EPiSERVER

CREATE+

A PRODUCT IN THE EPiSERVER CREATE+ PACKAGE

VERSION 3.3

ImageVault

ImageVault simplifies and improves the handling of files in EPiServer CMS. The basic function of ImageVault is to store files on a central server, so everyone in the organization can access and use the files

Installation Manual

Copyright
This manual is protected by international copyright law. No changes of content or reproduction of parts of this publication is allowed without permission from the publisher.

Installation Manual for ImageVault version 3.3
Created October 30, 2007
Last edited January 21, 2010
©Meridium AB

Contents

1. Introduction 2
1.1.1. News in ImageVault 3.3 2
1.1.2. Known problems 2
1.1.3. To consider before installation 3
1.1.4. To consider after installation 3
1.1.5. To consider when upgrading ImageVault 5

2. Installation 5
2.1.1. Installation program started 5
2.1.2. Introduction 5
2.1.3. License conditions 6
2.1.4. Select EPiServer installation 7
2.1.5. License file 8
2.1.6. Select Database 9
2.1.7. Installation of ImageVault 10

3. Changes in Installation 11
3.1.1. Move File archive to File server 11
3.1.2. Configure ASPNET to run under a domain account 11
3.1.3. Move Albums 11
3.1.4. Change of Web server 11
3.1.5. Change of web.config. 12

4. Web.config. reference 12
4.1.1. Cache 13
4.1.2. Support for Search engines 14

5. Running ImageVault in a 64 bit environment 15

6. Load balancing 17
6.1.1. Cache listener configuration 17
6.1.2. Cache listener service 17

7. Multiple sites using one ImageVault 18
7.1.1. Installation 18
7.1.2. File share configuration 18
7.1.3. Site and user configuration 19
7.1.4. Cache listener configuration 20

8. Mirroring 21

9. Image Editor 21

10. Other Configuration 23

1. Introduction

This document describes the different stages in the installation of ImageVault. The document also explains how ImageVault can easily be configured to store the images on a separate file server.

1.1.1. News in ImageVault 3.3

Any news and changes concerning the installation are described here.

- Limited support for running in 64 bit mode.
  Some parts of ImageVault is still required to run in 32 bit mode, but as these parts is now isolated the EPiServer site can be executed in 64 bit mode.
  This is achieved by running the web service responsible for image conversions in another application pool.

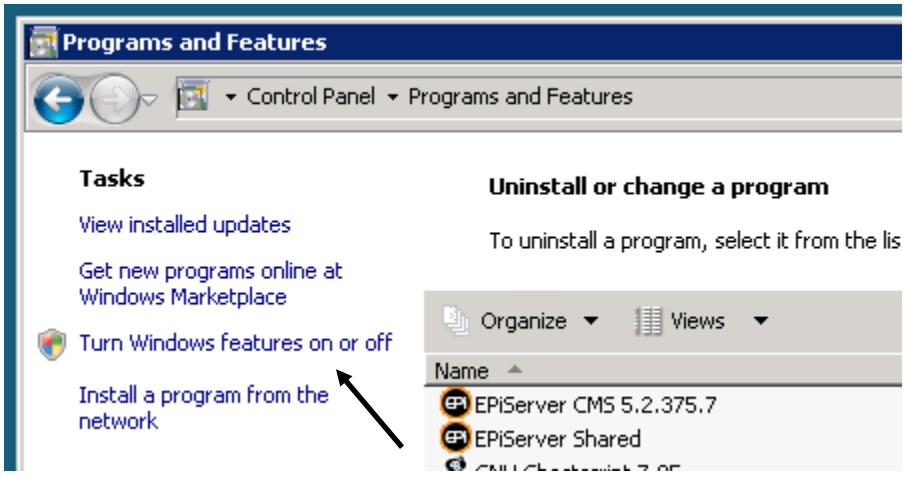
ImageVault 3.3 will run on both EPiServer 4.61 and .Net 2.0 and on EPiServer CMS; however different installation packages is required.

1.1.2. Known problems

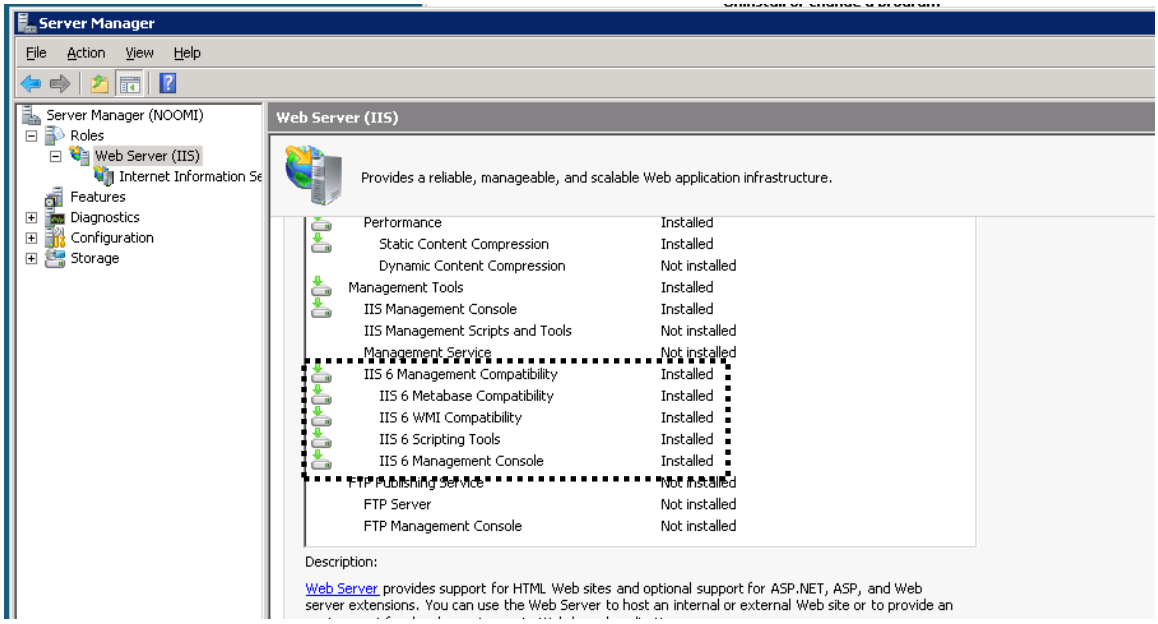
ImageVault will run on a 64 bit OS, but requires that the web service making the image convertions to be run on a site that executes in 32 bit mode. Se section 5.

When using IIS 7, ImageVault installation requires that the IIS 6 Management Compatibility features to be installed on your IIS server.

- Click the *"Programs and Features"* icon in *"Control Panel"*
- Click on *"Turn Windows features on or off"*.



- If on a Windows 2008 server, the *"Server Manager"* is shown.



Make sure that all the features under *"IIS 6 Management Compatibility"* is installed

1.1.3. To consider before installation

The user that you log in as, when running the installation, must have permissions to be able to create and change databases on the SQL server that is to be used for ImageVault.

If you indicate the SQL server with a computer name, Named Pipes must be activated, and if you indicate an IP address, TCP/IP must be activated. Right-click on the database server in the Enterprise Manager, and select Properties. Click Network Configuration to activate Named Pipes and/or TCP/IP.

If the installation is an upgrading of an earlier version of ImageVault, a complete backup of both database and files is recommended.

Note! When upgrading from an earlier version changes will be made to both the EPiServer database and the ImageVault database. These changes are not reversible so make backup copies of both databases.

1.1.4. To consider after installation

ImageGlue will require QuickTime for optimal performance.

Note! QuickTime is not included in the installation package but can be downloaded for free from http://www.apple.com/quicktime/download/.

When ImageVault is installed, a number of keys are added in the *ImageVault* section in the se.meridium.epiServer.extensions section of web.config. The keys that indicate a URL will be set to http://localhost by the Installation program, and must be changed to valid DNS names or IP addresses, for ImageVault to function properly. The IP address or DNS name must also be accessible from the server that ImageVault is installed on.

The installation has created two groups in EPiServer that are called *ImageVault Admins* and *ImageVault Editors*. To have access to administration in ImageVault, you have to be a member of *ImageVault Admins*, and to be able to add, change and delete images, you have to be a member of *ImageVault Editors*. You can replace these groups by changing *AdminGroups* and *EditorGroups* below the section *ImageVault*, in the section se.meridium.epiServer.extensions in web.config. Several groups can be indicated in a comma-separated list.

ImageVault has functionality to deliver notifications by e-mail. For this to work, the Web server must be enabled to use the Default SMPT Server to send e-mails.

Note! Make sure that the ASPNET process has access rights to the temp directory since this is required be some functions in the .Net framework.

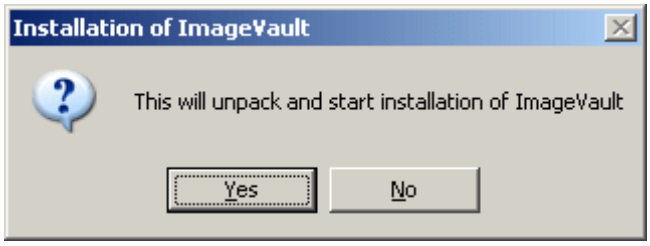
1.1.5. To consider when upgrading ImageVault

2. Installation

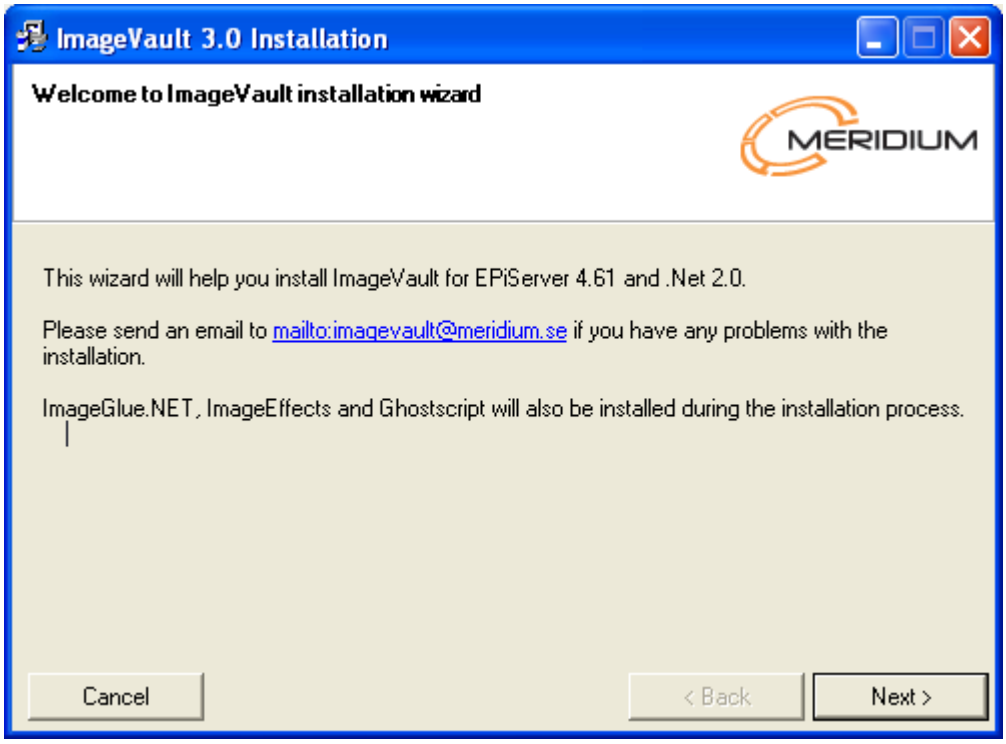
Download and run the Installation program.

During the installation ImageGlue, ImageEffects and GhostScript will also be installed. These components are necessary for ImageVault to function properly.

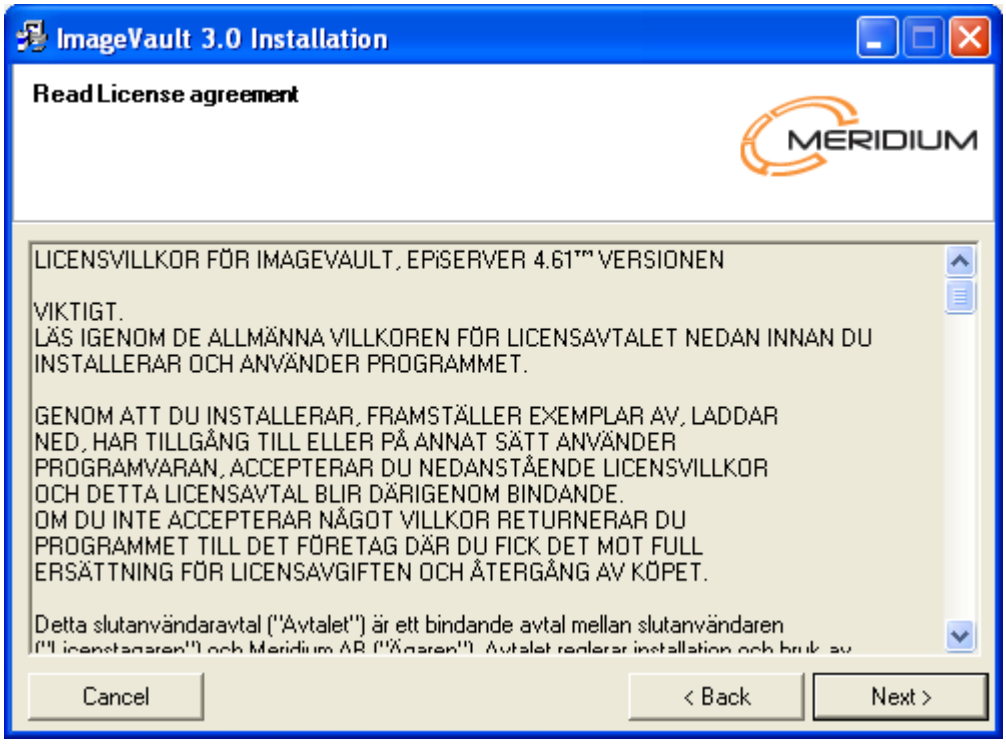
2.1.1. Installation program started



2.1.2. Introduction

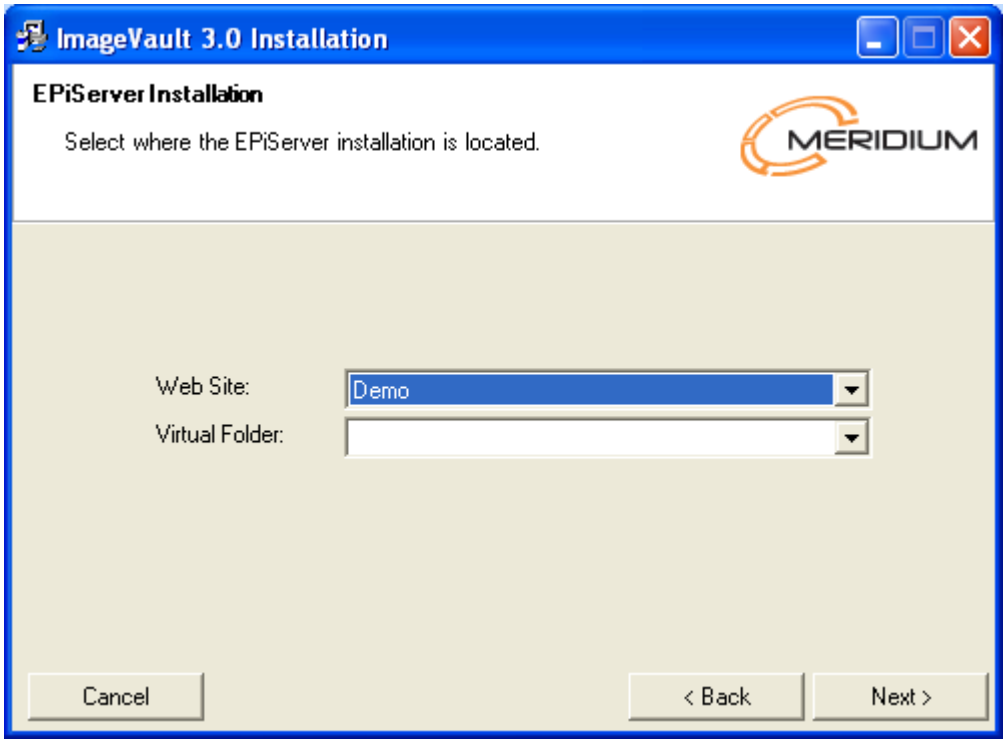


2.1.3. License conditions



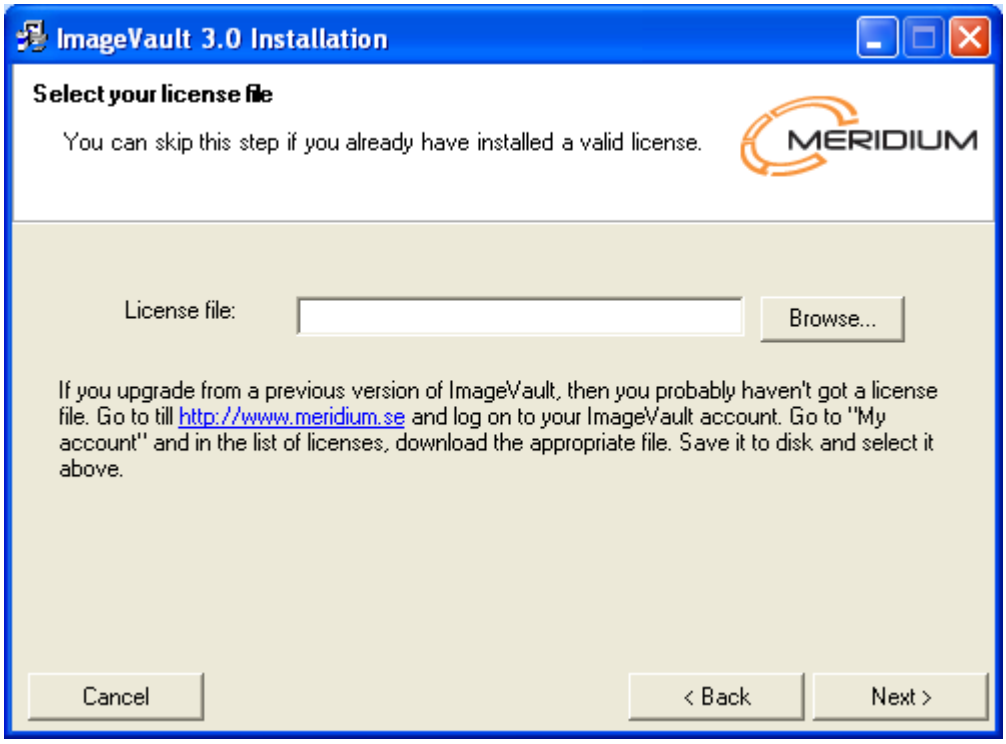
Read through the conditions and accept to continue.

2.1.4. Select EPiServer installation



Only web sites that have an EPiServer installation of version 4.61 or CMS will appear in the drop-down menu of the upper box. Select the desired web site and possible virtual album where you want ImageVault to be installed.

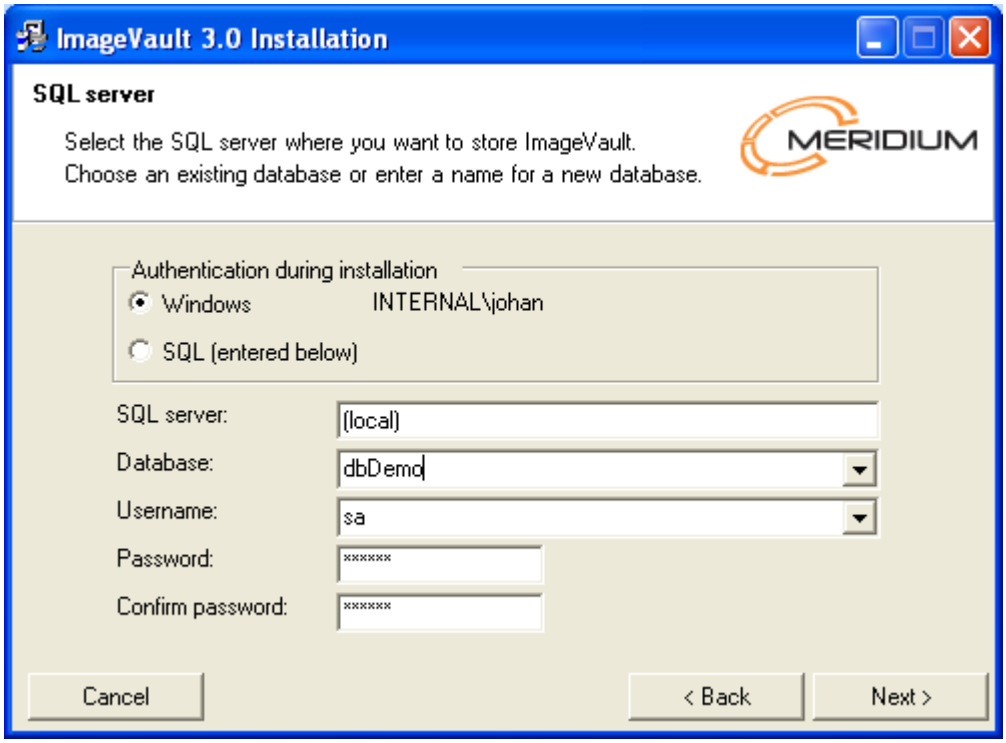
2.1.5. License file



Use the Browse button to locate your license file for ImageVault. If no file is selected, it can later be manually copied to the web sites root directory.

Note! If multiple Meridium products is installed on the same web site the license files has to be merged. This must be done manually if the file is not selected in this step.

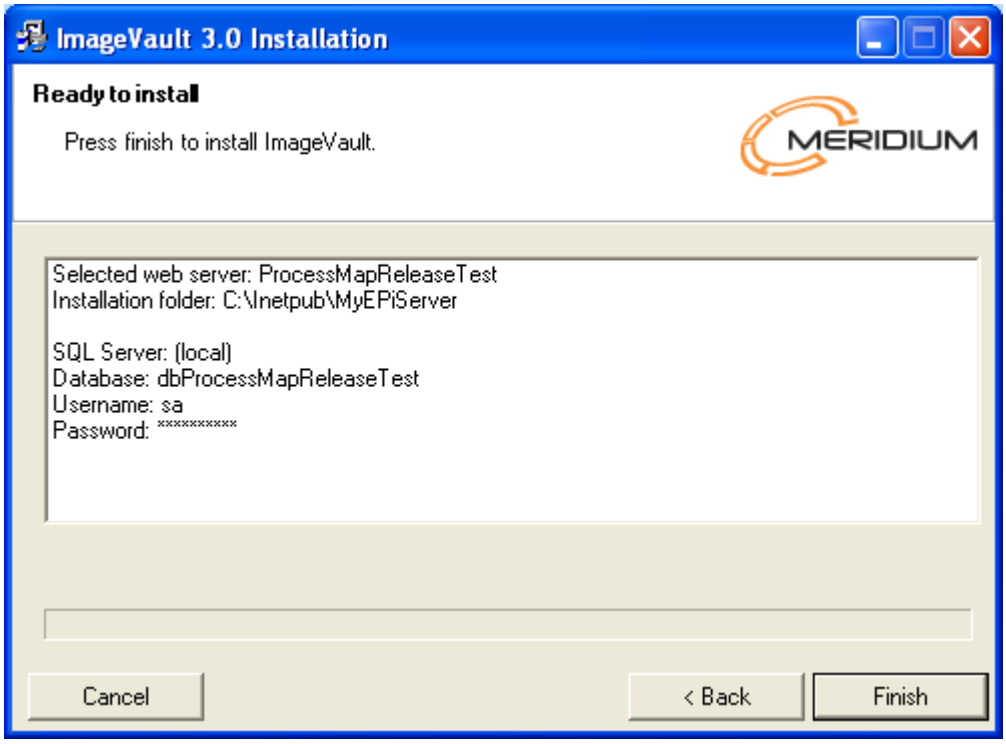
2.1.6. Select Database



As preset value the local SQL server is selected. To choose another SQL server, indicate its name in the text field. Existing databases on the selected SQL server will be displayed in a drop-down menu. If you want to create a new database, indicate the name of the database that you want to use. User name and password should be the ones that you want ImageVault to use after installation.

As preset the Installation program will use the database of the user running the Installation program. By selecting SQL in *Authentication during installation*, the installation may instead be run with the user indicated in the free text fields.

2.1.7. Installation of ImageVault



By now the Installation program has received all information about where and how ImageVault is to be installed. Click *Finish* to complete the installation.

During the installation ImageGlue, ImageEffects and GhostScript will be installed. ImageGlue is the component used for image handling in ImageVault. GhostScript is necessary for PostScript to function in ImageVault. Installation of these must be completed for ImageVault to function properly.

3. Changes in Installation

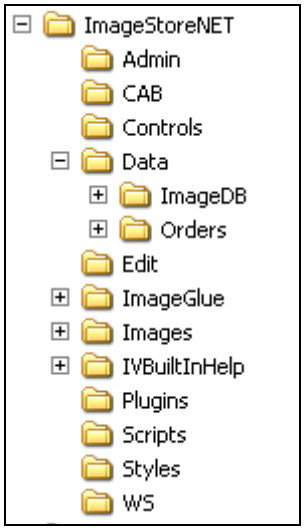
ImageVault is installed with a structure that fits most of our customers' basic requirements. This chapter describes how you can change structure and configuration to fit different applications.

3.1.1. Move File archive to File server

3.1.2. Configure ASPNET to run under a domain account

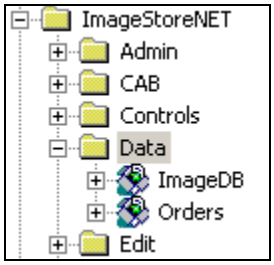
For ImageVault to reach files on another server, the account that is used to run the ASPNET process must be able to get permissions on the file server. This means that the account must be a domain account. For more information see http://support.microsoft.com/default.aspx?scid=kb;EN-US;317012

3.1.3. Move Albums



All files that are used in ImageVault are placed in an album structure under the ImageDB album.
Move these albums to the file server. Note that the account that runs the ASPNET process must have permission to the albums on the file server.

3.1.4. Change of Web server



When moving files to another file server, the ImageDB should be linked under the album *Data*, as virtual albums. The virtual albums only need to be given view permissions.

3.1.5. Change of web.config.

The values that may be configured in web.config are placed as attributes in the following Configuration section

```
  <se.meridium.epiServer.extensions>
    <imageVault ImagePathOriginal="C:\ImageDB" >
```

This key should be connected to the folder where the original images are archived.

```
ImagePathOriginal="C:\inetpub\Demo\ImageStoreNET\Data\ImageDB" />
```

Note! Please note that it's recommended to change this to a location on another disk, since files is created and removed in this location depending on user action. Also see MaxImageDataSizeRestriction in section 4 below.

4. Web.config. reference

A description of the keys that ImageVault uses.

Information about the database link to the ImageVault database.

```
ImageVaultConnStr=
"Server=(local);Database=ImageVaultNET;UID=sa;Password=;"
```

The parameter should be set in case the service that manages the conversion of images is run on a separate web server. If you do not use a dedicated server for the image handling, this key should not be included at all.

```
WebServiceURL="http://localhost:80/ImageStoreNET/WS"
```

A comma separated list with names of the groups that have Administration rights in ImageVault.

```
AdminGroups="ImageVault Admins"
```

A comma separated list with names of the groups that have Edit rights in ImageVault.

```
EditorGroups"ImageVault Editors"
```

A comma separated list of string to replace during extraction of IPTC and EXIF meta data. This could be used to handle national characters even when this is not permitted by the standard.

```
EXIFCharTranslation=" Ã¥=å,Ã¤=ä,Ã¶=ö"
```

If unexpected errors occur when accessing ImageVault images additional information about errors can be logged to a file.

```
DebugErrorLog=” c:\temp\log.txt”
```

Note! Make sure that the file exists and that the ASPNET process has rights to write to the file.

Should the new mirroring support be turned on or off.

```
ExportImageVaultProperties=”true”
```

Limit the amount of disk used by the handler to dynamically create converted images. The value entered is maximum number of GB that can be used.

```
MaxImageDataSizeRestriction=”5”
```

If the used disk space is exceeding the specified value no new conversions can be done by the handler and the image won’t be displayed.

This can be useful for preventing the disk from filling up if someone tries to request a lot of images in different formats.

Note! This is not a hard quota. It only set a limit for when the handler is allowed to convert new images. Though users can still upload new images, the conversion job still can make new conversions and so forth, hence the disk space used can exceed the specified value.

4.1.1. Cache

These parameters are used to control how ImageVault should handle the caching of images. The caching will function in the same way as EPiServer handles caching.

If this parameter is set to *true*, no caching of images will occur. The database will get a question every time an image is to be displayed (as before the caching was introduced). No notification about updates will be given to other cache listeners.

```
DisableCache="true"
```

Number of hours that an ”unused” IV-object remains “unused” in the cache, before it is thrown.

```
CacheTimeout="1"
```

Number of minutes that the image should be kept in the cache. This value is sent to Web clients and proxy servers and determines how long they should cache the content.

```
ImagesExpiresInMinutes="60"
```

4.1.2. Support for Search engines

To facilitate indexation for Search engines, e.g. Site Seeker, there is the possibility to send down metadata in the header information to these Search engines.

This should include a comma separated list of UserAgent for their crawlers, which have extra information as HTTP headers.

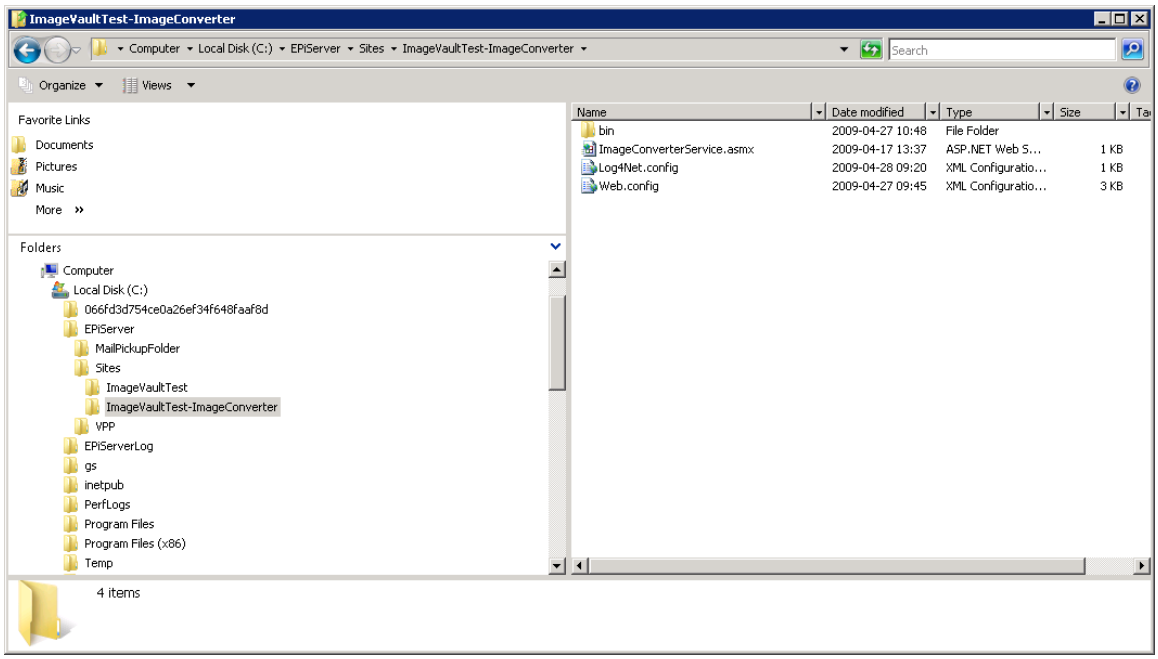
```
CrawlerUserAgents="SiteSeekerCrawler/1.0”
```

5. Running ImageVault in a 64 bit environment

ImageVault and the EPiServer site it is installed on can run in 64 bit mode, but the module converting images must execute in a 32 bit environment.

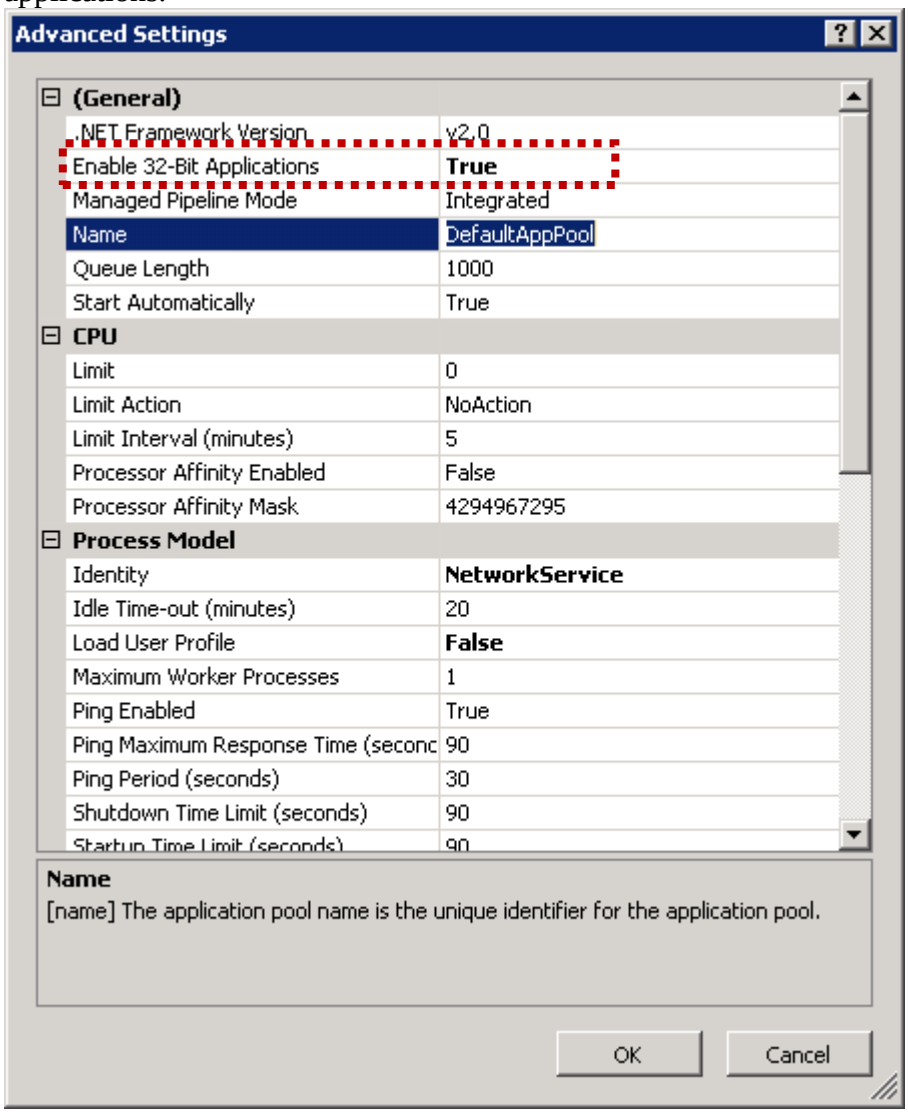
This can be achieved by running the required web service in a separate application pool.

1. Install ImageVault on the target EPiServer site using the installation program.
2. Create a new web site or virtual folder in an existing site.
3. Copy the files from folder wwwroot\ImageStoreNET\ImageConverter in the EPiServer site to the root of the new virtual folder.



4. Copy the <ImageVault> - tag from the web.config of the EPiServer site to the web.config in the new virtual folder.

5. Make sure that the application pool for the new virtual folder has enabled 32-bit applications.



6. And finally add the attribute ImageConverterWsUrl to the <ImageVault>-tag in the web.config for the EPiServer site.
   This should point to the web service ImageConverterService.asmx in the newly created virtual folder.

```
<imageVault ImageConverterWsUrl=
http://localhost:81/ImageConverter/ImageConverterService.asmx ...
```

6. Load balancing

When a site is split up over several servers ImageVault needs to communicate with the other servers to invalidate there cache when changes are made. In older versions of EPiServer ImageVault uses the same settings as EPiServer so no further changes are need to get the cache updates to work.

ImageVault does not support UDP cache updating that was introduced in EPiServer CMS 5 R1. In these cases ImageVault needs to be configured in the old way using "Remote Web Sites" that are set in EPiServer's admin mode.

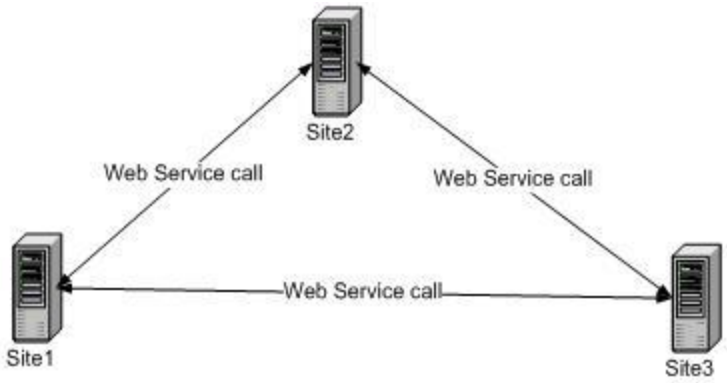
More information on how to do this can be found here:

http://world.episerver.com/en/Articles/Items/Configuring-the-Cache-in-Multi-Server-Scenarios/#Configuration

As an alternative to the method above this can also be configured in the ImageVault section of the web.config. This is the preferred way when working with multiple instances of EPiServer against a single ImageVault database, see section 7 below.

6.1.1. Cache listener configuration

Cache listeners are used by ImageVault to clear the sites cache when changes are made to the database.



6.1.2. Cache listener service

The cache listeners can be set up using anonymous access, but this is not recommended. To secure the web service we use EPiServer's BasicAuthentication Module. Every installation has a web service that has to be set to basic authentication. This is done by opening Internet Information Services and select properties for folder ~/ImageStoreNet/WS/AuthenticationService/ and disable Anonymous access.

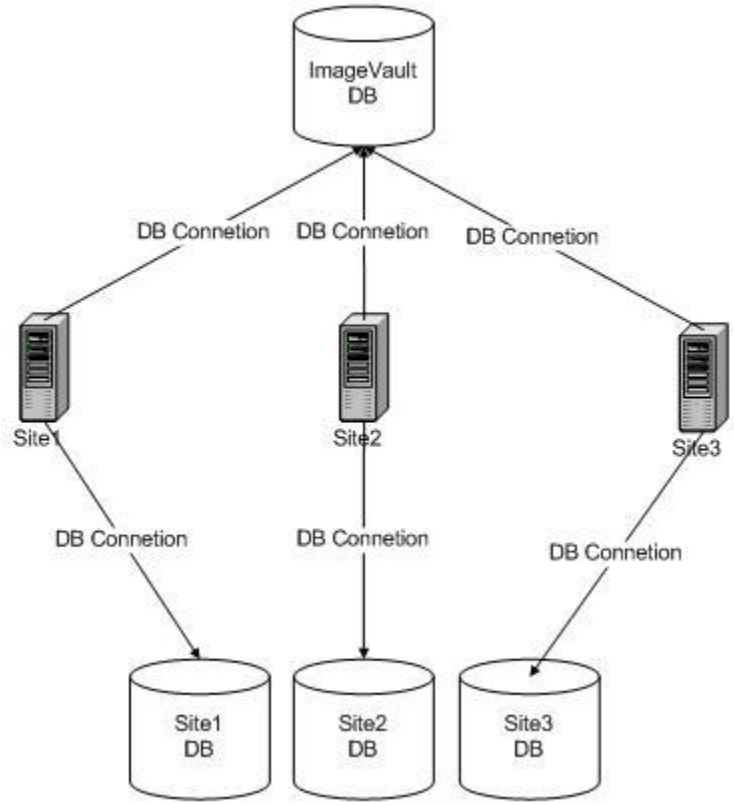
Se the document “Web Services.pdf”, downloadable from the EPiServer website, which describes how to configure Basic Authentication.

7. Multiple sites using one ImageVault

ImageVault can be used to share a single file repository between multiple EPiServer sites. This text describes how to set up ImageVault so that it will work with multiple sites. Note that this should not be confused with load balancing, i.e. multiple servers running the same site.

7.1.1. Installation

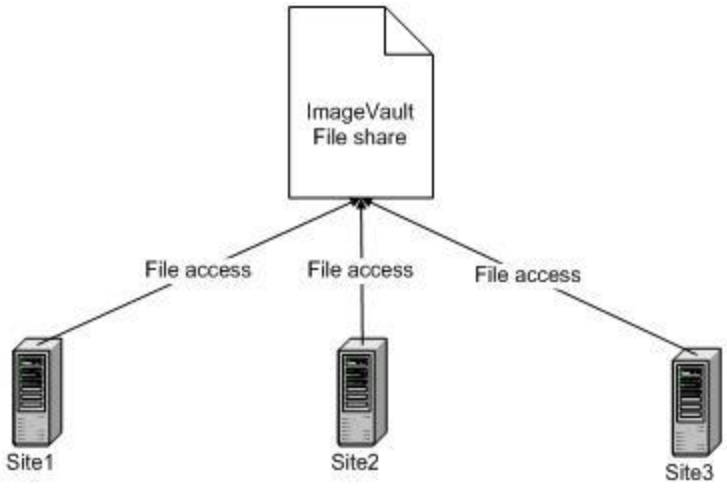
Install ImageVault on all servers. During the first installation create a database and in the following installations, select the database the first installation created. Note that you should use the exactly the same version of ImageVault on all the sites.



7.1.2. File share configuration

All the servers must have access to a common file share where the files will be stored. To change the location, modify this attribute in the web.config.

```
ImagePathOriginal="\\MyServer\ImageStoreNET\Data\ImageDB"
```



Note that the user running the .Net process on the servers needs to have read and write access to the file share.

7.1.3. Site and user configuration

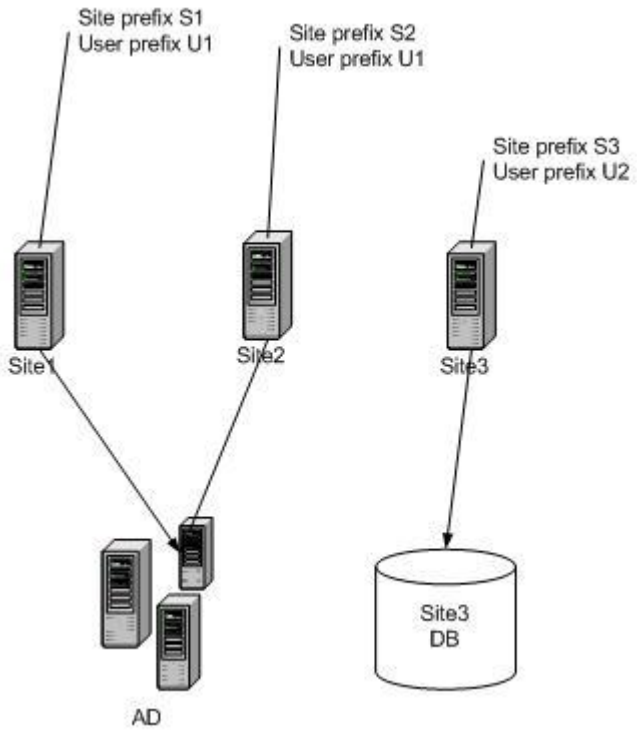
To be able to identify the servers each server must be configured whit a unique identifier, called site prefix. Depending on how users should be authenticated each site may also need a unique identifier for the users, this is called user prefix below.
The site prefix and user prefix has to be specified in each sites web.config. The prefixes should be added as attributes to the ImageVault node in web.config.

```
<imageVault UserPrefix="" SitePrefix="" …
```

The site prefix has to be unique for each site. If a site is running on more than one server, i.e. load balancing, then those servers should have the same site prefix.

The user prefix has to be unique for each user directory that is used. This is best illustrated with an example.

If we have tree sites S1, S2 and S3. S1 and S2 are using the same AD for authentication while S3 is using a local database. The S1 and S2 should have the same user prefix, and S3 would have a unique value.



Note that you should only set the user prefix to the same value if you want album rights to be the same for both sites. By setting different prefixes on site S1 and S2 you can set user rights independent on each server.

It's important that the settings for user and site prefix are made before ImageVault is taken into use. Applying prefixes to an ImageVault installation that is already in use will cause strange behavior regarding user rights and when tracking the usage of images. It's also important that all sites that are using the database has a site prefix and a user prefix. Otherwise it will not know that it is a shared database and might remove rights that are set by the other sites.

It is possible to set a site and user prefix for an already existing ImageVault site but that will require changes to the database and that the site is taken offline. There is a sql-script available to assist in making theses changers. See the comments in the script file for more information. The script MultipleEPiServersUpdate.sql can be downloaded from www.meridium.se

7.1.4. Cache listener configuration

It is important to set up cache listeners between the servers sharing the same ImageVault database to ensure that changes made in ImageVault on one of the EPiServer instance is shown correctly on the other EPiServer instances sharing the same database.

Se section 6 for instructions on configuration.

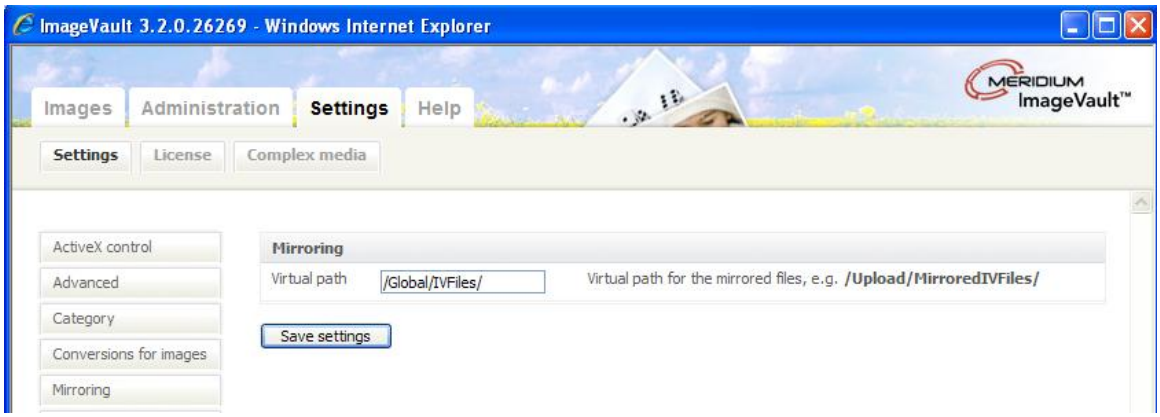
8. Mirroring

Older versions of ImageVault had limited support for mirroring.
Images used in the EPiServer free text editor could be exported to an image stored in the page folder on the receiving site.

From version 3.2 ImageVault can support mirroring of images to another site with ImageVault installed. In this case images, albums, metadata, conversion formats and categories for images published on pages will be replicated to the receiving site. This could also be used to export/import pages including ImageVault files to another site.

To activate this feature set the attribute ExportImageVaultProperties to true in the <ImageVault> tag in web.config on BOTH the sending and receiving site.

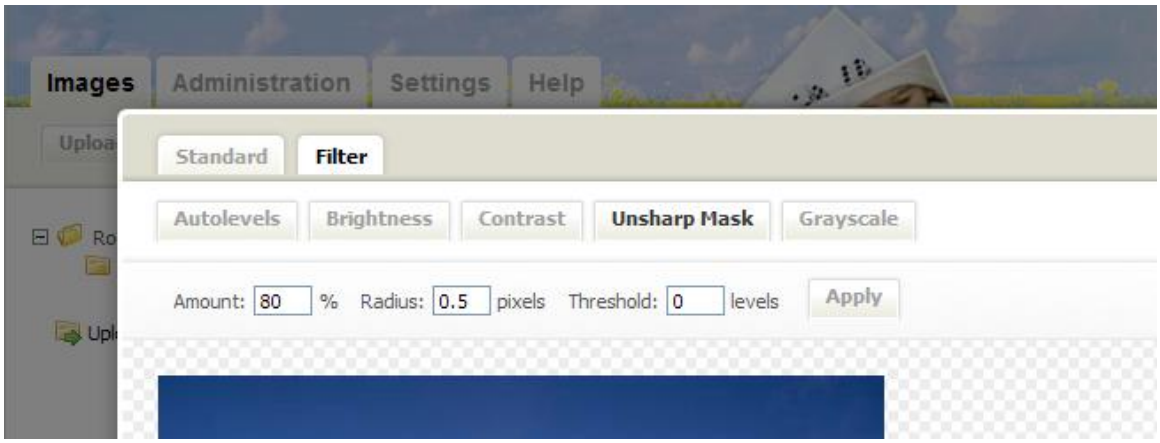
Also make sure that the "Virtual path" setting under "Mirroring" is set to the same value on both sites.



Note! This feature is only available in the CMS version of ImageVault.

9. Image Editor

It is possible to change the default values used in the filter section in ImageVaults Image Editor.


Images
Administration
Settings
Help
Standard
Filter
Autolevels
Brightness
Contrast
Unsharp Mask
Grayscale
Amount: 80 %
Radius: 0.5 pixels
Threshold: 0 levels
Apply

This is done in the database in the table IS_SystemSettings.

settingName	settingValue
UnsharpMaskSettings	Radius=0.5|Amount=80|Threshold=0
ContrastSettings	Amount=0
BrightnessSettings	Amount=0

These settings could also be made accessible in the admin section of ImageVault by setting the column systemSettingsGroup_id to a value corresponding to a value in IS_SystemSettingsGroups.

10. Other Configuration

All other configuration is managed from the System configuration interface in ImageVault (see Administrator Manual).