



Active Content Manager

Installation and Configuration Guide

Version 24.3
December 4, 2024

Active Network, LLC
3400 N Central Expy | Suite 300 | Richardson TX 75080

Email support: ACMSupport@ActiveNetwork.com
Web site: <https://activenetwork.my.salesforce-sites.com/acm/>

Active Network™

© 2006 – 2024 Active Network, LLC, and/or its affiliates and licensors. All rights reserved. Trademarks and logos are the intellectual property of Active Network, LLC.

The content of this document is for information purposes only and is subject to change without notice. Active Network, LLC assumes no liability or responsibility for any error that may appear within this document. The material herein should not be construed as a commitment or warranty.

All other copyrights, trademarks, and tradenames are the property of their respective owners.

The Digital Millennium Copyright Act of 1998 provides recourse for copyright owners who believe that material appearing on the Internet infringes their rights under U.S. copyright law. For more information on the subject, see [Copyright policy](#).

CONFIDENTIAL

Active Network, LLC
3400 N Central Expy
Suite 300
Richardson TX 75080
1.888.543.7223

About Active Network, LLC

Active Network, LLC is a technology and media company that helps millions of people find and register for things to do and helps organizations increase participation in their activities and events.

For more information about Active Network™ products and services, please visit ActiveNetwork.com.

Table of Contents

Introduction to Active Content Manager	1
Technical Support	1
Deployment Planning	2
Hardware Requirements.....	3
Software Requirements.....	4
Active Content Manager Software	4
Server.....	4
Admin and Visitor Browser.....	4
Server Diagrams	5
Server Diagram of a Single Site setup.....	5
Installation Checklist.....	6
IIS and ASP.NET 4.7 Install.....	7
Installing IIS on Windows Server 2012 (IIS 8)	7
Installing IIS on Windows Server 2012 R2(IIS 8.5)	7
Installing IIS on Windows Server 2016 or 2019	7
Installing ASP.NET 4.7	7
DNS Configuration	8
FQDN (Fully Qualified Domain Name)	8
Server Host Name	8
Application Files Install	9
Installing the ACM Application Files.....	9
Create the ACM Directory.....	9
Copying the ACM application files	9
Custom Page Types.....	9
License Files Installation	10
Database Setup: SQL Server.....	11
SQL Server Database Setup	11
Database Connection String.....	13
Default	13
SQL Server.....	13
Saving Database Connection Information to the Registry.....	13
Web.config Configuration	16
Setting CustomErrors	17

Setting Maximum Request Settings.....	18
Configuring IIS.....	19
Configuring Maximum Digital Asset Upload Size.....	19
Configure Directory Security.....	21
Configuring permissions for Windows Server.....	21
ACM Testing and Configuration	22
Default Administrator Password	22
System-wide Administrator Setup	22
Site Administrator Setup.....	22
Support Contact Information.....	23
Register Page Types and HTML Editors	23
FQDN Setup.....	24
Health Monitor Install	25
Installation and Configuration	25
The Health Monitor package	25
Installation	25
Enable Health Monitor Auto Start.....	26
Configuration	26
Uninstalling Health Monitor.....	26
Appendix A: IIS Bindings.....	27
Configuring IIS Bindings.....	27
Appendix B: Redirects.....	30
Authority Redirects	30
301 Redirects.....	30
FQDN Mapping	31
Appendix C: Multiple Sites on a Single Install.....	32
IIS Configuration.....	32
FQDN Mapping Method.....	32
Default.aspx Method	33
Appendix D: Search Page	35
How the Search Page Works	35
Static Export of SearchSite files	35
Installing Adobe's iFilter	36
Windows Search	36

Installation on Windows Server	36
Configuring Windows Search Service.....	36
Configuring the ACM Search Page.....	37
Removing Specific Pages from the Search.....	37
Appendix E: Static HTML Export.....	38
Definitions used in this section:.....	38
Advantages and Disadvantages of Static Sites	39
Advantages of a Static/Dynamic site setup	39
Disadvantages of a Static/Dynamic site setup	39
Configure the Site to Export and FTP Static HTML Pages	39
Schedule the Export to Automatically Occur.....	40
Configuring Server Side Includes (Static Server Setup).....	40
Configuring Server Side Includes (Dynamic Server Setup)	41
Appendix F: Shared Folders.....	42
Public vs. Private Shared Folders.....	42
Public Folders	42
Private Folders	42
Storing Shared Folders in a Non-Default Location.....	43
Windows Server 2008:.....	43
Windows Server 2008:.....	44
Appendix G: Multi-Server Setup.....	45
The Concept of Multiple Servers in ACM.....	45
Domains	46
Setting up State Server Service	46
Setting up the ACM Cache Invalidation Service	47
Configuring Shared Folders.....	48
ACM Server Configuration	49
Configure Additional Services	50
Windows SearchService.....	50
Health Monitor	50
Appendix H: Log File Locations	51
IIS Logs	51
Windows Server 2008:.....	51
ACM System Event Logs.....	51

Windows Event Logs.....	52
FileNotFound Logs.....	52
Appendix I: Backup and Restore Procedures.....	53
Backup Schedules and Data Retention	53
Backing up the Database	53
Backing up Shared Folders.....	53
Backing up ACM Application Files.....	54
Backing up the IIS Configuration.....	54
IIS 7.0, IIS 7.5:.....	54
Restoring the ACM Database	54
Restoring the ACM Application Files	55
Restoring the Other Services	55

Introduction to Active Content Manager

This guide provides detailed instructions for installing Active Content Manager (ACM) on a server to host one or more sites.

Note The audience for this guide is IT staff and/or system administrators who will be installing, configuring, and maintaining ACM. This guide assumes that you are skilled at configuring Microsoft Servers (Windows, IIS, SQL Server, etc.), and have a thorough understanding of DNS and TCP/IP.

Read through this document completely before installing Active Content Manager.

Installing and Configuring ACM involves the following high-level steps:

- Conduct deployment planning
- Review Hardware and Software Requirements
- Deploy the ACM Application Files
- Configure Microsoft SQL Server
- Configure IIS
- Configure Directory Security for ASP.NET
- Configure ACM through web.config
- Configure your DNS
- Test Correct Operations
- Configure Windows Search
- Install and Configure Health Monitor

Once ACM is operating correctly, remember to plan:

- Backup and Restore Procedures

Technical Support

Technical support is available through the online support center at:

- <https://activenetwork.my.salesforce-sites.com/acm/>

In order to use this site, you must have a valid user name and password. If you're unsure, please try the Forgot Password link to check for an existing user name.

Email support: ACMSupport@ActiveNetwork.com

Deployment Planning

Active Content Manager is a flexible enterprise software system that can service the needs of small/simple installations and very complex and highly scalable installations:

- In a small deployment, ACM and all supporting software will be installed on a single computer; this would be appropriate for small or departmental websites. This deployment scenario offers little in the way of scalability, redundancy, or survivability¹.
- In a large deployment, ACM can be installed in a web-farm scenario with multiple application servers connected to a SQL Server Cluster. This would be appropriate to support large site, many small sites, and/or high-traffic websites. This deployment scenario offers very high scalability, redundancy, and survivability.

It is important that you understand your requirements and plan the appropriate deployment environment for your needs. Some items to consider:

- How many websites will you be hosting?
- How much traffic will the websites get?
- Are you deploying internet, intranet or extranet sites? How many of each?
- How many authenticated viewers will you have?
- How many content contributors will you have?

It is beyond the scope of this guide to provide help with deployment planning. For detailed analysis of deployment options, please contact your Active Account Executive who will arrange an analyst to work with your organization.

¹ Survivability is the ability of the system to survive in the face of catastrophic (and therefore infrequent) failure. For example, can the system continue to service requests in the event of a disk-array failing, a computer ceasing to function, or in the event of a power outage?

Hardware Requirements

Active Content Manager can be installed in a variety of configurations depending upon load and scalability needs. In the simplest configuration, the application and all supporting software is installed on a single computer. In more demanding environments, the database can be moved to a separate server (and optionally in a cluster) and multiple application servers can be used.

We recommend working with your Active Content Manager Consultant to determine the hardware requirements for your organization. However, the ACM Hardware and Software Requirements document provides server specifications that will be sufficient for most ACM installations. If your site has high traffic, you may wish to consider multiple servers in a web farm.

Software Requirements

Active Content Manager Software

Before proceeding with installation, make sure you have received the following files:

Requirements	Specifications
ACM_X_X_X_X.zip	Standard ACM release build. Contact the ACM Upgrades team for information on the password to extract the file.
Licenses.zip	Standard ACM page type licenses

Server

Please refer to the ACM Hardware and Software Requirements document for ACM-specific installation requirements. Read the documentation accompanying each prerequisite software product for additional requirements. Before installing any prerequisite software, it is highly recommended that you read Microsoft documentation about how to ensure the security of your platform²:

Admin and Visitor Browser

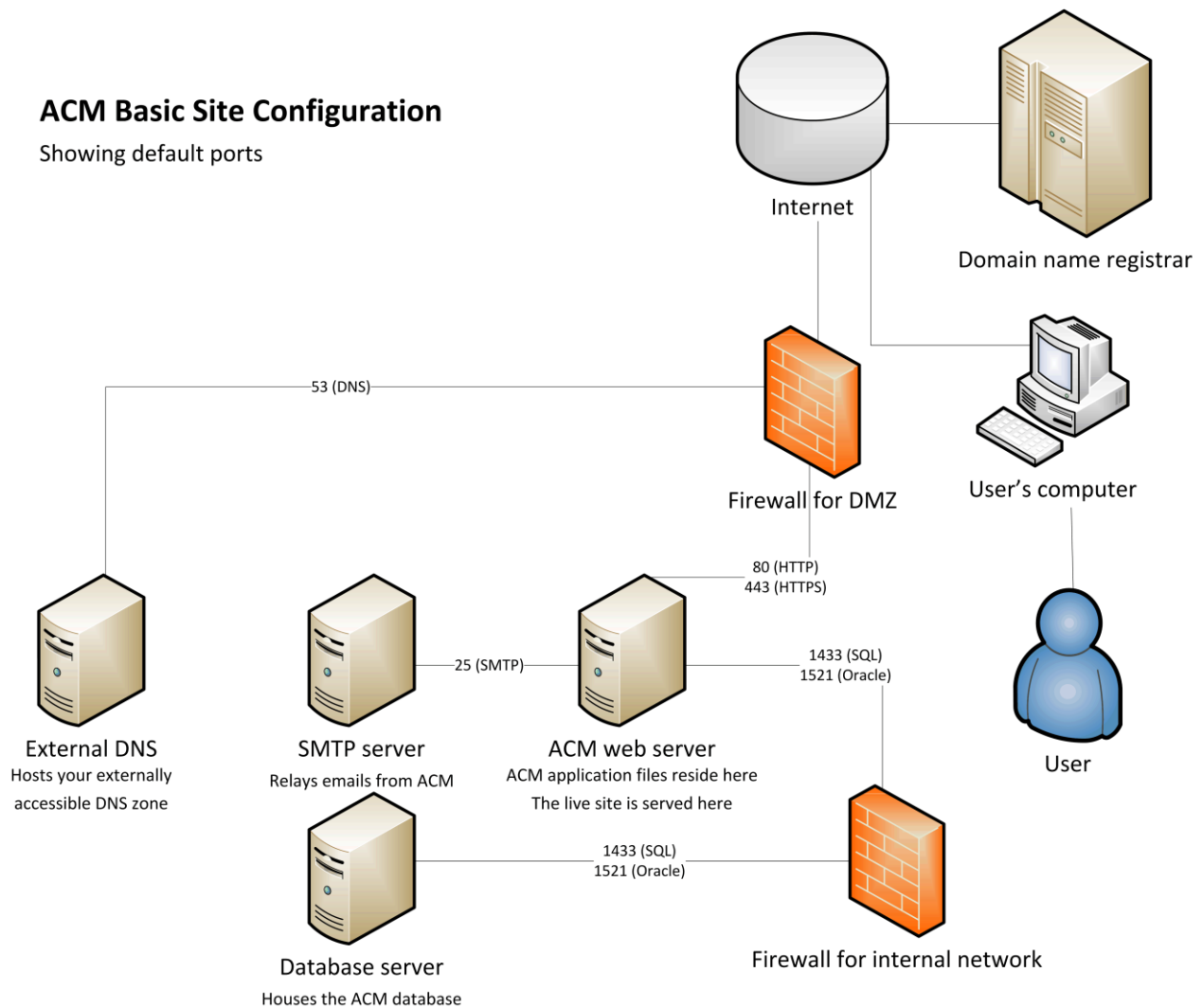
Please refer to the ACM Hardware and Software Requirements document.

² <http://www.microsoft.com/security/>

Server Diagrams

Server Diagram of a Single Site setup

This is a diagram of a simple single web server setup for an externally accessible site. Each customer's environment may be quite different, but this is the core part of the ACM architecture.



Installation Checklist

This checklist can be used as a guide for someone who is familiar with the installation process to follow. It is recommended to follow the full instructions to ensure nothing is missed during an installation. Any small item missed can result in the site not coming up.

1. Install IIS and ASP.NET.
2. Configure DNS for the site.
3. Extract ACM application files.
4. Copy Install files into the CMSroot.
5. Obtain license files and apply to CMSroot.
6. Set up Database.
7. Configure Database Connection in ACM.
8. Configure web.config file.
9. Apply directory permissions.
10. Configure the site in IIS.
11. Test and configure ACM.
12. Install and configure Health Monitor service.
13. Perform necessary Appendices.

IIS and ASP.NET 4.7 Install

Installing IIS on Windows Server 2012 (IIS 8)

Follow these instructions to install IIS, if your server is running Windows Server 2012, and IIS is not already installed. Microsoft has a number of resources for installing IIS that can be found at:

- <https://docs.microsoft.com/en-us/iis/get-started/whats-new-in-iis-8/installing-iis-8-on-windows-server-2012>

Installing IIS on Windows Server 2012 R2(IIS 8.5)

Follow these instructions to install IIS, if your server is running Windows Server 2012 R2, and IIS is not already installed. Microsoft has a number of resources for installing IIS that can be found at:

- <https://docs.microsoft.com/en-us/iis/install/installing-iis-85/installing-iis-85-on-windows-server-2012-r2>

Installing IIS on Windows Server 2016 or 2019

Follow these instructions to install IIS, if your server is running Windows Server 2012, and IIS is not already installed. Microsoft has a number of resources for installing IIS that can be found at:

- <https://docs.microsoft.com/en-us/iis/>

Installing ASP.NET 4.7

ACM uses ASP.NET Framework version .NET 4.7. You must download and install .NET 4.7 on the server running as the ACM Application server:

- <https://www.microsoft.com/en-us/download/details.aspx?id=56116>

DNS Configuration

The URL to access ACM will be the same as the one used to browse the site as a non-user, unless the site is using the Static Export functionality. It is important to choose the method which will be used to browse ACM.

FQDN (Fully Qualified Domain Name)

Selecting a fully qualified domain name is recommended. This will require you to register a DNS name for this site or configure your existing FQDN to point to the server hosting the ACM site once the site is finished implementing. If only used internally, the DNS zone may be configured on the internal network only. Consult your DNS administrator for more information on configuring this.

Server Host Name

You should be able to access the server using the host name of the server (the computer name of the server). For example, <http://myACMServer.active.com/>.

Using the server host name setting requires minimal configuration (from DNS) and may be a good interim solution until an FQDN is created or until the site is fully implemented and ready to go live.

Application Files Install

Installing the ACM Application Files

Installing the application files consists of creating a directory for the root of the site and the ACM application files. There are no changes to the registry required.

It is recommended that you close all other programs before you begin the ACM installation process.

Create the ACM Directory

Create a directory that will be used to install the ACM application files. For example, **C:\ACM\Site**. For the purposes of this document, we'll refer to the ACM Directory as **CMSroot**:

- You can use any local drive for the ACM application files directory.
- The ACM does not rely on the name used for the application files. However, this path will be used when configuring IIS.

The recommended structure of the ACM directory is:

C:\ACM\Backup – contains backups of the site before an upgrade

C:\ACM\Build Files – contains downloaded application files

C:\ACM\Site – contains the application files (the root of the ACM install)

Copying the ACM application files

Unzip **ACM_X_X_X_X.zip** and copy all of the files and folders within the **\Install files** folder into the **CMSroot** folder.

Custom Page Types

If you have any custom page types created by Active or through the SDK, these page type files are version specific, and must be compiled in the same version as this install of ACM. If the page type was created by Active, contact the ACM Support team to get the newer version. If the page type is managed by your organization, contact the developer for this page type to recompile the page type for this specific version of ACM.

Place the .dll of the page type(s) in the **CMSroot\bin** folder. Place the page type folder in the **CMSroot\CM\WebUI\PageTypes** along with the other page type folders.

License Files Installation

You will receive a separate .zip file of standard ACM licenses (e.g. Licenses.zip). If unsure, contact your Account Manager or the ACM Support Team to inquire about obtaining the license files. Proceed as follows:

1. Unzip the license file (The extracted files should contain a number of .lic files.
2. Copy all files (.lic format) into the CMSroot\bin folder of your ACM.

Database Setup: SQL Server

Configuring databases is beyond the scope of this guide. Consult Microsoft for more information on these processes.

It is also a good idea to set up a maintenance schedule to regularly back up, re-index and run other diagnostics on the databases. Consult your database administrator for more information on that.

SQL Server Database Setup

ACM requires two database logins: mcw and CM. The mcw login is the user which ACM uses to pull and manipulate data in the database. The CM user is the owner of all the objects in the database. The login names should not change; however, the passwords can be updated.

Note SQL Server *must* allow Mixed Mode Authentication (SQL Server and Windows Authentication mode).

Note The instructions on how to run queries are beyond the scope of this document. Please refer to Microsoft documentation for more details.

1. Open CMSroot\DatabaseScripts\IronPoint.Install.sql in a text editor.
2. Find and Replace IronpointCM with the desired database name.
3. Replace C:\Program Files\Microsoft SQL Server\MSSQL\Data\ with the desired physical location of the .mdf and .ldf files on the database server.
4. Locate this line and replace iggy with the desired password for the mcw database login:

```
exec sp_addlogin N'mcw', 'iggy', @logindb, @loginlang
```

Note SQL Server 2005 and higher by default, requires strict password complexity, which should be considered when creating this password.

5. Locate this line and replace the second CM with the desired password for the CM database login:

```
exec sp_addlogin N'CM', 'CM', @logindb, @loginlang
```

Note SQL Server 2005 and higher by default, requires strict password complexity, which should be considered when creating this password.

6. Open SQL Server Management Studio as a System Administrator.
7. Run the modified query from IronPoint.Install.sql above.
8. Once the query has run successfully, the ACM database has been created. If any errors occur, such as the SQL login passwords are not complex enough for the password policies, delete the database, fix the issues in the script, and run the script again.

Note It is recommended to disable password expiration for CM and mcw as the configuration in ACM must be changed if these passwords change.

Database Connection String

Default

By default, Active Content Manager is configured to connect to a SQL Server database running on the default instance of SQL Server on the same computer as the application. If your SQL Server is on a different computer, if you are using a named instance of SQL Server, or if you are using an Oracle Database, then you must follow the appropriate steps below to modify the data source identifier used by the ACM application to connect to the database.

SQL Server

Follow these instructions if ACM is connecting to a SQL Server database.

To connect ACM to a SQL Server:

1. Open **CMSroot\bin\IronPoint.DataAccess.dll.config** in a text editor.
2. In the section of the config file named **dataSources**, locate the **dataSource** element with the name attribute **"SQL"**.
3. The **connectionString** attribute configures how ACM connects to SQL Server. Modify the values of the parameters in the connection string to configure which database ACM looks at. There are four settings you can adjust in the **connectionString**:
 - a. **Data Source**: This is the host name or IP address of your SQL Server instance. Port number is also specified here using a comma to denote the port (e.g. 10.1.216.225,2433 for port 2433). If the SQL instance is a named instance, the name is denoted by a backslash (e.g. 10.1.216.225\SQLEXPRESS for a named instance called SQLEXPRESS).
 - b. **Initial Catalog**: This is the database name within the specified SQL Server instance.
 - c. **User ID**: This is the user that ACM will use to connect to SQL Server.
 - d. **Password**: This is the password of the User ID above.

Note You can also configure your database connection through a separate application. This application encrypts the **IronPoint.DataAccess.dll.config** file and writes the database connection information to the registry. You can then manage this information from within ACM. See the "Save Database Connection Information to the Registry" section for details.

Saving Database Connection Information to the Registry

This is an optional process. You can use the ACM Configuration tool to input the database connection information instead of using the **IronPoint.DataAccess.dll.config**. This method

also allows you to store the connection information in the registry rather than leaving it in the config file.

1. Open **CMSroot\bin\DataAccessSetupApp.exe** with the **Run as Administrator** option to launch the ACM Configuration Tool.

The screenshot shows the 'Active CM Configuration Tool' dialog box. It has a title bar with a minimize button, a maximize button, and a close button. The dialog is divided into three main sections: 'Installation', 'Storage Location', and 'Data Source Information'.
- The 'Installation' section contains a 'Name:' text box, a 'Root Dir:' text box, and a 'Browse ...' button.
- The 'Storage Location' section contains two radio buttons: 'Registry (Secure)' and 'IronPoint.DataAccess.dll.config'. The 'IronPoint.DataAccess.dll.config' option is selected.
- The 'Data Source Information' section contains two radio buttons: 'SQL Server' and 'Oracle'. The 'SQL Server' option is selected. Below these are four text boxes labeled 'Server:', 'Database Name:', 'Username:', and 'Password:'.
At the bottom right of the dialog is an 'Advanced >>' button. At the very bottom are 'Save' and 'Close' buttons.

2. Enter a name value for this ACM installation.
3. Click the **Browse** button to browse to and select the root directory of your ACM installation.
4. Select the **Registry (Secure)** option to write the connection information to the registry, or **IronPoint.DataAccess.dll.config** to write to the config file.

Simple Mode

1. Select SQL Server.
2. Fill in the other fields
 - a. **Server:** Host name or IP address of the database server (SQL Server only)

- b. **Database Name:** Name of the database
 - c. **Username:** The user that will connect to the database
 - d. **Password:** Password of the user that will connect to the database
3. Click Save to write the changes to either the registry or config file.

Advanced Mode

1. In the **Provider** field, type “SqlClient”
2. In the **Connection String** field use the Connection string found in CMSroot\bin\IronPoint.DataAccess.dll.config. For example:

```
Data Source=Pollock;Initial Catalog=IronPointCM71;User  
ID=mcw;Password=iggy
```

3. In the **Data Ops File Mask** field, type “SqlClient.CM.Commands*.config”
4. If there are any parameter name prefixes associated with the SQL files, type the prefix in the **Param Name Prefix** field; otherwise, you can leave it blank.
5. Click **Save** to write the changes to either the registry or config file.

Web.config Configuration

Some application-specific settings can be configured through the **CMSroot\web.config** file. The web.config variables can be edited through a text editor. Additionally, a limited set of variables in the web.config file can be edited through ACM: **System > Click Configuration Files > web.config** tab.

Most settings can be left with the default setting. Most installs need only configure the SMTP server, but here is a list of the ACM application specific settings that can be configured for further tweaking and configuration.

KEY	DESCRIPTION
PhysicalPathToSharedFolder	The physical path to the shared folder. This is the folder that contains all data that is not contained in the database. For example: C:\ACM\Sites. See Appendix F: Shared Folders for more information.
VirtualPathToSharedFolder	The virtual path to the shared folder. For example: CMSroot\Sites
PhysicalPathToPrivateSharedFolderLocal	The physical path to the private shared folder stores SearchSite files only. This setting only takes effect with the Force Export to Local Folder Search Export setting is enabled and is used specifically for multi-server environment, to export SearchSite files locally on each server. See Appendix G: Multi-Server Setup for more information.
CustomErrors	Determines if the ACM should show detailed error messages to users. Default value: RemoteOnly
Compilation	Determines if the application is set to ASP.NET "debug" mode. Recommended value: false:
Trace	Determines if the application should use ASP.NET tracing.
SessionState	Determines configuration settings for session management. Can be in-process for single application servers, or out-of-process for web farms.
Httpruntime	Sets the max size of an ASP.NET request and the timeout of an ASP.NET.
AsyncKeepAliveEnabled	False
AsyncJobServerEnabled	True
AsyncJobInterval	1
AsyncPollingInterval	1
AsyncJobRestartInterval	60
AsyncJobRemovalInterval	1440
SMTP_ServerHost	The SMTP server host to use for sending email from the system. For example: smtp.fusemail.com. Default value: none.
SMTP_ServerPort	The SMTP server port to use for sending email. Default value: 2500
SMTP_UserName	The SMTP user name to use to authenticate against the SMTP server. Default value: none.

SMTP_Password	The SMTP password to use to authenticate against the SMTP server. Default value: none.
SMTP_SAS	2
SMTP_SSLProtocol	0 - None, 240 - Default, 12 - SSL2, 48 - SSL3, 192 - TLS
DoDatabaseValidation	Determines if the application should perform a database validation check at application start. Default value: False
DoAssetAuthorizationCheck	Determines if the application should perform Digital Asset authorization checks based on the asset belonging to pages the user is able to see. This setting is only required if digital assets access must be controlled. Default value: False
UseRadEditor	Determines if the application with use the Telerik r.a.d. Editor for WYSIWIG content. If False, a normal text editor is used. Default value: True
RequestHistory	Sets the number of requests to record in logging: Default value: 5
EnableCaching	Determines if the application should enable caching. Default value: True
EnableMultiServerCache	Determines if the application instance will support a multi-server cache configuration. This setting is only required if multiple application servers are used. Default value: False
ShowQuickEditIcon	Determines if the application will show the QuickEdit icon when editing content. Default value: True
UseWindowsEventLog	Provides for the option to record ACM system events in the Windows Event Log. This can make supporting multiple instances of the ACM application on a single server more convenient. Default value: False
SessionTimeoutAnonymous	Provides the option to set the anonymous session timeout value in minutes. Default value: 60
SessionTimeoutAuthenticated	Provides the option to set the authenticated session timeout value in minutes. Default value: 60
SessionTimeoutAnonymousKeepAliveInterval	Provides the option to send an Ajax call to keep the anonymous session alive. If this value is less than the SessionTimeoutAnonymous value, the session will be preserved until the user closes their browser or browses off the site. Default value: 30
SessionTimeoutAuthenticatedKeepAliveInterval	Provides the option to send an Ajax call to keep the authenticated session alive. If this value is less than the SessionTimeoutAuthenticated value, the session will be preserved until the user closes their browser or browses off the site. Default value: 30

Note You can encrypt the information within the web.config file as it contains password information for your SMTP server. To do so, log on to ACM > **System** > **System Information** > expand the **WEB.CONFIG SETTINGS** section > click **Encrypt** or **Decrypt** to perform those operations on the web.config file.

Setting CustomErrors

This setting controls whether or not to show the full ASP.NET error message in a web browser when an error is encountered in the application. It is recommended to change the

CustomErrors setting to RemoteOnly or On as these error messages can contain parts of the code or information in the web.config (such as passwords).

1. Open **CMSroot\web.config** with a text editor.
2. Find this line:

```
<customErrors mode="Off"  
defaultRedirect="~/System/AdminAppError.htm"/>
```

3. Change the mode to **On** or **RemoteOnly**.

Note **RemoteOnly** allows internal users to see the error messages, while external users see a custom error page

4. Change the defaultRedirect to a custom error page if desired. By default it uses the default ACM error page located in **CMSroot\System\AdminAppError.htm**.
5. Save and exit.

Setting Maximum Request Settings

While ACM has a setting to limit the maximum size of a digital asset that can be uploaded, there is also a .NET limit that controls this. This setting governs any .NET request, and also limits things like the size and duration of uploading template packages if they are too big.

Limits are put in place to prevent extremely large requests, or malicious requests from affecting the system negatively,

By default, the limits set in ACM are: 20MB and 10 minutes. To change these values:

1. Open **CMSroot\web.config** with a text editor.
2. Find this line:

```
<httpRuntime maxUrlLength="1024"  
relaxedUrlToFileSystemMapping="true" maxRequestLength="2097151"  
executionTimeout="60000"  
requestValidationType="IronPoint.CM.Framework.CustomRequestValid  
ation" requestValidationMode="2.0" />
```

- a. Set the `maxRequestLength` to the desired value in kB.
 - b. Set `executionTimeout` to the desired value in seconds (maximum allowed: 2097151).
3. Save and exit.

Note There is also a default limit in IIS 7.5 that can bottleneck the maximum size of a digital asset that can be uploaded. Check the [Configuring IIS](#) section for more information configuring this.

Configuring IIS

It is recommended to run ACM in 64-bit mode to take advantage of the improved memory addressing capabilities. For these instructions we will assume this application will be configured in 64-bit mode; however, 32-bit mode will work as well with different configuration.

1. Open IIS Manager.
2. Right-click in the IIS instance > select Add Web Site. Assuming this is the only site in IIS on this server, then use the following settings. For IIS instances with multiple sites or more advanced configuration here, refer to Appendix A: IIS Bindings.
 - **Site Name:** Enter a name for your site that will appear in IIS
 - **Physical path:** click ... and select the CMSroot folder
 - **Type:** http
 - **IP address:** All Unassigned
 - **Port:** 80
 - **Host name:** *blank* (see Appendix B: Redirects if there is more than 1 IIS site on this server)
3. Click **OK**.
4. In the left pane, click on Application Pools.
5. Right-click the Application Pool with the name entered into Site Name above > Click Advanced Settings.
 - **.NET Framework Version:** Set to v4.0
 - **Enable 32-Bit Applications:** Set to *False*
 - **Managed Pipeline Mode:** Set to *Classic*
 - **Load User Profile:** Set to *True*
 - **Idle Time-out:** Set to a 0 to not shut down
 - Click **OK**.
6. Right-click the Application Pool with the name entered into **Site Name** above > Click **Recycling**.
 - a. Uncheck **Regular time intervals**.
 - b. Enable **Specific time(s)**.
 - c. Enter an off-peak time (e.g., 2:00 AM).
 - d. Click **Next**.
 - e. Click **Finish**.

Configuring Maximum Digital Asset Upload Size

By default, there is a setting in IIS 7.5 that limits the maximum content length for a request to IIS (20MB)

1. Expand **Sites** in the left pane.
2. Click the site that was just created.
3. Double-click **Request Filtering** on the Web Site Home pane.

4. Select the **Headers** tab in the **Request Filtering** pane.
5. Click **Add Header...**
6. In the **Header:** field, type in *Content-type*.
7. In the **Size limit:** field, type in the desired maximum size in the size limit in bytes (e.g. 100000000 would be 100MB).

Note A maximum content limit is in place for security purposes to prevent large requests from affecting the system. Do not extend this value to more than what is needed. More information can be found online:

<http://www.iis.net/ConfigReference/system.webServer/security/requestFiltering/requestLimits>).

If using Friendly URLs for this site:

1. Click on the site again.
2. Open **Handler Mappings**.
3. Click **Add Script Map**.
 - **Request path:** *
 - **Executable:**
C:\Windows\Microsoft.NET\Framework64\v4.0.30319\aspnet_isapi.dll
 - **Name:** ACM Wildcard Friendly URL
 - Click **Request Restrictions...** > Access tab > Set to None.
 - If you have installed IIS 8 on Windows Server 2012, disable **Invoke handler only if request is mapped to**.
4. Click **OK**.
5. Click Yes to allow this ISAPI extension.
6. Ensure there are no handler mappings for *.htm and *.html. Remove them if there are.

Configure Directory Security

The ACM application, via the ASP.NET worker process, needs file permissions to create and change files in the ACM application files.

Configuring permissions for Windows Server

In Windows, by default, the ASP.NET worker process runs the ApplicationPoolIdentity user (this user is created when the Application Pool is created). The ASP.NET worker process can also run as the NETWORK SERVICE user. We will assume the ApplicationPoolIdentity user is used as that is more option according to Microsoft.

To adjust the permissions of a file or folder:

1. Right-click **CMSroot** > Click **Properties**.
2. Click **Security** tab.
3. Click **Edit**.
4. Click **Add** (if ApplicationPoolIdentity is not present).
5. Make sure the location is set to the local server. Do not select domain name.
6. Type *IIS APPPOOL\NAME* (replace NAME with the name of the Application Pool created for this site in IIS).
7. Click **Check Name** button.
8. Click **OK**.
9. Enable the Modify checkbox for the ApplicationPoolIdentity user.
10. Click **OK**.
11. Repeat these steps for all files and folders listed above.

ACM Testing and Configuration

At this point, you are ready to test the site. There can be multiple ways to access the site and they all depend on how the IIS bindings and DNS are set up.

If an FQDN is set up via DNS, then browse the FQDN to access the site (e.g., <http://www.mySite.com>).

If the site is currently using the server host name, then browse the name of the host name of the server (e.g., <http://myServer/>) to access the site.

Note The first time a site starts up, it may take a few minutes as ACM loads all pages and digital asset into memory during startup to improve performance.

After starting the site, select the Login page and ensure you can log in.

Default Administrator Password

The Default Administrator username and password are:

Username: *activecm*

Password: *good*

Note It is important to change the default administrator password after you log in to the system for the first time.

System-wide Administrator Setup

The System-wide Administrator is notified of system-wide issues through email. This email address is the From address in some emails sent to users from the system.

To set an administrator:

1. Log on to ACM.
2. **System** > Click **Global System Variables**.
3. Select the **System Variables** tab.
4. Fill in the **System-wide administration email address** and **System-wide administration email name** fields.
5. Click **Save**.

Site Administrator Setup

The Site Administrator is notified of issues with the site and is the From address in some emails sent regarding a specific site (e.g. Approved for Publication emails).

To set an administrator:

1. Log on to ACM.
2. **System** > Click **Sites**.
3. Select the site (most likely Your Site #1 if this is a new install).
4. **Active**: enabled.
5. **Site Name**: the name of the site as shown in ACM.
6. **Default Email**: email address of Site Administrator.
7. **Email Display Name**: display name when emails are sent from the Site Administrator.
8. **Dynamic External FQDN**: The FQDN or hostname of the site (including http://). Leave blank if no current FQDN or host name is configured.
9. **Enable Friendly URLs**: Enable if the friendly URL settings were enabled during the IIS setup and you wish to use friendly URLs instead of /pageXXX.aspx URLs.

Support Contact Information

You can configure what support contact information users will see if they click on the Help > About ACM menu in the Admin Toolbar.

To configure the contact information:

1. Log on to ACM.
2. **System** > Click **Global System Variables**.
3. Select the **Support** tab.
4. Configure the information here.
5. Click **Save**.

Note The support contact information can be set at a site by site basis through the Sites Manager.

Register Page Types and HTML Editors

Registering page types will add all the page types which you have licenses for into the database and make them available for you to use. Registering HTML Editors will do the same for different HTML Editors in the system.

1. Log on to ACM.
 2. **System** > Click **Page Types**.
 3. Click **Register Page Types**.
 4. Ensure the expected pages are registered, not all pages will be registered if you do not have the licenses for those pages.
 5. **System** > Click **System Information**.
 6. Expand **HTML EDITORS**.
 7. Click **Register HTML Editors**.
 8. Ensure the RadAjax and TinyMCE editors are registered successfully. Contact the ACM Support team if you were not given the license for the TinyMCE editor.
-

FQDN Setup

ACM uses the FQDN when it needs to build a fully-qualified domain name. This field is typically not used in most operations, but in some that require the domain name of the correct site, it may not build the correct link without configuring this option.

1. **System** > Click **Sites**.
2. Select the site.
3. Fill in the FQDN in the **External FQDN** field (e.g. http://www.domain.com).
4. If the server cannot browse the internet or resolve the URL in the **External FQDN** field, fill in a URL that the server can resolve in the **Internal FQDN** field as the server will use this to build links for operations such as the static or search export (e.g., http://servername/).
5. If the site is using Friendly URLs, enable the FQDN checkbox. Make sure the Handler Mappings in the **Configuring IIS** section is enabled before doing this.
6. Click **Save**.

Health Monitor Install

Installation and Configuration

The Health Monitor is used to ensure that the system is alive in memory. ASP.NET applications by default will shut down after being idle for a certain amount of time. If it's shut off, then any scheduled tasks may not run as it requires the system to be running to poll what needs to be done at a specific time, and any users who do browse the site will have to wait for pre-caching of pages and digital assets upon the first startup of the site, which can take a while. The Health Monitor is implemented as a system service to prevent the site from being taken offline and keep all the cache in place.

The Health Monitor package

The Health Monitor is located in the Health Monitor folder of the extracted zip file containing the ACM application files. You will receive a separate .zip file that will contain all files necessary for the Health Monitor installation

Installation

1. Create folder to hold the Health Monitor (e.g. C:\ACM\Health Monitor).
Note Do not place this within the **CMSroot** folder as it will need to be uninstalled before an upgrade.
Note This folder cannot be moved without reinstalling Health Monitor.
2. Copy the contents of the Health Monitor folder into C:\ACM\Health Monitor.
3. Open the HealthMonitor.config file in a text editor.
 - a. Change **LogSuccess** to *False*.
 - b. Fill in the **SiteDownEmail** details if you would like the Health Monitor service to send an email if it detects the ACM site is down.
 - c. Change the **MonitorURL** to the URL of the site /keepalive.aspx (e.g. http://www.activenetwork.com/keepalive.aspx). This URL must be browsable on the ACM Server itself.
 - d. Add additional instances if the Health Monitor needs to monitor multiple sites by adding additional <IronpointInstance> tags.
4. Install the service using .NET InstallUtil.exe which is located in the following directory
 - a. Open Command Prompt with the **Run as Administrator** option.
 - b. Type in:

```
%SYSTEMROOT%\Microsoft.NET\Framework\v4.0.30319\installutil.exe  
/i "C:\ACM\Health Monitor\IronPoint.CM.HealthMonitorService.exe"
```

Enable Health Monitor Auto Start

1. Open Windows Services (**Start > Run > type *services.msc***).
2. Right-click **IronPoint Health Monitor** service > Click **Properties**.
3. Select *Automatic* for Startup type.
4. Click **Start**.
5. Click **OK**.

Configuration

If the HealthMonitor.Config is edited after installation, you must restart the **IronPoint Health Monitor** service to push the changes through. The service does not need to be reinstalled.

Uninstalling Health Monitor

If you need to uninstall the Health Monitor, you must utilize the `-u` flag on the `installutil.exe`:

1. Open Command Prompt with the **Run as Administrator**
2. Type in:

```
%SYSTEMROOT%\Microsoft.NET\Framework\v4.0.30319\installutil.exe  
/u "C:\ACM\Health Monitor\IronPoint.CM.HealthMonitorService.exe"
```

Appendix A: IIS Bindings

Configuring IIS Bindings

IIS can host multiple sites at a time. The way that IIS distinguishes requests for different sites is through the IIS Bindings. You can bind a specific combination of IP addresses, Port Numbers, and Host headers for each site so that a request to that combination is picked up by the right site.

The bindings are requested during the creation of a site in IIS but are also configurable from the site properties after the site is created.

If this ACM site is the only site on this server, you can keep the default bindings:

- **IP address:** All Unassigned (a request for any IP address)
- **Port:** 80
- **Host name:** blank (a request for any host name or URL)

Add Website ? X

Site name: ACM Site Application pool: ACM Site Select...

Content Directory

Physical path: C:\CMSRoot\ ...

Pass-through authentication

Connect as... Test Settings...

Binding

Type: http IP address: All Unassigned Port: 80

Host name:
Example: www.contoso.com or marketing.contoso.com

Start Website immediately

OK Cancel

If there are multiple sites, the bindings must be unique across different sites in the same instance of IIS. If you have another site that listens for the exact same IP, Port, and Host, then IIS will stop one of the sites.

Common use cases for host headers are if there are two IIS sites (e.g. www.ACTIVENetwork.com and www.ACTIVEgovernment.com), [?]both listening on the same IP address and port 80. And the server only has a single IP address. In this scenario,

you can distinguish the two using Host. To set this up, use these bindings when setting up the IIS site:

www.ACTIVEnetwork.com site:

- IP address: All Unassigned
- Port: 80
- Host name: www.ACTIVEgovernment.com

www.activgovernment.com site:

- IP address: All Unassigned
- Port: 80
- Host name: www.ACTIVEgovernment.com

It would be a good idea to set each site to add additional bindings for the non-www version of the domain name as well because you can create redirects in ACM to redirect requests to the non-www version of the URL to the www version using Authority Redirects. For more information, see [Appendix B: Redirects](#).

www.ACTIVEnetwork.com site:

- IP address: All Unassigned
- Port: 80
- Host name: activegovernment.com

www.activgovernment.com site:

- IP address: All Unassigned
- Port: 80
- Host name: activegovernment.com

For more information about bindings, see this Microsoft Blog:

<http://blogs.technet.com/b/chrad/archive/2010/01/24/understanding-iis-bindings-websites-virtual-directories-and-lastly-application-pools.aspx>.

There are also many resources with Microsoft that explain how bindings work if you require more detailed information.

Appendix B: Redirects

ACM allows you to create Authority Redirects, 301 Redirects and FQDN Mappings. These settings can be configured in the Ironpoint.config tab located in the Configuration Files section of the Admin Center.

These settings are all redirects that can be used to improve SEO (Search Engine Optimization), prevent 404 errors when moving domains, and provide an easy interface to create different types of redirects within the application.

These 3 types of redirects take priority in the order mentioned (and shown in the configuration screen).

Authority Redirects

Multiple authority redirects can be configured and each one takes an 'authorityIn' and a 'targetAuthority'.

Upon receiving a request, ACM will check the authority of the request URL to see if a redirect has been configured. If a match is found, it will rebuild the URL changing the requested domain with the domain in Target Authority.

Using the example below, a request to **http://domain.com/parks/animals.htm** would be 301 redirected to **http://www.domain.com/parks/animals.htm**.

AUTHORITY REDIRECTS		
AUTHORITY IN	TARGET AUTHORITY	DELETE
<input type="text" value="domain.com"/>	<input type="text" value="www.domain.com"/>	
<input type="button" value="Add"/>		


The most common use of Authority Redirects is to redirect requests to the non-www version of a URL to the www version.

301 Redirects

Multiple 301 redirects can be configured, each taking a 'targetURL' and defining multiple semicolon delimited URLs that will redirect to the target.

Upon receiving a request, ACM will see if the requested URL matches any of the 'Redirect301' path list URLs and performs a redirect to the target URL.

Using the example below, if a request is made to **http://www.domain.com/home.htm** or **http://www.domain.com/site3.aspx** the request will be redirected to **http://www.domain.com/**.

301 REDIRECTS		
TARGET URL	PATH LIST	DELETE
<input type="text" value="http://www.domain.com/"/>	<input type="text" value="http://www.domain.com/home.htm;http://www.domain.com/site3.aspx"/>	
<input type="button" value="Add"/>		

Two common uses of this redirect are:


- Redirect all the different ways of reaching the home page to a single way (as above).
Note You must set up an FQDN Mapping redirect in order for the above scenario to work as the default.aspx will redirect a request to www.domain.com to www.domain.com/site3.aspx.
- Shorten URLs by redirecting a simple, easier to type and remember URL to a longer one (e.g., www.domain.com/parks redirecting to www.domain.com/areas/green/parks.htm).

FQDN Mapping

Multiple FQDN mappings can be configured, each requiring an 'FQDN' and a target page. This will allow a user to click or type in the 'FQDN' and see that remain in the address bar when the actual page is loaded.

Upon receiving a request that contains only the authority, ACM will check if the request URL matches one of the configured FQDN settings in the list of mappings. If one is found, ACM will rewrite the path server side to the specified target page, but it will keep the same requested URL in the browser.

Using the example below, requesting **http://www.domain.com** will load the Home page while remaining **http://www.domain.com** on the address bar.

FQDN MAPPINGS		
FQDN	TARGET PAGE	DELETE
<input type="text" value="http://www.domain.com/"/>	<input type="text" value="Home Page"/>	
<input type="button" value="Add"/>		

A common use for this feature is in conjunction with the 301 redirects. Setting up the 301 redirects to redirect requests to www.domain.com/page4.aspx, www.domain.com/site3.aspx, www.domain.com/home.htm, and www.domain.com/default.aspx to www.domain.com, which is then caught by the FQDN Mapping and produces the home page. The net effect is that any way you access the home page will count as a hit towards www.domain.com.

Appendix C: Multiple Sites on a Single Install

ACM allows you to create multiple sites within a single install of ACM. If you wish to use different URLs or domains to access the individual sites created in ACM, there are two ways to accomplish it:

1. Set up FQDN Mappings for each domain to perform the redirect depending on the requested domain.
2. Use the default.aspx file to redirect to various sites depending on the requested domain.

IIS Configuration

Ensure that the IIS instance that ACM runs on is configured to listen for each domain for each site in this ACM install by configuring the IIS bindings. See [Appendix A: IIS Bindings](#) for more information.

Note: You can also configure an IIS site to listen for all domains by leaving the Host Header field blank. This allows you to add new sites without needing to update the IIS Configuration to add the new domain to the bindings list

FQDN Mapping Method

The FQDN mapping method is the preferred way to separate sites since it works in conjunction with setting up SEO-friendly URLs mentioned in “Appendix B: Redirects”:

1. Log on to ACM.
2. **System** > Click **Configuration Files**.
3. Select the **Ironpoint.config** tab.
4. Click **Add** in the **FQDN MAPPINGS** section.
5. Type in the domain name of the page in the FQDN column (e.g., http://www.WebSiteOne.com).
6. Click <<**Select Page**>> in the **TARGET PAGE** column and select the home page of the site this domain name should link to. Use the **Site** dropdown menu to select a different site if necessary.
7. Repeat steps 4 to 6 for each site using a different domain name.
8. OPTIONAL: “Follow Appendix B: Redirects” to complete all the redirects to minimize the number of different URLs somebody could browse the home page of your site (which increases search rankings on Google).

Default.aspx Method

As noted in the beginning of this section, **the CMSroot\default.aspx** can be configured to redirect a request to the desired site based on the requested domain.

The default.aspx can be controlled via the ACM Configuration Files interface or modified directly from the server using a text editor. By default, the CMSroot\default.aspx will redirect requests to Site ID 3 using this syntax:

```
<% Response.Redirect("~/site3.aspx") %>
```

Here is an example of syntax to redirect www.WebSiteOne.com to Site ID 3, and www.WebSiteTwo.com to Site ID 4 and to return the default error 404 page if the domain does not match either of these.

```
<%
dim strURL as string = UCase(Request.URL.ToString)
If InStr(strURL, UCase("www.WebSiteOne.com")) > 0 then
Response.Redirect("~/site3.aspx")
elseif InStr(strURL, UCase("www.WebSiteTwo.com")) > 0 then
Response.Redirect("~/site4.aspx")
else
Response.Redirect("~/System/Error404.htm")
end if
%>
```

There are many ways to achieve this using VB.NET code, the above is simply one example.

The Default.aspx can also be managed directly from the Admin Center of ACM. In order to configure this:

1. Open the CMSroot\Default.aspx file with a text editor
2. Replace the code in there with this:

```
<%@ Page Language="vb" AutoEventWireup="false"
Codebehind="Default.aspx.vb" Inherits="IronPoint.Default" %>
```

This tells the system that ACM will handle the default.aspx request. To configure the redirects within ACM:

1. Log on to ACM.
2. **System** > Click **Configuration Files**.
3. Select the **Default.aspx** tab.
4. Configure this page to add/edit/remove redirects for the default.aspx.

The example below will have the same result as editing the default.aspx file directly.

Web.config Ironpoint.config Default.aspx

WARNING: Changes made here only work with the system-supplied Default.aspx.

CONDITIONAL REDIRECTS

ORDER	URL COMPARISON	COMPARE TEXT	CUSTOM RESPONSE CODE	REDIRECT PAGE	DELETE
1	Contains	www.WebSiteOne.com	Response.Redirect ("~/site3.aspx")	<< Select Page >>	
2	Contains	www.WebSiteTwo.com	Response.Redirect ("~/site4.aspx")	<< Select Page >>	
3	None (Else)		Response.Redirect ("~/System/Error404.htm")	<< Select Page >>	

Add

Additionally, if you know what page you would like the **www.WebSiteOne.com** to load, you can exclude the custom response code and simply select the page in the redirect page column.

Appendix D: Search Page

The following steps are necessary to configure ACM and server to support the Search page type. If you plan to not use the ACM Search page, this does not need to be set up.

How the Search Page Works

The ACM Search page uses the Windows Search to index the site and perform the search operations. ACM needs to be configured to generate static versions of each index page.

When someone submits a search term, the ACM Search page then uses this index created from Windows Search to find the pages containing those words.

Static Export of SearchSite files

In order to create the SearchSite files, ACM must be configured to statically export the search files on each site that will be included in the search. This process must be scheduled in order to keep the SearchSite files and contents up to date.

1. Login to ACM.
2. **System** > Click **Sites**.
3. Click on the **Site Name** of the site you wish to enable search functionality on.
4. Enable the **Generate Search Files** checkbox.
5. Enter the FQDN (Fully Qualified Domain Name) that will resolve to this site in the **External FQDN** field (e.g., <http://www.domain.com>).

Note The FQDN must be browsable from the ACM Server itself. If the server is not connected to a DNS server or cannot resolve the External FQDN, enter the IP address or an FQDN that the server can browse to, to pull up this site in the Internal FQDN field. (e.g. <http://127.0.0.1>).

6. Click **Save**.
7. Perform a Static SearchSite Export:
 - a. **System** > Click **Export**.
 - b. Select the site this page is on in the **Sites** dropdown menu.
 - c. Ensure that the whole site is selected in **Page(s) to Export**.
 - d. Enable the **File Deletion** checkbox.
 - e. Ensure it says **SearchSite will be created** below the **Export Now** button. If it is not, the site will need to be configured to generate SearchSite files.
 - f. Click **Export Now**.
8. Check for errors during the export;
 - a. **System** > Click **System Events**.
 - b. Select **Export** in the **Available Logs** dropdown menu.
 - c. Ensure there are no errors from the export process.

Note Some page types will cause an error message here when exported because they do not support static search exports such as the Category and GoogleMiniSearch page types.

9. Schedule the Export to run automatically on a regular interval:
 - a. Select all pages in the site within the **Page(s) to Export** field.
 - b. Enable the days the export should occur in the **Export on Schedule** field.
 - c. Set the time the export should occur.

Note It is recommended to select a time outside of normal hours as a static export can use up CPU power on the server.

- d. Enable the **File Deletion** checkbox.
- e. Click **Save**.

Installing Adobe's iFilter

In order to index contents of PDF files (be able search for words inside of PDF files through the ACM search page), you must install the Adobe iFilter on the ACM Server. The Adobe iFilter is also bundled into Adobe Acrobat Reader; if Reader is installed on this server already, there is no need to install the iFilter.

Look in Adobe's site for the latest version of the Adobe iFilter.

Windows Search

Follow these instructions to configure the Windows Search Service to index your ACM site.

Installation on Windows Server

1. Log on to the ACM Server.
2. Open **Server Manager**.
3. Click **Roles** in the left menu.
4. Click **Add Roles** in the right pane.
5. Click **Server Roles** in the left menu.
6. Enable the **File Services** checkbox.
7. Click **Next** twice.
8. Enable the **Windows Search Service** checkbox > **Next**.
9. Enable the checkbox for the volume containing ACM to index > Click **Next**.
10. Click **Install**.

Configuring Windows Search Service

After installing the Window Search service:

1. **Start > Control Panel**.
 2. Search for and select **Indexing Options** (you may need to set **View by**: to **Small icons** to see this option).
 3. Click **Modify**.
-

4. Disable the checkboxes on all folders from search except the **CMSroot\Sites** folder and the location set in the web.config for PhysicalPathToPrivateShareFolderLocal.
5. Click **OK**.

The Windows Search Service will start indexing and automatically continue to index.

Configuring the ACM Search Page

This is how to configure the ACM Search page to utilize the service you've set up above.

1. Log on to ACM.
2. Create a new Search Page.
3. Configure any other necessary options.
4. Publish the page.

Removing Specific Pages from the Search

Pages can be configured to be removed from the search. To do this:

1. Log on to ACM.
2. **System** > Click **Site Design Manager**.
3. Navigate to the page that needs to be removed from search > Click **Edit**.
4. Uncheck **Include in Search**.

This will cause this page to not be exported during a Static Search Site Export. You must perform an export after this with the File Deletion option enabled.

An export will initiate a static export.

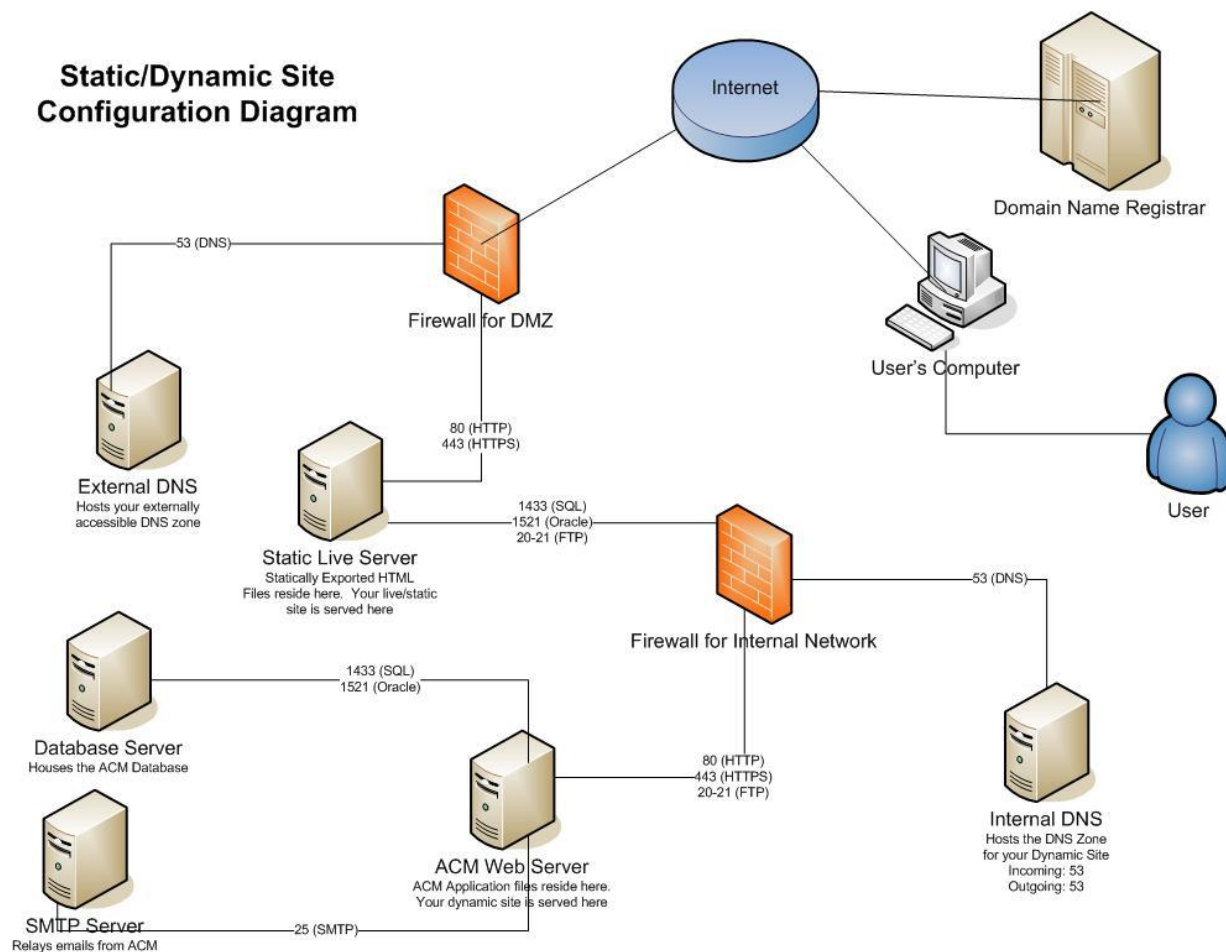
1. **System** > Click **Export**.
2. Select the site this page is on in the **Sites** dropdown menu.
3. Ensure that the whole site is selected in **Page(s) to Export**.
4. Enable the **File Deletion** checkbox.
5. Ensure it says **SearchSite will be created** below the **Export Now** button. If it is not, the site will need to be configured to generate SearchSite files.
6. Click **Export Now**.
7. The pages removed from search will be deleted from the **SearchSite** folder, and then Windows Search will remove them from the index.

Appendix E: Static HTML Export

These instructions are only for customers who plan on hosting their live website using statically generated HTML pages from ACM rather than the dynamic ACM pages.

ACM can be configured to export static HTML versions of each page in the system, and then FTP this to a remote web server hosting the website. This allows ACM to be hosted on a different server than the live site which is hosting the static copy of the site.

This setup is not recommended; please see the [Advantages and Disadvantages of Static Sites](#) section for more information.



All ports are assuming the use of default ports for that application.

Definitions used in this section:

Static: the website that runs the public facing content. This is a static site that receives its contents from a static HTML export from ACM. This server does not have ACM installed.

Dynamic: the website that runs ACM. Content Providers will edit content using this website, and this performs the export of static HTML files

Advantages and Disadvantages of Static Sites

There are some advantages and disadvantages of a static/dynamic setup as opposed to a traditional dynamic only ACM site.

Advantages of a Static/Dynamic site setup

- Ability to host the server which content providers edit content on, within a different server than the server the live site resides in.
- Ability to create new pages and entire sections of a site and view the results without publishing live.

Disadvantages of a Static/Dynamic site setup

- There are more systems and dependencies on scheduled processes, which leads to more points of failure and more maintenance involved.
- Some page types are not supported in a dynamic environment, and some require a hybrid mode in order to be supported, which is equivalent to browsing the dynamic server.
- The dynamic server needs to be externally accessible for hybrid pages to work. Some pages need to be in hybrid mode to keep the dynamic functionality available (e.g. Survey, Calendar, and Login page types).
- Pages are not published instantly. It requires content be pushed through a scheduled export process (typically once a day), or using Emergency Publish which only System Administrator users have access to.
- Exported links are different than what's shown in the dynamic site, which can cause confusion from the content provider end.
- Personalization (the ability to restrict pages from being viewed by users who aren't in a certain group(s)) does not work in a static/dynamic environment.
- The File Deletion checkbox (which is needed to remove pages from the static side when they are archived/inactivated/deleted in ACM) will also delete any other non-ACM created file in the HTMLSites folder preventing you from storing any non-ACM files there.

Configure the Site to Export and FTP Static HTML Pages

1. Log on to ACM.
 2. **System** > Click **Sites**.
 3. Select the site to be exported.
 4. Configure **STATIC PROPERTIES** settings:
 - a. **Generate HTML Files:** This needs to be enabled in order for the Export process to statically export HTML files.
 - b. **FQDN:** This is the fully qualified domain name used to build a full URL for relative links. If the dynamic site has a link like “/page123.aspx”, then the export process will render the link as FQDN/page123.aspx (e.g. http://www.staticdomain.com/page123.aspx).
-

- c. **Virtual Directory (Alias):** If the site will be exported into a virtual directory of the static server, type in the directory path here.
- d. **FTP URL:** This is the URL of the FTP server that the exported files will be transferred to. If this is blank, the FTP process is disabled.
- e. **FTP Protocol/Connection Mode/UserName/Password:** These options depend on the settings of your FTP server. Contact your FTP administrator for more information.
- f. **File Extension:** Files generated from the export will use this extension, typically set to .htm or .html.

Note Friendly URLs will also use this extension to render their URLs.

5. Configure **DYNAMIC PROPERTIES** settings:
 - a. **External FQDN:** The export process will browse each page in the site to export. The way it does this is, it uses the FQDN/pageXXX.aspx?pagemode=HTML The server will browse this link for each page. The dynamic server must be able to browse the FQDN.
 - b. **Internal FQDN:** If this is specified, it will override the External FQDN in order to build the links for the server to browse each page during the export process.

Schedule the Export to Automatically Occur

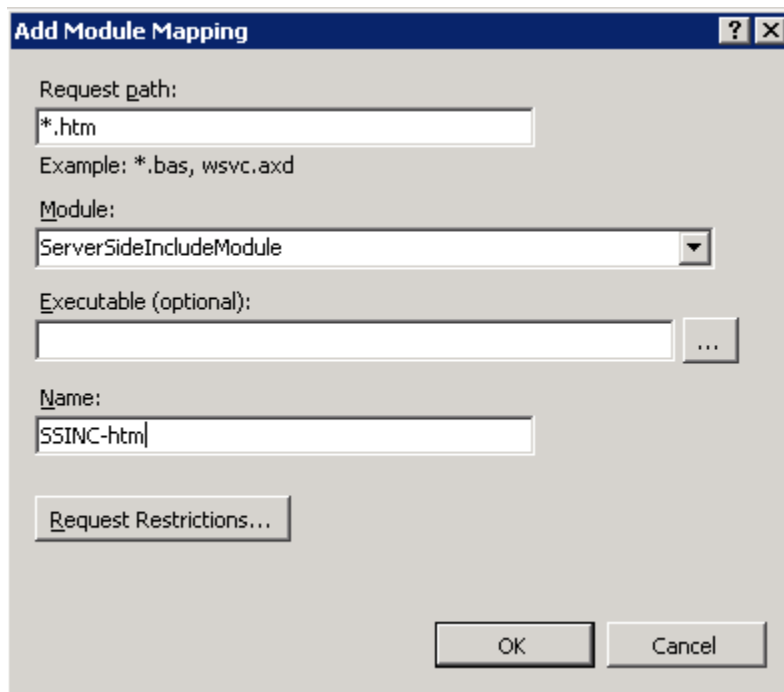
1. Log on to ACM.
2. **System** > Click **Export**.
3. Select the site this page is on in the **Sites** dropdown menu.
4. In the **SCHEDULED EXPORT** section, ensure that the whole site is selected in **Page(s) to Export**.
5. Enable the **File Deletion** checkbox.
6. Ensure it says SearchSite will be created below the **Export Now** button. If it is not, the site will need to be configured to generate SearchSite files.
7. Click **Save**.
8. Test the export by performing an unscheduled export with the same options. Check the files and System Events > Export Log to ensure the export completed successfully.

Configuring Server Side Includes (Static Server Setup)

Server Side Includes must be enabled for the dynamic elements of a static site, such as the Section Menu, to work properly. Static sites utilize server side includes to pull dynamic content to their pages. To enable server side includes for a static website running on IIS, follow these steps:

1. Install Server Side Includes feature following Windows documentation:
<https://docs.microsoft.com/en-us/iis/configuration/system.webserver/serversideinclude>
 2. Log on to the static ACM Server.
-

3. Open IIS Manager.
4. Expand the IIS instance (indicated by the server name).
5. Expand **Sites**.
6. Select the dynamic ACM site.
7. Open **Handler Mappings**.
8. Check if **.htm** and **.html** are included in the Handler Mappings list. If they are, ensure that the below settings are correct for these extensions. If not, add them following the next steps.
9. Click **Add Module Mapping...**
10. Fill in the details as per this screenshot:



11. Click **OK**.
12. Repeat step 8 to 10 for ***.html**.

Configuring Server Side Includes (Dynamic Server Setup)

1. Log on to the dynamic website server.
2. Open **Server Manager**.
3. Select Add Roles and Features.
4. Next, Next, Next until onto Server Roles
5. Expand Web Server (IIS) and Web Server and Application Development
6. Enable the Server Side Includes Checkbox.
7. Click **Next, Next, and Install**.

Appendix F: Shared Folders

In addition to storing data in a Database (SQL Server or Oracle), the ACM also stores data on an NTFS Volume we call Shared Folders. The Shared Folders contains all data that is not stored in the database such as digital assets and attachments uploaded from a Survey page type.

It is not necessary to configure the Shared Folders outside of what's default, but this allows you some flexibility as to where these files are stored (in addition to public vs. private files and folders in the Shared Folders in different locations on the ACM web server).

With no additional setup, all of the Shared Sites folder will automatically be stored in **CMSroot\Sites**. However, the Shared Folders can be configured to be stored in a remote location on the server or on an external file share.

Note A central Shared Folders location is required for the ACM Multi-server setup.

Public vs. Private Shared Folders

The Shared Folders are broken down into public folders and private folders. Public folders are those that need to be accessible via a browser. Private folders are those that do not need to be accessible via a browser and can be stored in a different location. ACM will access the private folders as necessary; they do not need to be accessible via HTTP.

By default, public and private folders are stored in the same location (CMSroot\Sites).

Public Folders

The location of public folders is configured in the web.config key: `PhysicalPathToSharedFolder`.

The contents of the public Shared Folders include:

- One folder for each site, denoted by Site ID (e.g., **CMSroot\Sites\3**).
- The current template design package for each site (e.g. **CMSroot\Sites\3\Templates**).
- Attachments from various page types and other functionality in ACM (e.g., **CMSroot\Sites\3\Attachments**).

Note In some older versions of ACM, the attachments folder was stored directly in the Sites folder instead of underneath a specific Site ID folder.

Private Folders

The location of private folders is configured in the web.config key: `PhysicalPathToPrivateSharedFolder`.

The contents of the private Shared Folders include:

- One folder for each site, denoted by Site ID (e.g., **CMSroot\Sites\3**).
- Archived template design packages for each site (e.g., **CMSroot\Sites\3\archive**).
- Digital Assets - that contains all documents, images, PDFs, etc. that are managed by the ACM (**CMSroot\Sites\DigitalAssets**).
- HTML files generated for Search functionality (e.g., **CMSroot\Sites\3\SearchSite**).
- HTML files generated for Static File Exports (e.g., **CMSroot\Sites\3\HTMLSite**).

Storing Shared Folders in a Non-Default Location

These instructions explain how to move the Shared Folders (/Sites) from being stored in the default **CMSroot\Sites** folder to a folder in a remote location, such as a File Server or a NAS.

Note If the Shared Folders are to be stored in a location remote to the ACM web server, the ACM web server must be part of a domain. This scenario requires domain user authentication between the ACM Server and the remote file server.

1. Create a file share on the desired location where the Shared Folders will be stored.
2. If using a remote location, create a domain user (that will be used as the impersonation user) whose password does not expire, and ensure this domain user has read/write access to the Shared Folders and read/write access to the file share for this Shared Folders.
3. Copy the contents of the current **CMSroot\Sites** folder into the Shared Folders location.
4. Create a Virtual Directory (this will take place of the **/Sites** folder).

Windows Server 2008:

- a. Open IIS Manager.
- b. Expand the IIS instance (indicated by the server name).
- c. Expand **Sites**.
- d. Right-click **Add Virtual Directory**.
- e. Type **SharedSites** in the **Alias** folder.
- f. Type the local or UNC path to the Shared Folder (e.g. \\FileServer\ShareName).
- g. Click **OK**.
5. Update the web.config file:
 - a. Open **CMSroot\web.config** in a text editor
 - b. Find the **PhysicalPathToSharedFolder** key and update the value parameter to the local or UNC path of the Public Shared Folder, e.g.:

```
<add key="PhysicalPathToSharedFolder"
value="\\FileServer\ShareName" />
```

- c. Find the **VirtualPathToSharedFolder** key and update the value parameter to the virtual directory created in step 4, e.g.:

```
<add key="VirtualPathToSharedFolder" value="/SharedSites" />
```

- d. If storing the Private Shared Folder separate from the Public Shared Folder, find the **PhysicalPathToPrivateSharedFolder** key, uncomment the line (remove the leading <!-- and ending --> in the line), and update the value parameter to the path of the Private Shared Folder, e.g.:

```
<add key="PhysicalPathToPrivateSharedFolder"
value="\\FileServer\PrivateShareName" />
```

- e. Add the following line on the line right after <system.web>

```
<identity impersonate="true" userName="DOMAIN\UserName"
password="Password" />
```

6. Add the impersonation user to the IIS worker process group:

Windows Server 2008:

In Windows 2008, the IIS worker process group is IIS_IUSRS. Add the impersonation user to this group. For more information on this group, refer to <https://www.iis.net/learn/get-started/planning-for-security/understanding-built-in-user-and-group-accounts-in-iis>.

7. Set the permissions on the ACM files for the impersonation user.
8. Right-click on the Shared Folders > click **Properties**.
9. Select the **Security** tab.
10. Ensure that the impersonation user has Modify permissions on this folder.
11. Click OK.
12. Repeat this process for the following folders on the ACM Server:
 - **CMSroot\CM\WebUI**
 - **CMSroot\System**
 - **CMSroot\web.config** (only if you want the web.config file to be configurable through ACM)\

Appendix G: Multi-Server Setup

This section discusses how to set up multiple ACM servers. Servers connected to the same database can be configured to work in a load balanced environment. For most uses of ASP.NET, a single server can handle all requests in a timely manner. However, in order to handle high volumes of traffic or to meet redundancy requirements, you may need to set up multiple ACM servers to handle the processing and traffic.

The instructions below assume ACM is already set up in a single server environment.

The Concept of Multiple Servers in ACM

A multi-server setup with ACM consists of multiple web servers with ACM installed, pointing at the same database and same shared /Sites folders. A file server should be provisioned to allow storage of the shared /Sites files and folders.

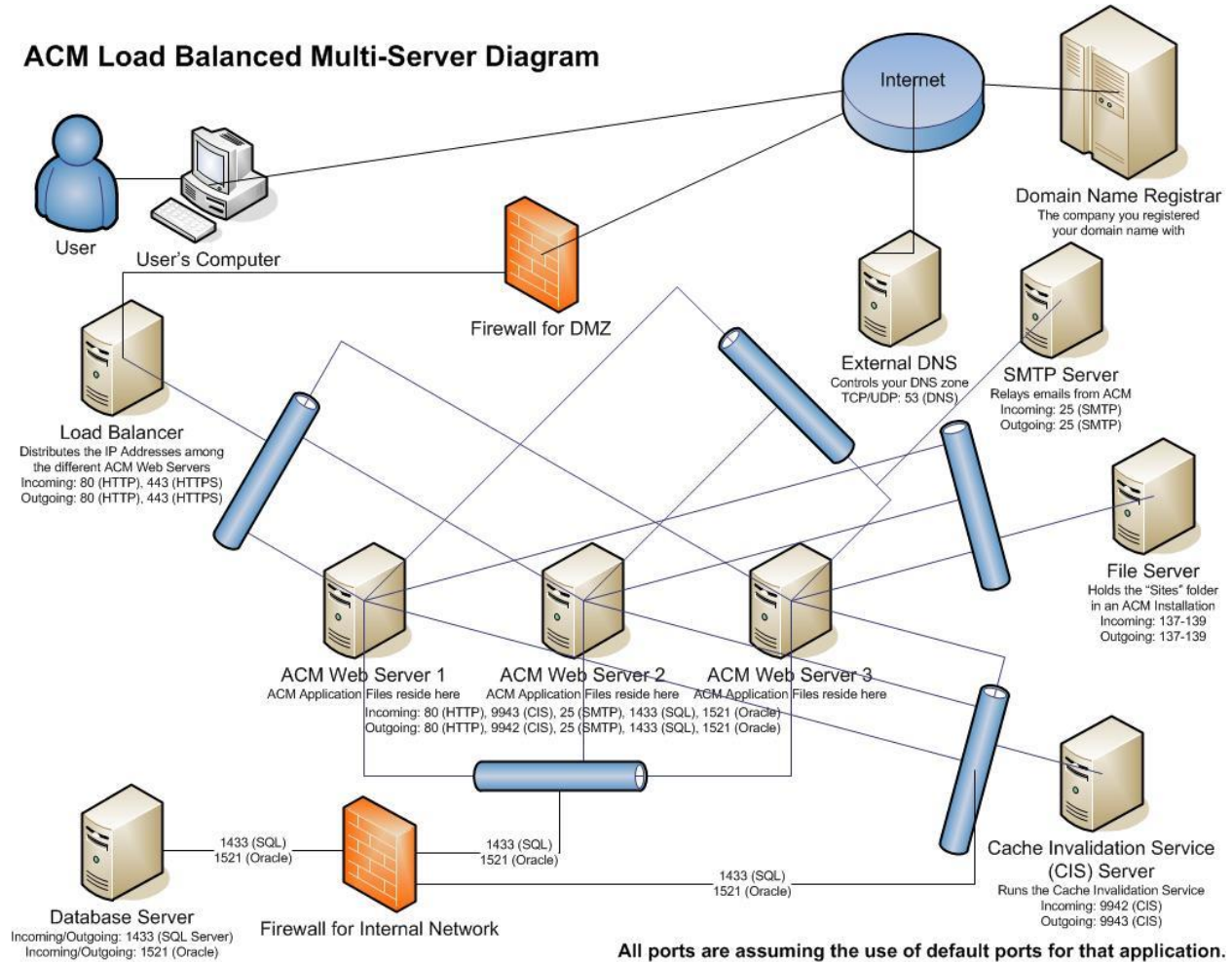
Additionally, a load balancing mechanism is used to receive requests from users, and then distribute them among the different web servers.

To handle caching across multiple servers, ACM uses the Cache Invalidation Service which allows servers to notify other servers; when data has changed on one server, it invalidates the cache on other servers.

Something else to consider is how to manage sessions across multiple servers. With the default in process (InProc) setting, sessions are stored on the web server; if a load balancer pushes the connection to another server, the session information will not exist on the next server, and that session would be lost.

There are a few configurations that handle this:

- Use the ASP.NET State Service. This is the optimal way to allow proper load balancing across servers because it does not rely on the load balancer to handle session affinity (same session always goes to the same web server). However, this introduces a new single point of failure. But the State Service is generally regarded as a more reliable service than having a single web server as a point of failure.
- Use SQL Server to set up a database to manage sessions. This requires additional setup on the SQL Server and behaves much like the ASP.NET state service. This method won't be covered in these instructions as it is recommended to use the ASP.NET state service due to its simplicity and ease to set up.
- Allow the load balancer to handle sessions along with balancing the load. This is the simplest way to handle multiple sessions as it does not require any additional setup within ACM.



Domains

If running in a multi-server environment, each ACM web server must be configured in a domain, as the application must be able to run as a specific domain user, which also has modify access to the Shared Folders that typically reside in a file server accessible by all web servers.

Setting up State Server Service

If the Load Balancer is set to handle session affinity, this section can be ignored. Otherwise, follow these instructions if the ASP.NET state service is set up to manage sessions across multiple servers. Perform these operations on the server that will manage the sessions which has to be one that each web server can communicate with over.

1. Install .NET 4.0 on the ASP.NET State Service server.
2. Configure the state server to allow connections from remote machines by setting the following registry key to 1:

- a. **Start** > Click **Run**
- b. Type in **regedit** > click **OK**
- c. Open:

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\aspnet_state\Parameters\AllowRemoteConnection

- d. Set Value = **1**
3. Start the Service
 - a. **Control Panel** > **Administrative Tools** > Open **Services**
 - b. Right-click on **ASP.NET State Service** > click **Start**
 - c. Right-click on **ASP.NET State Service** > click **Properties**
 - d. Ensure **Startup type** is set to **Automatic**

Note By default, the State Service listens on port 42424. This can be changed in the registry at the following location:

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\aspnet_state\Parameters\Port.

Setting up the ACM Cache Invalidation Service

Follow these instructions on the server that will run the ACM Cache Invalidation Service (CIS). The cache invalidation service ensures that pages that have changed on one server, notifies the other servers to invalidate the cache for that page.

The CIS will be a separate install of ACM that points to a separate ACM database. This allows the CIS to not need to be upgraded with an ACM upgrade with this setup, and this will also allow a separate ACM site to view log messages specifically with the CIS.

Note A single CIS service can serve multiple ACM instances.

1. Install .NET 4.0 on the server that will run the CIS.
2. Install ACM with a new ACM database on the CIS server with the instructions above. We will refer to this folder as the **CISroot** folder.

Note The files used for the CIS must be the same ACM version as the application files on the ACM Servers.

3. Ensure license file (**CacheInvalidationService.lic**) is located in the **CISroot\bin** folder. Contact the ACM Support Team if you do not have this license.
4. Open **CISroot\IronPoint.DataAccess.dll.config** in a text editor.
5. Ensure the Connection String is pointing to the correct ACM database.
6. Modify the default ports for CIS if you do not want to use the default ports.
 - a. Open **CISroot\bin\CacheInvalidationService.config** in a text editor.
 - b. Modify the **<webServerPort>** tag value to change which port each web server will listen on for the CIS traffic.
 - c. Modify the **<cisPort>** tag value to change which port the CIS server will listen for the CIS traffic.
 - d. Save and close **CISroot\bin\CacheInvalidationService.config**.

7. Register the ACM Cache Invalidation Service:
 - a. **Start > Run > type cmd.**
 - b. Run the following command:

```
%SYSTEMROOT%\Microsoft.NET\Framework\v4.0.30319\InstallUtil.exe  
/i CISroot\bin\CacheInvalidationService.exe
```

- c. Verify that the final lines written to the console by InstallUtil.exe are:

```
The Commit phase completed successfully.
```

```
The transacted install has completed.
```

8. **Control Panel > Administrative Tools > Open Services.**
9. Right-click **ACTIVE CM Cache Service > click Start.**
10. Right-click **ACTIVE CM Cache Service > click Properties.**
11. Ensure **Startup type** is set to **Automatic.**
12. Log on to ACM.
13. **System > Click System Events.**
14. Ensure a message is logged in the Application log indicating **The cache invalidation service has been started.**

Note The default ports which the CIS listens for traffic is port 9943, and each ACM Server listens on port 9942. These ports must be open between the CIS and ACM Servers in order for the CIS to function correctly. The ports used can be configured to use any port via the **CISroot\CacheInvalidationService.config** file.

If the CIS is not running, the ACM site will still continue to function; however, changes to various pages may not show up on the other server right away. This prevents the CIS from being a single point of failure, but it is imperative that the service is fixed and restarted in the event of an issue to prevent old data from being displayed on the ACM site.

Configuring Shared Folders

In order to access the same Shared Folder between the different web servers, a central location must be created to store these files. It is recommended to set up a share on a file server to store these files.

In a multi-server environment, the path to the Public and Private Shared Folders must be a centrally accessible location. Ensure that the SharedSites virtual directory and both web.config keys (PhysicalPathToSharedFolder and PhysicalPathToPrivateSharedFolder) point to locations that are accessible via CIFS from each web server. See Appendix G: Shared Folder, for more details on configuring the SharedSites.

If this central file share is not accessible, none of the templates and digital assets will be accessible on the site, which would likely leave the site displaying only text.

Note Ensure that the central file share is on a stable environment and is regularly backed up to ensure recoverability.

ACM Server Configuration

Follow these instructions on each ACM Server in the pool. These steps assume that the web server has been correctly configured for single-server ACM operation with the ACM files installed in **CMSroot**. Please note a number of these parameters are case sensitive.

1. Open **CMSroot\web.config** in a text editor
2. If utilizing the ASP.NET State Service to manage sessions, the machineKey must be set. Otherwise, it does not. Immediately after the **<system.web>** element, specify an encryption key to use for ViewState.

```
<machineKey
validationKey="0123456789ABCDEF0123456789ABCDEF0123456789ABCDEF"
decryptionKey="FEDCBA9876543210FEDCBA9876543210FEDCBA9876543210"
validation="SHA1"
decryption="AES" />
```

The validationKey and decryptionKey should be set to 48 random hexadecimal characters, and each web server should have the same values.

3. If the ASP.NET State Service was chosen as the method to handle sessions, follow these instructions. Otherwise, ignore this step. Modify the sessionState settings:
 - a. Open **CMSroot\Web.config** in a text editor
 - b. Find the **<sessionState>** tag
 - c. Set mode to **"StateServer"**
 - d. Set **stateConnectionString** to **"tcpip=IP:PORT"** (e.g. **"tcpip=127.0.0.1:42424"**)

Note IP is the host name or IP address of the server containing the state service and PORT is the port which the state service is listening on.

4. Find the EnableMultiServerCache key and set the value = **True**.

```
<add key="EnableMultiServerCache" value="True" />
```

5. Find the CacheInvalidationServiceAddress and change the value to the IP address of the CIS server.

```
<add key="CacheInvalidationServiceAddress" value="192.168.0.48" />
```

6. Find the CacheInvalidationWebServerAddress and change the value to the IP address of the web server that this web.config file is located on.

Note This will be different on each web server in a cluster.

```
<add key="CacheInvalidationWebServerAddress"
value="192.168.0.47" />
```

7. Add the following keys into the web.config right before the **</appSettings>** tag:

```
<add key="MultiServerWebServerPort" value="9942"/>
```

```
<add key="MultiServerCacheServicePort" value="9943"/>
```

Note These keys only need to be added if non-standard ports are used to communicate with the CIS server. The ports must match the corresponding ports in the `CISroot\bin\CacheInvalidationService.config` configured above. If not specified, ACM will use the default ports shown above.

Configure Additional Services

Windows SearchService

If you use the ACM search page type, you must configure ACM to export files locally to each web server for Windows Search to index rather than exporting to the `PhysicalPathToSharedFolder` (the `/Sites` folder). The Windows Search service cannot index remote locations easily. To configure ACM to export files locally (must be done for all servers used):

1. Log on to the ACM web server.
2. Open the **web.config** file in a text editor.
3. Add this line within the `<appSetting>` tag near the **PhysicalPathToSharedFolder** section, and replace the value with the desired location of the search files:

```
<add key="PhysicalPathToPrivateSharedFolderLocal" value="C:\ACM\SearchExport" />
```

4. Log on to ACM.
5. **System** > Click **Sites**.
6. Select the site that is exporting search files.
7. Enable the **Force Export to Local Folder** checkbox.

Note This checkbox only appears if the **Generate Search Files** checkbox is enabled

8. Click **Save**.
9. **System** > Click **Export**.
10. Do a full unscheduled export.
11. Check the export folder to confirm the files have been exported there.

Health Monitor

Health Monitor should be set up on each web server. It is recommended to customize the email template used in the Health Monitor configuration file to indicate which server it comes from (e.g. add the IP address of the server in the email), to help determine which server is having an issue when problems occur.

Appendix H: Log File Locations

ACM sites will generate a number of log files. Here is a listing in order to help you to troubleshoot issues, and to know where to look when space usage is being consumed by log files.

IIS Logs

IIS, by default, logs information about every request to IIS. This logging can be configured to include or remove various pieces of information and can be deleted safely without harming the site. To find the location of the log files:

Windows Server 2008:

1. Open IIS Manager.
2. Expand the instance (indicated by the server name).
3. Expand **Web Sites**.
4. Click the ACM site.
5. Open **Logging**.
6. The **Directory** field is the location of the IIS log files.

Follow these instructions on each ACM Server in the pool. These steps assume that the web server has been correctly configured for single-server ACM operation with the ACM files installed in **CMRoot**.

These files can grow rapidly depending on the amount of traffic, so developing strategy to maintain these files is recommended (e.g., scheduling the files be moved to another server or compressed on a regular basis).

ACM System Event Logs

These logs are stored in the database and will log up to a certain limit defined in the `ironpoint.config` setting. They log important information regarding events that occur within ACM. To access them:

1. Log on to ACM.
2. **System** > Click **System Events**.
3. Select the log type in the **Available Logs** dropdown menu.
4. Click on the log entry to view the details.

The logs are stored in the `CM.LogMessages` table in the database. If the site is having issues and ACM cannot be accessed, the database can be queried for information about each log entry.

Windows Event Logs

The Windows Event logs can store information about events that happen in .NET that can affect the site. Some errors will be logged in both ACM System Event Logs as well as the Windows Event Logs.

For more information about Windows Event Logs, refer to <http://technet.microsoft.com/en-us/library/cc722404.aspx>.

FileNotFound Logs

These logs are only created when Friendly URLs are created in ACM. Since ACM will handle all requests in IIS with Friendly URLs enabled, requests returning error 404 codes are not captured in the IIS logs anymore. In order to capture this information, these requests are logged in **CMSroot\Sites\FileNotFound**.

Appendix I: Backup and Restore Procedures

This document includes a recommendation for scheduled actions to be performed to protect ACM data in the event of complete failure of server hardware and/or databases for self-hosted customers. **Active Network, LLC is not responsible for maintaining the backups on your application or database servers**, so it is highly recommended to schedule backups for the items mentioned in the following sections.

ACM stores data in (up to) three locations: the database instance, the public shared folder, and the private shared folder (note that in most types of deployments, the public shared folder and the private shared folder will be the same location). To backup all data, each of these data repositories must be addressed.

It is also recommended to back up the IIS site configuration to make restoring the site easier. View Microsoft documentation on backup and restore procedures depending on the version of IIS you are using.

Backup Schedules and Data Retention

Active Network, LLC recommends that a backup of the ACM database is taken regularly on a scheduled basis. Backups should also be stored off-site to prevent larger scale disasters from affecting the recovery. The frequency of the backup and type of backup (full, differential, incremental, etc.), depends on your organization's data retention policies, storage capacity, and backup policies.

A sample backup schedule that could be followed is:

- Full backup – once a week
- Incremental backup – once per day

Backing up the Database

Refer to your database administrator for best practices on backing up the database to minimize the amount of storage space needed but maximize the ability to restore data.

Backing up Shared Folders

The Shared Folders should be backed up to a similar schedule as the database as the files should be about the same age; otherwise, unexpected problems could occur, such as:

- Missing template files that exist in the database, but do not physically exist in the Shared Folders causing any page to use it to not be browsable.
- Missing digital assets that exist in the database, but do not physically exist in the Shared Folders.

The default location of the Shared Folders is **CMSroot\Sites**. If the folder is in a non-standard location, follow these steps to locate the current **PhysicalPathToSharedFolder** location:

1. Open **CMSroot\web.config** in a text editor.
2. Find the **PhysicalPathToSharedFolder** key. The **value** parameter is the location of these files.
3. This folder should be backed up according to the backup schedule.

If private Shared Folders are enabled, follow these steps to locate the current private Shared Folder location:

1. Open **CMSroot\web.config** in a text editor.
2. Find the **PhysicalPathToSharedFolder** key and the value parameter is the location of these files.
3. This folder should be backed up according to the backup schedule.

Backing up ACM Application Files

The ACM application files (the content of **CMSroot** minus the Shared Folders) do not need to be regularly backed up, except prior to an upgrade. These files should not change often unless a configuration change is made directly on the files, which we typically do not recommend.

Backing up the IIS Configuration

The IIS configuration is likely not to change often, so backups may not need to be scheduled as often as the ACM database or application files.

IIS 7.0, IIS 7.5:

Follow these instructions to backup IIS configuration:

<http://blogs.iis.net/bills/archive/2008/03/24/how-to-backup-restore-iis7-configuration.aspx>

Restoring the ACM Database

Follow Microsoft's best practices for restoring a SQL Server database.

Restoring the ACM Application Files

Ensure that the database restore was done at the same or around the same time as the Site files restore. The Shared Folder files contain digital assets and templates that are mapped to entries in the database, so mismatching versions can cause some digital assets to not work, and some templates to not work (meaning the whole page using that template does not work).

In most cases, mismatching versions should work, but to ensure the easiest restore process, ensure that that Site Files and the database match up as much as possible.

The configuration should not change in most cases, but some areas to consider confirming, are:

- The ConnectionString in the ACM configuration files
- The SMTP IP address in the web.config

Restoring the Other Services

If a web server needs to be set up, follow the ACM Installation Guide to configure the server with ACM; however, use the recovered ACM application files instead of new ACM installation files for the ACM root folder.

If applicable, there may be other services to reinstall:

- Health Monitor
- Windows Search
- Cache Invalidation Service (for Multi-Server installations only)