Sitecore Experience Platform Installation Guide

Sitecore Experience Platform 8.2 Update 7

Installation guide for administrators and developers

sitecore®
Own the experience™

Table of Contents

Chapter 1 Introduction 4
1.1 Getting Started 5
1.2 Preparing to Install Sitecore XP 6
1.2.1 Install Latest Updates 6
1.2.2 UNC Share Not Supported 6
1.2.3 Sitecore Cannot Operate from a Virtual Directory 6
Chapter 2 Sitecore XP Requirements 7
2.1 Sitecore Hosting Environment Requirements 8
2.1.1 IIS Requirements 8
2.1.2 Operating System Requirements 8
2.1.3 .NET Framework Requirements 9
2.1.4 Visual Studio Requirements for Custom Solutions 9
2.1.5 Database Requirements 9
Oracle 9
2.1.6 Antivirus Software Considerations 9
2.1.7 Hardware Requirements for a Server Running a Single Sitecore Installation 10
2.2 Sitecore Client Requirements 11
2.2.1 Software Requirements 11
2.2.2 Hardware Requirements 11
Chapter 3 Installing the Sitecore XP from the .EXE File 12
3.1 The UAC (User Account Control) Configuration 13
3.2 The Installation Process 14
3.3 Troubleshooting 21
Chapter 4 Installing the Sitecore XP from the .ZIP File 22
4.1 Installation Process Overview 23
4.2 Configuring Folder and Registry Permissions 25
4.2.1 Updating the Security Settings for a Folder 25
4.2.2 File System Permissions for Anonymous Requests 25
4.2.3 File System Permissions for ASP.NET Requests 25
4.2.4 File System Permissions for System Folders 26
4.2.5 File System Permissions for SQL Server Files 26
4.2.6 Windows Registry Permissions 26
Adding an Account to a User Group 27
4.3 Configuring the IIS 28
4.3.1 IIS 7, 7.5, 8, 8.5 and 10 28
Create the Application Pool 28
Create the Website 29
Limit Access to Anonymous Users 30
4.4 Microsoft SQL Server Configuration 33
4.4.1 Creating a Database Account for Sitecore XP Databases on SQL Server 33
4.4.2 Granting Execute Rights to Stored Procedures 35
4.5 Oracle Database Storage Setup 37
Database Server Preparations 37
Website Server Setup 37
4.6 Uninstalling Sitecore XP 39
Chapter 5 Sitecore XP Post-Installation Steps 40
5.1 Content Expiration 41
5.2 Windows Authentication for Sitecore 43
5.3 Setting Microsoft SQL Server Database Properties 45
5.3.1 Warm up the Servers 46
5.4 Rebuild the Search Indexes and the Link Database 47
5.4.1 Cache Configuration 47
5.5 Configure the Sitecore Experience Database (xDB) 48

5.6 Deploy Marketing Definitions....49
5.7 Deploy Path Analyzer Maps....50

Chapter 1

Introduction

This document describes how to install Sitecore Experience Platform 8.2 rev. 180406 (Update-7).

The document contains the following chapters:

- **Chapter 1 – Introduction**
  An introduction to the Sitecore Experience Platform 8.2 installation process.
- **Chapter 2 – Sitecore XP Requirements**
  An outline of the installation requirements for Sitecore CMS.
- **Chapter 3 – Installing the Sitecore XP from the .EXE File**
  Describes the pre-installation steps, configuration and installation process.
- **Chapter 4 – Installing the Sitecore XP from the .ZIP File**
  Step-by-step instructions for installing Sitecore CMS from the .ZIP file.
- **Chapter 5 – Sitecore XP Post-Installation Steps**
  Information about necessary actions after installation.

1.1 Getting Started

Sitecore® Experience Platform™ (Sitecore XP) is the name of the entire Sitecore solution. It contains:

- Sitecore CMS – the content management capabilities.
- Sitecore xDB – the experience marketing and scalability, and analytics features.

These components are installed together. Sitecore XP is distributed as an `.exe` or `.zip` file.

The Sitecore Experience Database (xDB) is a central data repository for marketers. It collects and connects all customer interactions in a single view, and enables immediate, personalized, automated experiences. Sitecore xDB uses MongoDB as a collection database and it is included in the Sitecore XP package.

1.2 Preparing to Install Sitecore XP

For small implementations, including developer workstations and testing environments, Sitecore XP and the database server can be installed on a single computer. For larger implementations, the database server is typically separated from the application server. The content authoring environment for business users is also frequently separated from the content delivery environment that is accessed by Internet users.

For more information about scaling Sitecore XP solutions, refer to the Sitecore Documentation site – https://doc.sitecore.net/sitecore_experience_platform.

Sitecore XP can be broken down into three logical components regardless of the database technology that is used:

- The IIS document root ("the website").
- The supporting file system for logs and data.
- The supporting relational databases. These databases are often stored under the supporting file system.

The configuration is stored in files such as `web.config` and the files under the `\App_Config` folder.

For information about hardening security after you install Sitecore, see the manual Security Hardening Guide in the Sitecore Documentation site.

You can create an archive of your Sitecore XP solution to move it from one computer to another.

For assistance or to report any discrepancies between this document and the product, please contact http://support.sitecore.net/helpdesk/.

Before installing Sitecore XP, you should decide on the:

- Project name and acronym, for example, *ProjectName*.
- Acronym/prefix, for instance PN.
- Target installation file system root, for instance `C:\inetpub\wwwroot\ProjectName`.
- Database technology:
  - SQL Server
  - MongoDB
- Database location – locally or on a remote server.
- Appropriate Sitecore license.
- Hardware.
- Administrative access to computers and databases.
- IIS configuration – host headers, and so on.

1.2.1 Install Latest Updates

Run Windows Update to install all the appropriate service packs and security updates on all of your Sitecore XP host and client computers.

1.2.2 UNC Share Not Supported

Sitecore XP must be installed on a local drive, not a UNC share.

1.2.3 Sitecore Cannot Operate from a Virtual Directory

Do not create a virtual directory for Sitecore, because the product cannot operate from a virtual directory.

Chapter 2

Sitecore XP Requirements

This chapter describes the prerequisites, hardware, and software requirements for Sitecore XP 8.2 host and client computers. In this document, the term "server" refers to a Windows OS Web server and the term "host" refers to an installation of Sitecore XP on either a server or a workstation.

This chapter contains the following sections:

- Sitecore Hosting Environment Requirements
- Sitecore Client Requirements

2.1 Sitecore Hosting Environment Requirements

Sitecore XP 8.2 has specific requirements for the operating system, IIS Web Server, .NET Framework, and the database server.

Important
When you configure the Sitecore Experience Database (xDB), you must synchronize all the servers in your solution to a single reliable time source, for example, by means of the Network Time Protocol (NTP). The aggregation of engagement automation states depends on the system time and changing this can lead to incorrect aggregation results or loss of data.

2.1.1 IIS Requirements

Sitecore XP can be hosted on the following IIS versions:

- IIS 10
- IIS 8.5
- IIS 8.0
- IIS 7.5
- IIS 7.0

You must use the version of IIS that is supported by your operating system. For more information about IIS and operating systems, see Microsoft's documentation.

Sitecore XP does not officially support any other ASP.NET web servers such as IIS Express, or Mono Web Server.

Important
Sitecore neither supports nor permits multiple IIS website definitions pointing to the same Sitecore web root.

2.1.2 Operating System Requirements

Sitecore XP 8.2 is only compatible with the client and server operating systems that support .NET Framework 4.5.2.

Sitecore XP has been tested, and can be hosted on the following Microsoft operating systems:

- Windows Server 2016
- Windows Server 2012 R2 (64-bit)
- Windows Server 2012 (64-bit)
- Windows Server 2008 R2 (32/64-bit) SP1
- Windows 8.1 (32/64-bit)
- Windows 8 (32/64-bit)
- Windows 10 (32/64-bit)

Important
Run Windows Update and install all the appropriate service packs and security updates on all of your Sitecore XP host and client computers.

Windows 10

Sitecore XP supports the following editions of the Windows 10 operating system:

- Home

- Education
- Professional

Windows 8.0 and 8.1

Sitecore XP supports the following editions of the Windows 8.0 and 8.1 operating systems:

- Enterprise
- Professional
- Core

2.1.3 .NET Framework Requirements

Sitecore XP requires .NET Framework 4.5.2.

You must apply any available updates to the .NET Framework to every Sitecore installation.

2.1.4 Visual Studio Requirements for Custom Solutions

Sitecore XP 8.2 requires Visual Studio 2012 or later.

2.1.5 Database Requirements

Sitecore XP supports the following database servers:

- Microsoft SQL Server 2016 SP1
- Microsoft SQL Server 2014
- Microsoft SQL Server 2012
- Microsoft SQL Server 2008 R2 SP1
- MongoDB Database:
  - Mongo 2.6 MMAPv1
  - Mongo 3.0 MMAPv1 or WiredTiger
  - Mongo 3.2.1 MMAPv1 or WiredTiger
  - Mongo 3.2.1 Enterprise with data-at-rest encryption, WiredTiger only

Note
Sitecore recommends that you run the database on a separate server from the web server.

Oracle

The Oracle database storage setup is only available when Sitecore XP is running in Sitecore Experience Management mode (CMS-only mode), which lets you run Sitecore XP without the Experience Database (xDB) enabled.

Sitecore supports:

- Oracle Database 11g R2 (11.2)

2.1.6 Antivirus Software Considerations

Some antivirus software can have a detrimental effect on the performance of ASP.NET applications including Sitecore. We therefore recommend that you use only antivirus scanners that are certified for the operating system that you use.

For more information about the certified products, see the Windows Server Catalog website.

To ensure optimal performance, exclude the following folders from scanning by your antivirus software:

- The site root folder.
- The data folder that is defined in the `web.config` file.
- The folder that contains the actual Sitecore database files.
- The `C:\Windows\Temp or {app_pool user profile}\Temp` folder.

2.1.7 Hardware Requirements for a Server Running a Single Sitecore Installation

Minimum configuration requirements:

- 4 core processor
- 8GB of RAM

Recommended configuration:

- 4 core processor
- 16GB of RAM

Note
The recommended hardware requirements are for running the software on a single computer. For more information about running Sitecore on different kinds of hardware, consult your Sitecore partner or technical sales representative.

2.2 Sitecore Client Requirements

This section describes the software and hardware requirements.

2.2.1 Software Requirements

Browser

Sitecore XP clients are browser-based user interfaces and can run on the following browsers:

- Microsoft Internet Explorer 11
- Mozilla Firefox 17+
- Google Chrome 25+
- Microsoft Edge
- Apple Safari 6+ for Mac OS

Note
Sitecore XP 8.2 supports all the current stable versions of these browsers unless stated otherwise. Sitecore XP 8.2 does not support the Compatibility view in Internet Explorer 11.

For more information about configuring a browser, see the Browser Configuration Reference document on the Sitecore Developer Network.

Silverlight

Some applications, such as Engagement Automation, and Sitecore App Center require Microsoft Silverlight 5 or later.

2.2.2 Hardware Requirements

Sitecore XP 8.2 has the following hardware requirements:

- Processor: Intel Pentium 4, 2GHz or faster processor.
- RAM: 512 MB minimum, 1GB – recommended.
- TCP/IP connection at 512Kbps or faster to the Sitecore XP host.
- 1024 x 768 or greater screen resolution required for advanced operations.

You do not need to install any additional software on the Sitecore XP clients that access Sitecore XP servers.

Chapter 3

Installing the Sitecore XP from the .EXE File

This chapter describes how to use the installation program to install Sitecore XP.

You can download the Sitecore installation program from the Sitecore Downloads page – https://dev.sitecore.net.

This chapter contains the following sections:

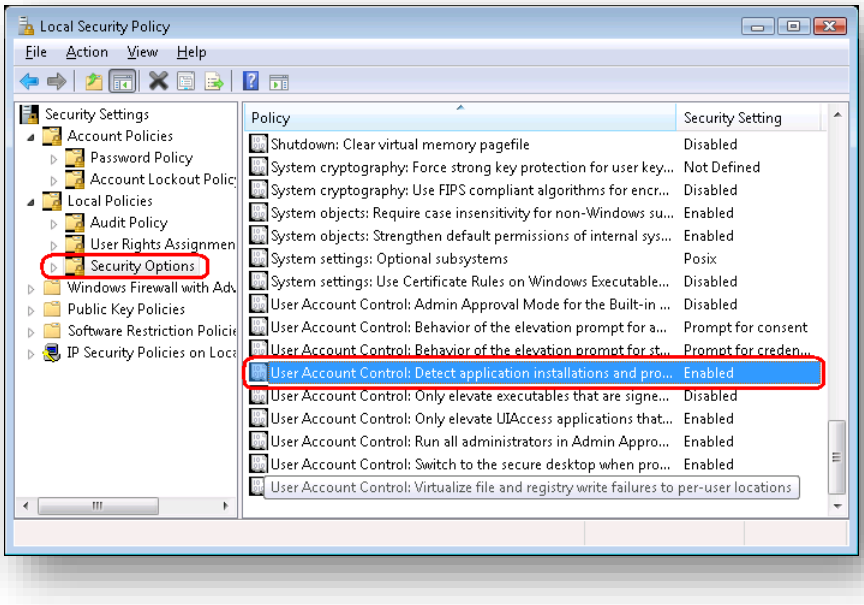
- The UAC (User Account Control) Configuration
- The Installation Process
- Troubleshooting

3.1 The UAC (User Account Control) Configuration

Before you run the installation program on Windows 8, Windows 8.1, or Windows 10, you must ensure that the UAC (User Account Control) configuration security policy *Detect application installations and prompt for elevation* is enabled.

To check if this setting has been changed on your computer:

1. Log in as an administrator.
2. Press WIN + R to open the **Run** window.
3. In the **Run** window, enter *secpol.msc* and press ENTER.
4. Click **Continue** to open the **Local Security Policy** window.



5. In the pane on the left side, expand the *Local Policies* node and select *Security Options*.
6. In the pane on the right side, scroll to the bottom of the list of policies.
7. Verify that the *User Account Control: Detect application installations and prompt for elevation* policy is set to *Enabled*.

   If this setting is not enabled, you must enable it.

3.2 The Installation Process

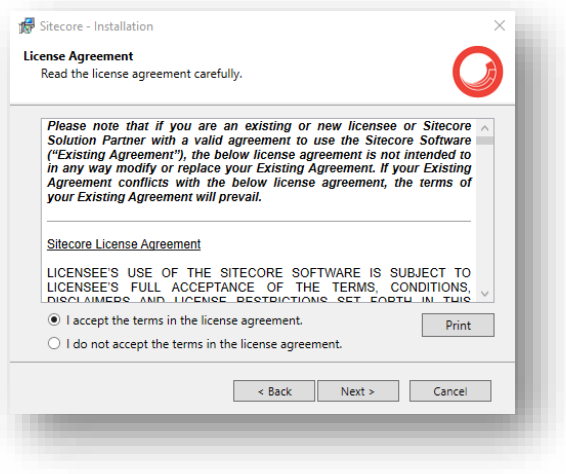
To start the installation, launch the *Sitecore.exe* application, which starts the **Installation Wizard**. The **Installation Wizard** guides you through the installation process.

This section describes the important steps in the wizard.

Note
The Sitecore XP configuration files are automatically modified according to the settings provided in the wizard.

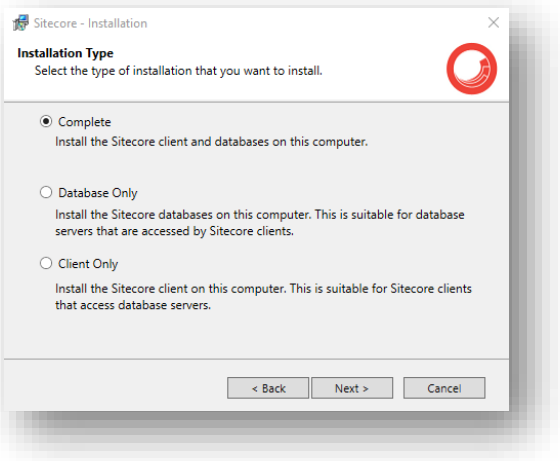
Welcome page

License Agreement



Read the license agreement and select the appropriate option. Click **Next**.

Installation Type

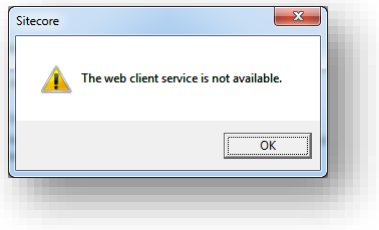


Select the type of installation that you want:

- Complete – Installs both the Sitecore databases and the Sitecore client.
- Database Only – Only installs the Sitecore databases.
- Client Only – Installs everything except the databases.

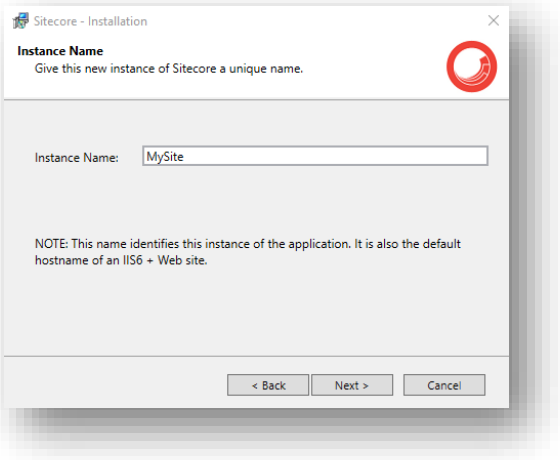
Click **Next**.

To install Sitecore WebDAV features, you must have the WebClient service on your computer. If the `WebClient` service is not available, the following message appears:



Disregard this message if you do not want to install WebDAV features. Click **Next**.

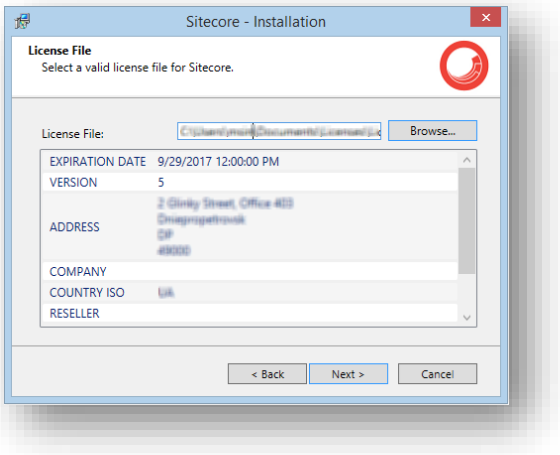
Instance Name



Give a unique name to this new instance of Sitecore. If you choose to create a new site at a later step, this name is also used as the hostname of an IIS web site.

License File

Note
If you selected the Database Only installation type, the **Installation Wizard** skips this step.

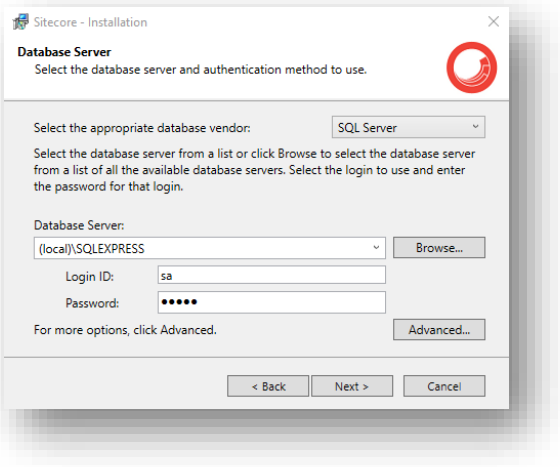


Click **Browse** to navigate to the license file that you want to use, and then click **Next**.

Database Server

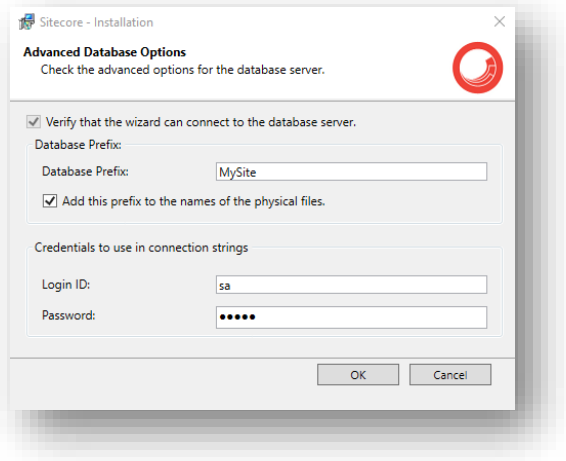
Select the database server.

If you select SQL Server, you see the following **Database Server** dialog box:



Select the database server from a list, or click **Browse** to see a list of all the database servers that are available. Enter the login ID (SQL server user name) and the corresponding password. The installation program uses these credentials to access the SQL server.

To set the database prefix options, click **Advanced** and the **Advanced Database Options** dialog box appears.



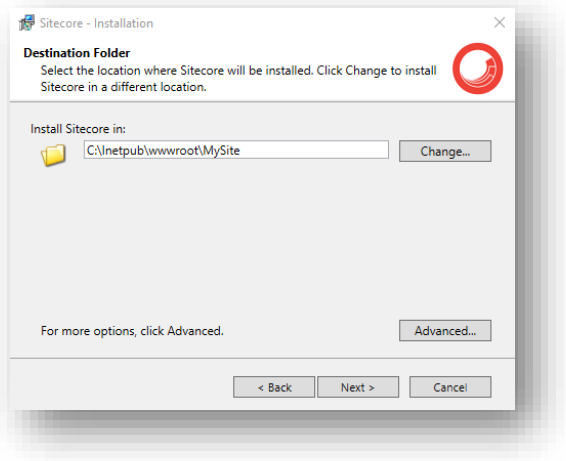
In the **Advanced Database Options** dialog box, specify the database prefix for this instance of Sitecore XP, and if you want the database prefix to be added to the names of the physical files, select the **Add this prefix to the names of the physical files** check box.

In the **Credentials to use in connection strings** section, enter the login ID (SQL server user name) and corresponding password. Sitecore applications will use these credentials to access the SQL server. Click **OK**.

Click **Next**.

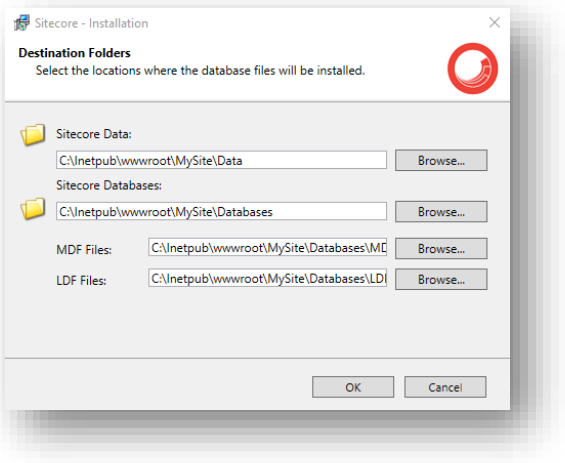
Note
If you selected the Database Only or Client Only installation, the **Credentials to use in connection strings** section is not shown.

Destination Folder



Select the folder where the Sitecore web site should be installed.

To change the database folder and the folder where various data files are stored — log files, packages, and so on — click **Advanced** and the **Destination Folders** dialog box appears:



Note
If you selected the Database Only installation, the **Sitecore Data** and **Sitecore Databases** fields are not shown.

If you selected the Client Only installation, the **Sitecore Databases** field is not shown.

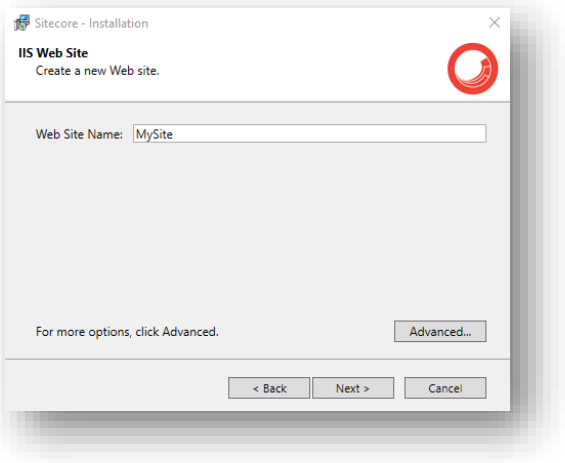
Enter new destination folders, and click **OK**.

Click **Next**.

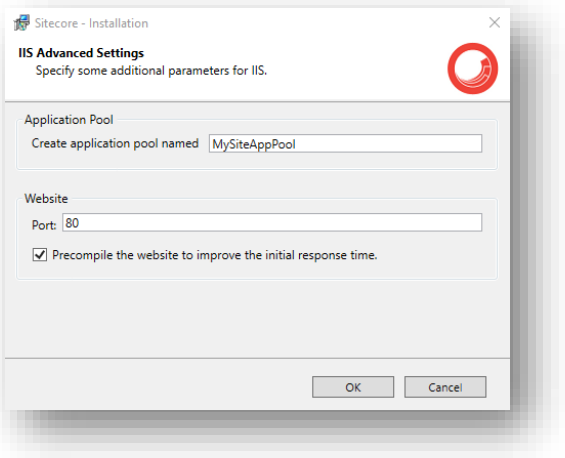
Note
Ensure that SQL server has the permissions to write to the destination folder(s).

IIS Web Site

Note
If you selected the Database Only installation, the IIS Web Site step is not shown.

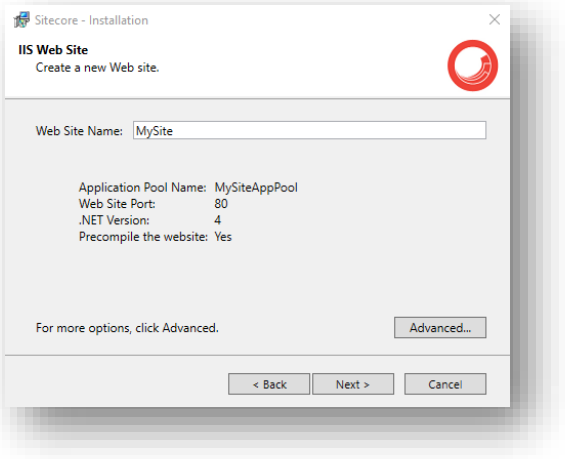


Enter the name of the new website. Click **Advanced** to edit a website port and create an Application Pool. Select the **Precompile the website to improve the initial response time** check box.



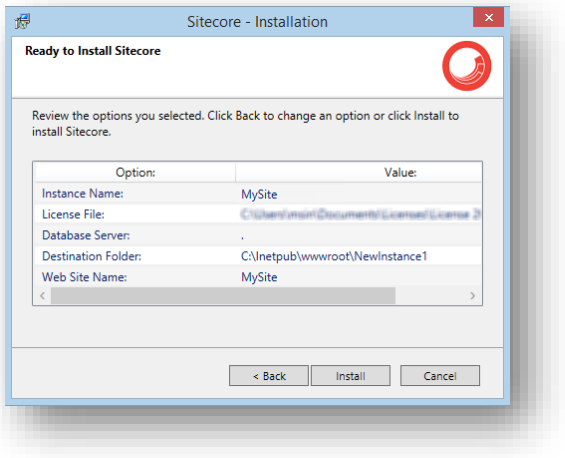
Click **OK**.

The **IIS Web Site** dialog box reflects the changes.



Click **Next**.

Ready to Install Sitecore



Review the options you selected. Click **Install** to install Sitecore XP.

A progress window informs you of the installation process.

Start the browser and enter the following URL:

`http://MySite/sitecore`

3.3 Troubleshooting

If there is an error during installation

If the installation program fails, a log file is automatically created. You can find the `SitecoreInstaller.log` file in the `\%Temp%\` folder.

Ensure that you save the log file, and if necessary, attach it to your support request.

Important
The log file is recreated during each installation attempt. If there are errors during installation, you must save and copy the log file to another location right after an unsuccessful installation. Otherwise, the file will be overwritten.

"Database '...' cannot be upgraded" error message while using SQL Server

When using SQL Server, the installer returns the "*Database '...' cannot be upgraded*" error message. To avoid this, the user account configured to run the SQL server application must have full access permissions to the website root folder.

Typically, the SQL server application runs in the context of the NETWORK SERVICE account.

For more information, see the section *File System Permissions for SQL Server Files*.

Chapter 4

Installing the Sitecore XP from the .ZIP File

This chapter describes how to install Sitecore XP 8.2 from the ZIP file of the website root.

This chapter contains the following sections:

- Installation Process Overview
- Configuring Folder and Registry Permissions
- Configuring the IIS
- Microsoft SQL Server Configuration
- Oracle Database Storage Setup
- Uninstalling Sitecore XP

4.1 Installation Process Overview

This section describes how to install Sitecore from a ZIP archive of the Sitecore XP site root.

You can download the Sitecore XP ZIP archive from the Sitecore Downloads page – https://dev.sitecore.net.

To install an instance of Sitecore:

1. Ensure that the ASP.NET feature is enabled for Internet Information Services (IIS).
2. Choose a parent directory path for the project, for example:

   `C:\InetPub\wwwroot\ProjectName`
3. Extract the `.zip` archive to create the following folders:

   ```
   C:\InetPub\wwwroot\ProjectName\Website
   C:\InetPub\wwwroot\ProjectName\Data
   C:\InetPub\wwwroot\ProjectName\Databases
   ```
4. Set permissions as described in the section *Configuring Folder and Registry Permissions*.
5. In the `Website\App_Config\Include` folder, enable the `DataFolder.config` file and change the following setting to the absolute location of the `\Data` folder.

   ```
   <sc.variable name="dataFolder">
         <patch:attribute name="value">/data</patch:attribute>
      </sc.variable>
   ```

   For example: `C:\InetPub\wwwroot\ProjectName\Data`
6. Copy the appropriate `license.xml` file to the `\Data` folder.
7. Attach the SQL databases from the `\Databases` folder to the SQL server.

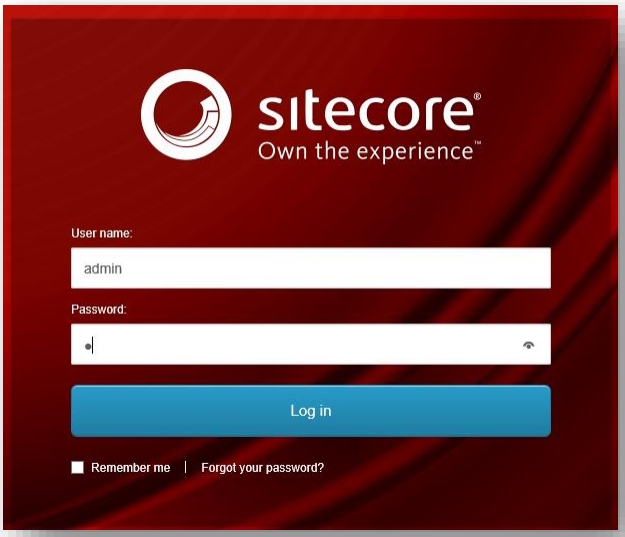
   For more information about working with SQL Server, see the section *Microsoft SQL Server Configuration*.
8. Edit the `ConnectionStrings.config` file located in the `\Website\App_Config\` folder.

   Enter a connection string for each database, for example:

   ```
   <add name="core" connectionString="user
   id= sql server user ;password= user password ;Data
   Source=_sqlserver_;Database=_core_database_name_" />
   ```
9. In the IIS, configure the website as described in the section *Configuring the IIS*.
10. Configure the Internet Explorer browser as described in the *Browser Configuration Reference* manual.
11. Start the browser and enter the following URL:

    `http://ProjectName/sitecore`

The Sitecore login screen appears:



The default password for the Admin user is "b".

4.2 Configuring Folder and Registry Permissions

This section describes the NTFS permissions for folders and files, as well as Windows registry permissions required by Sitecore XP to function properly. For practical advice on how to edit NTFS permissions and registry permissions, please refer to the Microsoft Developer Network website.

4.2.1 Updating the Security Settings for a Folder

To update the security settings for a folder:

1. In **Windows Explorer**, right-click the folder and select **Properties**.
2. In the **Properties** dialog box, click the **Security** tab.
3. In the **Group or user names** field, select a user or group and grant them the appropriate permissions in the **Permissions for** field.
4. In the **Security** tab, click **Advanced**.
5. In the **Advanced Security Settings** dialog box, select the **Replace permission entries** checkboxes.

4.2.2 File System Permissions for Anonymous Requests

Certain types of browser requests (generally for directory lookups and static resources) are executed with the permissions of the anonymous Internet user account. This account requires *Read* permission for all the files, folders, and subfolders under the `\Website` folder.

The following table contains the default name for this account in different versions of IIS:

IIS version	Default anonymous Internet user account name
7, 7.5, 8.0, 8.5	IUSR

If IIS is not configured to use the default IUSR account for processing anonymous requests or if Anonymous authentication is not enabled for the website, the previously mentioned permissions must be granted to the corresponding account(s).

4.2.3 File System Permissions for ASP.NET Requests

Sitecore XP executes requests for ASP.NET resources and all the .NET code running within the application with the permissions of the account configured as an identity for the website's application pool.

This account requires *Modify* permissions for all the files, folders, and subfolders under the `\Website` and `\Data` folders.

The following table contains the default account used for processing ASP.NET requests in the different versions of IIS:

IIS version	Default ASP.NET account name
7	NETWORK SERVICE
7.5	NETWORK SERVICE
8.0	NETWORK SERVICE
8.5	NETWORK SERVICE

IIS 7 and later lets you change the user account that is used for processing ASP.NET requests.

If you select a different user account to process the ASP.NET requests, you must also grant this account the *Modify* permissions.

Note
For more information about application pool identities and specifically about assigning rights to the AppPoolIdentity account, see the following article: http://learn.iis.net/page.aspx/624/application-pool-identities/

4.2.4 File System Permissions for System Folders

To be able to load .NET runtime and ASP.NET resources used for processing the ASP.NET requests, the worker process hosting the Sitecore XP application requires access to multiple system files and folders that are not distributed as a part of Sitecore XP, but are installed as a part of the Windows Operating System and the .NET framework.

Most of these permissions are granted by IIS to all ASP.NET applications, automatically making the application pool identity account a member of the *IIS_IUSRS* security group.

However, on certain environments it is necessary to manually grant permissions for the application pool identity to the following system locations:

Default location	Required permissions	Comments
%WINDIR%\temp\	Modify	
%WINDIR%\Globalization\	Modify	Required for registering custom languages by the .NET Framework correctly.
%PROGRAMDATA%\Microsoft\Crypto	Modify	Required for storing cryptographic keys used for encrypting/decrypting data.

Note
ASP.NET and/or IUSR account require the modify access rights to the `\temp` folder to install Sitecore XP.

These variables have the following default values:

Variable	Default value
%WINDIR%	`C:\Windows`
%PROGRAMDATA%	`C:\ProgramData for IIS 7 and later`

Note
For more information about built-in groups and accounts in IIS, refer to http://learn.iis.net/page.aspx/140/understanding-built-in-user-and-group-accounts-in-iis-7/

4.2.5 File System Permissions for SQL Server Files

If you are using SQL Server, the user account configured to run the SQL server application requires *Full access* to the `\Databases` folder containing MDF and LDF database files. Typically, the SQL Server application runs within the context of the *NETWORK SERVICE* account.

4.2.6 Windows Registry Permissions

Sitecore XP contains a built-in functionality that reads and updates the Windows performance counters that can be used for monitoring and troubleshooting the Sitecore application. This functionality requires access to Windows registry keys. This access can be granted by making the application pool identity a member of the built-in *Performance Monitor Users* group.

Note
If the required registry permissions are not granted, the "*Access to the registry key 'Global' is denied*" error will be registered in the Sitecore log files whenever the application attempts to access Windows performance counters.

To avoid this error, you must prevent Sitecore from updating the performance counters. To do that, in the `web.config` file, set the `Counters.Enabled` setting to *false*.

Adding an Account to a User Group

To add a Windows account to a user group, use the `net localgroup` command.

For example, to add the *NETWORK SERVICE* account to the *Performance Monitor Users* group, run the following command on the server hosting the Sitecore XP web application:

```
net localgroup "Performance Monitor Users" /add "NETWORK SERVICE"
```

An IIS reset is required for the changes to take effect.

For more information about the `net localgroup` command, refer to http://www.microsoft.com/resources/documentation/windows/xp/all/proddocs/en-us/net_localgroup.mspx

4.3 Configuring the IIS

The following sections describe how to configure different versions of IIS (versions 7, 7.5, 8 and 8.5) for Sitecore XP.

The following list shows which operating systems use which IIS versions:

- IIS 7.5 – Windows Server 2008 R2 SP3, Windows 7
- IIS 8 – Windows Server 2012, Windows 8, Windows 10
- IIS 8.5 – Windows Server 2012 R2, Windows 8.1
- IIS 10 – Windows 10, Windows Server 2016.

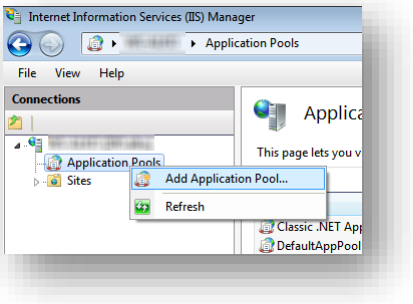
4.3.1 IIS 7, 7.5, 8, 8.5 and 10

This section contains the instructions for IIS 7, 7.5, 8, 8.5, and 10 for Sitecore XP.

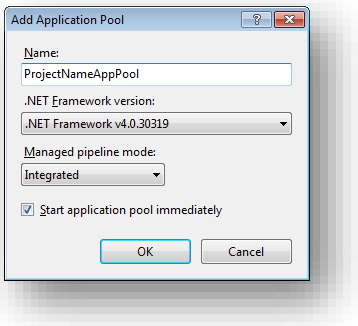
Create the Application Pool

To create the application pool for Sitecore on IIS 7, 7.5, 8, 8.5 and 10:

1. Start the **IIS Manager**.
2. In the **Connections** panel, right-click **Application Pools** and click **Add Application Pool**.



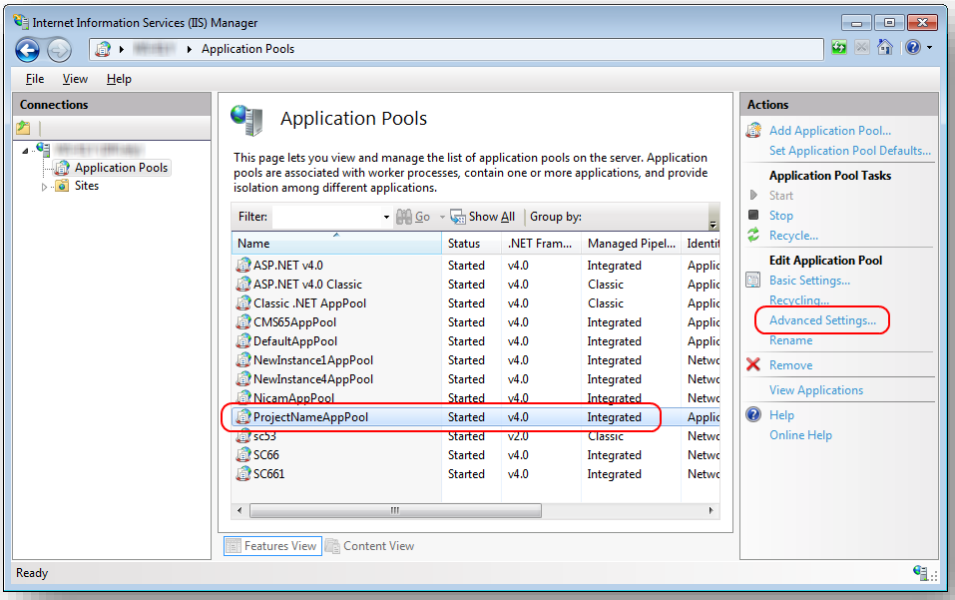
3. In the **Add Application Pool** dialog box, enter the following values:



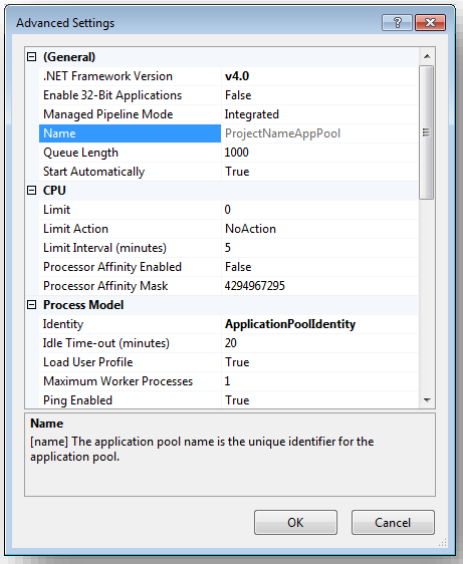
Field	Value
Name	ProjectNameAppPool
.NET Framework version	4.X.X
Managed pipeline mode	Integrated

4. Click **OK**.

5. In the **Application Pools** panel, click the application pool you have just created, and in the **Actions** panel, click **Advanced Settings**.



6. In the **Advanced Settings** dialog box:
   - Set the .NET Framework version to *v4.0*
   - Set the **Identity** setting to the built-in account called *ApplicationPoolIdentity* or *NetworkService*.
   - Ensure that the **Maximum Worker Processes** setting is set to *1*.
   - Set the **Load User Profile** setting to *True*

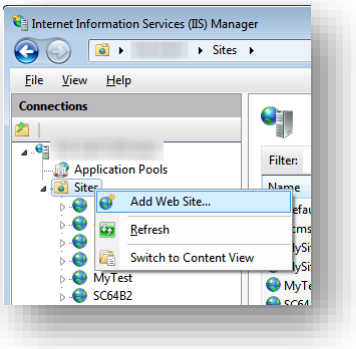


Create the Website

To configure the website:

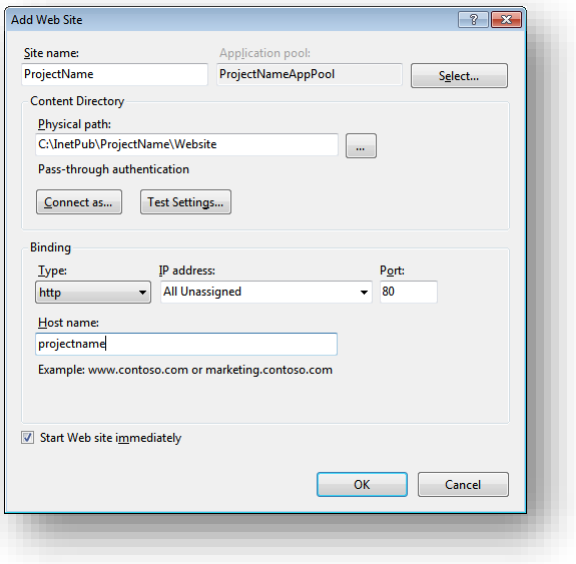
1. Start the **IIS Manager**.

2. In the **Connections** panel, expand *Sites*.



3. Right-click **Sites** and then click **Add Web Site**.
4. In the **Add Web Site** dialog box, fill in the required fields, for example:

Field	Value
Site name	ProjectName
Application pool	ProjectNameAppPool
Physical path	`C:\InetPub\ProjectName\Website`
Host name	Projectname



5. Click **OK**.
6. Open the hosts file – `C:\Windows\System32\drivers\etc\hosts` – and add the following line:

```
127.0.0.1       projectname
```

Now you can access your website by entering the *http://projectname* address in your browser.

Limit Access to Anonymous Users

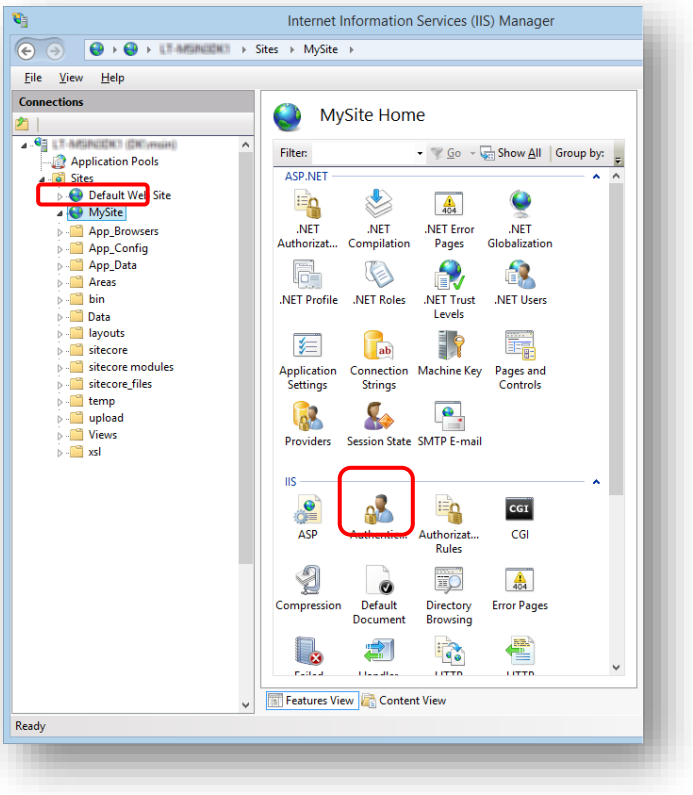
You must prevent anonymous users from accessing the following folders:

- `\App_Config`

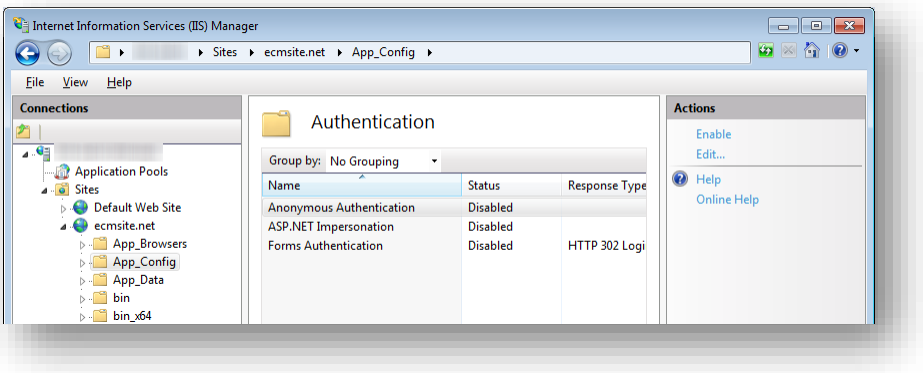
- `\sitecore\admin`
- `\sitecore\debug`
- `\sitecore\shell\WebService`

For example, to limit anonymous access to the `\App_Config` folder:

1. Open the IIS.
2. In the **Connections** panel, browse to your Sitecore site and select the `\App_Config` folder.
3. In the **App_Config Home** panel, double-click **Authentication** to configure the authentication settings for this folder.



4. In the **Authentication** panel, ensure that **Anonymous Authentication** is disabled.



To prevent anonymous users from accessing the `\sitecore\admin\`, `\sitecore\debug`, and `\sitecore\shell\WebService` folders, repeat steps 2-4 for each folder.

5. Restart IIS.

Note
If you have one or more processing, dedicated publishing, and/or indexing servers that does not handle requests, you must ensure that you use Application Initialization (IIS 8.0) or the appropriate module (IIS 7.5), so you can successfully start Sitecore after you recycle the application pool.

If you do not do this, Sitecore will not be launched and its application pool can be shut down due to inactivity.

4.4 Microsoft SQL Server Configuration

When you install Microsoft SQL Server, select **Mixed Mode** authentication (Windows Authentication and SQL Server Authentication).

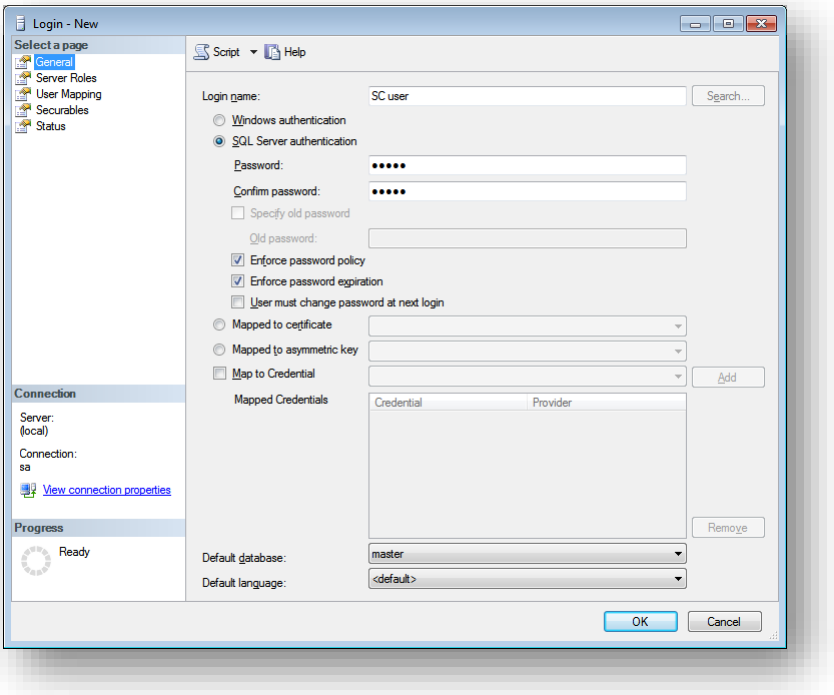
If you have already installed MS SQL and you need to change the security authentication mode, refer to the following: http://technet.microsoft.com/en-us/library/ms188670.aspx

Note
You can also use Windows authentication to connect to the SQL server. For more information on connecting to the SQL server, see http://msdn.microsoft.com/en-us/library/ff647396.aspx

4.4.1 Creating a Database Account for Sitecore XP Databases on SQL Server

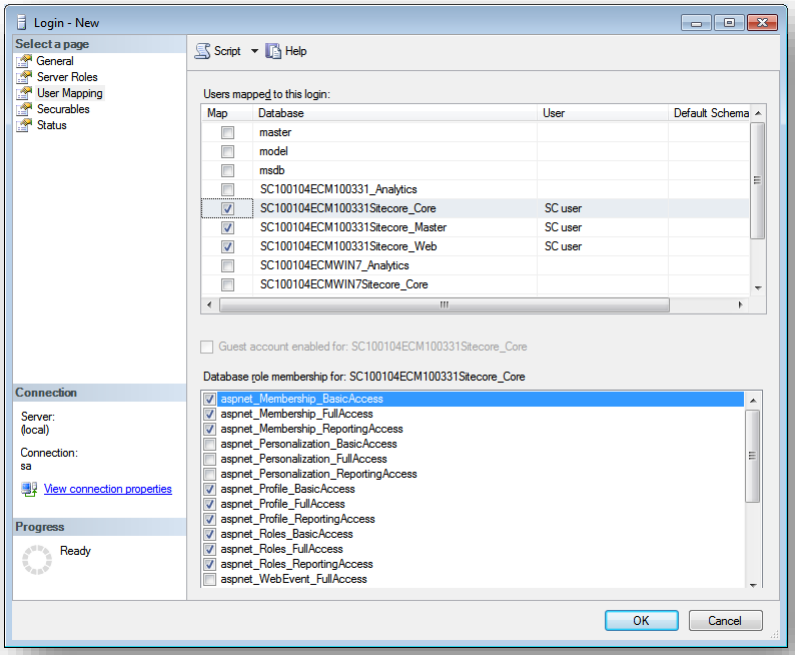
This section describes how to create a database account for Sitecore databases under SQL Server 2008 (or later) with the minimum required permissions.

1. Start **SQL Server Management Studio**.
2. Make sure the Sitecore databases are attached.
3. In **SQL Server Management Studio**, in the **Object Explorer** pane, expand the *Security* folder.
4. Right-click the *Logins* folder and click **New Login** to create a new login.
5. In the **Login – New** window, select the **General** page and specify the login name.



6. Select the **SQL Server authentication** option and clear the **User must change password at next login** check box (this option is not supported on some Windows operating systems).

7. Click the **User Mapping** page and in the **User mapped to this login** field, select the Sitecore databases.



8. In the **Database role membership for** field, select the following check boxes:

   For the Master and the Web databases:

   - db_datareader
   - db_datawriter
   - public

   For the Analytics database:

   - db_datareader
   - db_datawriter
   - public

   If you perform Historical Aggregation:

   1) In SQL Server Management Studio, connect to the SQL Server instance that will be used for Historical Aggregation.
   2) Find the database that will be used for Historical Aggregation.
   3) Open a New Query window.
   4) Execute the following:

      GRANT ALTER TO [Username];

      GRANT REFERENCES TO [Username];

   For the Core database:

   - db_datareader
   - db_datawriter
   - public
   - aspnet_Membership_BasicAccess

- aspnet_Membership_FullAccess
- aspnet_Membership_ReportingAccess
- aspnet_Profile_BasicAccess
- aspnet_Profile_FullAccess
- aspnet_Profile_ReportingAccess
- aspnet_Roles_BasicAccess
- aspnet_Roles_FullAccess
- aspnet_Roles_ReportingAccess

For the Sitecore.Sessions database:

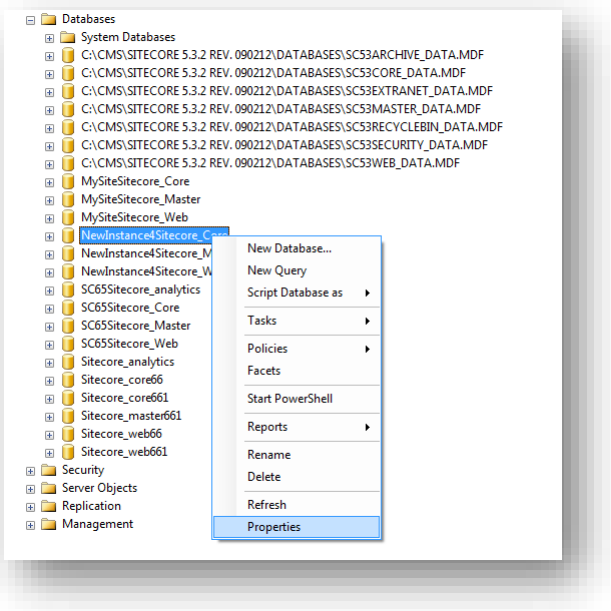
- db_datareader
- db_datawriter
- public

9. Click **OK** to create the user.

4.4.2 Granting Execute Rights to Stored Procedures

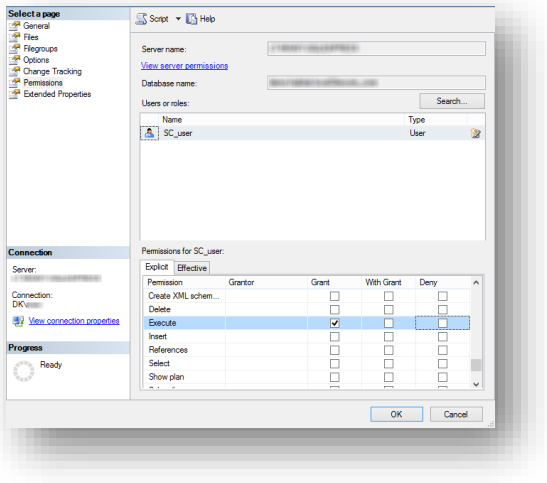
To grant execute rights to stored procedures:

1. In the **SQL Server Management Studio** window, expand Databases.
2. Right-click a Sitecore database, and then click **Properties.**



3. In the **Database Properties** window, click the **Permissions** section and select the user that you want to grant rights to.

4. In the **Users or roles** field, select the user and in the **Explicit permissions for** field, in the **Grant** column, give the user *Execute* permission.



Repeat this procedure for the other Sitecore databases.

Important
The analytics database for Sitecore xDB runs on MongoDB. For more information about configuring Sitecore xDB on MongoDB see the xDB Configuration Guide.

4.5 Oracle Database Storage Setup

This section describes how to set up the Oracle database storage for Sitecore CMS.

The Oracle database storage setup is only available when Sitecore XP is running in Sitecore Experience Management mode (CMS-only mode), which lets you run Sitecore XP without the Experience Database (xDB) enabled.

For more information about Experience Management, refer to the Sitecore Documentation site – https://doc.sitecore.net.

Database Server Preparations

To set up the Oracle database storage, the database server must run a supported version of an Oracle database instance that meets the following requirements:

- The Oracle instance must be configured in shared server mode.
- The Oracle instance must contain a tablespace for Sitecore, having at least 200MB of free hard disk space.
- Additional hard disk space will be needed as the Sitecore database grows.

To load Sitecore data into the Oracle database, to set up Oracle databases for Sitecore CMS, on the Oracle Database Server (or a workstation PC with the administration tools for the Oracle instance):

1. Download the `Sitecore 8.2 rev. 171121 (oracle dbs).zip` file from the Sitecore Downloads page and unzip the file.
2. In a text editor, open the `createdata.bat` file located in the `\Databases\Oracle` folder and find the following node:

```
if "%DATABASE%" == "" set DATABASE=sitecore
if "%PASSWORD%" == "" set PASSWORD=manager
if "%TBS%" == ""      set TBS=sitecore
if "%PREFIX%" == ""   set PREFIX=sc
```

User-level variables	Description
DATABASE	The name of the Oracle Database instance that will host Sitecore data (as set up on the PC).
TBS	Name of the tablespace for Sitecore data.
PASSWORD	Password for the system user of the Oracle instance.
PREFIX	Prefix to use for user schema names and passwords. (Default: sc)

3. Specify the user-level variables as required and save your changes.
4. Launch the `createdata.bat` file from the `\Databases\Oracle\` folder.

At this point, installation scripts set up the following 3 user schemas:

- %PREFIX%core
- %PREFIX%master
- %PREFIX%web

Website Server Setup

The following instructions describe a web server setup where the Sitecore CMS file system is hosted.

Note
When you install the Oracle Data Provider in a 64-bit environment, you must use Oracle Client for Windows x64.

To set up a web server:

1. Install Sitecore XP. If you are installing Sitecore XP from a ZIP archive (and not from an `.exe` file setup) you must configure the `dataFolder` setting and copy a valid `license.xml` file to the `\Data` folder before proceeding to the next steps.

   For more information, see the section *Installing the Sitecore XP from the .ZIP File.*

2. Back up the original `Web.config` file and rename the `Web.config.Oracle` file to `Web.config`.

3. Back up the original `Sitecore.config` file and rename the `Sitecore.config.Oracle` file to `Sitecore.config`.

4. Modify the connection strings in the `\App_Config\ConnectionStringsOracle.config` file to point to the valid Oracle instance.

   - The Data Source parameter must point to the local Oracle instance name as configured in the Oracle Net Manager.
   - User names and passwords must be adjusted to match those installed on the database server. The default usernames and passwords are:

Username	Password
sccore	sccore
scmaster	scmaster
scweb	scweb

   Configure IIS as you normally would for a Sitecore installation, and then run Sitecore.

4.6 Uninstalling Sitecore XP

Before uninstalling Sitecore:

- Delete the IIS Website or update it to reference a different directory, or stop IIS by executing the following command:
  - `net stop w3svc.`
- Remove the Sitecore file system.
- Remove any databases on remote or local servers that are used by this instance of Sitecore.

Chapter 5

Sitecore XP Post-Installation Steps

This chapter describes the recommended steps that you should take after installing Sitecore XP 8.2 rev. 180406 (Update-7).

This chapter contains the following sections:

- Content Expiration
- Windows Authentication for Sitecore
- Setting Microsoft SQL Server Database Properties
- Rebuild the Search Indexes and the Link Database
- Configure the Sitecore Experience Database (xDB)
- Deploy Marketing Definitions

5.1 Content Expiration

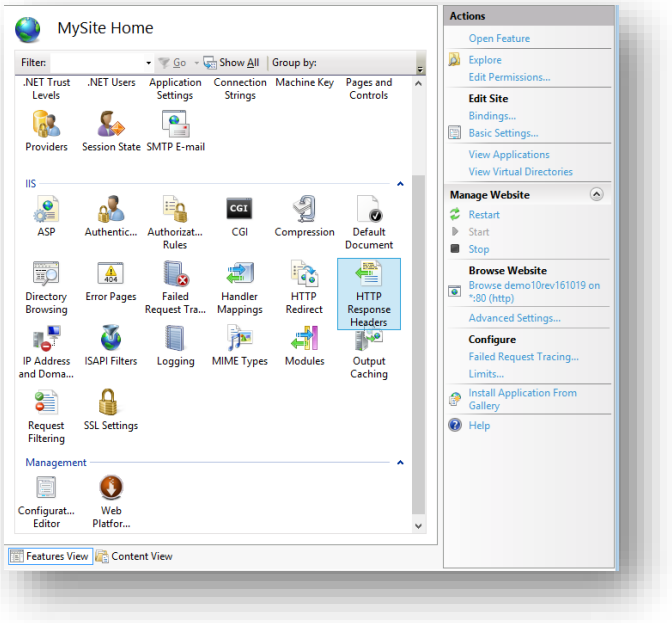
You set the IIS Expire Web content header to improve performance.

The Expire Web content header (located in common HTTP Response headers) is how IIS determines whether to return a new version of the requested web page if the request is made after the web page content has expired. IIS marks each web page before it is sent, using the settings that you provide for content expiration. The website visitor's browser translates the expiration mark.

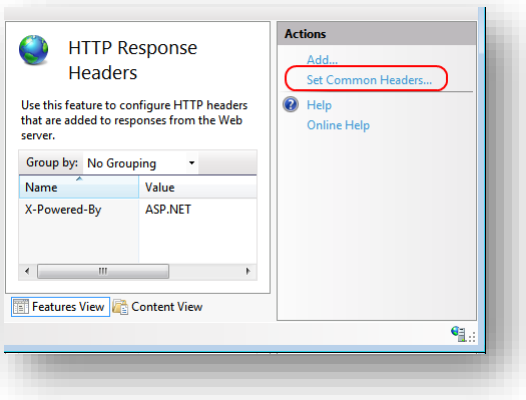
By setting Expire Web content to something other than *immediately*, you can reduce second-access load times by 50 to 70 percent. This setting does not affect dynamically-generated content.

To enable the Expire Web content header in IIS 7.x:

1. Launch IIS Manager.
2. Navigate to the site that you want to enable the Expire Web content header for.
3. In the **IIS** section, double click **HTTP Response Headers**.

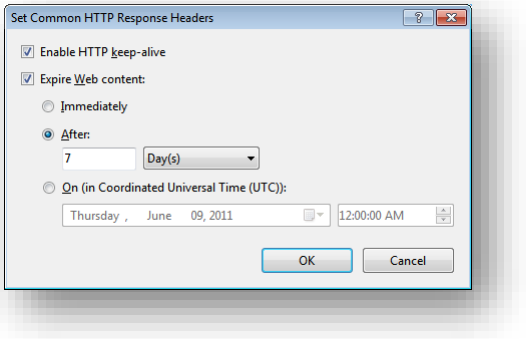


4. In the **Actions** panel, click **Set Common Headers...**



5. Select the **Expire Web content** check box.

6. Click **After,** and set the number of days to its default (7).

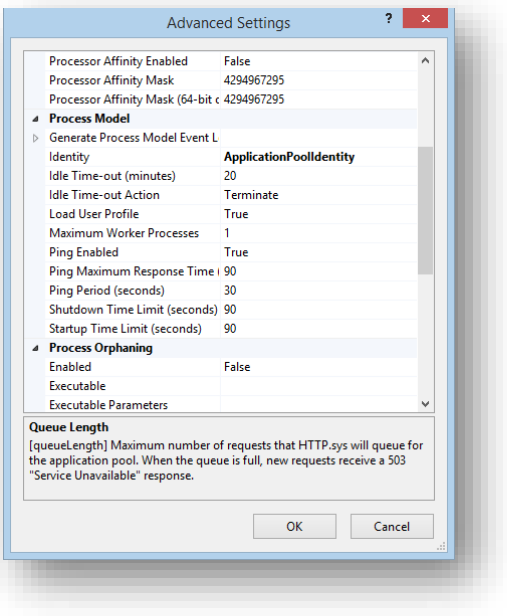


7. Click **OK**.

5.2 Windows Authentication for Sitecore

To configure Sitecore to use Windows Authentication for SQL connection (Windows Server 2008 R2 SP3) and effectively remove the user name and password from the `connectionStrings.config` file:

1. Open IIS and find the application pool that Sitecore is running under. Open **Advanced Settings** and in the **Identity** field, set the identity to the domain user.



2. In the SQL Server, register the domain user and grant the appropriate security permissions to the Sitecore databases for the domain user.

   For more information, see the section *Creating a Database Account for Sitecore XP Databases on SQL Server*.

3. On the computer that hosts Sitecore XP, add the domain user to the IIS_IUSRS group.

   Note
   You must adjust the permissions for the IIS_IUSRS group according to the instructions in the sections: *File System Permissions for ASP.NET Requests* and *File System Permissions for System Folders.*

4. Edit the `\App_Config\ConnectionStrings.config` file and replace the `user id` and `password` parameters with `trusted_connection=Yes`.

```
        <?xml version="1.0" encoding="utf-8"?>
        <connectionStrings>
        <add name="core" connectionString="Data
Source=.\sql2008;Database=Sandbox6 Core;Trusted Connection=Yes" />
        <add name="master" connectionString="Data
Source=.\sql2008;Database=Sandbox6_Master;Trusted_Connection=Yes" />
        <add name="web" connectionString="Data
Source=.\sql2008;Database=Sandbox6_Web;Trusted_Connection=Yes" />
        </connectionStrings>
```

5. Prepare your identity so that it can be used as a service account with the `aspnet_regiis.exe` file and the -ga switch.

6. To add your domain service account to the local *Performance Monitor Users* group, follow the instructions in the *Windows Registry Permissions* section.

Note
If you use the Sitecore Experience Database (xDB), the same configuration must be repeated for the Reporting database. The connection string for the Analytics database must be:

```
        <add name="reporting" connectionString="Data
        Source=<Data-Source>;Database=Sandbox6_Analytics;Trusted_Connection=Yes" />
```

5.3 Setting Microsoft SQL Server Database Properties

The following Microsoft SQL Server database properties, when configured correctly, can increase performance.

Compatibility Level

Compatibility Level affects SQL syntax and query parsing, and should have no impact on performance. Setting the Compatibility Level to a value of *SQL Server 2008(100)* takes advantage of new T-SQL features, which are used in many of the scripts/commands.

Auto Close

When MS SQL Server opens a database, resources are allocated to maintain that state. Memory for locks, buffers, security tokens, and so on are all assigned. These operations take time. The *Auto Close* property defines how these resources are handled. If it is set to *true* or *ON*, then when the last connection is closed, these resources are deallocated. If a new connection comes in within a short period of time ($1/10^{th}$ of a second or quicker), then all of those resources need to be activated again. Setting the *Auto Close* property to *false* or *OFF* prevents this from happening.

Auto Shrink

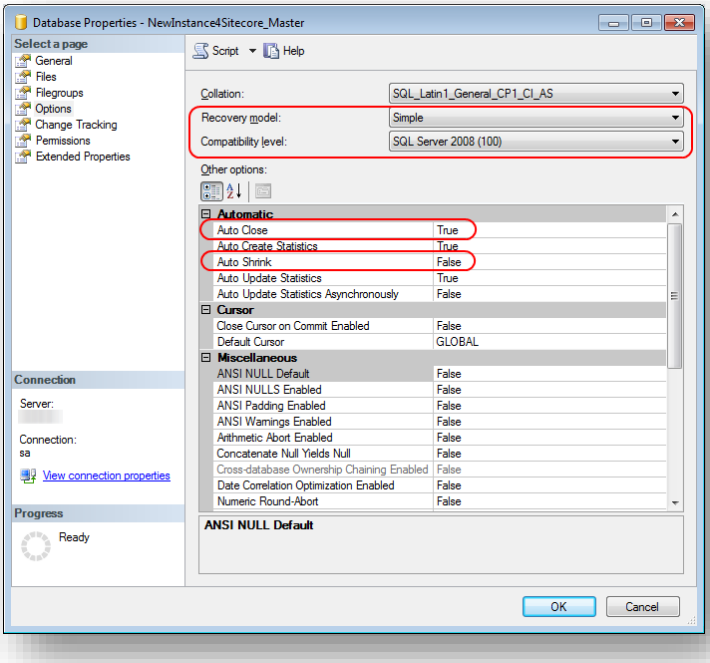
The Auto Shrink property uses many resources when it is called, and you have no control over when it is called. If you combine *Auto Shrink* with *Auto Growth*, you can get into a spiral of constantly growing and shrinking the database, taking valuable resources away from other database tasks as well as causing fragmentation issues. If a database or file requires a SHRINK command, it should be done via a script, a command, or a scheduled maintenance plan. Setting the *Auto Shrink* property to *false* or *OFF* disables this feature.

Recovery Model

In the Simple Recovery Model, MS SQL Server records a minimal amount of information in the transaction log. MS SQL Server truncates the transaction log whenever the transaction log becomes 70 percent full or the active portion of the transaction log exceeds the size that MS SQL Server can recover in the amount of time that is specified in the Recovery Interval server level configuration. Setting the Recovery Model to Simple has the lowest amount of overhead compared to *Full* and *Bulk-logged*, which is crucial to the performance requirements needed for the Sitecore XP databases.

To configure all these Microsoft SQL Server database properties to improve performance:

1. Open **MS SQL Server Management Studio**.
2. In the **Object explorer**, right click the **CMS Master** database and click **Properties**.
3. Click the **Options** page and set the following parameters:
    - In the **Compatibility Level** field, select *SQL Server 2008(100).*
    - Set the **Auto Close** property to *false*.
    - Set the **Auto Shrink** property to *false*.
    - In the **Recovery Model** field, select *Simple.*



4. Click **OK**.

Repeat these steps for the Sitecore Core and Web databases.

5.3.1 Warm up the Servers

To ensure that your Sitecore websites are available at all times, even after restarting a server, you should enable the IIS auto-start feature for the application pools on all the servers that you have configured.

For more information about the auto-start feature, see Microsoft's documentation.

5.4 Rebuild the Search Indexes and the Link Database

After you install Sitecore, you must rebuild the search indexes, and the Link database.

To rebuild all the indexes:

1. On the **Sitecore Launchpad**, click **Control Panel**, and in the **Indexing** section, click **Indexing manager**.
2. In the **Indexing Manager** dialog box, click **Select all**, and then click **Rebuild**.

To rebuild the Link database for the Master and Core databases:

1. On the **Sitecore Launchpad**, click **Control Panel**, and in the **Database** section, click **Rebuild Link Databases**.
2. Select the Master and Core databases and click **Rebuild**.

5.4.1 Cache Configuration

When Sitecore XP is installed, you must optimize caching to improve solution performance, reduce server load, and increase system capacity.

For more information, see the cache API changes topic for Sitecore 8.2 in the Sitecore Documentation Site – https://doc.sitecore.net/

For more information about cache configuration, see the Cache Configuration Reference document on the Sitecore Developer Network. – https://sdn.sitecore.net/

5.5 Configure the Sitecore Experience Database (xDB)

This section describes how to configure the Sitecore Experience Database (xDB).

Note
If you do not plan to use Sitecore xDB, you do not have to perform these steps.

There are three steps to configure Sitecore xDB:

- **Configure the reporting database** – attach the `Sitecore.Analytics.mdf` database to SQL Server. This database is used for reporting.
  - In the `ConnectionStrings.config` file, customize the following string:

```
<add name="reporting" connectionString="user id= username ;password= password ;Data Source=_server_;Database=Sitecore_Analytics" />
```

Note
You can create a database account with the minimum required permissions for the Analytics database as described in the section *Microsoft SQL Server Configuration*.

- **Install MongoDB** – download and install the latest version of MongoDB. For more information on how to install MongoDB, see http://docs.mongodb.org/manual/.
- **Configure Sitecore to use Mongo databases** – perform the actions outlined in the following procedure for each Mongo server.

To configure Sitecore to use Mongo databases:

1. In the `ConnectionStrings.config` file, specify the connection strings to your MongoDB server:

```
<add name="analytics" connectionString="mongodb://localhost/analytics" />
<add name="tracking.live" connectionString="mongodb://localhost/tracking live" />
<add name="tracking.history" connectionString="mongodb://localhost/tracking history" />
<add name="tracking.contact" connectionString="mongodb://localhost/tracking contact" />
```

2. In the `Sitecore.Analytics.config` file, in the relevant settings, update the `Analytics.EmailFromAddress` (for sending out reports as emails) attributes value.
3. In the `Sitecore.config` file, in the relevant settings, update the `MailServer` attributes value.
4. If you want to use Sitecore XP with Sitecore MVC, in the `\App_Config\Include` folder, rename the `Sitecore.MvcAnalytics.config.disabled` file to `Sitecore.MvcAnalytics.config`.
5. To limit the lookup activity to one server, in the `\App_Config\Include\Sitecore.Analytics.config` file, update the `Analytics.PerformLookup` setting.
6. To hash IP addresses before they are stored in xDB, in the `Sitecore.Analytics.config` file, update the `RedactIpAddress` setting.

   Important
   To ensure that hashing is secure, you must also change the default salt value in the `Sitecore.Analytics.config` file, in the `geoIpManager` section.

7. Restart Internet Information Services (IIS).

For more information about configuring Sitecore xDB to use with Sitecore XP, as well as the scalability of Sitecore xDB architecture, see the Sitecore Documentation site – https://doc.sitecore.net/.

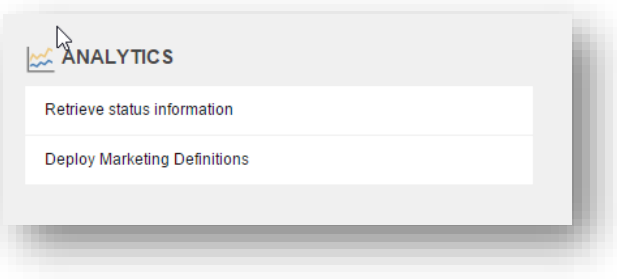
5.6 Deploy Marketing Definitions

If you want to use the Sitecore Experience Marketing functionality, you must deploy marketing definitions.

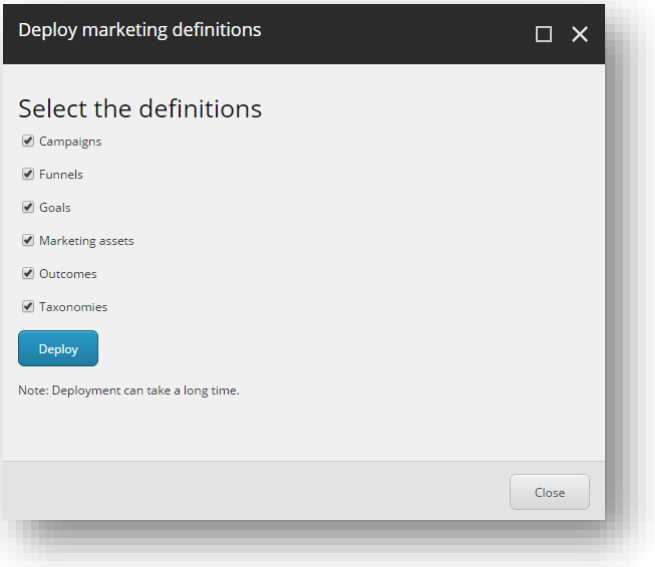
Note
If you do not plan to use Sitecore xDB, you do not have to perform these steps.

To deploy marketing definitions:

1. On the **Sitecore Launchpad**, click **Control Panel**, **Analytics**, and then click **Deploy Marketing Definitions**.



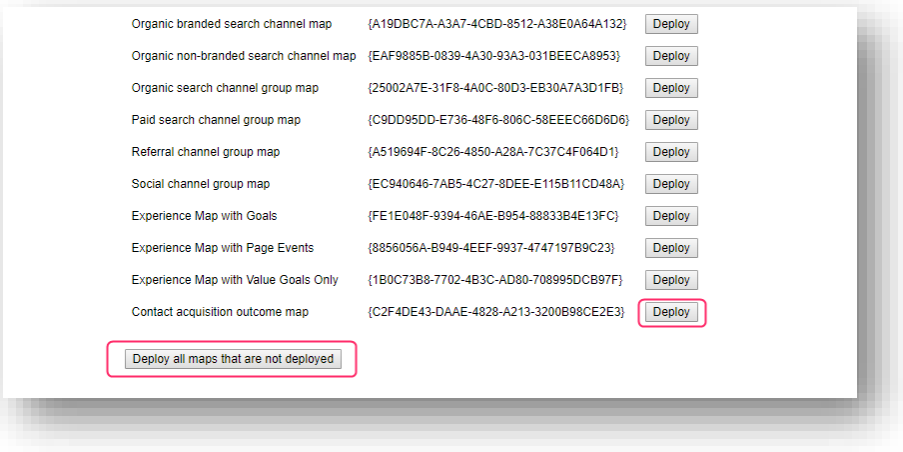
2. In the **Deploy marketing definitions** dialog box, select all the definitions and taxonomies and click **Deploy**.



5.7 Deploy Path Analyzer Maps

You must deploy the Path Analyzer maps before they can aggragate data.

To deploy the Path Analyzer maps :

1. Open the Path Analyzer's administration page: /sitecore/admin/pathanalyzer.aspx
2. To deploy all the maps, in the **Maps Manager** section, click **Deploy all maps that are not deployed**.

   Alternatively, to deploy individual maps, click **Deploy** next to the maps that you want to deploy.