



Understanding Updates for SharePoint Products and Technologies

This document is provided “as-is”. Information and views expressed in this document, including URL and other Internet Web site references, may change without notice. You bear the risk of using it.

- Some examples depicted herein are provided for illustration only and are fictitious. No real association or connection is intended or should be inferred.

This document does not provide you with any legal rights to any intellectual property in any Microsoft product. You may copy and use this document for your internal, reference purposes. You may modify this document for your internal, reference purposes.

© 2010 Microsoft Corporation. All rights reserved.

Understanding Updates for SharePoint Products and Technologies

Sam Hassani
Neil Hodgkinson

Microsoft Corporation

January 2010

Applies to: Microsoft® SharePoint® Products and Technologies

Summary: This white paper addresses the often-misunderstood subject of SharePoint Products and Technologies patching. We examine the various types of updates that are available and the release cycle for each type of update, and how these updates depend on — and interact with — each other. We cover the recommended, tried-and-true process for installing updates. We describe how to dissect a SharePoint Products and Technologies update and the typical components that you will find contained within it; this information is crucial for you to understand how the update mechanism works. The final section of the white paper discusses how to monitor, verify and troubleshoot the application of SharePoint Products and Technologies updates. This white paper is meant to be read in conjunction with TechNet articles, and includes references to these articles where relevant.

Contents

Contents	iii
Understanding Updates	1
Terminology	1
Packages	1
Updates	2
Transforms	2
Types of Updates	3
Hotfix	3
Cumulative Update	3
Service Pack	3
Understanding Farm Build Numbers	4
Build Number Format	5
Identifying Build Numbers	6
Update Releases	10
Patching post-Infrastructure Update	10
Cumulative Updates Model	10
Service Pack Support Lifecycle	13
Service Pack and Cumulative Update Interaction	14
Service Pack and Cumulative Update Installation Order	15
Dismantling an Update	16
Modifying Files	18
Deployment of Updates	19
Updating to	19
Which Build to Update To?	19
How to Deploy Updates	20
Patching with Minimal Downtime	21
Slipstream Creation	21
Sequence of Installation	22
Installing Updates	22
Uninstalling Updates	22
Installing Language Packs	22
Determining Upgrade Order	22
Parent-Child Farms	23
Post-Installation Monitoring and Troubleshooting	24
Verify That the Upgrade Was Successful	24

Monitoring the Upgrade 24

Troubleshooting 26

 Running Upgrade After Failure 26

Understanding Updates

Terminology

To understand how software updates are implemented in Microsoft® SharePoint® Products and Technologies, it is important to clearly understand the terminology for the core components that are contained in an update.

Packages

A package — sometimes called a “patch” — is a compiled, executable installer file that contains updates to one or more products. Examples of packages are the executable (.exe) files that you download to install a service pack, cumulative update (CU), or critical on-demand (COD) hotfix. Packages are also known as MSI files.

A particular update package is released for different processor architecture and language versions. The processor architecture is identified in the package file name. The following packages are available:

- **x86 architecture** Suitable for servers that run a 32-bit installation of SharePoint Products and Technologies.
- **x64 architecture** Suitable for servers that run a 64-bit installation of SharePoint Products and Technologies.

When you install a particular update for a given deployment of SharePoint Products and Technologies, you need to install both the global and the localized package(s) to ensure that your deployment is up to date:

- **Global package** This updates the core components of SharePoint Products and Technologies.
- **Localized package** This updates the language-specific components of SharePoint Products and Technologies.

Historically, localized packages were released only in languages that were specifically requested, but — in response to feedback from customers — they are now released in every language.

The core components of SharePoint Products and Technologies are language-agnostic. Any language-specific items of code are stored in separate DLL files or resource files. Many farm deployments have additional SharePoint language packs installed in the following scenarios:

- Where English is not the primary language spoken in the region where the farm is deployed, and therefore a region-specific language pack has been installed — for example, a French language pack installed by an organization based in France.
- Where a global deployment of SharePoint Products and Technologies has one or more farms that span and support multiple regions that require language packs to support the languages of those regions. For

example, a SharePoint Products and Technologies deployment that provides services to users in Germany, Spain, and the Middle East may have installed German, Spanish, and Arabic language packs.

In summary, an out-of-the-box installation of SharePoint Products and Technologies is like having a localized version in the core language. For example, an installation of Office SharePoint Server 2007 in U.S. English will include the region-agnostic core components plus the U.S. English localized components. This example deployment is commonly (and incorrectly) viewed as a “non-localized” version of Office SharePoint Server.

We have language packs for both Windows® SharePoint Services and Office SharePoint Server. In an Office SharePoint Server deployment, we must install both the Windows SharePoint Services and Office SharePoint Server language packs for any language that needs to be available in a given deployment. For instance, an out-of-the-box installation of Office SharePoint Server in the German language without any language packs will default everything to German. If the English language pack is applied for just Windows SharePoint Services and not Office SharePoint Server, users could see some parts of the UI in English, where other areas of the UI would fall back to the server’s out-of-the-box installation as a default language when the additional localized resources are not available.

Updates

The actual update itself is contained in the package in the form of a series of MSP files. You can run these MSP files directly, rather than execute the installer package that contains them. The drawback to running these MSP files directly is that, after the update is installed, the SharePoint Products and Technologies Configuration Wizard automatically runs in silent mode and invokes a build-to-build upgrade. This may be a problem in a multi-server farm environment, not only because the binaries will be out of sync between the servers, but also because the version numbers of the back-end databases might be incremented (depending on which MSP file is run and what changes were implemented by the update in the version that was running before the MSP file was run). Scenarios where you may want to interact directly with the MSP files are discussed in “Slipstream Creation,” later in this white paper.

Transforms

A transform is contained in the update (MSP file) itself, in the form of an MST file. The transform guides the installation of the update based on the environment. An update can include different transforms to support different environments; for example, an update may have one transform to be used with an RTM build of Office SharePoint Server and another for a Service Pack 1 (SP1) build of Office SharePoint Server. If a transform for the current installation of the product is not available, you will get an error message that the product is not recent enough to be updated. For example, you will receive this error if you try to apply the April Cumulative Update (April CU) to an RTM build, because the April CU was released after SP2 was released, and RTM support ended after SP2 shipped.

Types of Updates

Updates can be grouped into three categories based on their purpose and intended audience: hotfixes, cumulative updates, and service packs.

Hotfix

A hotfix is generally created to address a specific problem raised by a customer. There are three different types of hotfixes:

- **Critical on-demand** A critical on-demand (COD) hotfix is available to address critical problems that cannot be handled via the cumulative update delivery cycle. COD hotfixes are limited to emergency situations, for example, an issue that blocks normal business operations for the customer or an issue for which there is no effective workaround. COD hotfixes are included in the next cumulative update that is released.
- **Public update/Microsoft update** Occasionally, based on the criticality of an update, a COD hotfix is made available for public download or a Microsoft security update is released as a public update for SharePoint Products and Technologies. The US DST Hotfix - KB941422 is an example of a security update that was released as a public update.
- **Post-service pack rollup** This update package includes any “SPLock hotfixes,” which are hotfixes that were developed during the SPSLock period. The SPSLock period is a lockdown on service pack development that is meant to help achieve stability in the development process. Any hotfixes that have been produced before the SPSLock period is declared are integrated into the next service pack. SPSLock hotfixes are never distributed publicly and are only made available during Microsoft Customer Service and Support engagements. SPSLock hotfixes require special handling because if the next service pack is applied without the post-service pack rollup, the fix is lost. This could cause data loss, so Microsoft is very careful about releasing these SPSLock hotfixes, and provides detailed guidance for each customer scenario

Cumulative Update

Cumulative updates include all updates that broadly affect support issues that have been released since the last service pack. Since December 2008, cumulative updates have become truly cumulative, and are released on schedule every two months. For more information about cumulative updates, see “Update Releases,” later in this white paper.

Service Pack

Service packs include all of the updates for SharePoint Products and Technologies (that is, all hotfixes and cumulative updates, but not the SPSLock hotfixes described above). They also deliver important stability and performance improvements that might not have been requested specifically by customers, but were found internally by the product group. These improvements incorporate further enhancements to user security.

Understanding Farm Build Numbers

In SharePoint Products and Technologies, the *farm build number* varies in proportion to the number and variety of updates that have been installed in the farm. Not every IT department installs the exact same kind of updates at the same time; for this reason, figuring out the build number of a SharePoint farm can be more of an art than a science.

Before deploying any update, it is important to be aware of the current build number of the SharePoint Products and Technologies deployment. It is also important to ensure that all servers within a SharePoint farm consistently share the same build number. Having the same build number is defined as “specific SharePoint Products and Technologies installed from an MSI file that is at the same update level.” For example, if you have an Office SharePoint Server farm with five servers, you could have Windows SharePoint Services Service Pack 2 (SP2) plus the June Cumulative Update (June CU) and Office SharePoint Server SP2 without any cumulative update installed on top of it. This is acceptable as long as all five servers are configured in the same way. If three of the servers have Windows SharePoint Services SP2 plus the June CU, and two of the servers only have Windows SharePoint Services SP2 without the June CU, you have not achieved consistency. The June CU should be applied to the two servers that are out of sync from a build number perspective.

To clarify this point: We are discussing the build number on a per-product basis. The build numbers in the content databases, site settings, and Central Administration site are based on the Windows SharePoint Services build level; they are not necessarily the same for other products such as Office SharePoint Server, Project Server, and so on. You can update the different products to different build numbers, but we recommend that you keep them in sync to maximize the benefit of applying fixes that cross product boundaries. For example, it will not harm your environment if you do not keep Windows SharePoint Services and Office SharePoint Server in sync, but if Microsoft releases a hotfix that requires both Windows SharePoint Services and Office SharePoint Server builds to be past a certain build number, you will not gain the benefit of that fix, and your environment will remain “broken” in the same way it was before the partial fix was applied.

Build Number Format

The format of a SharePoint Products and Technologies build number is shown in the following illustration.



The build number is composed of the following four numbers:

- **Major** This indicates the major version of the product. For Windows SharePoint Services 3.0 or Office SharePoint Server 2007, this is always 12.
- **Minor** This indicates the minor version of the product. For any version of Windows SharePoint Services 3.0 or Office SharePoint Server 2007, this is always 0 for our binaries when our major version is 12, and will not be incremented when we install service packs. (In some contexts, this number will change; for example, in the upgrade log, you may find references to 12.1, where the .1 represents Service Pack 1).
- **Build** This is what we can use to indicate the version number. This is the number that is incremented when we apply a service pack or any other update. Some examples of build numbers and their corresponding versions are given below:
 - 4518 is the RTM build
 - 6219 is SP1
 - 6320 (Windows SharePoint Services) and 6322 (Office SharePoint Server) are the Infrastructure Updates
 - 6421 is SP2
 - 6520 is the October 2009 CU

- **Revision** The revision number indicates the type of update that has most recently been applied. Some examples are given below:
 - 5000 is a fully supported, COD hotfix suitable for a production environment
 - 1000 is a service pack
 - 300x is private release build for a customer who has been working with Microsoft Premier Support. This type of update should not be deployed in a production environment, because it has not been properly integration-tested. However, if this type of update is applied to a production environment, the final hotfix or later hotfixes will replace it as a final release and will have a higher revision number in the format of 500x.

Caution Using a private release in production can cause irreversible data loss. In many cases, Customer Service and Support can work through non-data loss scenarios, but the risk of data loss is so high with private release builds that Microsoft has taken the stance that these types of builds should never be put into production.

Identifying Build Numbers

There are various methods you can use to identify build numbers, including:

- Check SharePoint Products and Technologies special directories and storage locations.
- Check the Windows SharePoint Services content databases version in SQL Server®.
- Check the Knowledge Base (KB) updates that have been applied on each server that is running SharePoint Products and Technologies .
- Check the upgrade log.

Check Special Directories and Storage Locations

You can check the version attribute value of certain key files contained in the Office SharePoint Server 2007 installation directory (%COMMONPROGRAM FILES%\Microsoft Shared\Web Server Extensions\12) to determine the build number of the SharePoint Products and Technologies environment. Note that when these build numbers are identified for corresponding components, they only relate to global updates that have been applied. It can be difficult to identify localized updates that have been applied, because the files that are updated by localized updates are not usually files that have a version number. To check these files manually, you can do one of the following:

- Validate the contents of the files against the same files in a non-modified environment that has had the update applied.
- Generate an MD5 hash of the current file and compare it against what is stored in the MSP file by opening the MSP file in Orca.exe. For more information about using ORCA, see “Dismantling Updates,” later in this white paper.

Files that are typically updated by the localized updates are resource files, JavaScript files, and so on. For more information about identifying specific localized updates that have been applied, see “Check the KBs that have been applied on each server running SharePoint Products and Technologies.”

The following TechNet article describes the special directories and storage locations:

<http://technet.microsoft.com/en-us/library/cc721635.aspx>.

The following table lists the files that correspond to components of SharePoint Products and Technologies.

Component	Filename
Windows SharePoint Services	%COMMONPROGRAM FILES%\Microsoft Shared\Web Server Extensions\12\ISAPI\OWSSVR.DLL
Office SharePoint Server (core server)	%COMMONPROGRAM FILES%\Microsoft Shared\Web Server Extensions\12\ISAPI\Microsoft.SharePoint.Portal.DLL
OSRV	%COMMONPROGRAM FILES%\Microsoft Shared\Web Server Extensions\12\ISAPI\ Microsoft.Office.Server.DLL
DLC (Document Lifecycle Conversions)	Highest version of: Microsoft.Office.Policy.DLL, Mirosoft.Office.Workflow.Pages.DLL located in: %COMMONPROGRAM FILES%\Microsoft Shared\Web Server Extensions\12\ISAPI\
PJSRV (Project Server)	%COMMONPROGRAM FILES%\Microsoft Shared\Web Server Extensions\12\ISAPI\Microsoft.Office.Project.Server.dll
IFS (InfoPath® Forms Server)	%COMMONPROGRAM FILES%\Microsoft Shared\Web Server Extensions\12\ISAPI\Microsoft.Office.InfoPath.Server.dll
XLSRV (Excel® Services)	%COMMONPROGRAM FILES%\Microsoft Shared\Web Server Extensions\12\ISAPI\Xlsrvint.dll

Looking at these file versions is very helpful, but does not always give you an exact picture. Consider the Windows SharePoint Services Infrastructure Update. This update was released with a build number of 6320. However, the build number of OWSSVR.DLL is actually 6318 after the Windows SharePoint Services Infrastructure Update has been successfully applied. This can cause a certain amount of confusion, because the perceived build number in this case is 6318, but the actual build number is 6320.

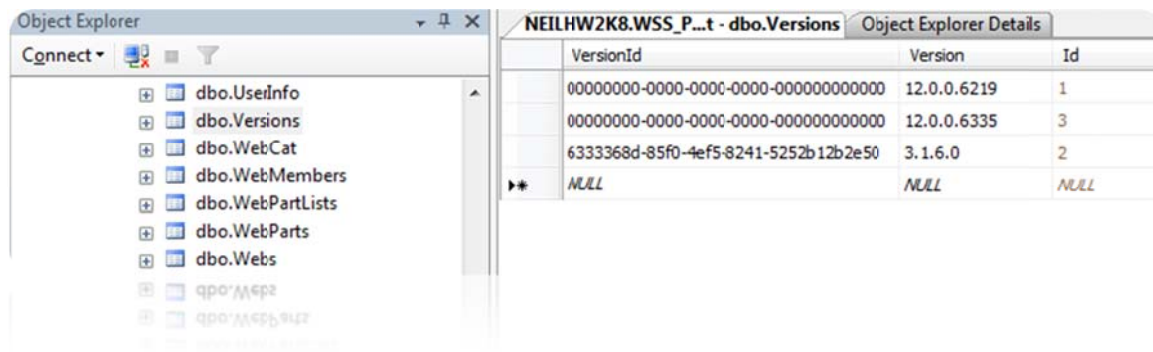
Important Using the OWSSVR.DLL and other key files to identify our build number is typically a reliable approach to take, but it is important to be aware that it is not an exact science.

The most accurate method for identifying the build number is to look at the KBs that have been installed. The KB numbers should correspond to the actual build number. For more information, see “Check the KBs That Have Been Applied on Each Server Running SharePoint Products and Technologies,” later in this white paper.

Check Windows SharePoint Services Content Databases Version in SQL Server

Another way to identify the build number of Windows SharePoint Services is to look up the build number from the version table in a content database. It is in fact OWSSVR.DLL that defines this value.

For a particular content database, locate the table named `dbo.versions`. In this table, find the records that have a **VersionID** attribute equal to all zeroes. If multiple updates have been applied, there will be multiple records showing the different intermediate builds. The record that has the highest **Id** value shows the latest build number in the **Version** field.



A preferred way to retrieve this information is to issue a read query on this table by using the **NOLOCK** option. This option ensures that no shared locks are issued, which prevents other transactions from modifying data being read by the current transaction. Use a query such as the following:

```
SELECT * from Versions with (nolock)
```

This is a great way to identify all the updates that have been applied. For example, if you check OWSSVR.DLL and see that the environment has a build number of 12.0.0.6335 (the build corresponding to the December 2008 CU), but you do not know whether SP1 has ever been applied, you can use this table to identify any record that has a value in the version field that corresponds to SP1 (12.0.0.6219).

Check the KBs That Have Been Applied on Each Server Running SharePoint Products and Technologies

Another way to identify the build number is to identify the KBs that have been installed on the server. In Control Panel, open **Add/Remove Programs**, and then click **View installed updates**. To identify the build number that corresponds to a KB number, do one of the following:

- Look up the KB on the Knowledge Base site (use the format <http://support.microsoft.com/kb/<kbnumber>> for the URL), and look for the build number in the KB description.
- Raise a Customer Service and Support case, provide the KB number, and request the build number.

As previously described, any localized updates that have been applied do not have build numbers. The only way to determine which localized updates have been applied is to identify which KBs have been installed.

You can use this method of checking for installed KBs as an alternative way to see the set of updates that have been applied to a server and to identify any servers that do not have all the updates that the rest of the servers have.

Check the Upgrade Log

You can use the upgrade log to identify what components have been updated, and — more importantly — to what build level.

There are two stages of upgrade:

- Binary installation
- Build-to-build (B2B) upgrade

It is possible for upgrade to reach a successful first stage and a partially successful second stage without the entire environment being in an unavailable state. For example, in the scenario where we have 100 site collections, four might have failed to be upgraded. For every subsequent update applied in the future, the upgrade will continue to fail on those sites because it can't move forward to the new TargetSchemaVersion until it completes upgrading the site it is currently on. In fact, it has to upgrade each site in order. This reinforces a point mentioned earlier that there is no such thing as a "farm build number." There's not a single number. You could come close to a single number if you have updated all of the MSI files installed relating to separate components of SharePoint Products and Technologies to the exact same build number. We come close to this with the new server-package cumulative updates (described in "Cumulative Updates Model," later in this white paper), but even there we can have different builds inside those packages.

In summary, we should view our farm update level from a component (MSI)–specific standpoint, but also understand that individual components may actually be older if they failed to be upgraded when an update was applied.

Update Releases

Since the release of Microsoft® Office SharePoint® Server 2007 and Windows® SharePoint Services 3.0, there have been changes in the way updates are released and improvements to how they are applied to a SharePoint environment. These include the following:

- Updating after an Infrastructure Update
- Cumulative updates model
- Service Pack support lifecycle
- Service Pack and cumulative update interaction

Patching post–Infrastructure Update

The Infrastructure Update was released in July 2008 (<http://support.microsoft.com/kb/951695/>). It drastically reduced the time that is required to update and upgrade SharePoint sites. Any update applied to SharePoint before the Infrastructure Update upgrades each site collection in the farm by updating its build number to reflect the most recent value. When you apply updates after installing the Infrastructure Update, only those site collections that have a schema object that needs to be updated — such as an update to the template schema — will be upgraded. This type of update is far less frequent, because updates rarely modify the site collection schema (however, the Infrastructure Update does support the integrated search features). Consequently if an update does not require a schema update, the SharePoint Products and Technologies Configuration Wizard does not have to go through each site collection and update the build number. This drastically reduces the amount of time necessary to perform the upgrade.

For more information about the improvements to the patching process that the Infrastructure Update has made possible, see: <http://support.microsoft.com/kb/953749/>

Cumulative Updates Model

Historically, it has been hard to manage which SharePoint Products and Technologies cumulative updates needed to be applied for your systems to be up-to-date. The reason for this lay in the fact that individual components had to be updated separately. Administrators had to consider not only which SharePoint Products and Technologies components needed to be updated, but which localized updates to apply for these components. Cumulative updates were often released for each of these components individually. It was difficult for administrators to identify the current build of their environment and which updates had (or had not) been applied; this often resulted in inconsistencies across the farm. An environment would appear to be at a certain update level to an administrator who used accustomed methods to check the build number, but because certain components would be missing, the environment would still experience bugs that the administrator had applied updates to fix.

This cumulative update model is illustrated in the tables below. Before December 2008, for both Windows SharePoint Services and Office SharePoint Server, different updates were released for different components of the server products.

Windows SharePoint Services (before December 2008):

Build		STS-glb	STS-loc
6324.5000	July 2008	955594	
6327.5000	Aug 2008		957109
6332.5000	Oct 2008	957691	

Office SharePoint Server (before December 2008):

Build		CoreServer-glb	CoreServer-loc	DLC-glb	DLC-loc	pjsrvwfe-glb	pjsrvapp-glb	lfswfe-glb	Xlsrvapp-glb
6324.5000	July 2008	955593		955586	955586				
6327.5000	Aug 2008	956056							953397
6331.5000	Oct 2008	957693	958567	958569		957696	957694		

The SharePoint product group at Microsoft has heard your feedback about how hard it has been to manage which cumulative updates needed to be applied for systems to be up-to-date. From December 2008 onward, for each cumulative update release, we also produce a "server package" that contains the latest of every hotfix update that has ever shipped.. Consider a scenario where you want to build a new Office SharePoint Server server. You can apply the latest service pack, the latest Windows SharePoint Services cumulative update server-package, and the latest Office SharePoint Server cumulative update server-package, and be completely up-to-date. (There might have been a COD hotfix beyond the cumulative update. However, COD hotfixes receive the least testing of all updates, and we do not recommend that you install them simply to keep your environment up-to-date.). However, while it is recommended, you are not required to install the latest service pack. If it is not possible to install the latest service pack, you can install a cumulative update on top of an older service pack that is still within lifecycle. For more information, see <http://www.microsoft.com/lifecycle>.

A few key points should be noted on the structure of the new update format for SharePoint Server 2007:

- Windows SharePoint Services continues to remain separate and is not included in the Office SharePoint Server package
- All of the latest global and localized updates for Windows SharePoint Services are in the Windows SharePoint Services package
- All of the latest global and localized updates for Office SharePoint Server (Excel Server, Document LifeCycle, and so on are part of Office SharePoint Server), InfoPath Forms Server, and Project Server are in the Office SharePoint Server package.

- The list of what is in the package is an accumulation over time of what has shipped since RTM. It is key to understand that cumulative updates only affect components for which a hotfix has actually been built; by contrast, a service pack updates all of the MSI files.

Typically a Windows SharePoint Services cumulative update server-package contains the following files (“x-none” means global, where a regional code like “en-us” indicates localized):

- Dw20w-x-none.msp
- Sts-x-none.msp
- Wssmui-en-us.msp (and every other language)

Typically an Office SharePoint Server cumulative update server-package contains the following files:

- Coreserver-x-none.msp
- Coreservermui-en-us.msp (and every other language)
- Dlc-x-none.msp
- Dlcmui-en-us.msp (and every other language)
- Ifswfe-x-none.msp
- Lpsrvwfe-x-none.msp
- Msxml5s-x-none.msp
- Pjsrvapp-x-none.msp
- Pjsrvwfe-x-none.msp
- Xlsrvapp-x-none.msp

This can be illustrated more clearly in the tables below for Windows SharePoint Services and Office SharePoint Server, respectively. Notice how the releases differ after December.

Windows SharePoint Services (after December 2008):

Build		STS-glb	STS-loc
6324.5000	July 2008	955594	
6327.5000	Aug 2008		957109
6332.5000	Oct 2008	957691	
6335.5000	Dec 2008	960010	960010
6341.5000	Feb 2009	961755	961755

Office SharePoint Server (after December 2008):

Build		CoreServer-glb	CoreServer-loc	DLC-glb	DLC-loc	pjsrvwfe-glb	pjsrvapp-glb	lfswfe-glb	Xlsvapp-glb
6324.5000	July 2008	955593		955586	955586				
6327.5000	Aug 2008	956056							953397
6331.5000	Oct 2008	957693	958567	958569		957696	957694		
6335.5000	Dec 2008	960011	960011	960011	960011	960011	960011	960011	960011
6341.5002	Feb 2009	961756	961756	961756	961756	961756	961756	961756	961756

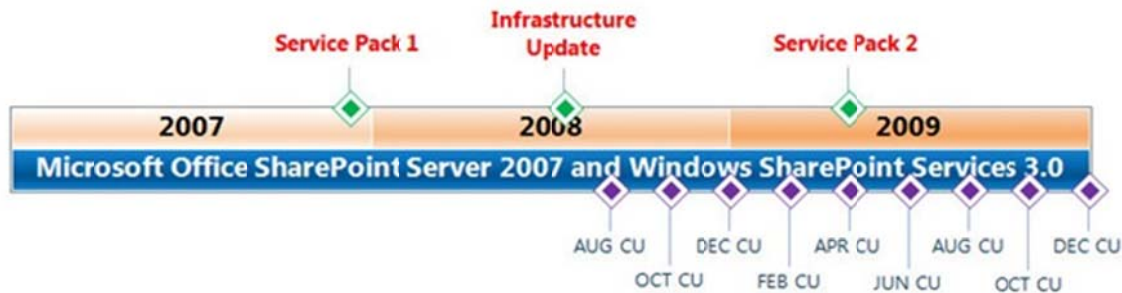
Cumulative updates are released every two months, which allows for better preparation and testing of new updates. Any customer who needs an emergency hotfix can still request a critical on-demand (COD) hotfix. For information, see <http://support.microsoft.com/kb/953878>.

The above list of updates is not exhaustive and may be out of date in places. For an accurate and up to date list of updates please check the Updates Resource Center for SharePoint Products and Technologies (<http://technet.microsoft.com/en-us/office/sharepointserver/bb735839.aspx>)

Service Pack Support Lifecycle

After a service pack is released, the n-1 build is supported for 12 months. After 12 months, updates for an n-1 build will not typically be shipped. RTM for Windows SharePoint Services 3.0 and Office SharePoint Server 2007 went out of support on January 13, 2009, which resulted in Service Pack 1 (SP1) becoming the minimum supported build. From this date onward, cumulative updates will typically not ship with a transform for any builds previous to SP1. We do reserve the right to still ship these transforms for limited periods of time past the 12 months, but it is a choice we will make under certain circumstances.

Below is an example timeline to help understand the update schedule during 2007 through 2009. Cumulative updates are released every two months.



Since April 2009, cumulative updates cannot be applied directly to RTM version of SharePoint installations. SP1 is the minimum requirement, while SP2 is recommended.

Service Pack and Cumulative Update Interaction

Occasionally, a service pack and a cumulative update will be released almost simultaneously. An example of this is the release of the April CU on April 30, 2009, and the Service Pack 2 (SP2) release on April 28 2009.

SP2 includes every hotfix, security update, infrastructure update, service pack or any other update that was released for SharePoint Products and Technologies through February, 2009. This follows the behavior of any service pack for SharePoint, in that all updates released since RTM will be included in the service pack until SP2Lock begins. Therefore, all hotfixes that were released in cumulative updates prior to the April CU are included in SP2.

However, if only the April CU is installed, and not SP2, some of the updates included in SP2 will not be present in the environment. For the fullest set of updates to be installed, we recommend that you install both SP2 and the April CU.

To explain further, the April CU includes only a subset of SP2 files — those that were updated in response to a hotfix request. On the other hand, SP2 contains product improvements and updates to many other files that are not impacted by a hotfix request. The volume and diversity of fixes in SP2 is much greater than those in the April CU. The general guideline is to install the latest service pack and the latest cumulative update server-packages to get the full set of the latest updates.

Service Pack and Cumulative Update Installation Order

Typically, it is possible to install a service pack and a cumulative update in any order. This is the case for SP2 and the April CU.

To explain further, the installer checks to see whether the environment already has a newer version of the file it is attempting to replace. If not, it replaces the old file with the newer version; otherwise it skips the installation and retains the newer version. It is key to understand that the installer does this when processing an MSP for each file; the installer does not simply look at the version stamped on the package and check a registry key or a specific file and then move on. This is a crucial point to understand, because it explains why applying an update that you don't need takes as long as it does. It is necessary for the installer to behave this way to guarantee that the binaries on disk are as accurate as possible.

For example, say that the April CU version of a file is newer than the SP2 version of the same file and always takes precedence during an installation. Therefore, if you install SP2 followed by the April CU, the April CU version of the file replaces the SP2 version. On the other hand, if you install the April CU first and then SP2, the April CU version of the file is retained in the environment. The April CU version of a file contains all updates to that file that were in SP2 also. Hence, installation order does not matter.

Below is an example table for a file (microsoft.sharepoint.portal.dll), in various installation scenarios.

Installation Scenario	Patches Installed	Example File	File Version after installation
SP2 only	coreserverwvsp2.msp	microsoft.sharepoint.portal.dll	12.0.6420.1000 (SP2 version)
April CU only	coreserver.msp	microsoft.sharepoint.portal.dll	12.0.6504.5000 (April CU version)
SP2 + April CU	coreserverwvsp2.msp + coreserver.msp	microsoft.sharepoint.portal.dll	12.0.6504.5000 (April CU version)
April CU + SP2	coreserver.msp + coreserverwvsp2.msp	microsoft.sharepoint.portal.dll	12.0.6504.5000 (April CU version)
SP1 + April CU	Osrchwfewwvsp1.msp + coreserver.msp	microsoft.sharepoint.portal.dll	12.0.6504.5000 (April CU version)

If you attempt to install an update from an older cumulative update after you applied an update for the same set of files from the newer cumulative update, the installer will prevent installation and display an error that says: "This patch or one that supersedes it is already installed." Note that because the updates are

cumulative in nature, the newer cumulative update always contains the latest version of the set of files and also encompasses fixes in those files from the older cumulative update.

Dismantling an Update

When you download an update, you may want to understand the internal structures of the package. For example, you might want to verify that a particular update has a particular version number.

The following table maps updates (MSP files) to the MSI files and translates MSI files to the products, and illustrates which components are updated by a particular package.

Update	Components
sts.msp	sts.msi (Windows SharePoint Services)
coreserver.msp	spswfe.msi (Core Office SharePoint Server components), osrchapp.msi, osrchwfe.msi (Search components), osrv.msi (Shared components)
dlc.sp	dlc.msi, dlcapp.msi (Document Life Cycle components (includes policies, workflow, and so on))
pjsrvapp.msp, pjsrvwfe.msp	pjsrvapp.msi, pjsrvwfe.msi (Project Server)
ifswfe.msp	ifswfe.msi (InfoPath Forms Server)
xlsrvapp.msp	xlssrvapp.msp (Excel Calculation Services)

A tool that can be used to dismantle an update and is freely available to all is Orca.exe, from the Windows Installer Development Tools area of the Windows Software Development Kit. The tool is available from the following link:

<http://msdn2.microsoft.com/en-us/library/aa370557.aspx>

To use Orca.exe to examine the internal structure of an update, follow the instructions below. This example shows how to validate the version of the December CU for Windows SharePoint Services:

1. Copy the files from the released version source media for the product to a folder (in this example, sts.msi).
2. Download the appropriate software update package.
3. Extract the software update files by using this command:

```
<package> /extract:<path>
```

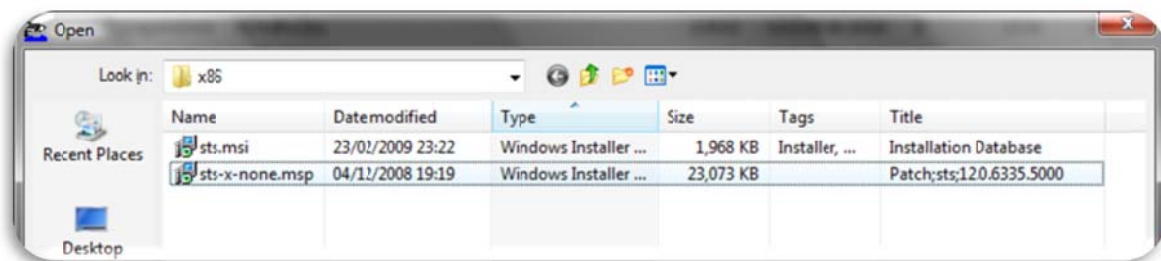
The **/extract** switch prompts you to provide a folder name for the files. The example below is for x86 systems.

4. `wssv3sp2-kb953338-x86-fullfile-en-us.exe /extract:<C:\WSS\Updates>`

`<C:\WSS\Updates>` is the location to which you copied the files that you extracted from the Windows SharePoint Services 3.0 released version (sts.msi).

5. Start Orca.exe.
6. Click **Open** on the menu bar, navigate to the sts.msi file in the above folder, and then click **Open**.
This opens the MSI database.
7. Click **File** in the left pane, click **Edit**, and then click **Find**. Locate the Microsoft.sharepoint.dll file and note the version number.
8. Click **Transform**, click **View Patch**, and then navigate to the sts-x-none.msp file in the folder you specified in step 4.
9. In the Select dialog box, change the view to details, and make sure the title field is included (right click the fields, and then select **Title**).

Observe the titles of the MSI file in comparison to the update. Sts.msi should display **installer database** and the MSP update should display **Patch** beside a version number, as illustrated below. Sts-x-none.msp should show the build number we are aiming for.



10. Click **open**.

As illustrated below, after we have applied the update to the original transform, we can see changes in the different sections in the left pane. Green lines indicate a change in a particular section, and green boxes indicate a new section.

The database can be explored to identify different changes and modifications incorporated into the update. An interesting section to explore is the **File** section in the left pane.

11. Click the **File** section in the left pane, and then click the **Version** column to sort by version.
12. Scroll down until you reach 12.0.4518.1016

All components that have the version of 4518.1016 were part of the full release at RTM. Various different releases (SP1, Infrastructure Update, and so on) can be viewed as you scroll through list of files, which makes it easy for you to identify when files were last updated.

In this example, we explored the December CU for Windows SharePoint Services. The most recent version that you observe should be a build number of 12.0.6335.5000. In the December CU, notice that the majority of the core DLL files were updated.

File	Size	Version	Status
SPWRITER.EXE	37416	12.0.6329.5000	0
STSAADM.EXE	527256	12.0.6331.5000	0
STSSOAP.DLL	310128	12.0.6331.5000	0
WSSADMOP.DLL Microsoft.SharePoint.Administratio...	469872	12.0.6332.5000	0
TQUERY.DLL tquery.dll	2345800	12.0.6333.5000	0
ONETUTIL.DLL	1981296	12.0.6335.5000	0
STSWEL.DLL	1932144	12.0.6335.5000	0
STSLB.DLL Microsoft.SharePoint.Library.dll	129904	12.0.6335.5000	0
STSOM.DLL Microsoft.SharePoint.dll	9374576	12.0.6335.5000	0
STSOM.DLL Microsoft.SharePoint.dll	9374576	12.0.6335.5000	0
OWSSVR.DLL	3019120	12.0.6335.5000	0
STSA.P.DLL Microsoft.SharePoint.ApplicationPages.dll	625552	12.0.6335.5000	0
SEARCHO1.DLL Microsoft.SharePoint.Search.dll	1440120	12.0.6335.5000	0
SEARCHOM.DLL Microsoft.SharePoint.Search.dll	1440120	12.0.6335.5000	0

The above screenshot of the Orca user interface shows the raw internal file name of STSOM.DLL which as a part of the installer action gets renamed to Microsoft.SharePoint.dll.

Modifying Files

It is unsupported to modify core SharePoint files. Consequently, upgrading modified files is not supported, and core Windows Installer behavior is followed for these non-versioned files (for example, bform.js).

The hash of all core files are stored in the registry. Whenever we install an update, Windows Installer checks the registry and compares the hash stored with the hash of the file. If they match, the file is updated and the new hash is added to the registry. After a file has been modified, the hash of the file changes and Windows Installer treats the file as data and will not update it again.

For more information about this behavior, see the following articles:

- [http://msdn.microsoft.com/en-us/library/aa368599\(VS.85\).aspx](http://msdn.microsoft.com/en-us/library/aa368599(VS.85).aspx)
- [http://msdn.microsoft.com/en-us/library/aa370531\(VS.85\).aspx](http://msdn.microsoft.com/en-us/library/aa370531(VS.85).aspx)

Deployment of Updates

Updating to...

With the frequent release of cumulative updates and regular service pack release cycles, an administrator must carefully decide which updates to apply to a SharePoint environment, and when, to help ensure the stability and availability of the environment.

In the field, there are three categories into which most SharePoint environments fall, in terms of the current build number of the environment:

- Minimum required build
- Microsoft recommended build
- Latest and greatest build

Minimum required build The minimum required build is the oldest supported build. Currently, this is SP1 for Windows SharePoint Services 3.0 and Office SharePoint Server 2007. After a service pack is released, the $n - 1$ version is supported for 12 month. After this period, updates will not be shipped for the $n - 1$ version.

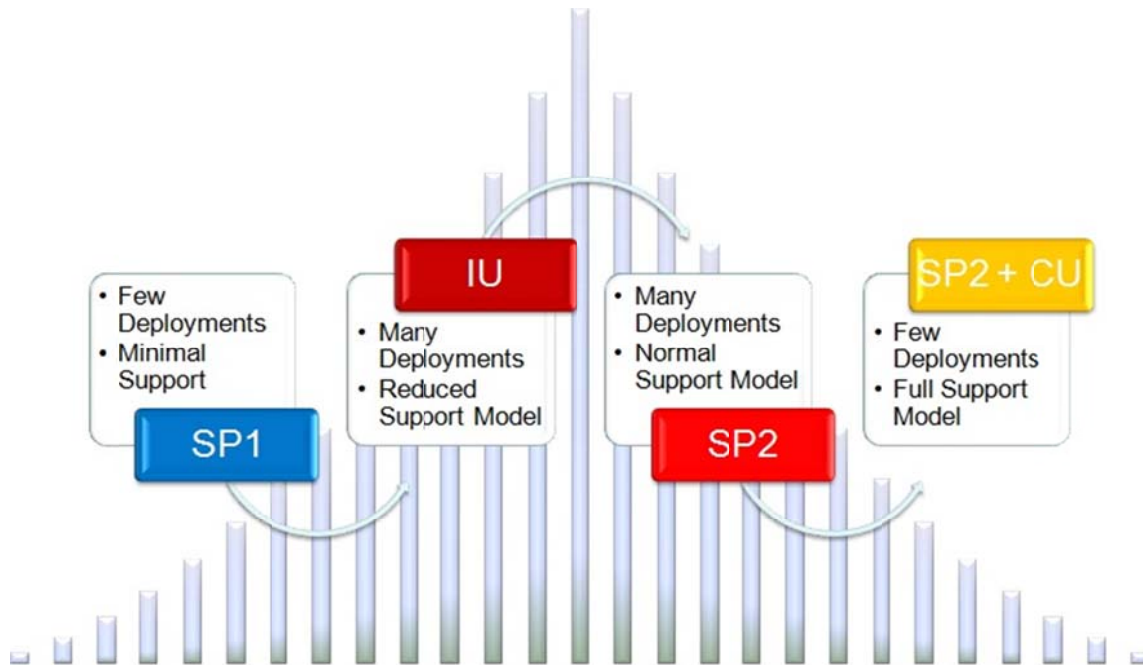
Microsoft recommended build The Microsoft recommended build is the build level the SharePoint product group recommends to the public. Typically, this build will be more recent than the minimum required build, and will be announced on the product group's blog at <http://blogs.msdn.com/sharepoint>. The current recommended build is SP2 for Windows SharePoint Services 3.0 and Office SharePoint Server 2007. For customers who want to use the Stsadm **mergecontentdbs** operation, the recommended build is SP2 plus the April CU.

Latest and greatest build In this build, all the latest updates have been applied, regardless of whether all the features that are fixed by a particular update are used in the environment. Generally, administrators who maintain "latest and greatest" builds find over time that this approach results in the best performance and the fewest problems. Within our own Microsoft environments, we have embraced this model not only because of these benefits but also because when we update between a smaller gap of build numbers, the upgrades go much faster. For example, when we update from post-SP2 April CU to the post-SP2 June CU, it is much faster than if we update from SP1 to the post-SP2 June CU. There are fewer code changes from CU to CU than there are when we skip multiple CUs or only update at service pack intervals. Therefore, fewer upgrade actions need to run and, consequently, there are fewer things to do to complete the upgrade.

Which Build to Update To?

There appear to be cultural changes in attitudes about patching SharePoint environments. We are seeing a more continual upgrade model in the field. This represents a culture shift, moving away from the attitude of only installing an update if one is affected by the problems solved by that particular update. The consumer is more accustomed to being told when to update by the software vendor rather than waiting for a problem to

arise and then responding. Today, software vendors are more diligent, which leads to increased consumer trust.



With SharePoint updates, we see a bell curve like the illustration above. Most environments are currently at SP2. It is important to be aware of the consequences of running older builds of SharePoint Products and Technologies. If you experience a particular issue and require a fix, you may need to take a big upgrade step, for example from the Infrastructure Update all the way to the latest and greatest build. Not only does this incur the maximum downtime, but because so many performance-related and other issues have been addressed between those two builds, there is a significant chance that the problem being seen would go away on the later builds. Because of this, Microsoft Customer Service and Support will often verify whether the problem still exists in the latest and greatest builds by using a reproduced environment. It is always faster to apply an existing hotfix than get a new one and apply it.

How to Deploy Updates

For in-depth guidance on the update process, refer to the following articles:

[Deploy software updates for Windows SharePoint Services 3.0](http://technet.microsoft.com/en-us/library/cc288269.aspx) (<http://technet.microsoft.com/en-us/library/cc288269.aspx>)

[Deploy software updates for Office SharePoint Server 2007](http://technet.microsoft.com/en-us/library/cc263467.aspx) (<http://technet.microsoft.com/en-us/library/cc263467.aspx>)

Patching with Minimal Downtime

For in-depth guidance on how to update your environment with minimal downtime, refer to the following article:

[Configure a server farm for minimal downtime during software updates \(Office SharePoint Server 2007\)](http://technet.microsoft.com/en-gb/library/ee514459.aspx)

(<http://technet.microsoft.com/en-gb/library/ee514459.aspx>)

Slipstream Creation

Typically, there are three different scenarios when updates may be applied to a SharePoint Products and Technologies environment:

1. **New farm:** Implementing a new collection of SharePoint Products and Technologies servers, all at a specified build level.
2. **New server in an existing farm:** Adding a new server to an existing farm as part of a scale-out process or replacing an existing server. The new server must be same build version as existing servers in the farm.
3. **Existing server in an existing farm:** In this, the most common patching scenario, updates are applied to servers by following new Microsoft recommendations, or to correct an error that has been experienced (per a KB article), or simply to be running the latest code.

For scenarios 1 and 2 above, updates can manually be applied after SharePoint Products and Technologies has been installed on the servers, or you can follow an approach known as "slipstreaming." This involves creating an installation source that contains a copy of the original, released version of the software, along with the software updates to match the build level that you want. For example, in scenario 2, we want to add a new front-end Web server to an existing farm that is at an SP2 build. We create an installation source that contains a copy of the original Office SharePoint Server installation media and SP2. Setup then runs from this updated installation source, and the new server will have the same software update version as the rest of the Web servers in the server farm. Maintaining a network installation source that has been updated by using the same updates that you applied to a production farm is a recognized best practice, one that will improve the turn-around time of deploying new servers to the farm.

For more information about how to create a slipstreamed installation source, see the following articles:

[Create an installation source that includes software updates \(Windows SharePoint Services 3.0\)](http://technet.microsoft.com/en-us/library/cc287882.aspx)

(<http://technet.microsoft.com/en-us/library/cc287882.aspx>)

[Create an installation source that includes software updates \(Office SharePoint Server 2007\)](http://technet.microsoft.com/en-us/library/cc261890.aspx)

(<http://technet.microsoft.com/en-us/library/cc261890.aspx>)

Sequence of Installation

Installing Updates

It is possible to successfully install service packs, hotfixes, and language packs and their updates all at the same time, and only run the SharePoint Products and Technologies Configuration Wizard one time. However, the environment will not be accessible between installing the first update and the final update and running the configuration wizard.

To reduce troubleshooting complexity, you may choose to run the configuration wizard after you apply service packs and again after you apply language packs. You may choose to do this a third time if you do updates in addition to those. We have flexibility here, but the more frequently you run the configuration wizard between installing updates, the better. If something isn't upgraded successfully, a more complicated process creates a more complicated analysis of a problem encountered during that process.

Uninstalling Updates

It is not possible to uninstall any type of SharePoint Products and Technologies update. This is because it is not possible to reverse the database upgrade operations that take place when you apply an update. Schema changes to the databases are made that cannot be reversed. It is therefore crucial to have good, tested backups available before you carry out any upgrade operations in SharePoint Products and Technologies. If you have a catastrophic problem during the application of an update that Customer Service and Support cannot remediate, the solution is to restore the server by using the backups that you took before you applied the update.

Installing Language Packs

For Office SharePoint Server environments, it is necessary to install both Windows SharePoint Services and Office SharePoint Server language packs. There is often confusion about this point because at RTM it was possible just to install the Office SharePoint Server language pack and updates: the Office SharePoint Server language pack also contained the Windows SharePoint Services language pack files. From the release of SP1 and later, it is necessary to install both Windows SharePoint Services and Office SharePoint Server language packs, along with updates for both components, separately.

Determining Upgrade Order

When applying updates in a multi-server farm environment, you have to decide which server to upgrade first — more specifically, which server to run the SharePoint Products and Technologies Configuration Wizard on first, after the update binaries have been installed on each of the servers in the farm. This upgrade operation can only be run on one server in the farm at a time in Office SharePoint Server 2007 and Windows SharePoint Services 3.0. There are many possible upgrade scenarios. A few examples are discussed below.

A point to bear in mind is that the first server where the upgrade operation is run will initiate the SharePoint content database upgrades, which is the longest running operation within the upgrade. The services that have been enabled on servers will affect which components will be upgraded; for example, a server on which the Search service is enabled will upgrade the search components. It makes sense to optimize what is

happening in an environment by upgrading multiple components at the same time when the first server is being upgraded. Depending on the way the environment is structured, it may not be possible to upgrade every single component at the same time. The first server you choose should upgrade as many components as possible, and the next server you choose to upgrade should maximize the number of components it is upgrading.

After all the major roles have been upgraded, the SharePoint Products and Technologies Configuration Wizard can be run on the remaining servers one at a time. Upgrading these remaining servers will be a comparatively quick process.

If one server exists that hosts the Central Administration site, it is generally a good idea to target this server first. The reason for this is to ensure that you have a working Central Administration site in case you need to manage your configuration because one of the subsequent servers failed to be upgraded.

If more than one server hosts the Central Administration site, it is important to first upgrade the server that hosts the Central Administration site that was created first. The reason for this is that the Central Administration site that was created second will have links back to the first site, so it is important not to upgrade the second site Central Administration when the first is not available.

Considering larger farms, it can be more difficult to choose which server to upgrade first. There are many different possible upgrade scenarios, and a strategy should be put in place following the logic described above.

Parent-Child Farms

In parent-child farm scenarios, our recommendation is to update parent farms first, and then child farms. The reason for doing this is that the farms could be out of sync with each other. The updated parent services should be resilient to work with both updated child farms and non-updated child farms, whereas if you update a child farm beyond the parent, it is possible that the child farm could return data in a way that the parent does not understand or expect, which could cause inconsistent results. Although this is likely to be a rare occurrence, it is theoretically possible, so the best practice is to keep the parent at the same or later version than the child farms.

To illustrate this with an example: If an update changes how the crawler works, the target would not be aware of these code changes, where the source would. In other words, if the children were updated first, the behavior of the child `sitedata.aspx` could potentially be altered through an update, and a parent farm at an older build could potentially not be ready for those changes. Conversely, if the parent is updated first, a crawler update would have to be created and tested to work against an older version of the code as a target (that is, a separate farm scenario). This has been tested by Microsoft.

Post-Installation Monitoring and Troubleshooting

Verify That the Upgrade Was Successful

After an update has been installed, the success of the installation can be verified by using one of the following techniques:

- View the upgrade log file. In addition to viewing the results of the installation in Upgrade.log file, you can use this log file to troubleshoot a failed installation. (See "Monitoring the Upgrade," later in this paper, for more information about the upgrade log file.)
- Check version numbers on certain files and registry keys. If you have to investigate the success of the software update installation in more depth, use this procedure to verify version numbers of certain files and verify certain keys in the registry.
- Examine the SQL Server schema. You can also verify that the software update installation was successful by using SQL Query Analyzer to examine the SQL Server schema. Although the version of the DLL files and the registry are updated during the first part of an upgrade — when the files are being copied — the SQL Server schema is only upgraded after the SharePoint Products and Technologies Configuration Wizard is run. Use this procedure to determine whether the SharePoint Products and Technologies Configuration Wizard was run after the software update.
- View the version number on the Servers in Farm page in the SharePoint Central Administration Web site. However, note that this page only displays the version number for Windows SharePoint Services 3.0.

For a detailed description of the above methods, refer to the section "Verify update completion and success" in the following TechNet articles:

[Deploy software updates for Windows SharePoint Services 3.0](http://technet.microsoft.com/en-us/library/cc288269.aspx) (<http://technet.microsoft.com/en-us/library/cc288269.aspx>)

[Deploy software updates for Office SharePoint Server 2007](http://technet.microsoft.com/en-us/library/cc263467.aspx) (<http://technet.microsoft.com/en-us/library/cc263467.aspx>)

Monitoring the Upgrade

After you apply an update, upgrading the environment (by starting the SharePoint Products and Technologies Configuration Wizard) can take several hours. One may be interested in knowing what is happening as the upgrade is taking place. The upgrade.log file is written to during the upgrade process.

By default, the upgrade.log is stored in C:\Program Files\Common Files\Microsoft Shared\web server extensions\12\LOGS. However if the SPTimerv3 account cannot write to the default Upgrade.log, it may write to \Documents and Settings\SPTimerv3 account\Local Settings\Temp\Upgrade.log. The upgrade.log file is cumulative and could potentially grow very large. We therefore recommend that you relocate this file if

its located on a system drive. It cannot be moved while the upgrade is in process. We also recommend that you rename upgrade.log before you begin a build-to-build upgrade, so that it is easier to identify any problems specific to that session.

Upgrade.log can be viewed as the upgrade is taking place. Third-party tools are available to automatically refresh the file as it is being written to. Alternatively, you can use a simple tool such as Notepad to open and view the file. Close Notepad and open the file again to view the latest actions that have taken place.

In the upgrade.log file, it is possible to view the different sequences that guide various actions to be run. It can be somewhat complex to understand the upgrade.log file. The following KB article describes how to look at an upgrade.log file and the different sequences and actions: <http://support.microsoft.com/kb/948780>.

In upgrade.log , various sequences and actions are repeated. This is by design. If multiple content databases or multiple Web applications exist within a SharePoint environment, each of these objects will be upgraded in turn.

A commonly asked question is, "What does the log file entry 'SyncUpgradeTimerJob: sleeping for 10 seconds' mean?"

This is an entry often written to upgrade.log as the upgrade is taking place. As the SharePoint Products and Technologies Configuration Wizard is run, it makes a timer job that reports back its status to the upgrade process. While upgrade actions are being carried out, the wizard is going to write this message to the log file if it hasn't received a notification of the current status in 10 seconds. If these messages are being received for long periods of time, and other errors appear within the same time frame, this could indicate that issues are being experienced in the upgrade process.

```
[SPManager] [DEBUG] [2/7/2009 9:33:40 PM]: Creating exclusive upgrade reg key
[SPManager] [DEBUG] [2/7/2009 9:33:49 PM]: SyncUpgradeTimerJob: sleeping for 10 seconds
[SPManager] [DEBUG] [2/7/2009 9:33:59 PM]: SyncUpgradeTimerJob: sleeping for 10 seconds
[DEBUG] [2/7/2009 9:34:09 PM]: SyncUpgradeTimerJob: sleeping for 10 seconds
[SPManager] [DEBUG] [2/7/2009 9:35:13 PM]: Logging enabled Saturday, February 07, 2009 at 9:35:13 PM.
[SPManager] [INFO] [2/7/2009 9:35:13 PM]: upgrade Boot Strap starts. Initialized in Mode: InPlace, as
MachineAdmin: True.
```

Troubleshooting

Running Upgrade After Failure

If an upgrade attempt failed, and the underlying issue has been resolved, it may be the case that the upgrade needs to be forced to be completed. One way of achieving this is by running the command-line version of the SharePoint Products and Technologies Configuration Wizard.

To force an upgrade

1. Open a Command Prompt window.
2. Change to the following directory: <system drive>\Program Files\Common Files\Microsoft Shared\web server extensions\12\BIN.
3. Type the following at the command prompt:

```
psconfig –cmd upgrade –inplace b2b –wait –force
```

-WAIT = No timer job so the upgrade happens in real time in the psconfig process.

-FORCE = Removes all existing upgrade jobs incase a timer job is stuck.

An alternative way to complete an upgrade is to detach content databases, upgrade the farm, and then reattach the content databases. More information about this method can be found in the section entitled 'To detach content databases' in the following TechNet articles:

[Deploy software updates for Windows SharePoint Services 3.0](http://technet.microsoft.com/en-us/library/cc288269.aspx) (<http://technet.microsoft.com/en-us/library/cc288269.aspx>)

[Deploy software updates for Office SharePoint Server 2007](http://technet.microsoft.com/en-us/library/cc263467.aspx) (<http://technet.microsoft.com/en-us/library/cc263467.aspx>)