity on IIS 6.0 and Windows Server 2003, the appropriate libraries are also installed. There should be minimal changes to the dynamic content.



Note

This method assumes that you are installing a version of the database, such as MySQL, on the target server.

Convert the Dynamic Content to use Windows Database Connectivity

When you use this method, the assumption is that you are connecting to MSDE or SQL Server 2000 as the database engine through ODBC or ADO. If you are installing a version of the database engine on IIS 6.0 and Windows Server 2003, use the database connectivity libraries that accompany the database engine to reduce the effort required in the migration.

For example, there are ODBC drivers for MySQL; however, if you install MySQL on the target server, there is little or no conversion if you continue to use the MySQL libraries. In this case, converting your dynamic content to use ODBC has no long-term advantage unless you are planning to migrate to another database engine.

For more information about selecting ODBC, see the Open Database Connectivity (ODBC) link on the Web Resources page at <http://www.microsoft.com/windows/reskits/webresources>. For more information about selecting ADO, see the ActiveX Data Objects link on the Web Resources page at <http://www.microsoft.com/windows/reskits/webresources>.

Migrating External Modules

External modules, such as Cold Fusion and Wpoison, provide extended features to Apache Web sites. The Apache to IIS 6.0 Migration Tool transfers external modules to the target computer if they are in the same directory with the other Web site content. When the external modules are stored in other directories, you must manually transfer the external modules to the target server. After the external modules are transferred to the target server, you must complete additional steps so that the external modules run on IIS and Windows Server 2003. Table 7.8 describes the advantages and disadvantages of the methods for migrating external modules.

Table 7.8 Comparison of Methods for Migrating External Modules

Migration Method	Advantages	Disadvantages
Obtain a version of the external module that runs on IIS 6.0 and Windows Server 2003.	<ul style="list-style-type: none"> Requires minimal migration effort. Source code for the external module is not required. 	<ul style="list-style-type: none"> Might not run as efficiently as external modules written in ASP or ASP.NET. Requires purchase of another version of the external module. Might require minor modifications to the Web site content to accommodate version differences.
Convert the external modules to run on the same preprocessor as IIS 6.0 and Windows Sever 2003.	<ul style="list-style-type: none"> Can require minimal migration effort, depending on the extent of changes to the code for the external module. 	<ul style="list-style-type: none"> Might not run as efficiently as external modules written in ASP or ASP.NET. Might require the purchase of an additional preprocessor. Requires the source code of the external module so that the module can be converted.
Rewrite the external module in ASP or ASP.NET.	<ul style="list-style-type: none"> Runs the dynamic content produced by the original external module more efficiently. Requires no additional purchases. 	<ul style="list-style-type: none"> Requires the most migration effort because the external module must be completely rewritten.

Obtain Version of External Module that Runs on IIS 6.0 and Windows Server 2003

When you select this method for migrating the external module, you need to purchase a version of the external module for Windows. Install and configure the external module on the target server using the documentation that accompanies the external module.

For example, if an Apache Web site uses Cold Fusion, you would purchase a version of Cold Fusion for IIS 6.0 and Windows Server 2003. After installing and configuring Cold Fusion on the target server, you might have to adjust the code, such as PHP, CGI, or Perl, that calls the external module to accommodate any difference between Cold Fusion on Apache and IIS.

Convert External Modules to Run on Same Preprocessor as IIS 6.0 and Windows Sever 2003

This method for migrating the external module assumes you have the source code to the external modules. In addition, you must have a preprocessor on IIS that supports the code in which the external module is written, such as PHP, CGI, or Perl. You might have to adjust the code to accommodate any differences between preprocessors on Apache and IIS.



Important

When you run external modules written in preprocessors, you need to enable the Web service extensions for the preprocessor. For more information, see “Configuring Web Service Extensions” later in this chapter.

Rewrite the External Module in ASP or ASP.NET

Select this method if no version of the external module exists for IIS 6.0 and Windows Server 2003 and you do not have the source code to the external module, or if you want to rewrite the external module to take advantage of the enhanced performance offered by ASP or ASP.NET. You need to evaluate the functions of the external module, draft a functional specification for the external module, and then develop a version of the external module in ASP or ASP.NET.

For more information about selecting either ASP or ASP.NET, see the Active Server Pages and ASP.NET links on the Web Resources page at <http://www.microsoft.com/windows/reskits/webresources>.



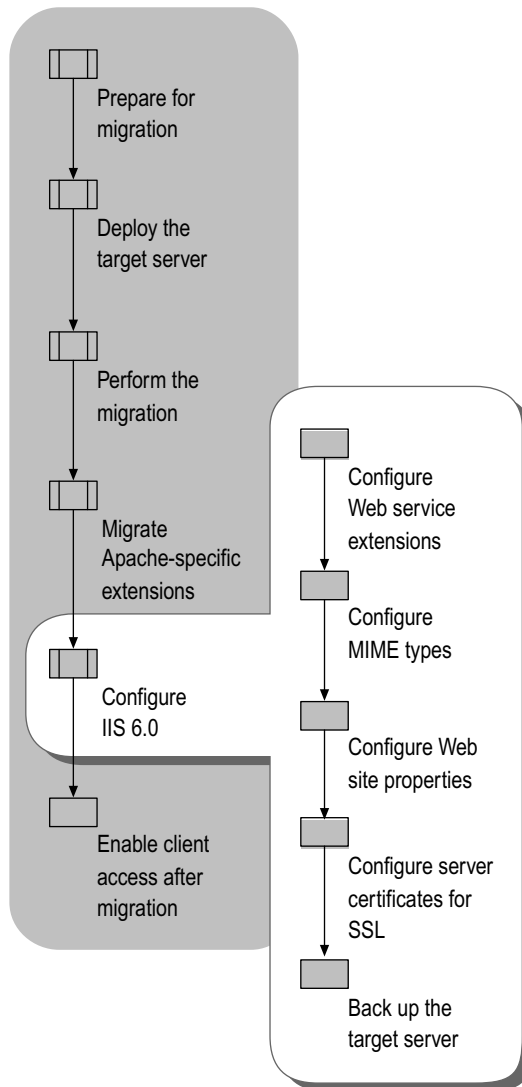
Important

When you write ASP or ASP.NET applications, you must enable Web service extensions for ASP or ASP.NET. For more information, see “Configuring Web Service Extensions” later in this chapter.

Configuring IIS 6.0

After migrating the Web site content, configuration, and Apache-specific extensions, you must further configure IIS to allow proper operation of the Web sites. This post-migration configuration of IIS ensures that your Web sites run as they did on Apache before they were migrated to IIS 6.0.

Figure 7.9 illustrates the process for configuring IIS 6.0 after completing the migration of the Web sites from Apache.

Figure 7.9 Configuring IIS 6.0 After Migration

Configuring Web Service Extensions

Many Web sites and applications that you migrate have extended functionality beyond static pages, including the ability to generate dynamic content. Any request handlers, such as Perl and CGI, that extend IIS functionality beyond serving static pages are known as *Web service extensions*. Web service extensions allow you to enable and disable the serving of dynamic content and easily determine the content types on the Web server that can initiate code.

If you installed IIS 6.0 as described in “Installing and Configuring IIS 6.0” earlier in this chapter, all Web service extensions are disabled by default. If you used another method, such as using Manage Your Server, the configuration of IIS might be different.

Enabling all Web service extensions ensures the highest possible compatibility with your Web sites. However, enabling all Web service extensions creates a security risk because it increases the attack surface of IIS by enabling functionality that might be unnecessary for your server.

Web service extensions allow you to enable and disable the serving of dynamic content. *MIME types* allow you to enable and disable the serving of static content. For more information about enabling and disabling the serving of static content, see “Configuring MIME Types” later in this chapter.



Tip

If the appropriate Web service extension is not enabled, the Web server returns a 404 error when attempting to serve the dynamic content.

Configure the Web service extensions by completing the following steps:

1. Enable the essential predefined Web service extensions based on the information in Table 7.9.

Table 7.9 Predefined Web Service Extensions

Web Service Extension	Enable When
Active Server Pages	Enable this extension when one or more of the Web sites or applications contains ASP content.
ASP.NET version 1.1.4322	Enable this extension when one or more of the Web sites or applications contains ASP.NET content.
FrontPage Server Extensions 2002	Enable this extension when one or more of the Web sites are FrontPage extended.
Internet Data Connector	Enable this extension when one or more of the Web sites or applications uses the Internet Data Connector (IDC) to display database information (content includes .idc and .idx files).
Server-Side Includes	Enable this extension when one or more of the Web sites uses server-side include (SSI) directives to instruct the Web server to insert various types of content into a Web page.
WebDav	Enable this extension when you want to support Web Distributed Authoring and Versioning (WebDAV) on the Web server, but it is not recommended for dedicated Web servers.



Important

Enable only the Web service extensions that are required by the migrated Apache Web sites.

2. For each Web service extension used by your applications that is not a one of the default Web service extensions, add a new entry to the Web service extensions list and configure the status of the new entry to **Allowed**.

For information about how to add a Web server extension to the list, see “Configure Web Service Extensions” in “IIS Deployment Procedures” in this book.

3. Use a Web browser on a client computer to verify that the Web sites and applications run on the server.

Configuring MIME Types

IIS serves only static files with extensions registered in the Multipurpose Internet Mail Extensions (MIME) types list. IIS is preconfigured to recognize a default set of global MIME types, which are recognized by all configured Web sites. You can define MIME types at the Web site and directory levels, independent of one another or the types defined globally. IIS also allows you to change, remove, or configure additional MIME types. For any static content file extensions used by the Web sites hosted by IIS that are not defined in the MIME types list, you must create a corresponding MIME type entry.

The Apache to IIS 6.0 Migration Tool migrates the MIME types for the Apache Web sites; however, you must add any MIME types for content that was added earlier in the Apache migration process by completing the following steps:

1. For each static file type used by your Web site, ensure that an entry exists in the MIME types list.

When your application uses the standard MIME types included in IIS 6.0 or the MIME types migrated by the Apache to IIS 6.0 Migration Tool, no new MIME type entries are required. However, if you added new static content in earlier in the Apache migration process, then add the new MIME types. For information about how to add a MIME type to the MIME types list, see “Configure MIME Types” in “IIS Deployment Procedures” in this book.

2. Use a Web browser on a client computer to verify that the Web sites and applications run on the server.

Configuring Web Site Properties

After the Apache Web sites are migrated, they run in a configuration that approximates their configuration on the Apache Web server. All of the Web sites run in the default application pool with the default security identity, NetworkService. In this configuration, the Web sites do not take advantage of the enhanced security and availability of IIS 6.0.

You can further configure the Web sites to provide the following:

- **Improved security.** In many cases, you can improve the security of the Web sites by isolating and running each one with unique identities. For more information about improving the security of the Web sites, see “Securing Web Sites and Applications” in this book.
- **Enhanced availability.** In many cases, you can to improve the availability of your Web sites. For more information about improving the availability of your Web sites, see “Ensuring Application Availability” in this book.

Configuring Server Certificates for SSL

Some of the Web sites on the source server might use Secure Sockets Layer (SSL) to encrypt confidential information exchanged between the Web server and the client. For each SSL-encrypted Web site that you migrated from the source server, you need to ensure that you have a certificate for that Web site installed on the target server. You can either renew the existing certificates on the source server in a format that is compatible with IIS 6.0 or you can request new certificates. After you have installed the server certificates on the target server, you can import the certificates into the certificate store on the target server, and then configure your Web sites to use the appropriate certificates.



Note

Server certificates are installed on the Web server and typically require no additional configuration on the client servers. Server certificates allow the clients to verify the identity of the server. Alternatively, some Web sites and applications might require client certificates. Client certificates are installed on the client servers and allow the server to authenticate the clients. For more information about configuring client certificates, see “Certificates” IIS 6.0 Help, which is accessible from IIS Manager.

For each Web site and application that uses SSL, configure the server certificate for SSL by completing the following steps:

1. Obtain an SSL server certificate that is compatible with IIS 6.0 by doing one of the following:

- Renew an existing certificate in a format that is compatible with IIS 6.0.

The format of the SSL server certificate on the source server is in a format that is incompatible with IIS. You can renew an existing certificate with your certification authority in a format that is compatible with IIS 6.0.

For more information about renewing existing server certificates in a format that is compatible with IIS 6.0, contact your certification authority.

- Request a new certificate from a certification authority in a format that is compatible with IIS 6.0.

You can use the Web Server Certificate Wizard either to generate a certificate request file (Certreq.txt, by default) that you send to a certification authority, or to generate a request for an online certification authority, such as Microsoft Certificate Services in Windows Server 2003. Depending on the level of identification assurance offered by your server certificate, you can expect to wait several days to several months for the certification authority to approve your request and send you a certificate file.

For more information about requesting a server certificate by using the Web Server Certificate Wizard, see “Request a Server Certificate” in “IIS Deployment Procedures” in this book.

2. Install the SSL server certificate to be used by the Web site on the target server.

After you obtain an SSL server certificate, you need to install the certificate on the target server. Install the SSL server certificate on the target server by using the Certificate MMC snap-in.

For more information about installing the SSL server certificate on the Web server, see “Install a Server Certificate” in “IIS Deployment Procedures” in this book.

3. Assign the SSL server certificate to the Web site.

For more information about assigning the SSL server certificate to the Web site, see “Assign a Server Certificate to a Web Site” in “IIS Deployment Procedures” in this book.

Backing Up the Target Server

Before you enable client access to the target server, perform a complete image backup of the target server. Performing this image backup provides you with a point-in-time snapshot of the Web server. If you need to restore the target server in the event of a failure, you can use this backup to restore the Web server to a known configuration.



Important

Do not continue with the migration process unless you have a successful backup of the entire target server. Otherwise, you can lose Web sites, applications, or data that you migrated to the target server.

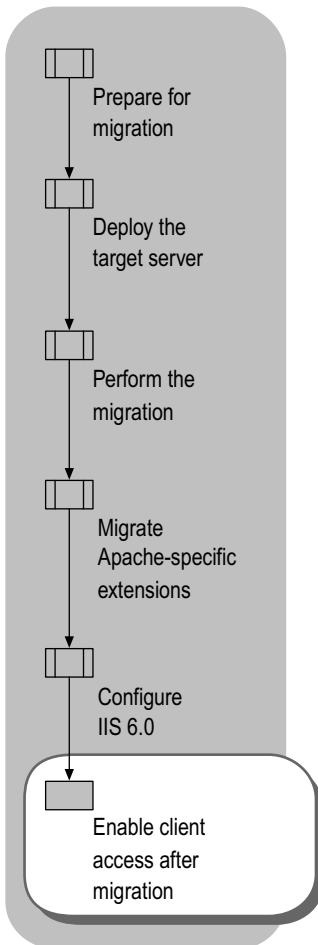
For more information about how to back up the Web server, see “Back Up and Restore the Web Server to a File or Tape” in “IIS Deployment Procedures” in this book.

Enabling Client Access After Migration

After you have migrated your Web sites from the source server to the target server, you are ready to enable client access to the Web sites on the target server, while maintaining the DNS entries to the source servers. After a period of time that meets your business needs, you can remove the DNS entries that point to the Web sites on the source server.

Figure 7.10 illustrates the final step of the process for migrating Web sites from Apache to IIS 6.0.

Figure 7.10 Enabling Client Access After Migration to IIS 6.0



Enable client access to the Web sites on the target server by completing the following steps:

1. Create the appropriate DNS entries for the Web sites and applications running on the target server.

For more information about how to create DNS entries for your Web sites and applications, see “Managing resource records” in Help and Support Center for Windows Server 2003.

2. Monitor client traffic to determine whether clients are successfully accessing the target server.

For more information about how to monitor client traffic to Web sites on the target server, see “Monitor Active Web and FTP Connections” in “IIS Deployment Procedures” in this book.

3. Establish a monitoring period, such as a few hours or a day, to confirm that clients accessing Web sites on the target server are experiencing response times and application responses that meet or exceed your requirements.

4. Remove the DNS entries pointing to the Web sites and applications on the source server.

For more information about how to remove DNS entries for your applications, see “Managing resource records” in Help and Support Center for Windows Server 2003.

5. Monitor client traffic to the Web sites on the source server to determine when clients are no longer accessing the source server.

For more information about how to monitor client traffic to Web sites on the source server, see “Monitor Active Web and FTP Connections” in “IIS Deployment Procedures” in this book.

6. When clients are no longer accessing the Web sites on the source server, decommission the hardware for the source server.

Additional Resources

These resources contain additional information and tools related to this chapter.

Related Information

- “Securing Web Sites and Applications in IIS 6.0” in this book for more information about securing your Web sites after migration.
- “Ensuring Application Availability” in this book for more information about improving the availability of your Web sites after migration.
- “Upgrading an IIS Server to IIS 6.0” in this book for information about upgrading IIS Web servers.
- “Migrating IIS Web Sites to IIS 6.0” in this book for information about migrating Web sites on earlier versions of IIS to IIS 6.0.

- “IIS Deployment Procedures in this book for more information about specific procedures for migrating Apache Web sites to IIS 6.0.
- The Active Server Pages and ASP.NET links on the Web Resources page at <http://www.microsoft.com/windows/reskits/webresources> for more information about selecting either ASP or ASP.NET to rewrite dynamic content.
- The Migration of Linux-Apache-MySQL-PHP Platform to Windows 2000 link on the Web Resources page at <http://www.microsoft.com/windows/reskits/webresources> for more information about migrating Apache Web sites to IIS.
- The SQL Server link on the Web Resources page at <http://www.microsoft.com/windows/reskits/webresources> for more information about importing data into SQL Server 2000.

Related IIS 6.0 Help topics

- “Certificates” in IIS 6.0 Help, which is accessible from IIS Manager, for more information about configuring client certificates.

Related Windows Server 2003 Help topics

- For best results in identifying Help topics by title, in Help and Support Center, under the **Search** box, click **Set search options**. Under **Help Topics**, select the **Search in title only** check box.
- “Managing resource records” in Help and Support Center for Windows Server 2003 or more information about how to create DNS entries for your applications.