

MediaRich[®] SERVER

MediaRich CORE 6.2

Installation and Administration Guide

Published: September 2020



© 2004-2020 Automated Media Processing Solutions, Inc. dba Equilibrium.

All Rights Reserved. U.S. Patent Numbers 6,964,009, 8,381,110 and 8,656,046 for the automated content processing, automatic derivative content management, dynamic on-the-fly visualization of content, bandwidth and device dependent optimization of visual media, on-the-fly auto-conformance and assembly of video content and dynamic on-demand generation of digital media assets. Other patents pending. MediaRich and Equilibrium are registered trademarks and EQ Network, MediaScript, MediaBatch, MetaBatch MediaCart, and AMPS are trademarks of Automated Media Processing Solutions, Inc. All other company, product or service names mentioned herein are the property of their respective owners.

MediaScript contains ScriptEase™, a JavaScript compatible, ECMAScript compliant interpreter developed by Nombas, Inc., <http://www.nombas.com>. All Rights Reserved.

Contents

CHAPTER 1 MediaRich CORE Overview	5
The MediaRich Platform	6
Deploying and Administering MediaRich	6
MediaRich Architecture	7
CHAPTER 2 Installing MediaRich Servers	11
Before You Install	12
MediaRich CORE Platform Installation	15
Setting Up the Initial Server Configuration	17
CHAPTER 3 Upgrading MediaRich Servers	19
Performing a Minor Upgrade	20
Performing a Major Upgrades	22
CHAPTER 4 Managing MediaRich Servers	27
Working with the Administration Utility	28
Performing Server Administration Tasks	35
Modifying Additional Properties in a Text File	49
Using Command Line Administration Tools	56
APPENDIX A File Format Support	59
MediaRich Image File Formats Natively Supported	60
MediaRich Audio and Video File Formats Natively Supported	64
LibreOffice Expanded File Formats Supported	67

Office File Formats Supported	67
APPENDIX B Administration Best Practices and Troubleshooting	71
Best Practices for Installation and Configuration	72
Troubleshooting the MediaRich Server	74

MediaRich CORE Overview

The MediaRich CORE (Content On-demand Repurposing Engine) Platform is designed to work with your existing infrastructure, integrating seamlessly with existing network servers and Web systems, as well as asset management solutions, HTML authoring tools, Content Management Systems, and Content Distribution Networks.

If you are licensing the MediaRich Media Server, MediaRich Merchandizer, MediaRich ECM for SharePoint, or a combination of these media processing server products, use the MediaRich CORE Installer to install the MediaRich CORE Platform.

Video, imaging, documents, and Adobe .pdf, .eps, and .ai files are now available for rendering by the MediaRich CORE.

Important: The initial release of MediaRich CORE 4.0, which includes A/V CORE 2.0, is only available in the Windows Server Product at this time. References to other operating systems within this document are correct for MediaRich functionality; however, support for additional operating systems that include A/V CORE 2.0 will not be available until the 4.1 cross-platform release.

This chapter explains the basics of the MediaRich CORE Platform architecture, the specific server products built upon it, and the MediaRich CORE Administration Utility.

Chapter summary

“The MediaRich Platform” on page 6

“Deploying and Administering MediaRich” on page 6

“MediaRich Architecture” on page 7

The MediaRich Platform

The MediaRich suite of client/server products is built on a common server platform called the MediaRich CORE. The MediaRich CORE is a scriptable, scalable, and extensible server architecture that implements and executes media processing requests initiated by a client application.

Two client/server solutions are currently offered in the MediaRich suite; MediaRich Merchandizer and MediaRich ECM for SharePoint. These products provide targeted solutions for specific image processing needs.

The MediaRich suite also includes the MediaRich Media Server. With this product, application developers and solution integrators can utilize the power of MediaRich to build their own media processing solutions or to add the power of MediaRich to existing enterprise workflows.

MediaRich ECM for SharePoint

This product is a high-volume media processing distribution and re-purposing server engine that seamlessly integrates with Microsoft SharePoint Server to create a powerful self-service Digital Asset Management (DAM) and visualization system.

MediaRich Merchandizer

Web servers utilizing MediaRich Merchandizer can provide custom zoom and pan capabilities by adding simple JavaScript and proprietary Resource Locators within the Web pages, which instruct Merchandizer to generate images from source image files and deliver them directly to the browser.

MediaRich Media Server

Application and Web site developers can use the MediaRich Media Server to deliver dynamic imagery and rich content to users. Images can be produced on the fly, based on multiple input parameters. The MediaRich Media Server can also be used as a batch processing system for large numbers of derivative assets. MediaRich Media Server incorporates a caching technology that identifies frequently requested objects and caches them locally in the MediaCache folder, which speeds up the delivery of media.

Deploying and Administering MediaRich

Solutions built on the MediaRich CORE platform will often consist of multiple components requiring multiple installation procedures, often on multiple machines.

The MediaRich server side for all four solutions is delivered via a single installer package called MediaRich CORE. Regardless of which MediaRich solution you have purchased or are evaluating, begin by installing this package.

MediaRich servers are designed to work either individually or in a clustered deployment. In a clustered solution, multiple MediaRich server machines access a common file repository to share

both source and derivative assets. A shared cache allows all machines in the cluster to instantly serve derivatives that have already been generated by another server in the cluster.

The MediaRich Server is configured using the MediaRich CORE Administration Utility. This utility is launched automatically at the end of the installation process on Macintosh and Windows servers. It is used to install license files and to configure and start and stop MediaRich servers.

For Linux systems, you can use the Administration Utility in a Linux graphical desktop environment. Information about the final setup steps for MediaRich CORE, including how to run the administration tool, is also displayed at the end of the MediaRich installation on Linux.

MediaRich Architecture

The following is an overview of the components of the MediaRich CORE Platform.

Shared folder

The shared folder is where MediaRich obtains global properties, scripts, fonts, ICC profiles, media, and writes logs. A single shared folder can be used by multiple MediaRich servers. Alternatively, each MediaRich server may have its own copy of the shared folder. The choice of whether to use a single shared folder or to have separate copies is up to the system administrator. Throughout this document, the shared folder is referred to as MediaRichCore/Shared. The exact location depends on platform and installation.

Services

The MediaRich CORE comprises two services: the MediaRich CORE Media Generator and the MediaRich CORE System Monitor. On MacOS (10.5 or greater) and Linux, these services run as a special user named “mediarich”. On Windows, these services can be run as LocalSystem, or for greater security you can choose to run them as a specific user for image generation. The user requires read and write access to the shared folder and all its subfolders.

Web server extension

MediaRich has the ability to operate behind a standard HTTP server, allowing it to serve URL requests via the standard HTTP protocol. To do this, an extension is installed on the Web server that will forward these requests to MediaRich.

On Windows, MediaRich is designed to operate behind IIS, Microsoft’s Web Server. The URLs are processed through an IIS filter that forwards MRL requests to the MediaRich CORE Media Generator. For security reasons, this filter impersonates the IIS anonymous user and has only the security privileges of that user. Therefore the user designated as the IIS anonymous user must have write access to the Logs subfolder of the shared folder.

On MacOS X and Linux, MediaRich can operate behind versions 1.X, 2.0, or 2.2 of the Apache Web Server. Extensions, called Apache MODs, are provided for each of these environments. These extensions are run as the “apache” user. This user must have write access to the Logs subfolder of the shared folder.

Virtual file systems (VFSs)

The MediaRich Server implements a file system abstraction by defining Virtual File Systems (VFSs) that map a symbolic name to a particular place on the server's physical filesystem. The VFS system is extensible, allowing any number of VFSs to be defined by the MediaRich administrator.

Each VFS mapping is given a unique name. In addition, a VFS definition specifies if MediaRich can write to and/or execute scripts on that portion of the filesystem. For more information about the use of Virtual File Systems, refer to the *MediaRich CORE Programmer's Guide*.

Important: Windows "Map drive" feature is unsuitable for using with Service applications for two reasons:

1. Drive mappings are stored per-user.
2. Drive mappings are re-established at the beginning of a user's interactive session.

Even though the service can be set to run as a particular user, services do not usually initiate an interactive session, so the drive mappings will still not be available. Combined, these issues make mapped drives unsuitable for use with Service applications.

Log files

MediaRich maintains several log files containing information about errors that could occur during image processing. The locations of these log files are as follows:

- **MediaGeneratorStartup.log** (Windows only) - this log file is located in the MediaRich installation directory. It logs the success or failure of the Web Server Extension to impersonate the IIS Anonymous user. This file will not be created unless the user running the W3SVC service has write access to this directory.
- **MediaGenerator.log** - this file is located at `MediaRichCore/Shared/Logs/<computername>`, where `<computername>` is the network name for the server. Each image server connected to `MediaRichCore/Shared` has its own log file in such a folder.
- **ScriptErrors.log** - this is located at `MediaRichCore/Shared/Logs`.

Note: Individual products might create their own log files, which will also appear in the Logs directory.

The easiest way to view these log files is with the MediaRich Administration Utility, which is accessible from the **Logs** menu.

The MediaRich dependency system

The dependency subsystem of the MediaGenerator allows you to manage outputs in your cache based on when the inputs were updated. The system is composed of two parts:

- **Dependency Tracking** - During dependency tracking, the MediaGenerator records files that were used as inputs (including scripts) for a given output and when those input files were last updated. For more information about configuring the Dependency Tracking properties, see [“Viewing and Modifying the Global Properties” on page 39](#).
- **Dependency Checking** - This is the process of checking the tracked dependency information against the file system to see if any inputs were updated. If they true, the corresponding outputs are removed from the filesystem. Consequently, the next time the MRL that generated those outputs is referenced, the output is regenerated using the updated inputs. For more information about Dependency Checking, see [“CheckDependencyTool” on page 56](#).

Dependency tracking is not enabled by default because most MediaRich installations do not need it. The MediaRich Caching system can be used without it and managed through other mechanisms. There is a performance cost to using the Dependency System and it is currently only suitable for small MediaRich installations, such as a single Generator. The CheckDependencyTool should be run during times of light or no Generator load; otherwise, you could encounter errors because the system is in use by the Generator for Dependency Tracking. If you do enable and configure Dependency Tracking, you should investigate using your operating system’s task scheduler to periodically run the CheckDependencyTool to keep the amount of tracked dependency information manageable.

Installing MediaRich Servers

The MediaRich CORE Installer installs the MediaRich CORE Platform. If you are licensing the MediaRich Media Server, the server should be installed on the server where original graphics and MediaScripts will be uploaded for the live Web site.

This chapter provides information about the system requirements for MediaRich and explains the step-by-step installation process. The following sections provide a detailed description of the installation process. You should read these instructions carefully before starting the installation.

Chapter Summary

“Before You Install” on page 12

“MediaRich CORE Platform Installation” on page 15

“Setting Up the Initial Server Configuration” on page 17

Before You Install

Installation of MediaRich CORE must be performed by an administrator for the server. System requirements may vary depending on server load. Before installing MediaRich you should carefully plan your configuration.

Windows Installations

Minimum recommended configuration for imaging, documents, office applications (no heavy video transcoding, 4k or UHD video transcoding):

- Dual 3+ GHz Intel® Xeon Sandy Bridge and later processors
- 8 GB memory
- Microsoft Windows Server 2012R2 and later
- Microsoft Internet Information Services® (IIS) and ASP.NET with Sub-components

Minimum recommended configuration when processing large HQ videos such as 4k, UHD or 1080P, images, multi-page documents (Bare metal is highly recommended for video transcoding!):

- 8 Core Intel Xeon Sandy Bridge and later
- Dedicated MediaRich Server 8 Core License in shared cache configuration
- Dedicated cache volume 4 x 250 GB Disk 15K SAS RAID-10
- OS-Volume 2x150 GB/SAS/10K mirror
- 32 GB Ram

HARDWARE NOTE: If you intend on handling very large videos in HD 720P or higher, and are looking to maximize throughput with MediaRich's optional GPU capability in the future, we HIGHLY recommend purchasing Intel Sandy, Ivy or Haswell capable chipsets. This will provide the best option for increasing throughput in the future!

Linux Installations

Minimum recommended configuration:

- Dual 3.0 GHz P4 processors
- 4 GB memory
- Red Hat ES 4 or Red Hat EL 5 (Red Hat 3 not supported)
- Apache 1.x, 2.x, or 2.2.x

Obtaining a License Key

In order for a MediaRich server installation to function, you need a license key from Equilibrium. The license key sets various properties in MediaRich and determines which MediaRich products are enabled. After installation, use the administration utility to install the license key .bin file and activate the licensed product(s). If you do not already have a license key (usually named “license.bin”), contact 1-855-EQUILIB or +1 415-332-4343 and Dial 4 for Product Information and Sales Support.

In addition to the primary products, iKits can be purchased and enabled in the license key to provide advanced functionality in the server.

Digimarc ImageBridge 4.0 iKit

This iKit enables easy integration between MediaRich and Digimarc technologies. An Embed function can be added to a MediaScript to enable every image generated by the server to be protected by the Digimarc forensic watermarking technology. A watermarked image can communicate copyrights, enhance digital asset management, and be tracked across the Internet.

This is currently available and supported on Windows, Macintosh, and Linux.

Important: Both the Digimarc iKit from Equilibrium, as well as, Digimarc Server + Maintenance licenses are required for the Digimarc Imaging Server to operate using MediaRich Server. Equilibrium is an authorized Digimarc reseller and can fulfill all Digimarc Server needs. If you already have your Digimarc License directly with Digimarc, please call Equilibrium to purchase the Digimarc iKit that enables the automatic Digimarc processing to occur within a MediaRich MediaScript environment.

Enabling Office (LibreOffice) Document Support

The MediaRich Media Server has the ability to dynamically image Microsoft Office and other Office-related file types and process them the same way MediaRich can for other files. It can save these processed images to any supported image file format and it can dynamically zoom and pan on these files for remote viewing using the UniZoom universal viewer.

Note: When MediaRich is running, LibreOffice cannot be used as a separate application.

MediaRich support for Office documents and other file types will not be available unless LibreOffice is installed. Libre office 6.0 or later is required with the current versions of MediaRich.

You can install/reinstall LibreOffice at any time after MediaRich is installed (or before you install MediaRich, if desired). If you change the LibreOffice installation after MediaRich is installed, you must use the MediaRich CORE Administration utility and stop and start MediaRich. This is not required if you are simply upgrading LibreOffice—MediaRich restart is only required when going from an installed -> uninstalled state or an uninstalled -> installed state with LibreOffice.

To install LibreOffice on a Windows system:

- From <http://www.libreoffice.org>, navigate to and download LibreOffice for your system.
- In the window that prompts for the type of installation you wish to perform, choose a Custom setup.
- In the Installation Wizard, clear the selections for the following Optional Components:
 - Online Update
 - Active-X Control
 - QuickStarter
- Click **Next** > and proceed with the installation.

Starting LibreOffice Functionality

After you install MediaRich it will automatically enable LibreOffice. It is a good idea to reboot after this initial run. On the Macintosh, you must be logged on to the account where you run LibreOffice as a service for MediaRich, and stay logged on, to enable the LibreOffice functionality. On a Windows system where MediaRich is already installed, you install LibreOffice and then stop and start MediaRich.

Note: `OpenOffice.LockExpireTime` is now available in `local.properties`. This function defaults to 10 minutes if not present. This feature was put in to cover any “unforeseen” issues with the sometimes unstable Open/LibreOffice 6.0 application which runs as a “headless server” and enables MediaRich to provide 200+ file types to be rendered and thumbnailed. This property assures that LibreOffice is restarted in the event it stops functioning for whatever reason. Even though errors could show in the logs, files are automatically re-attempted—processing of all files should automatically complete and do not need to be re-submitted.

IIS Configuration on a Windows Server 2012 or Later

Before you install MediaRich on a Windows server 2012 or later, you must configure the server features so that they are compatible with MediaRich; otherwise, the installation of MediaRich Server will not work properly.

Important: These steps are required only for using MediaRich in conjunction with Internet Information Services (IIS). Installations of CORE for only Hot Folder operations or only for the Java/C#/C++ API do not require these steps.

To configure the Roles and Features:

MediaRich for Windows requires that the IIS Application Server, including the ASP.NET Subcomponent, be installed and enabled for some licenses. Specifically, using MRLs or the AVCORE 2 Settings File Generator requires IIS. Also, many of the Merchandizer demos require IIS.

Use the Windows Server Manager to install:

IIS Updates:

- IIS
- IIS 6 Scripting
- IIS 6 WMI Compatibility
- IIS 6 Metabase Support.
- ASP.NET with sub-components

MediaRich CORE Platform Installation

The MediaRich CORE Platform is designed for deployment across a shared network. A shared network deployment also allows the server to receive requests from Web servers utilizing MRLs, generate the requested image from an original source file, and return the image to the Web browser.

Creating a Service User for the MediaRich Server (Windows)

We recommend using a Domain User for this Service. At the Domain level the user could be a standard User. However, they should be part of the local Admin Group where the MediaRich Server will be installed.

This Service account is requested during the Installation of the MediaRich Server and will be used to Monitor and installed the Services Server.

Also, the user running as MediaRich must have service access rights. These rights can be enabled for an account by opening the Group Policy editor, going to Computer Configuration > Windows Settings > Security Settings > Local Policies > User Rights Assignment > Log on as a service and adding the account to the list of service users.

Running the MediaRich Server Installer

Important: For installation on Windows Server 2012 or later, the server must be configured according to the guidelines described in [“IIS Configuration on a Windows Server 2012 or Later” on page 14](#) BEFORE installing MediaRich Server; otherwise, the installation of MediaRich Server will not work properly.

After running the installer, the MediaRich CORE Administration Utility appears, with a Host Name of localhost.

MediaRich Installer (Windows)

The MediaRich CORE installer is supplied as a standard Windows installer package. To begin the installation, double-click on the installer package to run it. The installer will walk you through the installation process.

Important: If you choose to run MediaRich CORE as a specific Service User, be aware that, for security reasons, this dialog does not check the validity of either the account name or the password. It only checks the password against itself so that you can be assured that it was entered correctly. This dialog does not guarantee that the password itself is the correct one or that the account name you choose has adequate permissions.

The Windows installer can install MediaRich for the following languages: English, German, French, and Dutch.

To install the MediaRich on a Windows server:

1. Run the Server Setup with Admin Privileges.
The Setup will request a User Account.
2. Use the User you created for the MediaRich Server.
For example, <Domain>\User and the Password.
3. Complete the initial configuration in the MediaRich CORE Administration Utility (see “Initial Configuration” on page 17).

MediaRich Installer (Mac)

For Mac installations, the MediaRich CORE installer is supplied as a standard MacOS X (10.5 or greater) installer package. To begin the installation, double-click the installer package to run it. The installer will guide you through the installation process.

MediaRich Installer (Linux)

For Linux installations, the MediaRich CORE installer is supplied as a standard Red Hat Package Manager (RPM) package file. To perform the installation, open a terminal window, navigate (cd) to the directory containing the MediaRich installer package, and execute the following command line while logged in as the “root” user:

```
rpm -i MediaRichUniv_3.8-4.XXX.i386.rpm
```

At the end of the installation process, instructions for completing the installation process are displayed to the terminal window, as well as in the Installation file. Follow these instructions to complete the installation.

MediaRich Permissions (MacOS X and Linux)

On the MacOS X (10.5 or greater) and Linux platforms, the MediaRich Server runs with the permissions of the “mediarich” group. To allow applications running as other users access to the

MediaRich Server shared folder, add those users to the “mediarich” group. To allow other users to administer the server, add those users to the “mediarich” server group.

Installing FFmpeg for Video Transcoding (Optional)

Installing FFmpeg is only required if you will be transcoding videos into a Flash-based output. It is not required to upload, ingest, or play Flash videos with MediaRich CORE Server. MediaRich Server must be fully installed and operational prior to executing the FFmpeg install.

To install FFmpeg for the MediaRich server:

1. Download the Windows version of ffmpeg.exe

<http://ffmpeg.zeranoe.com/builds>

Use the static 32-bit build (even on a 64-bit OS). Generally, the build with the most recent timestamp will work.

2. Open the tarball using 7-Zip or another archive utility.
3. Navigate to the “bin” folder and copy the ffmpeg.exe file.
4. Add the file to ..\MediaRichCore\Bin\MediaEngine folder.

The default path is C:\Program Files\Equilibrium\MediaRichCore\Bin\MediaEngine

Note: If you receive a (missing DLL error) message during the FLV transcodes, it means you could have installed the shared library version. You must go back and install the correct one.

Setting Up the Initial Server Configuration

Before proceeding with licensing your MediaRich installation, ensure that the service user you selected during installation has Read/Write permission for the MediaRichCore\Shared folder and all subfolders. If installing the Web Server extension, also ensure that the anonymous user for the root of the default Web server has read permission for the MediaRichCore\Shared folder and all subfolders, and write permission for the MediaRichCore\Shared\Logs folder.

Initial Configuration

When you first install MediaRich server, complete the following tasks to enable the system:

1. If it is not already open, launch the MediaRich CORE Administration Utility by double-clicking **MediaRich CORE Admin** in the Equilibrium\MediaRich CORE Admin folder.
2. Login using the default credentials: User **admin**, Password **admin**.
3. When the User Interface appears, select **Install a License File**.
4. Locate and select the unzipped license file.
5. Click the **Configure** button to open the MediaRich Server Options dialog.
6. Check if the Virtual Drives points to the desired Drive you like to use for the Server Cache:
 - VideoProxy: file Location for the preview Clips

- batchout: target location for the MediaCart Results
 - merchandizer (no changes needed)
 - mrdocs (no changes needed)
 - pyramid: file location for the Pyramid Zoom files
7. Click **Start Server**.
- If the three indicators are green, the Server portion of the installation was successful.



Figure 1: Successful start of the server

You can make additional configuration changes and additions after you verify a successful installation. To make these changes, you must stop the server. Click **Stop Server**; the status of the server will change when the server is stopped.

For more information about server configuration, see [“Working with the Administration Utility”](#) on page 28.

Upgrading MediaRich Servers

If you are upgrading an existing MediaRich 4.0.x Server, you will use the MediaRich CORE Installer to install the 6.2 (or later) release of the MediaRich CORE Platform. However, there are additional steps to take to ensure that you retain your existing settings and modified files.

Important: You should ensure that your server meets the system requirements for MediaRich CORE 6.2 and perform any system upgrades that are required before upgrading MediaRich. For more information, see [“Before You Install” on page 12](#).

Before you perform a MediaRich upgrade, you must copy the .zip file to the MediaRich CORE server.

- Log into the server with the appropriate administrator permissions.
- Copy the file to the server.
- Extract the .zip file on the server.

Also, review the included README file. This file contains important information about the MediaRich CORE installation and changes to the system.

This chapter provides information about major and minor upgrade processes for MediaRich servers.

Chapter summary

[“Performing a Minor Upgrade” on page 20](#)

[“Performing a Major Upgrades” on page 22](#)

Performing a Minor Upgrade

A minor upgrade involves updating your MediaRich server from a previous 4.0 version. This procedure varies depending on your existing file structure and level of customization. Review both to determine which is appropriate for your system.

Minor-Simple Upgrade

For most installations of MediaRich 6.2, where everything resides within the `..\Equilibrium` folder and where the two `.properties` files were NOT manually-edited, you can use the simplest upgrade procedure.

To perform a minor-simple upgrade of a MediaRich CORE server:

1. Review your current installation to make sure that you know where all of its components are located, as well as any MediaRich settings that you changed to accommodate those locations or any other changes to the settings changes from the defaults.

2. Uninstall the old version.

If the original MediaRich installer is still on the server, you can run it again and choose uninstall/remove. Alternatively, you can use the Programs and Features control panel.

All custom files, default MediaRich files that are modified, and other important files are left behind after the uninstall. In a default installation, this location is the `..\Program Files\Equilibrium` folder.

3. Install the new MediaRich CORE version.

For more information, see [“Running the MediaRich Server Installer” on page 15](#).

4. Open the Administration utility and confirm that the configured settings for the previous version are intact.

5. Install the license.

For more information, see [“Installing a License File” on page 30](#).

If you do not have the original license file as shipped to you from Equilibrium Sales, you can locate this file inside the `..\Shared\Properties` folder. It is named `"license.bin"`.

Note: You should not copy the old `.properties` files to overwrite the new ones in their respective Properties folders. The contents of these files change from version to version, and replacing them with out-of-date versions will lead to unexpected behavior.

6. Start the MediaGenerator.

It is a best practice to reboot the server as well.

Minor-Complex Upgrade

For more complex installations of MediaRich 6.2, where everything does not reside within the `..\Equilibrium` folder, and where the two `.properties` files were manually-edited, you should use the complex upgrade procedure.

To perform a minor-complex upgrade of a MediaRich CORE server:

1. Review your current installation to make sure that you know where all of its components are located, as well as any MediaRich settings that you changed to accommodate those locations or any other changes to the settings changes from the defaults.

2. Uninstall the old version.

If the original MediaRich installer is still on the server, you can run it again and choose uninstall/remove. Alternatively, you can use the Programs and Features control panel.

All custom files, default MediaRich files that are modified, and other important files are left behind after the uninstall. In a default installation, this location is the `..\Program Files\Equilibrium` folder.

3. Move the `..\Equilibrium` folder to a safe location.

Note: Cached data can take up a considerable amount of space, so having fast access to a large amount of space will make this step, along with many of those to follow, much easier.

4. Delete the old installation (Equilibrium) folder.
5. Install the new MediaRich CORE version but do not apply the license.
For more information, see [“Running the MediaRich Server Installer” on page 15](#).
6. Locate the SavedPropertyFiles folder (inside the Equilibrium folder you moved) and move it to another location outside of the Equilibrium folder.

Important: Make sure to **move** the folder—do not copy it.

7. Copy all remaining contents of the old, saved Equilibrium folder to the new Equilibrium folder, overwriting everything.

Note: Any Hot Folder MediaScripts created in Hot Folder locations will still be there, if they were not located under the original `..\Equilibrium` folder.

8. Transfer the property settings from the old files to the new:
 - Navigate to the SavedPropertiesFiles folder in its new location and open the old `.properties` files in a text editor.
 - Open the new ones in their respective subfolders in the new installation.
 - Copy only the changed settings from the old file to the new file.

Note: This step can be skipped if the `.properties` files were not manually-edited. Do NOT copy the old files over the new ones because the contents of these files change from version to

version, and replacing them with out-of-date versions will lead to unexpected behavior.

9. Open the Administration utility and confirm that the configured settings for the previous version are intact.

10. Install the license.

For more information, see [“Installing a License File” on page 30](#).

If you do not have the original license file as shipped to you from Equilibrium Sales, you can locate this file inside the `..\Shared\Properties` folder. It is named "license.bin".

Note: You should not copy the old .properties files to overwrite the new ones in their respective Properties folders. The contents of these files change from version to version, and replacing them with out-of-date versions will lead to unexpected behavior.

11. Start the MediaGenerator.

It is a best practice to reboot the server as well.

Performing a Major Upgrades

A major upgrade involves updating your MediaRich server from a previous 3.9.x version. This procedure varies depending on your existing file structure and level of customization. Review both to determine which is appropriate for your system.

MediaRich CORE Version 6.2 is 64-bit only, while version 3.x is 32-bit only. If you did not make changes to IIS to accommodate the 32-bit requirement for version 3.x, no changes to IIS are required. (Most Hot Folder and MediaRich ECM for SharePoint licenses do not include Web capabilities, so the changes to IIS might not have been required.) However, you should check the status of the server before moving on to the overinstall of MediaRich itself.

Important: Do **not** enable 32-bit IIS worker processes when using 64-bit MediaRich; it will cause all MRL access to fail. If you did this in the past for previous versions of MediaRich, go into the IIS manager and locate the "Application Pools" entry in the middle pane. Go through each application pool entry, select "Advanced Options" and make sure that 32-bit mode is set to false. Also, locate the "Application Pool Defaults" action in the right pane and set that flag to false. Reboot the machine after changing these settings.

LibreOffice Support

The last 3.9.x MediaRich version supported no later than LibreOffice version 4.1.4. That LibreOffice version is now superseded by later versions. As of this writing, MediaRich 4.x supports LibreOffice version 6.0. You might need to uninstall the old LibreOffice version and install the new one for continued support of Office-type files in MediaRich.

Legacy QuickTime Support

In MediaRich 4.0, Equilibrium is introducing our new A/V Core 2.0, which initially supports the latest ffmpeg and Quicksync GPU transcoding. However, for historical workflows, or in order to use any of the QuickTime video features on the Windows platform, such as specific plug-ins available for QuickTime, you must ensure that QuickTime is installed. You can download the QuickTime player for Windows at <http://www.apple.com/quicktime>. Run the setup file to install QuickTime. The MediaRich server may need to be restarted after installation.

Major-Simple Upgrade

For most installations of MediaRich 3.9.x, where everything resides within the ..\Equilibrium folder and where the two .properties files were NOT manually-edited, you can use the simpler major upgrade procedure.

To perform a major-simple upgrade of a MediaRich CORE server:

1. Before you start, make sure you have your NEW license file.
3.9.x licenses do not work with MediaRich CORE 6.2.
2. If you use LibreOffice as a helper to MediaRich, uninstall the old LibreOffice version and install the new 6.0 version.

For more information, see “Enabling Office (LibreOffice) Document Support” on page 13.

3. Review your current installation to make sure that you know where all of its components are located, as well as any MediaRich settings that you changed to accommodate those locations or any other changes to the settings changes from the defaults.
4. Uninstall the old version.

If the original MediaRich installer is still on the server, you can run it again and choose uninstall/remove. Alternatively, you can use the Programs and Features control panel.

All custom files, default MediaRich files that are modified, and other important files are left behind after the uninstall. In a default installation, this location is the ..\Program Files (x86)\Equilibrium folder.

5. Install the new MediaRich CORE version.
For more information, see “Running the MediaRich Server Installer” on page 15.
6. Copy or move the folder named "MediaRichCore" (found under the ..\Program Files (x86)\Equilibrium folder) to ..\Program Files\Equilibrium.

Allow Windows to replace all files.

Note: Any Hot Folder MediaScripts created in Hot Folder locations will still be there, if they were not located under the original ..\Equilibrium folder.

7. Open the Administration utility and confirm that the configured settings for the previous version are intact.
8. Install the new 6.2 license.

For more information, see “Installing a License File” on page 30.

9. Start the MediaGenerator.

It is a best practice to reboot the server as well.

10. When you confirm the proper operation of MediaRich, you can delete the old Equilibrium folder within `..\Program Files (x86)`.

Major-Complex Upgrade

For more complex installations of MediaRich 3.9.x, where everything does not reside within the `..\Equilibrium` folder and where the two `.properties` files were manually-edited, you should use the complex upgrade procedure.

To perform a major-complex upgrade of a MediaRich CORE server:

1. Before you start, make sure you have your NEW license file.
3.9.x licenses do not work with MediaRich CORE 6.2.
2. If you use LibreOffice as a helper to MediaRich, uninstall the old LibreOffice version and install the new 6.0 version.

For more information, see [“Enabling Office \(LibreOffice\) Document Support” on page 13](#).

3. Review your current installation to make sure that you know where all of its components are located, as well as any MediaRich settings that you changed to accommodate those locations or any other changes to the settings changes from the defaults.
4. Uninstall the old version.

If the original MediaRich installer is still on the server, you can run it again and choose uninstall/remove. Alternatively, you can use the Programs and Features control panel.

All custom files, default MediaRich files that are modified, and other important files are left behind after the uninstall. In a default installation, this location is the `..\Program Files \Equilibrium` folder.

5. Move the `..\Equilibrium` folder to a safe location.

Note: Cached data can take up a considerable amount of space, so having fast access to a large amount of space will make this step, along with many to follow, much easier.

6. Install the new MediaRich CORE version but do not apply the license.
For more information, see [“Running the MediaRich Server Installer” on page 15](#).
7. Locate the SavedPropertyFiles folder (inside the Equilibrium folder you moved) and move it to another location outside of the Equilibrium folder.

Important: Make sure to **move** the folder—do not copy it.

8. Copy all remaining contents of the old, saved Equilibrium folder to the new Equilibrium folder, overwriting everything.

Note: Any Hot Folder MediaScripts created in Hot Folder locations will still be there, if they

were not located under the original `..\Equilibrium` folder.

9. Transfer the property settings from the old files to the new:
 - Navigate to the `SavedPropertiesFiles` folder in its new location and open the old `.properties` files in a text editor.
 - Open the new ones in their respective subfolders in the new installation.
 - Copy only the changed settings from the old file to the new file.

Note: This step can be skipped if the `.properties` files were not manually-edited. Do NOT copy the old files over the new ones because the contents of these files change from version to version, and replacing them with out-of-date versions will lead to unexpected behavior.

10. Open the Administration utility and confirm that the configured settings for the previous version are intact.
11. Install the new 6.X license key.
For more information, see [“Installing a License File”](#) on page 30.
12. Start the MediaGenerator.
It is a best practice to reboot the server as well.
13. When you confirm the proper operation of MediaRich, you can delete the old Equilibrium folder within `..\Program Files (x86)`.

Managing MediaRich Servers

The MediaRich Media Server is the engine for all MediaRich functionality. It is used for processing each image and movie file processing request, as well as a batch processing system for large numbers of derivative assets. MediaRich Media Server incorporates a caching technology that identifies frequently requested objects and caches them locally in the MediaCache folder to speed up the delivery of media.

This chapter provides detailed information about managing MediaRich servers, setting options for access to multiple servers, and managing cached media objects.

Chapter summary

“Working with the Administration Utility” on page 28

“Performing Server Administration Tasks” on page 35

“Modifying Additional Properties in a Text File” on page 49

“Using Command Line Administration Tools” on page 56

Working with the Administration Utility

On the MediaRich server, use the MediaRich CORE Administration Utility to perform the various administrative tasks, including the following:

- Configuring the server
- Managing servers and server properties
- Viewing server fonts and logs
- Managing administrative access

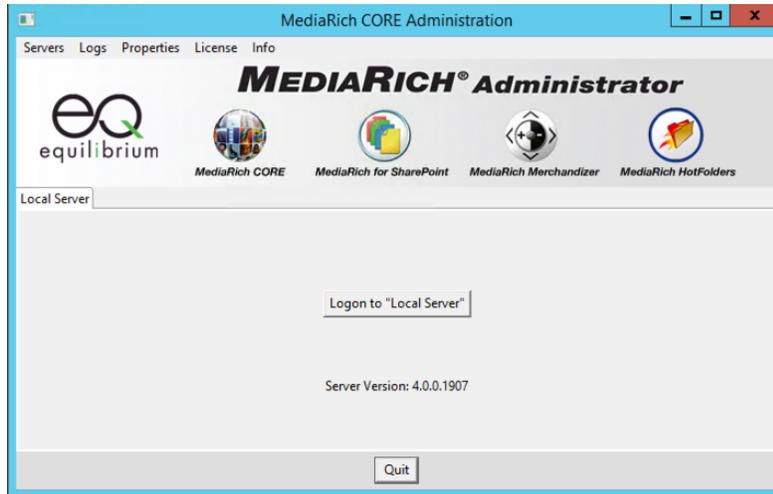
Accessing the MediaRich CORE Administration Utility

The MediaRich CORE Administration Utility provides tools for activating a license file, configuring FTP and email for the server, defining virtual file system settings, and starting and stopping the server. This utility opens automatically at the end of the installation process on Windows or Macintosh, or you can open it at any time to administer the server. You can use the administration utility to configure and administer the MediaRich Server on the local machine, as well as monitor other MediaRich Servers available on your network.

To open the administration utility:

1. Launch the utility application.
 - On Windows and Linux, double-click **MediaRich CORE Admin** in the Equilibrium\MediaRich CORE Admin folder.
 - On Windows, you can also choose **Start > Programs > Equilibrium > MediaRich > MediaRich CORE Administration**.
 - On Macintosh, double-click **MediaRich CORE Admin Application** in the Applications/Equilibrium folder.

By default, the Host Name is `localhost`, which connects the utility to the local installation of the MediaRich Server.



2. Click **Logon to “Local Server”** (or the display name assigned to the local server).
3. In the Login to Local Server dialog, enter the default username and password (`admin`) and click **OK**.

Note: You must use `admin` as both the username and password the first time you log into the server. It is recommended that you change the username and password when you set the server configuration options.

Logging Into the Server

When you are logged in to a server you have full permission to perform any administrative functions. The first time you log in using the MediaRich CORE Administration utility, use the default of `admin` (lower case only) for both the username and password. Be sure to change these defaults so that you can restrict access to authorized personnel only.

Equilibrium strongly recommends that a single person administer a server.

You can open the Administration utility from the Equilibrium folder: Equilibrium Server Admin (Mac) or Equilibrium Server Administration (Windows). To open the utility on Linux, execute the shell script located at the following path:

```
/usr/lib/Equilibrium/EquilibriumServerAdmin/EquilibriumServerAdmin.sh
```



Installing a License File

Note: The default license key is a demonstration key that allows you to test all the features of all products connected to MediaRich Server, including MediaRich Server/Publisher and all APIs, MediaRich Hot Folder, MediaRich ECM for SharePoint/Visualizer. The demonstration license key enables full functionality, however all images have a visible watermark and video transcode and auto-assembly requests will only process 1/2 the video.

Before using any product that runs on the MediaRich CORE Platform, you must have the proper license key installed. For a permanent license key, call 1-855-EQUILIB or +1 415-332-4343 and Dial 4 for Product Information and Sales Support.

After obtaining a license, copy the license file received from Equilibrium to the machine/server and extract/unzip the file. Use the MediaRich CORE Administration Utility to install your license key and start the licensed MediaRich server.

Click the **Install a New License File** button and locate and select the license file. (This button is only available if the server is not running.) Click **Open** to select and install the license file you obtained from Equilibrium.

When the Administration tool loads a valid license file, a license agreement for the product appears. Read the license agreement and click **Accept** to finish installing.

When you accept the license agreement, a ReadMe file appears, which provides important information and news about the product version. Review this document before you proceed with configuring and running the server.

Installing Another License File

If you purchase another MediaRich product or iKit, click the **Install a New License File** button to locate and select the new license file (you must stop the server in order to activate a new license). Click **Open** in the browse window to open and install the license.

If using Linux without a GUI, login as “root”, stop the MediaRich service, delete (do not rename) the old license from the ../Shared/Properties directory, copy the new one into the same directory, and

restart the service.

When you start the server, it reads the license file contents and enables all licensed features.

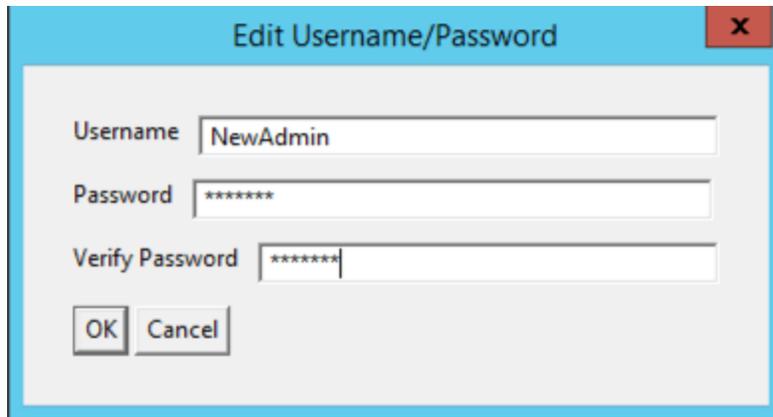
Viewing the Licensed Products and Features

To view which products and iKits are licensed on the server, select **License > Support Information**.

Changing the Username and Password

To change the administrative username/password from the default:

1. Choose **Servers > Change Username/Password**.
2. In Edit Username/Password dialog, enter the desired username and password.
3. Enter the same password in the **Verify Password** text box and click **OK**.



Starting and Stopping the Server

The MediaRich CORE Administration tool displays the status of the server. You must start the server to make it available for processing submitted job requests. When you have set the configuration options, click the **Start Server** button. The indicator light flashes green to red while the server is starting. When the server is running, the administration utility displays a green indicator and a **Server is running** status.

If you need to change any of the server configuration options or install a new license file, you must stop the server. Click **Stop Server**; the status of the server will change when the server is stopped.

Configuring the Server Options

Click the **Configure** button to open the MediaRich Server Options dialog. Use this dialog to specify server options, including enabling or disabling FTP protocol, email configuration, and specifying virtual file systems.

Note: The **Configure** button is available only if the server is not running. Click **Stop Server** to

stop the server before changing these options.

MediaRich Server Options

FTP Protocol: Disabled Enabled HTTP Protocol: Disabled Enabled

Email Options:

SMTP Host:

Sender E-mail Address:

SMTP Username:

SMTP Password:

Log Options:

E-mail Addresses:

E-mail Log Level: Severe

Name	Path	Sentinel	Writable	Executable
batchout	\\eqqapc\images	false	true	false
videoproxy	C:\Program Files\Equilibrium\Me...	false	true	false
mrdocs	C:\Program Files\Equilibrium\Me...	false	false	false
pyramid	C:\Program Files\Equilibrium\Me...	false	true	false

New VFS Remove Edit

OK Cancel

FTP Protocol

MediaRich has the ability to read from and write to remote FTP servers. Choose **Enabled** to enable this feature, or **Disabled** to turn the feature off.

HTTP Protocol

MediaRich has the ability to read from remote web (HTTP) servers. Choose **Enabled** to enable this feature, or **Disabled** to turn the feature off.

Email Options

In order to enable email delivery of job status information and zip files, you must specify the SMTP host, username, password, and the sender email address.

Log Options

You can also send log event information automatically to a server administrator by entering an email address and indicating the notification severity level. Send this information to multiple addresses by using a semi-colon between each email address.

Note: This option is extremely useful and much more convenient than examining log files every day. With the E-mail Log Level set to 'severe,' notification will be sent only for the higher-priority incidents, like the MediaGenerator stopping, advanced notice of an about-to-expire license, and so forth.

Virtual File Systems

A Virtual File System (VFS) assigns a symbolic name to a directory on the MediaRich server. MediaRich accesses both local and remote storage solely through symbolic VFS names. Defining a VFS allows MediaRich to access a particular place on its local filesystem or on a network.

A VFS definition controls what MediaRich is allowed to do on that part of the filesystem. MediaRich is able to read from any defined VFS. A VFS definition can, however, either allow or deny write access to the portion of the filesystem it points to. Likewise, it can either allow or deny the execution of scripts from that portion of the filesystem.

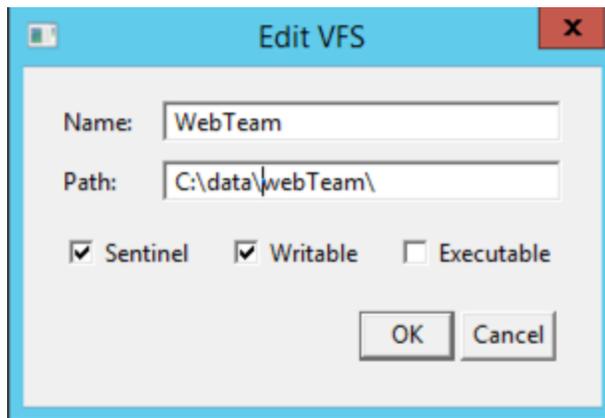
Using the MediaRich CORE Administration Utility, you can define any number of VFS directories for use in job submissions.

Note: If you expect to cause MediaRich to write files to a VFS directory, that directory must have its permissions set so that MediaRich has read and write permissions for all folders and sub-folders below the root of the VFS.

To define a Virtual File System:

1. In the MediaRich Server Options dialog, click the **Virtual Filesystems** tab.
2. Click the **New VFS** button.
3. In the Edit VFS dialog, enter a name for the VFS and its path.

You can assign a name for the VFS according to your needs. You then specify the path of the directory you want to associate with this VFS. This path must be valid from the server's perspective, not from the perspective of how that same directory might be mounted on the network.



4. If you want to create a read-only directory that the MediaRich Server can use as a source directory but not as a save directory, clear the **Writable** checkbox.

The **Writable** checkbox is selected by default, so if you want to create a standard read/write VFS directory leave this option selected.

The **Executable** checkbox determines if scripts can be executed on this filesystem. This option is disabled by default. Enable this option if you want to store MediaScript script files on this filesystem that MediaRich can execute.

The **Sentinel** checkbox is required in order for the server to share files with clients. If you aren't sure if you need this option enabled, leave it enabled.

5. Click **OK** to set the new VFS definition.

To change or remove a VFS definition, select it in the MediaRich Server Options dialog and click **Edit** or **Remove**.

Hot Folder Roots

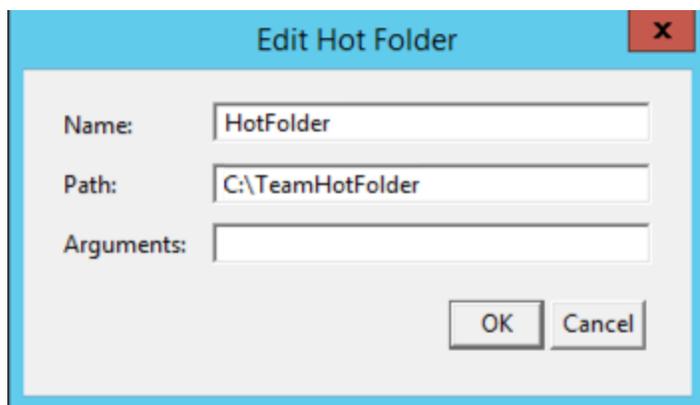
Hot Folder Roots are locations where MediaRich Hot Folders can be established. MediaRich monitors these locations looking for potential Hot Folders to be processed.

A Hot Folder is a directory that has one or more MediaRich MediaScripts associated with it. When files are dropped into this directory, they are processed using the associated MediaScript. Any number of Hot Folders can be established below a single Hot Folder Root directory and any number of Hot Folder Roots may be defined.

For information about creating and writing scripts for MediaRich Hot Folders, refer to the Hot Folders section of the *MediaRich CORE Programmer's Guide*.

To define a Hot Folder Root:

1. In the MediaRich Server Options dialog, click the **Hot Folder Roots** tab.
2. Click the **New Root** button.
3. In the Edit Hot Folder Root dialog, enter a name for the Hot Folder Root and its path.
Assign a unique name for the Hot Folder Root. You then specify the path of the directory you want to associate with this Hot Folder Root. This path must be valid from the server's perspective, not from the perspective of how that same directory might be mounted on the network.



The screenshot shows a dialog box titled "Edit Hot Folder". It has a light blue header bar with the title and a red close button. The main area is white and contains three input fields. The first is labeled "Name:" and contains the text "HotFolder". The second is labeled "Path:" and contains the text "C:\TeamHotFolder". The third is labeled "Arguments:" and is currently empty. At the bottom right of the dialog are two buttons: "OK" and "Cancel".

4. (Optional) Enter an arguments string.
This string will be passed to the MediaScript scripts that are run in this root, allowing you to pass global settings to them. Refer to the *MediaRich Programmer's Guide* for more information about passing arguments to scripts.
5. Click **OK** to set the new Hot Folder Root definition.

To change or remove an existing Hot Folder Root definition, select it in the list of existing Hot Folder Roots and click **Edit** or **Remove**.

Performing Server Administration Tasks

The MediaRich CORE Administration utility enables you to perform the following administrative functions for MediaRich servers:

- [Monitoring Additional Servers](#)
- [Managing the MediaRich Server](#)
- [Working with Local and Global Server Properties](#)
- [Working with Log Files](#)
- [Viewing Available Fonts](#)

Managing the MediaRich Server

You can view the status of the MediaRich server for all administered servers, and stop or restart the MediaRich server as desired.

Viewing server status

When you are logged in to a MediaRich server, the MediaRich CORE Administration utility displays the status of the server and indicates whether it is currently running or stopped.

Note: A `Not Responding` status indicates that the MediaRich Server is started but can't connect. An `Unavailable` status indicates that the system monitor is not started.

Starting or stopping the server

The administration utility displays the status of the server. You must start the server for it to be available for processing submitted batch jobs or Web site requests. When you have set the configuration options, click the **Start Server** button. The indicator light flashes green to red while the server is starting. It may take a few minutes to start, at which point the administration utility displays a green light indicator and a `Server is running` status.

To change any of the server configuration options or install a new license file, you must stop the server. Click **Stop Server**, and the administration utility indicates when the server is stopped.

Monitoring Additional Servers

If your network has more than one MediaRich server, you can use the administration utility to monitor or administer all of them, regardless of the operating system on which they are running.

Adding or removing MediaRich servers

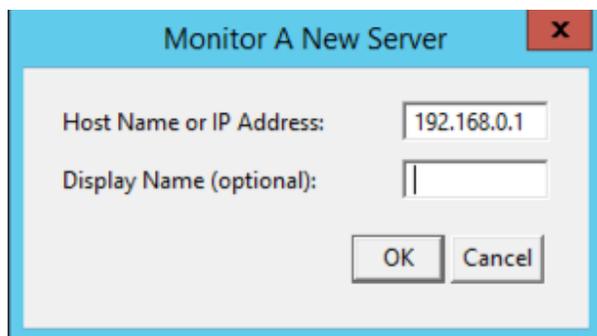
If you are running multiple MediaRich servers across your shared network, you can monitor other servers from a single system by adding the networked servers in the MediaRich CORE Administration utility.

Any MediaRich server that will be remotely administered or monitored must have the Media Socket port and System Monitor port open. The default ports are 9877 and 9878, but you should verify these first. For more information, see “[Viewing and Modifying the Local Properties](#)” on page 37.

To add another MediaRich server:

1. Choose **Servers > Monitor a New Server**.
2. In the dialog enter the Host name or IP address.

You can use the Display Name field to assign a name to the server, which appears in the MediaRich CORE Administration utility.



3. Click **OK**.

The names of monitored servers appear as tabs next to the Local Server. Click the server tab to view its information, or choose it by name from the **Servers** menu.



To remove a monitored server:

1. Select the server by clicking its name in the MediaRich CORE Administration utility or choosing the server from the **Servers** menu.
2. Choose **Servers > Stop Monitoring The Current Server**.

3. Click **OK** to confirm removal of the server.

Working with Local and Global Server Properties

Properties files contain some important operational settings for the MediaRich CORE server. These are separated into two types: local properties and global properties.

Properties that should be set specifically for each individual server are in the local properties file. Properties that can be shared across server installations, where multiple MediaRich servers can use a shared file location, are stored in a global properties file.

Viewing and Modifying the Local Properties

You can view and set the local properties for the current server in the Local Properties dialog. These properties include setting local paths (for example, fonts), file server paths (for example, to the image cache and original files), and various other settings such as the number of Web server and MediaRich Server threads that can run simultaneously.

To work with the local properties:

1. Choose **Properties > Local Properties**.

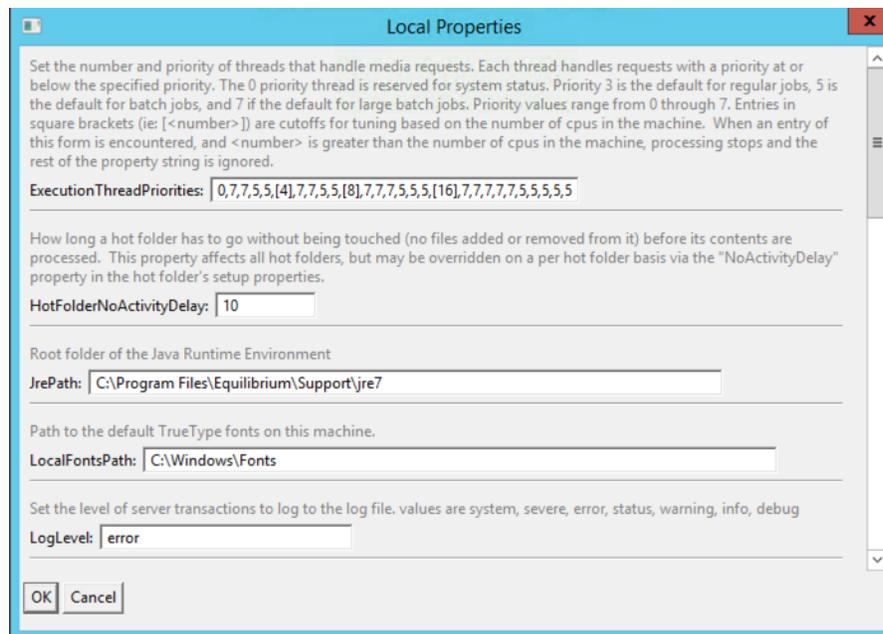


Figure 2: Defining the local properties

2. Confirm or modify the following settings:

Note: When a path is not specified for one OS, Equilibrium\ means C:\Program Files\Equilibrium on Windows, /Library/Equilibrium on the Macintosh, and /usr/lib/Equilibrium on Linux.

- **ExecutionThreadPriorities:** Specifies the number and priorities of execution threads that will be generated to handle requests.
This setting will determine how many jobs the server processes simultaneously.
- **HotFolderNoActivityDelay:** Determines how long a Hot Folder has to go unmodified (nothing added to it or removed from it) before its contents are processed.
- **LocalFontsPath:** Sets the path to the fonts folder.
For Windows servers, the default path is `C:\WINDOWS\Fonts`.
For Mac OS X servers, the default paths are `/Library/Fonts` and `/System/Library/Fonts`.
- **LogLevel:** Sets the level of server transactions to log. Available values are system, severe, error, status, warning, info, and debug.
During normal production use, this value should be set to “error”. This will keep the size of the log files to a minimum, while still reporting on critical events that an administrator should be aware of.
The “info” and “debug” levels are used for troubleshooting issues with particular MediaRich scripts or with the MediaRich system as a whole. If you are experiencing a problem, you should set the Log Level to “debug”, restart MediaRich, cause the problem to occur again, and then check the latest information added to the log files.
- **LogRolloverFrequency:** Sets the frequency of rollover for log files.
- **MediaPath:** Sets the path to original media.
The default path is `Equilibrium\MediaRichCore\Shared\Originals\Media`.
- **MediaSocketPort:** Specifies the TCP/IP port used by the MediaRich Server.
The default port is 9877.
- **MediaSocketThreads:** Specifies the number of connections that a Web server makes with the MediaRich Server.
- **ModulesPath:** Specifies one or more locations where MediaRich will look for add-on modules.
If multiple locations are specified, they should be separated by semicolons. An asterisk at the end of a location’s path specifies that subdirectories below that location should be searched for modules as well.
The default path is `Equilibrium\MediaRichModules*`.
- **OutputPath:** Specifies the path to generated images and dependencies.
The default path is `Equilibrium\MediaRichCore\Shared\Generated`.
- **PhotoshopPath:** (Windows only) Deprecated - Photoshop plug-ins are no longer supported in MediaRich.
- **ProgramPath:** Specifies the root folder of this MediaRich installation.
For Windows, the default is `C:\Program Files\Equilibrium\MediaRichCore`.
For Macintosh, the default is `/Library/Equilibrium/MediaRichCore/Properties`.
On Linux, the default path is `/usr/lib/Equilibrium/MediaRichCore`. These settings should not be changed.
- **PSPluginExecutionTimeout:** (Windows) Deprecated - Photoshop plug-ins are no longer supported in MediaRich.
- **ScriptsPath:** Specifies the path to MediaScript scripts.
The default path is `Equilibrium\MediaRichCore\Shared\Originals\Scripts`.
- **SharedColorProfilesPath:** Specifies the path to shared ICC profiles.

The default path is `Equilibrium\MediaRichCore\Shared\Originals\Profiles`.

- **SharedFontsPath:** Specifies the path to shared fonts. The default path is `Equilibrium\MediaRichCore\Shared\Originals\Fonts`.
- **SharedLogsPath:** Specifies the path to shared log files.
The default path is `Equilibrium\MediaRichCore\Shared\Logs`.
- **SharedPath:** Specifies the path to properties and logs.
The default path is `Equilibrium\MediaRichCore\Shared`.
- **SysPath:** The path to system scripts. The default path is `Equilibrium\MediaRichCore\Shared\Originals\Sys`.
- **SystemMonitorPort:** Specifies the TCP/IP port used by the system monitor process.
The default port is 9878.
- **TempPath:** Specifies the path to temporary files.
The default is `Equilibrium\MediaRichCore\Temp`.

3. When finished, click **OK** to save your changes.

Note: You must stop and restart the MediaRich Server before saved changes take effect.

Viewing and Modifying the Global Properties

You can modify global properties for the server in the Global Properties dialog. These properties include error and dependency handling, as well as cache management.

To modify global properties:

1. Choose **Properties > Global Properties**.

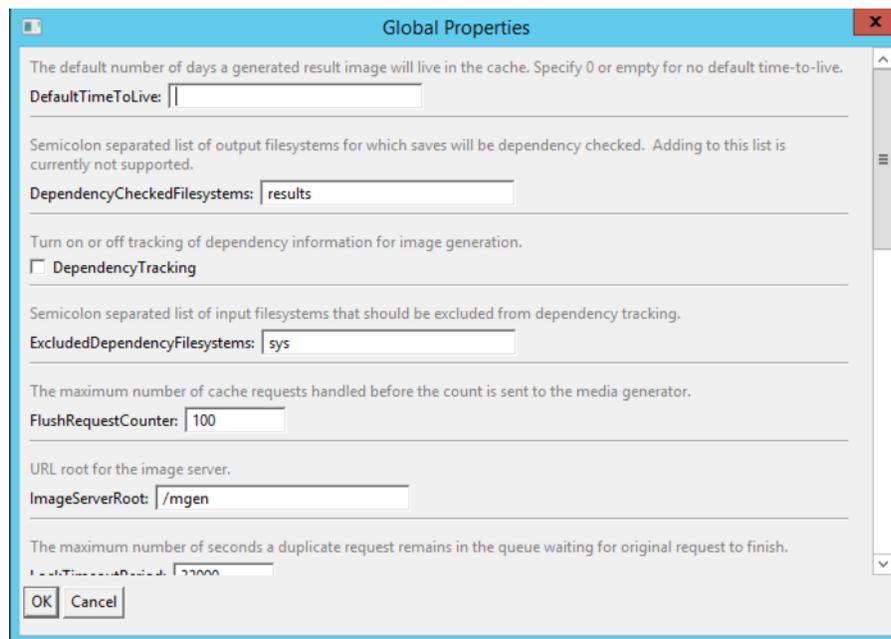


Figure 3: Defining the global properties

2. Confirm or modify the following settings:

- **DefaultTimeToLive:** Enter the number of days the server should keep media objects in the cache before removing them. The System Monitor keeps track of Time to Live information, but this information is lost if you restart the System Monitor.
- **DependencyCheckedFilesystems:** Enter a list (separated by semicolons) of the output locations for which saves will be dependency checked.
- **DependencyTracking:** Select this box to enable this feature; required for the `CheckDependencies` function to work. If not enabled, dependency information is not generated. This feature uses a database to keep track of the entries so that there is no limit on the number of files that can be tracked. (Versions previous to 3.8 had Dependency Tracking ON by default. The default in 3.8.x and later is for it to be turned OFF.)
- **ExcludedDependencyFilesystems:** Enter a list (separated by semicolons) of the input filesystems to exclude from dependency checking. The default is `sys`.
- **FlushRequestCounter:** Enter the number of cache requests that must be made before the count is sent to the MediaGenerator. The default is `100`.
- **ImageServerRoot:** Enter the root for the image server. The Web server extension uses this property to determine which URLs are processed by MediaRich. MediaRich processes only URLs with paths starting with this root. Default is `/mgen`.
- **LockTimeoutPeriod:** Enter the maximum number of seconds a cache entry can be locked. This is the maximum time that duplicate requests will wait in the queue for the original request to finish. This value should not be less than the default.
- **LookupFilesystems:** Enter the output filesystems (separated by semicolons) for which the file request command is allowed.
- **MaximumBatchInactiveTime:** Enter the maximum number of seconds a batch can be inactive before it is terminated. This is the maximum time that batches remain open for accepting new jobs. This applies only to applications that use the batch API. In general, this value should not be reduced from the default.
- **MaximumExecutionTime:** Enter the maximum number of seconds that a MediaScript can run. The default is `32000`.
- **MediaLoadMaxHeight:** Enter the maximum height of the image for TIFF, JPEG, and Photoshop files. If set to 0 (zero), downsizing is disabled and MediaRich will attempt to load at full height resolution. `MediaLoadMaxWidth` and `MediaLoadMaxHeight` work independently, and scaling is always proportional. So if, for example, width is specified as 0 and the height is specified as 4000, the image will be reduced proportionally as necessary so that the height is not larger than 4000. The default is `8192`. `MediaLoadMaxHeight` can be overridden by the `maxHeight` parameter in the `load()` method.
- **MediaLoadMaxWidth:** Enter the maximum width of the image for TIFF, JPEG, and Photoshop files. If set to 0 (zero), downsizing is disabled and MediaRich will attempt to load all images at full width resolution. `MediaLoadMaxWidth` and `MediaLoadMaxHeight` work independently, and scaling is always proportional. So if, for example, width is specified as 0 and the height is specified as 4000, the image will be reduced proportionally as necessary so that the height is not larger than 4000. The default is `8192`. `MediaLoadMaxWidth` can be overridden by the `maxWidth` parameter in the `load()` method.

Important: Disabling this automatic downsizing can result in errors or crashes due to a low-memory situation for large files. It is *strongly* recommended that you do not disable this

feature unless your system has 32 GB of physical RAM.

- **ProfileScriptPath:** Enter the virtual file system path to the profile script. The default is `sys/ProfileScript.ms`. Refer to the *MediaRich Programmer's Guide* for a description of the profile script and parameters.
 - **ReturnHtmlErrors:** Tick this box to have MediaRich return a separate HTML file for each failed image generation.
 - **ScriptErrorLifetime:** Enter the number of seconds that an HTML script error file remains in the `Script Errors` folder. The default is 3600.
 - **ScriptErrorLogging:** Select to enable logging of MediaScript errors. If cleared, no errors will be logged.
 - **TimeToLiveTracking:** Turn time-to-live tracking on or off. The default setting is on (checked).
3. When finished, click **OK** to save your changes.

Note: Both the System Monitor service and the MediaGenerator service must be restarted after editing the properties files in order for the changes to take effect. You must also restart all MediaRich servers and Web servers when changing properties.

Modifying properties files in a text editor

The settings you specify in the Local Properties and Global Properties windows in the MediaRich CORE Administration utility are saved as text files in the MediaRich server directory. Modifying local and global settings in these files using a text editor has the same effect as modifying the settings using the MediaRich CORE Administration utility interface.

The text file provides additional advanced options for modifying a server's properties.

The original property values are stored in the `local.properties` file, located at `MediaRichCore\Properties` and in `global.properties`, located at `MediaRichCore\Shared\Properties`. The Administration utility saves modifications made to these files in a separate set of files named `local.properties_user` and `global.properties_user` respectively. These files are located in the same directories as the original files they modify. This mechanism allows future versions of MediaRich to update the default set of property values without overwriting any changes made by the local administrator.

Note: MediaRich Server supports Unicode. If a properties file contains any Unicode characters, it must be saved as Unicode.

Moving properties files to a new installation

If you transfer `*.properties_user` files to a new MediaRich installation, you must restart SystemMonitor to force it to load the new files. This can be done either using the Services control panel or by rebooting the computer.

Configuring the execution threads

The `ExecutionThreadPriorities` property is a local property that specifies the number and priorities of execution threads that will be generated to handle requests. This setting determines how many jobs the server will process simultaneously.

A job can be given a priority number between 0 and 7 when it is submitted, with 0 being the highest priority, and 7 the lowest. The 0 priority is reserved for administrative jobs that are not expected to require a significant amount of processing time. Higher priority jobs are given precedence over lower ones.

Ignoring the allowance for multiple CPU configurations, the `ExecutionThreadPriorities` property consists of a list of priority numbers separated by commas. An execution thread is generated for each number in this list, and will run with the priority value specified by that number. A thread that runs at a specific priority will only execute jobs that have a priority equal to or higher than that number. A lower number specifies a higher priority.

Equilibrium suggests that the `ExecutionThreadPriorities` list ALWAYS contain at least one 0 priority thread. This thread can be ignored when determining the appropriate load for a host, since the thread will not process normal jobs, but only lightweight administrative functions. If no 0 priority thread exists, administrative functions such as status page refreshes will hang whenever the server is fully loaded with other kinds of jobs.

The priority assigned to a job is determined by the client that submits the job. The following guidelines are suggested to interpret various priority levels:

Regular Job = Priority 3 - These are dynamic requests, such as from a Web page or other interactive user interface. They have a relatively high priority since they are expected to be executed in real time. MediaRich requests submitted through a Web browser have this priority.

Batch Job = Priority 5 - These are jobs that aren't expected to return immediately. Processing of batches of files, such as jobs submitted by MediaRich HotFolder, are submitted at this priority level.

Large Batch Job = Priority 7 - These are jobs that require a long processing time regardless of what priority they are assigned. They are given a lower priority to allow the creation of threads that will not run these jobs and can be reserved for other job types.

The following examples illustrate these configurations:

```
ExecutionThreadPriorities=0,5,5,7,7
```

This is the standard configuration for a two-processor host. It specifies an administrative thread, two threads for running regular and batch jobs, and two threads for running any kind of job. This is a good configuration for a machine that is being used EITHER for processing dynamic requests OR for processing batch jobs. It will allow dynamic requests and most batch jobs to run on all four threads, but will only allow large batch jobs to run on two of the threads, preserving two threads for more lightweight jobs.

```
ExecutionThreadPriorities=0,3,3,5,5,7
```

This would be a good configuration for a single server that will be used for both dynamic and batch processing. It will preserve two threads for interactive requests, regardless of how many batch jobs

are waiting to be processed. It will provide three additional threads that will take either interactive or batch jobs.

```
ExecutionThreadPriorities=0,7,7,7
```

This configuration can be used when the server will be used almost exclusively for large batch jobs such as video processing. In this case, all available threads must be able to process these large jobs... no threads should be reserved for higher priority jobs. Note that higher priority jobs can still be submitted, but must wait in line for any large jobs that were submitted before them to complete.

One other contingency to be aware of is that a single `ExecutionThreadPriorities` value can specify multiple configurations based on the number of virtual CPUs (cores) present in the system. If a number in square brackets is inserted into the list of priority values, this value is used as a “processor count cutoff.” If the machine on which the server is running does not have at least that many virtual CPUs, the remainder of the list of values is ignored.

MediaRich users should rarely have a reason to specify multiple CPU thread configurations. Since the number of CPUs in a machine is usually expected to remain constant, there is no need to reconfigure this value to support multiple CPU configurations. This feature exists primarily to allow Equilibrium to provide a single initial value for this property that is appropriate for machines with multiple processors.

The following example specifies multiple configurations based on CPU count and is the initial value of this property when MediaRich is installed:

```
ExecutionThreadPriorities=0,5,5,7,7,[4],5,7,7,[8],5,5,7,7
```

This configuration specifies that five threads (of priorities 0, 5, 5, 7, and 7) be used on a single or two-processor machine, eight threads be used on a four-processor machine, and that 12 threads be used on an eight-processor machine.

Working with Log Files

The MediaRich server utilizes two primary log file types:

- **Media Generator log files:** this is a local log file that records MediaGenerator exceptions and start messages, as well as Web server-MediaGenerator connector information for a specific MediaRich server.
- **Script Error log files:** this is a global log file that records script errors from a MediaRich server.

The MediaGenerator.log file is a local log file: Each server generates its own separate files. The ScriptErrors.log file is a global file, logging script errors from all active servers, including those that are not being monitored.

The log files are located at `C:\Program Files\Equilibrium\MediaRichCore\Shared\Logs` on Windows, and `Library/Equilibrium/MediaRichCore/Shared/Logs` on the Macintosh.

When running the Web server extension for the Apache Web server, the Apache Web server extension also writes to Apache’s error.log file.

Note: You can view and manage all log files at the global level.

Working with local log files

Local log files are generated locally, so that each server has its own log file for recording errors, exceptions, start messages, and connector information specific to that server.

To view the Media Generator log for the current server:

- è Choose **Logs > Open Media Generator Log**.

To clear the contents of the Media Generator log file:

1. Choose **Logs > Clear Media Generator Log**.
2. When the confirmation dialog opens, click **OK**.

As with the local MediaGenerator log file, you can view and/or clear the ScriptErrors.log file. It contains error messages for all servers (not only those being administered).

To view Script Errors log file:

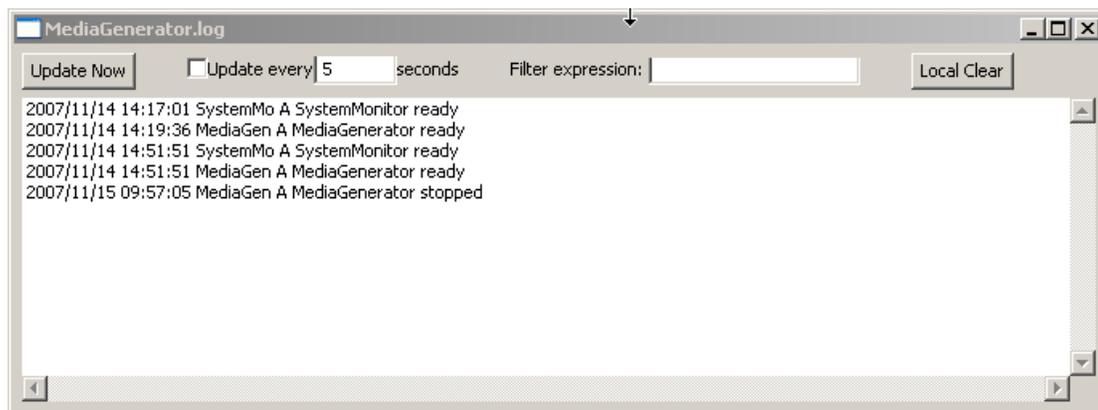
- è Choose **Logs > Open Script Errors Log**.

To clear the contents of the Script Errors log file:

1. Choose **Logs > Clear Script Errors Log**.
2. When the confirmation dialog open, click **OK**.

Working with the log file display window

The log file display window provides a number of features that make viewing log files easier.



When first displayed, the log window does not update its contents when MediaRich writes new information to the log file it is viewing. There are two ways to cause the window to update; that is, to display any recent additions to the log file.

The **Update Now** button forces a single update of the window's contents.

Select the **Update Every ___ seconds** checkbox and enter a number in the field to cause the log window to automatically update. The number determines how often the display is refreshed.

Use the **Filter expression** field to filter the information displayed by the log window. When you enter text in this box, only lines that contain that text will be displayed in the window. You can also enter Java regular expressions: for example, [A-Z]r will find any word with a capital letter followed by “r”, such as “Error”.

Click the **Local Clear** button to clear the current contents of the log window. This is useful when combined with the **Update Now** or **Update Every ___ seconds** features; it allows you to easily see new information as it is added to the log file.

Note: This button does not affect the contents of the log file. Rather, it affects only the current display of the log file. If you want to again view the information hidden by this button, close the log window and reopen it.

Viewing Available Fonts

MediaRich utilizes the standard system fonts installed on your server. On Windows, the path to the system fonts directory is usually C:\Windows\Fonts. On the Macintosh, the system fonts are found in two directories: <user>/Library/Fonts and /System/Library/Fonts. On Linux, there are a number of possible places that system fonts might be installed. The MediaRich installer will locate the system fonts directories and set up the system properties appropriately. Any fonts added to your system can be utilized by MediaRich. After installing fonts, restart the MediaRich server so it recognizes the new fonts.

You can also add fonts for use exclusively by MediaRich by adding the font files to the MediaRichCore/Shared/Originals/Fonts directory within your MediaRich installation directory.

Viewing fonts on a specific server

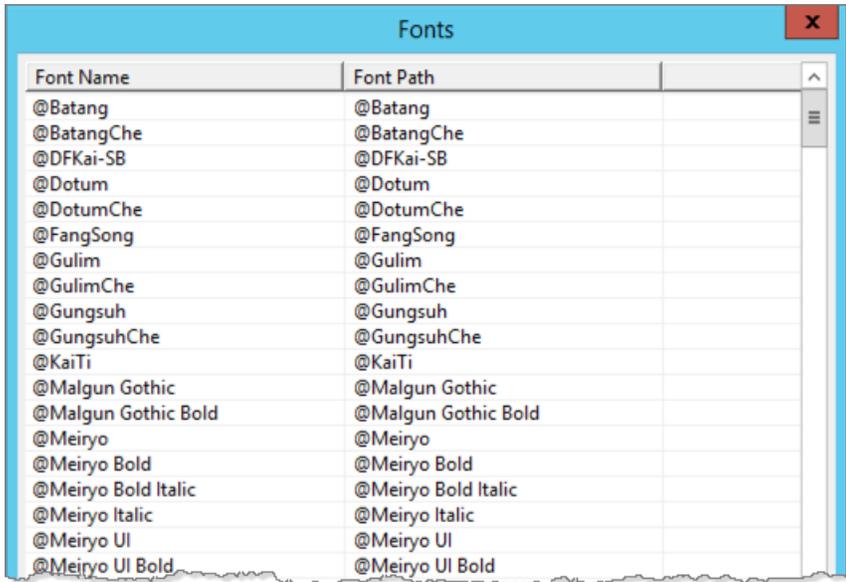
You can view the TrueType fonts (.ttf files) available on a specific server or view all fonts available on all servers. Ideally, all MediaRich servers should possess the same sets of fonts.

Note: You can use PostScript fonts with MediaRich CORE, but these fonts are not available from the Administration utility in Windows.

To view the fonts on the server:

Choose **Info > Show Installed Fonts**.

The Fonts window displays the fonts available on the selected server.

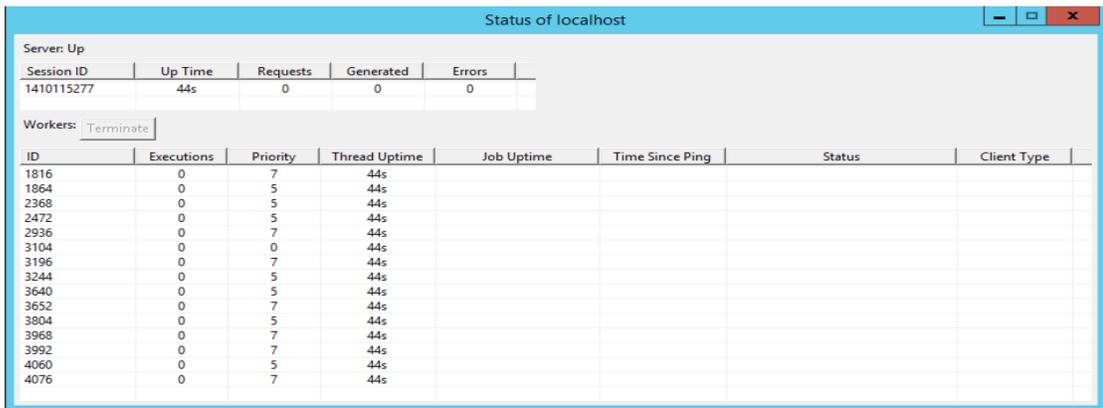


Viewing Worker Process Status

The Server Status window displays the current status of each of the worker processes running in the MediaRich Server being monitored. When a job is running, its status can be viewed in this window.

Displaying the server Status window

To access the Server Status window for the currently monitored server, choose **Info > Show Server Status** item. If the window is not already open, this opens the window. If it is open but obscured by other windows, this brings it forward.



Interpreting the data in the server Status window

The top section of the window displays information about the server itself:

- **Session ID** - An identifier that represents this execution of the MediaRich Server. It is normally not useful to users.
- **Up Time** - How long the server has been running since it was last launched or restarted.
- **Requests** - How many requests the server has served since it was last restarted.

- **Generated** - How many requests have been made that could not be serviced from the cache, and therefore required a MediaScript execution.
- **Errors** - How many script executions failed due to an error occurring.

The bottom section of the window displays information about each of the worker processes running in the server. Each worker process can be working on a single job at any particular moment, so the number of jobs a MediaRich server is capable of performing simultaneously is determined by the number of worker processes. The number of worker processes available is configurable via the local MediaRich properties. See [“Working with Local and Global Server Properties” on page 37](#) for more information.

The following information is displayed about each worker process:

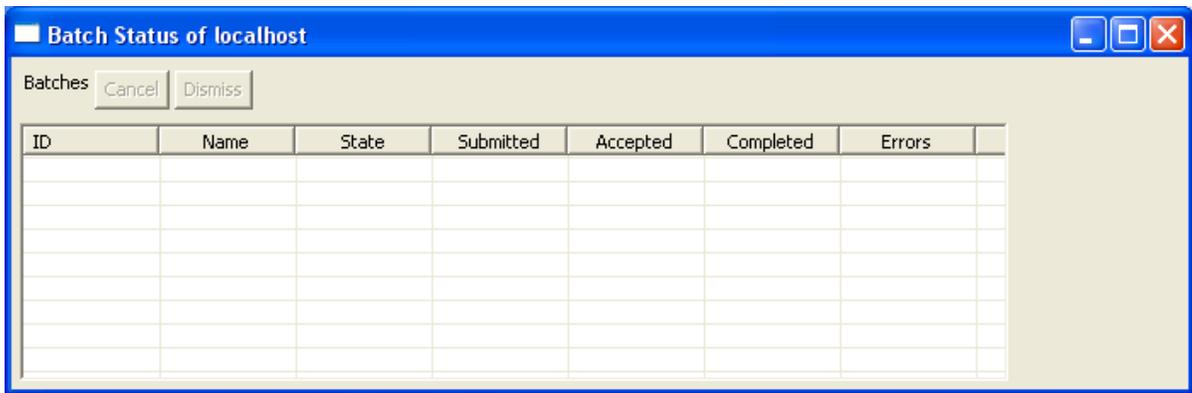
- **ID** - A unique identifier for the process. On most platforms, this is the actual operating system process identifier of the worker process.
- **Executions** - The number of script executions that this worker has performed.
- **Priority** - The priority of this worker. A worker will only process jobs that have a priority that is equal to or lower than its priority. So, for example, a worker with a priority of 3 will only handle jobs with priority 0, 1, 2 or 3.
- **Thread Uptime** - How long this worker has been running. In most cases, this number will be about the same as the server uptime value. If this value is significantly smaller than the server uptime value, it means that this worker died at some point and had to be relaunched.
- **Job Uptime** - How long the job currently being run by this worker has been running. If this value is empty, then the worker is not currently processing a job.
- **Time Since Ping** - How long since this thread last performed an action, either a job or an administrative function.
- **Status** - An area that can display a status string that has been set by a running script. What exactly shows up in this area is a function of the script being run. Many short-lived scripts don't bother to write anything here. This area is primarily used by video and audio processing scripts that can run for significant amounts of time. Such scripts use this area to convey how far along they are in the process they are performing.
- **Client Type** - This column is currently not used.

Viewing Batch Process Status

The Batch Status window displays the current status of each of the batch processes running in the MediaRich Server being monitored. When a job is running, its status can be viewed in this window.

Displaying the Batch Status window

To open the Batch Status window for the currently monitored server, choose **Info > Show Batch Status**.



Interpreting the data in the Batch Status window

- **ID** - An identifier that represents this batch execution of the MediaRich Server. It is normally not useful to users.
- **Name** - A name that represents this batch execution of the MediaRich Server. It is normally not useful to users.
- **State** - The state of the batch job: executing or done.
- **Submitted** - The number of batch jobs that have been submitted to the server.
- **Accepted** - The number of batch jobs that have been accepted by the server.
- **Completed** - The number of batch jobs that completed successfully.
- **Error** - The number of batch jobs that completed with errors.

Canceling or dismissing batch jobs

You can cancel or dismiss a batch job with the **Cancel** or **Dismiss** buttons. The Cancel button cancels a running batch job. The Dismiss button discards the messages for the batch jobs that are completed.

Viewing the License

Choose **License > Support Information** to view the MediaRich License information. Click the **Copy** button to copy this information to the clipboard. This is useful information for technical support to have when diagnosing your MediaRich system.

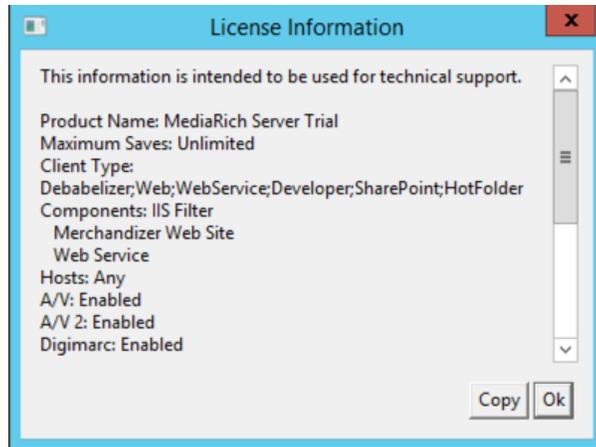


Figure 4: MediaRich license information

Modifying Additional Properties in a Text File

The settings you specify in the Local Properties and Global Properties windows in the MediaRich CORE Administration Utility are saved as text files in the MediaRich server directory. Modifying local and global settings in these files using a text editor has the same effect as modifying the settings using the MediaRich CORE Administration Utility interface.

The text file provides additional options for modifying the server properties.

Properties are saved in the `local.properties` file (located at `MediaRichCore\Properties`) and in `global.properties` (located at `MediaRichCore\Shared\Properties`).

MediaRich Server supports Unicode. If a properties file contains any Unicode characters, it must be saved as Unicode.

HTTP and FTP File Support Using the FSNet Plug-In

MediaRich accesses files by defining a number of virtual filesystems. A virtual file system indicates the root of a file tree located somewhere on either the local filesystem or on the network. When referencing a file via a virtual file path, the virtual filesystem on which the file is located is specified by prepending the virtual filesystem name followed by a colon (:) onto the path to that file.

The FSNet plug-in can implement HTTP and FTP access via standard URLs by defining virtual filesystems named `http` and `ftp`. Because normal HTTP and FTP URLs consist of these names followed by a colon followed by a file path, normal HTTP and FTP URLs are valid MediaRich virtual file paths.

In addition to the default HTTP and FTP filesystems, it is possible to set up other filesystems that refer to resources on HTTP and FTP servers. Their URLs would look the same as the standard URLs except that the `http` or `ftp` keys at the beginning of the URLs would be some other name to specify that an alternate filesystem is being accessed. The process of defining additional HTTP and FTP filesystem is described later in this document.

For HTTPS, simply specify port 443 in the path for your `Media.load()` op, like: `image.load(name @ "http://www.google.com:443/images/srpr/logol1w.png");`

Note: This only affects the Windows version because it uses the OS-provided HTTP services.

Enabling standard HTTP and FTP URL access

To allow standard HTTP URLs to be passed to MediaRich as file paths, add the following line to the `local.properties` file:

```
filesystem.fsnet.http.Specifier=http
```

To allow standard FTP URLs to be passed to MediaRich as file paths, add the following line to the `local.properties` file:

```
filesystem.fsnet.ftp.Specifier=ftp  
filesystem.fsnet.ftp.Ftp=1
```

HTTP and FTP URLs

There are a number of MediaScript methods, such as the media object's `load()` and `save()` methods, that accept file paths as parameters. The paths passed via these parameters will often be paths to files on the local filesystem, but can also refer to resources on a network via the HTTP and FTP protocols.

By enabling standard HTTP and FTP URLs, references to files on HTTP and FTP servers take the same form as standard URLs used to access those files via a Web browser. All of the information normally encoded in HTTP and FTP URLs can be passed to MediaRich. MediaRich uses the supplied information to connect to the specified network resource.

A fully specified HTTP or FTP URL looks like this:

```
(http|ftp)://<username>:<password>@<server name>:<port>/<path to resource>
```

The `<username>`, `<password>`, and `<port>` portions of the URL are optional. The following are some examples of well formed URLs:

```
http://www.pixar.com/images/pixar_bw.gif  
http://joeuser:abcdefg@acomputer/myimages/product1.jpg  
ftp://ftpserver:1234/public/images/photograph.tif
```

FSNet properties

A number of properties can be set in the `local.properties` file to influence the behavior of HTTP and/or FTP requests. These properties can be specified such that they effect all FSNet filesystems or so that they effect only a single filesystem.

To set a property that effects all FSNet filesystems, specify that property as follows:

```
filesystem.fsnet.<property name>
```

To set a property to only effect a single virtual filesystem, specify the property as follows:

```
filesystem.fsnet.<virtual filesystem name>.<property name>
```

This example sets up a proxy host for all FSNet filesystems:

```
filesystem.fsnet.ProxyHost=ourproxyhost:3322
```

This example sets the user name and password for just FTP access:

```
filesystem.fsnet.ftp.UserNameAndPassword=joeuser:tokyo
```

The specific properties that effect FSNet operation are described below. As you're reading the rest of this section, remember that each of these properties can be specified such that they effect either a single virtual filesystem (http or ftp), or all FSNet filesystems (http and ftp).

Loading and saving files using FTP and HTTP

To load a file via an HTTP or FTP URL (using the `load()` function), or to save a file using an FTP URL (using the `save()` function), you can enable FTP and HTTP functionality by modifying the `filesystem.properties` file.

Additional FSNET properties	Description
<code>filesystem.fsnet.<ftp or http>.Disabled=</code>	Set this property to 1 to disable FTP or HTTP
<code>filesystem.fsnet.<ftp or http>.UserName=</code>	Specify the default username
<code>filesystem.fsnet.<ftp or http>.Password=</code>	Specify the default password
<code>filesystem.fsnet.<ftp or http>.UserNameAndPassword=</code>	Colon-separated value containing both the username and password. E.g., janesmith:frobozz
<code>filesystem.fsnet.<ftp or http>.ResponseTimeout=</code>	Specify the amount of time to wait (in seconds) for a response from a server before giving up
<code>filesystem.fsnet.<ftp or http>.LockTryDelay=</code>	Specify the amount of time to wait (in seconds) when a cache file is locked
<code>filesystem.fsnet.<ftp or http>.MaxLockTries=</code>	Specify the maximum number of times to retry a locked cache file before giving up
<code>filesystem.fsnet.<ftp or http>.BreakLockOnFailure=</code>	Set this property to true to break the lock on a cached file if it fails to become available

Additional FSNET properties	Description
<code>filesystem.fsnet.<ftp or http>.RefreshInterval=</code>	Specify the frequency (in seconds) with which to check the Web for a newer image - the system default is 600 seconds (15 minutes)
<code>filesystem.fsnet.<ftp or http>.Debugging=</code>	Set to 1 to enable debugging output.
<code>filesystem.fsnet.<ftp or http>.ProxyHost=</code>	The name of a host to use as a proxy server. hostname:port.
<code>filesystem.fsnet.<ftp or http>.ProxyUserName=</code>	The username for the proxy server
<code>filesystem.fsnet.<ftp or http>.ProxyPassword=</code>	The password for the proxy server
<code>filesystem.fsnet.<ftp or http>.ProxyUserNameAndPassword=</code>	The username and password for the proxy server
<code>filesystem.fsnet.<ftp or http>.UseWinHttp=</code>	(Windows only) Set to 1 to use Win Http as the FSNet provider.
<code>filesystem.fsnet.<ftp or http>.AllowBasicAuthentication=</code>	Set to 1 to allow basic authentication.

Loading and saving files using virtual file systems

The MediaRich server includes default locations for all the filenames specified inside a MediaScript. For example, when loading an image in MediaScript, the system looks for the image relative to the default root directory for images:

```
MediaRichCore\Shared\Originals\Media.
```

When specifying an ICC profile, MediaRich looks for the file relative to the assumed root directory for ICC profiles:

```
MediaRichCore\Shared\Originals\Profiles.
```

You can easily change these default root directories using the MediaRich CORE Administration tool, as well as in the properties text file. Then you can load files from and save files to these virtual filesystems from within a MediaScript. Refer to the detailed information about the `load()` and `save()` functions in the *MediaRich CORE Programmer's Guide* for more information.

To create a virtual filesystem:

1. Open the `local.properties` file located in the Properties folder in the MediaRich directory.
2. Enter a new line at the bottom of the file using the following syntax:

```
virtualfilesystem.<name>=<path>
```
3. For name, enter a name for the virtual filesystem, such as `Sports`.

This name is used to specify the filesystem from within a MediaScript.

4. For path, enter a complete, native filesystem path, such as `C:/Images/Sports`. The complete entry for the example would be:

```
virtualfilesystem.Sports=C:/Images/Sports
```

Note: The filesystem name is case-sensitive and cannot contain spaces.

5. Save the changes.

You must stop and restart the MediaGenerator before the changes will take effect.

Enabling UNC paths

The *file* filesystem allows standard file URLs to be supplied to MediaScript functions that take filenames. This filesystem allows access to any file on the MediaRich server, including files available via UNC paths (files on the network).

Note: Because this filesystem allows such broad access to resources, it is read-only and disabled by default.

To enable the *file* filesystem, the following line should be added to the `local.properties` file:

```
filesystem.file.enabled=true
```

To make the *file* filesystem writeable, this line should be added also:

```
filesystem.file.writeable=true
```

Here are two examples of the use of the *file* filesystem:

```
var image = new Media();
image.load(name @ "file:C:/Documents and Settings/All Users/Documents/My
Pictures/Sample Pictures/Sunset.jpg");
var image2 = new Media();
image2.load(name @ "file://eqfileserv/home/eng/testbed/images/tif/32bit.tif");
```

Authentication

Authentication is performed by supplying a user name and password to HTTP and FTP requests. These values can be supplied in the URL, or they can be set globally so that all HTTP and/or FTP requests use the same user name and password. This is done using the `UserNameAndPassword` property.

Separate `UserName` and `Password` properties can also be used; these values are used only if the URL does not contain a `username:password` value.

Here are two examples, one that sets a user name and password for all FSNet filesystems, and another that sets them only for FTP access:

```
filesystem.fsnet.UserNameAndPassword=joeuser:tokyo
filesystem.fsnet.ftp.UserNameAndPassword=tomjones:apassword
```

File caching

The process of reading a file over a network can be time consuming. For this reason, files read via the FTP and HTTP protocols are stored in a local file cache. Subsequent accesses to the same file that occur shortly after the file was last downloaded will be served from the cache rather than being reread from the network.

The `RefreshInterval` property determines how HTTP and FTP files are cached. This value is interpreted as a time interval in seconds. This value indicates how often MediaRich should check with a HTTP or FTP server to see if a newer version of a file exists. The default value of this property is 900, or 15 minutes. If the value of the `RefreshInterval` property is a negative value, then caching is disabled and every HTTP or FTP request results in a file being retrieved from the network.

Unless caching has been disabled, caching behavior works as follows. Whenever a file is requested via HTTP or FTP, MediaRich first looks to see if the same file exists in the local cache. If the file does not exist in the cache, it is fetched. If it does, then MediaRich computes the age of that file (how long it has been since that file was last updated). If the age of the file is less than the `RefreshInterval` property, then the cached file is used. If the age of the file is greater than the `RefreshInterval`, MediaRich contacts the HTTP or FTP server to see if a newer version of the file exists. If so, MediaRich downloads a new copy of the file to the cache. If no newer version exists, then the age of the file in the cache is simply reset to 0. In all cases, the cached file is eventually returned to the caller.

Here is an example of how to disable file caching for FTP accesses:

```
filesystem.fsnet.ftp.RefreshInterval=-1
```

This example sets the refresh interval for both HTTP and FTP accesses to five minutes:

```
filesystem.fsnet.RefreshInterval=300
```

Proxy server support

To route HTTP and FTP file requests through a proxy server, use the `ProxyHost` property. A proxy host specification consists of a host name and port number, separated by a colon (:). The following is an example that designates that FTP and HTTP requests be routed to port 1133 of the server named *ourproxyserver.ourcompany*:

```
filesystem.fsnet.ProxyHost=ourproxyserver.ourcompany:1133
```

Defining additional FSNet virtual file systems

Any number of virtual filesystems can be defined in addition to the standard http and ftp filesystems. One reason to define additional HTTP or FTP filesystems would be to associate different property settings with the different filesystems. For example, if you needed to access network files via two different proxy servers, you'd need two different filesystems since the proxy server can not be specified in a virtual file path (in a URL). Another reason to define additional filesystem would be to associate different user names and passwords with each filesystem, or to differ the caching behavior between filesystems.

To define an HTTP filesystem, only a single property, `Specifier`, must be defined. The following is an example that defines a new HTTP filesystem named *foo* and sets the default user info for that filesystem:

```
filesystem.fsnet.foo.Specifier=foo
filesystem.fsnet.foo.UserNameAndPassword=joeuser:tokyo
```

A file path to a resource on this filesystem would look like the following:

```
foo://www.eq.com/images/applelogo.jpg
```

To define an FTP filesystem, an additional property, the `Ftp` property, must be specified to indicate that the filesystem will utilize the FTP protocol. The following is an example that defines a new FTP filesystem named *bar* that has caching disabled:

```
filesystem.fsnet.bar.Specifier=bar
filesystem.fsnet.bar.Ftp=1
filesystem.fsnet.bar.RefreshInterval=-1
```

The name of the filesystem, as specified after the `fsnet` portion of property name, need not match the value of the `Specifier` property. The filesystem name is used to refer to the filesystem in the properties file. The `Specifier` defines what word appears at the front of a virtual file path to indicate that filesystem. By convention, and for clarity, these two values should always be the same.

FontAlias Property

For users running MediaRich on different installation with different fonts, the `FontAlias` property can be used to map different fonts to a single common font name. This property must be added by directly editing the *local.properties* file.

Some fonts will present themselves as having one name on one OS, and as a different name on another. This behavior is inherent in the font-handling mechanisms of those operating systems, and the `FontAlias` property can be used to code around that behavior.

This feature can also be used to write scripts using abstract font names that can be globally changed. For example, you could write the script using the font name `\ "san serif bold\ "` and from within the properties file map that to Arial Bold. If you decide to switch to Univers 55, change the properties file, clear any cache and restart the MediaGenerator. All text designated by `\ "san serif bold\ "` will now be rendered as Univers 55.

Syntax

```
FontAlias.<fontname>=<real font name>
```

Where `<fontname>` is the name of the font to use in your script. The `<real font name>` is the name of the alternative font to be used when `<fontname>` is not found.

Example

```
FontAlias.Arial=Helvetica Neue
```

On a system where a font named "Arial" does not exist, the font "Helvetica Neue" is used instead. So a MediaScript command such as `makeText` or `drawText` can use "Arial" as the font name. If the "Arial" font does exist, it is used.

Using Command Line Administration Tools

In addition to the administration utility, there are MediaRich administration tools available from the command line. In the default installation, these command line tools are located at `...\Equilibrium\MediaRichCore\Admin\Tools`.

CheckDependencyTool

When dependency tracking is enabled, use this tool to verify dependencies in MediaRich cache to ensure that the content is up-to-date after modifications have been made to the originals. This feature uses a database to keep track of the entries so that there is no limit on the number of files that can be tracked. For more information, see [“MediaRich Architecture” on page 7](#).

Important: In the default installation, dependency tracking is now disabled. For more information about configuring dependency tracking, see [“Viewing and Modifying the Global Properties” on page 39](#).

If one or more virtual file paths are specified, only those paths are checked. Otherwise, all dependencies are checked. Files can be specified directly on the command line or in a file. File names should include the proper virtual file system specifier. If a specifier is not included, the "read" file system is used by default.

```
CheckDependencyTool -u <username> -p <password> [-h -f <file name> -P <port> -I <IP Address>] [<path list>]
-u <username> System Monitor administrator username
-p <password> System Monitor administrator password
-f <file name> File containing files to check.
-P <port> System Monitor port
-h Print help
-I <IP Address> System Monitor IP Address
<path list> List of files to check
```

Username and password are required.

Important: This tool relies on the timestamp of the file to determine if there are updates to be tracked. If an administrator changes a file and then later decides to overwrite that file with a previous version, dependency tracking cannot recognize the change because it is looking for a later timestamp. If you need to “roll back” changes in a file, you should overwrite the content within the file and save it to generate a later timestamp for the file.

ClearCacheTool

Use this tool to clear media cache of results older than `<age>`.

MRLs may be specified directly on the command line or in a file. If any MRLs are specified, the images associated with those MRLs are cleared (age is ignored). Otherwise, all generated images are checked using age.

If age is specified as 0, all checked images are cleared.

MRLs can be either full or partial. The following are equivalent:

```
http://server/mgen/mediaRichTest.ms
```

```
/mgen/mediaRichTest.ms
```

```
/mediaRichTest.ms
```

```
ClearCacheTool -u <username> -p <password> [-a <age> -c -f <file name> -h -P <port> -I <IP Address>] [mrl list]
```

```
-u <username> System Monitor administrator username
```

```
-p <password> System Monitor administrator password
```

```
-a <age> Delete cached media older than age (days)
```

```
-f <file name> File containing MRLs to clear
```

```
-P <port> System Monitor port
```

```
-h print this message
```

```
-c Only list out-of-date media
```

```
-I <IP Address> System Monitor IP Address
```

```
... List of mrls to check
```

Username and password are required.

Example

To clear all derivatives on the 192.168.0.0 machine generated by the MRL `"/mgen/myfolder/myscript.ms?f=main&x=42"`, use the following:

```
ClearCacheTool -u <username> -p <password> -I 192.168.0.0 /mgen/myfolder/myscript.ms?f=main&x=42
```

MRLHashTool

Use this tool to print the path to the image associated with `<MRL>`. If an image exists, the image extension is included in the path. Otherwise, the path is printed with no extension.

The MRL may be either full or partial. The following are equivalent:

```
http://server/mgen/mediaRichTest.ms
```

```
/mgen/mediaRichTest.ms
```

```
/mediaRichTest.ms
```

```
MRLHashTool [-h -P <port> -I <IP Address>] <MRL>
```

```
<MRL> MediaRich url
```

```
-P <port> System monitor port
```

```
-h print this message  
-I <IP Address> System monitor IP Address
```

ServiceTool

Use this tool to start, stop, or get the status of a MediaRich service. If multiple services are specified, the operation(s) are applied to all services specified. Selecting both stop and start is the same as selecting restart.

```
ServiceTool -u <username> -p <password> [-h -mgen -web -start -stop -restart -P  
<port> -I <IP Address>]  
-u <username> System Monitor administrator username  
-p <password> System Monitor administrator password  
-P <port> System Monitor port  
-I <IP Address> System Monitor IP Address  
-h print this message  
-mgen Target is media generator.  
-web Target is web server.  
-start start the service.  
-stop stop the service.  
-restart restart the service.
```

File Format Support

MediaRich supports over 450 400 image, video, audio, Office, Drawing, and RAW camera file types.

This appendix provides information about the read-write file format support in MediaRich, which also powers the high-end digital asset management features in , such as image transformation, rich thumbnails, and video playback.

For a regularly updated file format list, please visit our Supported Formats page on the Equilibrium web site:

<http://equilibrium.com/mediarichserver/features/#mr-filetypes>

<http://equilibrium.com/mediarichsharepoint/features/#formats>

<http://equilibrium.com/equilibriumnew/supported-formats/>

<http://equilibrium.com/equilibriumnew/oneviewer/supported-formats>

Appendix summary

“MediaRich Image File Formats Natively Supported” on page 60

“MediaRich Audio and Video File Formats Natively Supported” on page 64

“LibreOffice Expanded File Formats Supported” on page 67

MediaRich Image File Formats Natively Supported

The MediaRich CORE provides support for the following image file formats.

MediaRich Visualizer for SharePoint does not include support for key Adobe file types (such as PhotoShop, Illustrator, and InDesign) and image export is limited to the original format and JPEG. The full MediaRich ECM for SharePoint product is required to enable full image file format support within SharePoint. See "MediaRich for SharePoint Features" on page 1 for a comparison of Visualizer and the full MediaRich ECM for SharePoint product.

Note: If not indicated by a “yes”, Text Extraction may or may not work.

File extension and format	Category	Read/Write	Color space support	Metadata ingest	Metadata embed	Text extraction (all text)
PDF: Adobe Acrobat	Portable Document	Read/Write (PDF Image)	Renders to RGB, CMYK	Basic		yes (also one page, page range)
AI: Adobe Acrobat	Drawing-Vector	Read	Renders to RGB, CMYK	Basic		
PIX: Alias Workstation Image	Image	Read/Write	RGB	Basic	None	
RAW: Panasonic Camera Raw	RAW Camera Image	Read	Converts to RGB	Enhanced		
CR2: Canon Camera Raw	RAW Camera Image	Read	Converts to RGB	Enhanced		
CRW: Canon Camera Raw	RAW Camera Image	Read	Converts to RGB	Enhanced		
DNG: Adobe Digital Negative	Digital Negative	Read	Converts to RGB	Enhanced		
EPS: Encapsulated Postscript	Drawing-Vector	Read/Write	Renders to RGB, CMYK	Basic		

File extension and format	Category	Read/Write	Color space support	Metadata ingest	Metadata embed	Text extraction (all text)
EPSF: Encapsulated Postscript	Drawing-Vector	Read/Write	RGB and CMYK	Basic		
RAF: Fuji FinePix Raw	RAW Camera Image	Read	Converts to RGB	Enhanced		
GIF: Graphics Interchange Format	Image/Animation	Read/Write	RGB	Basic	Basic	
JPE: JPEG	Image	Read	RGB, CMYK, with ICC Support	Enhanced		
JPEG: JPEG	Image	Read	RGB, CMYK, with ICC Support	Enhanced		
JPG: JPEG	Image	Read/Write	RGB, CMYK, with ICC Support	Enhanced	Enhanced	
J2K: JPEG 2000	Image	Read	GRAY, RGB, YCBRC Convert	Basic		
JP2: JPEG 2000	Image	Read/Write	GRAY, RGB, YCBRC Convert	Basic	None	
JPX: JPEG 2000	Image	Read	GRAY, RGB, YCBRC Convert	Basic		
J2C: JPEG 2000 Stream	Image	Read	GRAY, RGB, YCBRC Convert	Basic		

File extension and format	Category	Read/Write	Color space support	Metadata ingest	Metadata embed	Text extraction (all text)
JCC: JPEG 2000 Stream	Image	Read	GRAY, RGB, YCBRC Convert	Basic		
JPC: JPEG 2000 Stream	Image	Read/Write	GRAY, RGB, YCBRC Convert	Basic	None	
DCR: Kodak Digital Raw	RAW Camera Image	Read	Converts to RGB	Enhanced		
PCT: Mac PICT	Image	Read/Write	RGB	Basic	None	
PICT: Mac PICT	Image	Read	RGB	Basic	None	
MOS: Creo Leaf Mosaic Camera Raw	RAW Camera Image	Read	Converts to RGB	Enhanced		
NEF: Nikon Raw	RAW Camera Image	Read	Converts to RGB	Enhanced		
ORF: Olympus Raw	RAW Camera Image	Read	Converts to RGB	Enhanced		
PCX: PC Paintbrush	Image	Read/Write	RGB	Basic	None	
PS: Photoshop	Image	Read/Write	RGB, CMYK with ICC Support	Enhanced	Enhanced	
PSB: Photoshop Large Image Format	Image	Read	RGB	Enhanced		
PNG: PNG	Image	Read/Write	RGB	Enhanced	Basic	
PBM: Portable Bitmap	Image	Read	B&W	Basic		
PPM: Portable Pixel Map	Image	Read/Write	RGB	Basic	Basic	

File extension and format	Category	Read/Write	Color space support	Metadata ingest	Metadata embed	Text extraction (all text)
PPMB: Portable Pixel Map	Image	Read	RGB	Basic		
PBMA: Portable Pixel Map	Image	Read	B&W	Basic		
PBMB: Portable Pixel Map	Image	Read	B&W	Basic		
PS: Postscript	Portable Document	Read	Renders to RGB, CMYK	Basic		Yes (also one page, page range)
BW: Silicon Graphics Image	Image	Read	RGB	Basic		
RGBA: Silicon Graphics Image	Image	Read	RGB	Basic		
SGI: Silicon Graphics Image	Image	Read/Write	RGB	Basic		
CS1: Sinar Camera Raw	RAW Camera Image	Read	Converts to RGB	Enhanced		
ARW: Sony Raw	RAW Camera Image	Read	Converts to RGB	Enhanced		
TGA: Targa	Image	Read/Write	RGB	Basic	Basic	
TIF: TIFF	Image/Multi-page image	Read/Write	RGB, CMYK with ICC support	Enhanced	Enhanced	
TIFF: TIFF	Image/Multi-page image	Read	RGB, CMYK with ICC support	Enhanced		
BMP: Windows Bitmap	Image	Read/Write	RGB	Basic	Basic	
WBMP: Wireless Bitmap	Image	Read/Write	RGB	Basic	None	

File extension and format	Category	Read/Write	Color space support	Metadata ingest	Metadata embed	Text extraction (all text)
3FR: Hasselblad Camera Raw	RAW Camera Image	Read	Converts to RGB	Enhanced		
ERF: Epson Camera RAW	RAW Camera Image	Read	Converts to RGB	Enhanced		
FFF: Hasselblad Camera RAW	RAW Camera Image	Read	Converts to RGB	Enhanced		
KDC: Kodak Camera Raw	RAW Camera Image	Read	Converts to RGB	Enhanced		
MEF: Mamiya Camera Raw	RAW Camera Image	Read	Converts to RGB	Enhanced		
MRW: Minolta Camera Raw	RAW Camera Image	Read	Converts to RGB	Enhanced		
PEF: Pentax Camera Raw	RAW Camera Image	Read	Converts to RGB	Enhanced		
SR2: Sony Camera Raw	RAW Camera Image	Read	Converts to RGB	Enhanced		
STI: Sinar Capture Shop Camera Raw	RAW Camera Image	Read	Converts to RGB	Enhanced		
X3F: Sigma Camera Raw	RAW Camera Image	Read	Converts to RGB	Enhanced		

MediaRich Audio and Video File Formats Natively Supported

AVCore 2 includes enhanced performance and GPU transcoding support. It reads all existing formats (EXCEPT FLI, FLC OR ANY VIDEO CONTAINING PALETTED DATA) and writes several existing formats, plus native Flash Video, Ogg Theora, GXF, MTS, and VPX/webm.

MediaRich CORE supports the following audio and video file formats. The video and video/audio formats support 24 bit RGB.

MediaRich Visualizer for SharePoint does not provide support for full file transcoding and video export is limited to MP4. The full MediaRich ECM for SharePoint product is required to enable full video and audio file format support within SharePoint. See "MediaRich for SharePoint Features" on page 1 for a comparison of Visualizer and the full MediaRich ECM for SharePoint product.

Important: Currently, AVCore 2 is only capable of reading non-paletted video frames.

Audio/Video File Extension and Format	Category	Read/Write	Metadata ingest	Metadata embed
AAC: AAC Audio	audio	Read	Enhanced	None
AC3: AC3 Audio Stream	audio	Read	Enhanced	None
AMR: AMR Audio	audio	Read	Enhanced	None
AIF: Audio Interchange File Format (Mac and SGI)	audio	Read, Write	Enhanced	None
AIFF: Audio Interchange File Format (Mac and SGI)	audio	Read, Write	Enhanced	None
DV: Digital Video Stream	video	Read, Write	Enhanced	None
FLV: Flash Video	video	Read, Write	Enhanced	None
F4V: Flash Video	video	Read, Write	Enhanced	None
FLC: Autodesk FLIC Animation File	animation	Read, Write	Enhanced	None
3G: iPhone cell, Mobile 3G	video	Read, Write	Enhanced	None
M4V: iPod, iPhone wifi, Apple TV	video	Read, Write	Enhanced	Enhanced (Transcode only)
MXF: Material Exchange Format	video	Read, Write	Basic	None
3GP: Mobile 3G Audio/Video	video	Read, Write	Enhanced	None
3G2: Mobile 3G Audio/Video	video	Read, Write	Enhanced	None
3GPP: Mobile 3G Audio/Video	video	Read, Write	Enhanced	None
3GP2: Mobile 3G Audio/Video	video	Read, Write	Enhanced	None
MJPG: Motion Jpeg	video	Read, Write	Enhanced	None
MP2: Mpeg Audio Layer 2	audio	Read	Enhanced	None
MPEG: Mpeg1 or 2 Video Program Stream	video	Read, Write	Enhanced	None
MPG: Mpeg1 or 2 Video Program Stream	video	Read, Write	Enhanced	None
M1V: Mpeg1 Video	video	Read, Write	Enhanced	None
MP3: Mpeg2 audio Layer 3	audio	Read	Enhanced	None

Audio/Video File Extension and Format	Category	Read/Write	Metadata ingest	Metadata embed
VOB: Mpeg2 DVD (unencrypted file required)	video	Read	Enhanced	None
PS: Mpeg2 Program Stream	video	Read, Write	Enhanced	None
MPE: Mpeg2 Standard	video	Read, Write	Enhanced	None
MPEG2: Mpeg2 Standard	video	Read, Write	Enhanced	None
MPG2: Mpeg2 Standard	video	Read, Write	Enhanced	None
MP2: Mpeg2 Standard, RAM and RM	video	Read, Write	Enhanced	None
M2T: Mpeg2 Transport Stream	video	Read, Write	Enhanced	None
M2V: Mpeg2 Video	video	Read, Write	Enhanced	None
M4A: Mpeg4 AAC Audio	audio	Read, Write	Enhanced	None
Mpeg4 (RAW): Mpeg2 RAW Video	video	Read	Enhanced	None
MP4: Mpeg4 with AAC and other standard Audio encode	video	Read, Write	Enhanced	None
OGG: OGG Vorbis Audio	audio	Read	Enhanced	None
MOV: Quicktime, Hinted movie * See Codec supported detail	video	Read, Write	Enhanced	Enhanced
RAM: Real Media (writes only 1.0/2.0)	video	Read, Write	Basic	None
RM: Real Media (writes only 1.0/2.0)	video	Read, Write	Basic	None
GXF: Sony Mpeg2-XDCAM SD/HD	video	Read, Write	Basic	None
AU: Sun Microsystems Audio File	audio	Read, Write	Basic	None
AVI: Windows Audio Video Interface	video	Read, Write	Enhanced	None
ASF: Windows Media Active Server (2) (3)	audio/video	Read, Write	Enhanced	Enhanced
WMA: Windows Media Audio (3)	audio	Read, Write	Enhanced	Enhanced
WMV: Windows Media Video (1) (3)	video	Read, Write	Enhanced	Enhanced
Wave: Windows PCM audio format	audio	Read, Write	Enhanced	None

(1) Windows Media 11 for WMV in HD (64-bit Windows Server 2003 not supported)

(2) Sharp Audio Codec must be installed.

(3) When exporting, CBR or Quality-based VBR output settings for audio are the only types supported. Bitrate VBR for audio is not supported.

Basic Metadata support means that MediaRich can extract size, bit depth, frames, and time.

Enhanced Metadata support means there is complete extraction and embedding of metadata with field handlers shown in the documentation. It can embed some kind of information other than width/height/depth/time/samplerate. The types of metadata vary from file format to file format: for instance, not every file format can embed Exif data, or author, etc.

Codecs we have tested and support with legacy QuickTime: Animation, Apple Intermediate Codec, BMP, Cinepak, Component Video, DV - PAL, DV- NTSC, Graphics, H.261, H.263, H.264, JPEG 2000, Motion JPEG A, Motion JPEG B, MPEG-4 Video, Photo - JPEG, Planar RGB, PNG, Sorenson Video, Sorenson Video 3, TGA, and TIFF.

Audio Codecs: 24-bit Integer, 32-bit Floating Point, 32-bit Integer, 64-bit Floating Point, A-Law 2:1, AMR Narrowband, Apple Lossless, IMA 4:1, MACE 3:1, MACE 6:1, MPEG-4 Audio (AAC), Qualcomm PureVoice.

LibreOffice Expanded File Formats Supported

Office File Formats Supported

MediaRich supports the following file formats when LibreOffice 6.0 is installed. They are rendered to RGB. These are read-only on Macintosh and Windows.

Note: If not indicated by a “yes”, Text Extraction may or may not work.

File extension and format	Category	Text extraction (Windows only)
PDB: AportisDoc (Palm)	Word Processing	yes
DXF: AutoCAD Interchange Format	Drawing-Vector	
CGM: Computer Graphics Metafile	Presentation	
XML: DocBook, Microsoft Excel/Word 2003 XML	Word Processing/Spreadsheet	yes
EMF: Enhanced Metafile	Drawing-Vector	
HWP: Hangul WP 97	Word Processing	yes
HTM, HTML: HTML Document	HTML	yes
OTH: HTML Document Template	HTML	yes

File extension and format	Category	Text extraction (Windows only)
PCD: Kodak Photo CD (192x128, 768x512, 384x256)	Drawing-Raster	
123: Lotus 1-2-3	Spreadsheet	yes
WK1: Lotus 1-2-3	Spreadsheet	yes
WKS: Lotus 1-2-3	Spreadsheet	yes
XLSB: Microsoft Excel 2007 Binary	Spreadsheet	yes
XLSM: Microsoft Excel 2007 XML	Spreadsheet	yes
XLSX: Microsoft Excel 2007 XML	Spreadsheet	yes
XLTM: Microsoft Excel 2007 XML Template	Spreadsheet	yes
XLTX: Microsoft Excel 2007 XML Template	Spreadsheet	yes
XLM: Microsoft Excel 4.x-5.0/95/97/2000/XP	Spreadsheet	yes
XLC: Microsoft Excel 4.x-5.0/95/97/2000/XP	Chart	yes
XLW: Microsoft Excel 4.x-5.0/95/97/2000/XP	Spreadsheet	yes
XLT: Microsoft Excel 4.x-5.0/95/97/2000/XP	Spreadsheet	yes
XLS: Microsoft Excel 4.x-5.0/95/97/2000/XP	Spreadsheet	yes
PPTM: Microsoft PowerPoint 2007 XML	Presentation	
PPTX: Microsoft PowerPoint 2007 XML	Presentation	
POTM: Microsoft PowerPoint 2007 XML Template	Presentation	
POTX: Microsoft PowerPoint 2007 XML Template	Presentation	
PPS/PPSX: Microsoft PowerPoint 97/2000/XP	Presentation	
PPT: Microsoft PowerPoint 97/2000/XP	Presentation	yes
POT: Microsoft PowerPoint 97/2000/XP Template	Presentation	
DOC: Microsoft WinWord, 5, 6.0/95, 97/2000/XP	Word Procession	yes
DOCM: Microsoft Word 2007 XML	Word Procession	
DOCX: Microsoft Word 2007 XML	Word Procession	yes
DOTM: Microsoft Word 2007 XML Template	Word Procession	yes
DOTX: Microsoft Word 2007 XML Template	Word Procession	yes

File extension and format	Category	Text extraction (Windows only)
DOT: Microsoft Word 95, 97/2000/XP Template	Word Processing	yes
OTG: ODF Drawing Template	Drawing-Vector	
ODG: ODG Drawing, ODF (Impress)	Drawing-Vector	
ODM: ODF Master Document	Word Processing	yes
ODP: ODF Presentation	Presentation	
OTP: ODF Presentation Template	Presentation	
ODS: ODF Spreadsheet	Spreadsheet	yes
OTS: ODF Spreadsheet Template	Spreadsheet	yes
ODT: ODF Text Document	Word Processing	yes
OTT: ODF Text Document Template	Word Processing	yes
STD: OpenOffice 1.0 Drawing Template	Drawing-Vector	
SXD: OpenOffice 1.0 Drawing, OpenOffice Impress	Drawing-Vector	
STW: OpenOffice HTML/Text Template	HTML	
SXG: OpenOffice 1.0 Master Document	Word Processing	yes
SXI: OpenOffice 1.0 Presentation	Presentation	
SXC: OpenOffice 1.0 Spreadsheet	Spreadsheet	yes
STC: OpenOffice 1.0 Spreadsheet Template	Spreadsheet	yes
SXW: OpenOffice 1.0 Text Document	Word Processing	yes
MET: OS/2 Metafile	Drawing-Vector	
PXL: Pocket Excel	Spreadsheet	yes
PSW: Pocket Word	Word Processing	yes
PGM: Portable Graymap	Drawing-Raster	
WB2: Quattro Pro 6.0	Spreadsheet	yes
RTF: Rich Text Format, RTF (OpenOffice Calc)	Word Processing	yes
SVG: Scalable Vector Graphic	Drawing-Vector	
SGV: StarDraw 2.0	Drawing-Vector	
SDP: StarImpress 4.0/5.0	Presentation	

File extension and format	Category	Text extraction (Windows only)
SVM: StarView Metafile	Drawing-Vector	
SDW: StarWriter 1.0, 2.0, 3.0-5.0	Word Processing	yes
SGL: StarWriter 4.0/5.0 Master Document	Word Processing	yes
SGF: StarWriter Graphics Format	Drawing-Vector	
STW: StarWriter doc	Word Processing	
SXD: OpenOffice 1.0 drawing	Drawing	
STD: OpenOffice 1.0 drawing	Drawing	
RAS: Sun Raster Format	Drawing-Raster	
SLK: SYLK	Spreadsheet	yes
602: T602 Document	Word Processing	yes
UOS: Unified Office Format spreadsheet	Spreadsheet	yes
UOF: Unified Office Format spreadsheet, text, presentation	Spreadsheet	yes
UOP: Unified Office Format presentation	Presentation	
WMF: Windows Metafile	Drawing-Vector	
WPD: WordPerfect Document	Word Processing	yes
XBM: X Bitmap	Drawing-Raster	
XPM: X PixMap	Drawing-Raster	

Administration Best Practices and Troubleshooting

As a MediaRich administrator, you should evaluate your environment and the needs of your organization so that you can provide a system that supports it. There are administrative best practices that you can implement, as well as troubleshooting information for installation and server administration.

[Appendix summary](#)

[“Best Practices for Installation and Configuration” on page 72](#)

[“Troubleshooting the MediaRich Server” on page 74](#)

Best Practices for Installation and Configuration

MediaRich CORE is a flexible platform, with many options to configure and utilize its features. This section provides some guidelines on how to best deploy MediaRich products for integration with your organization.

Multiple Processor Machines

The MediaRich server is multi-threaded and supports multiple-processor machines. Equilibrium recommends a dual-processor machine for optimum performance. When the choice between adding more processors and adding more machines is an option, we recommend adding more machines. This is because most media-serving applications are not restrained by CPU but bandwidth and disc access. Adding more machines increases the overall performance of a server farm.

Configuration for Large Image Sizes on Windows Servers

Setting a custom paging file size in the OS with a small minimum can cause "out-of-memory" errors to be returned by MediaScript when working with large images. Make sure to set a large enough minimum paging size to handle your images, or simply use "System Managed Size". This currently only applies to Windows.

Load Balancing/Failover

The MediaRich Media Server operates essentially as a Web server with additional media processing capabilities. MediaRich can be load balanced with any standard HTTP load-balancing solution. Failover can also be managed with any standard load-balancing solution.

Development/Staging/Production Servers

The MediaRich Media Server can be treated like any other Web server configuration. Typically the production site will have at least two MediaRich servers for load balancing and failover. One server will be required for staging. Each developer will not need to run a full MediaRich installation. Pushes from development to staging to production can be done using any standard file copy program.

Web Server Configuration

If you are licensing the MediaRich Media Server or MediaRich Merchandizer, you must have a Web server to be used in conjunction with the MediaRich CORE server. A Web server handles the HTTP requests and makes calls to the MediaRich system to process scripts and generate media. MediaRich integrates with IIS on Windows Server 2012R2 or later using an ISAPI filter and with Apache on Solaris or Linux using an Apache mod. The separation of systems allows the system administrator to configure the Web server separately from MediaRich. This allows the system administrator to optimize and secure the Web server according to their IT guidelines.

To enable IIS 6.0 to run 32-bit applications on 64-bit Windows:

1. Open a command prompt and navigate to the %systemdrive%\Inetpub\AdminScripts directory.
2. Enter the following command:
3. Press ENTER.
4. Restart the machine.

The system should return the following:

```
Microsoft (R) Windows Script Host Version 5.6
Copyright (C) Microsoft Corporation 1996-2001. All rights reserved.
Enable32BitAppOnWin64 : (BOOLEAN) True
```

Important: Do not do this where the 64-bit machine is running SharePoint at the same time, or where IIS websites are using 64-bit .NET. Making the above change kills these other applications.

Depending upon the particular server, it is possible you will need to precede the above with these steps:

1. Navigate to the **Advanced Settings** in the main Application Pool and set **Run 32-bit Applications** to TRUE.
2. Start the Server Manager (`run mmc.exe` or click on the toolbox icon in the desktop task bar).
3. Navigate to **Roles > Web Server (IIS)**, look in the far right column, and click **Add Role Services**.
4. In the wizard, navigate to **Management Tools > IIS Management Compatibility** and select the check box so that all sub-features are also installed.
5. Scroll down, click **Install** and wait for the system to install and exit.
6. Launch the MediaRich CORE Administration tool, choose **License > Uninstall license**, and install the license.bin again.

Note: Depending upon the server configuration, you might have to launch the Command Prompt by right-clicking it and choosing 'Run as Administrator.' In that case, you would likely also have to do that when running the MediaRich CORE installer.

Shared File System

MediaRich Media Server utilizes three types of files: original images, scripts, and derivative image files (cached files). The first two types are source files and MediaRich generates the third type. You can deploy MediaRich so that all load-balanced servers can share a common file system that contains these files. This can be accomplished through a network file share. The advantage of this system is that all cached images are shared. That means when an image is generated on server A for an image request, server B does not also need to generate the image because it is available on the common file

share. Also, pushing original images and scripts is easier because only one file system must be kept up-to-date. However, if the shared file system fails, all servers are unable to deliver images. We strongly recommend using a fault-tolerant file-sharing system.

Distributed File System

If a fault-tolerant system is not an option, it is possible to have each MediaRich server maintain its own local copy of the original images, scripts, and cached files. The benefit is that you remove the possibility of a single point of failure. The disadvantage is that each server must be mirrored for each push. Another disadvantage is that generated images are only generated on the requested server, with each server maintaining its own cache. To work around these drawbacks, you can utilize any standard file-mirroring software to synchronize the servers. Because the MediaRich cache is file-based, it is also possible to copy the cache from staging to production, thus reducing any image-generation overhead.

File System Plug-in

The primary file system for MediaRich is the location where it reads and writes files. This is usually the local server drives or a mounted network drive. It is possible for MediaRich to read/write files on other file systems. Currently, MediaRich supports reading and writing files via HTTP and FTP. Using these alternate file systems makes it possible for MediaRich to grab images from a remote location such as a vendor's site or to write files to a location to be used for another purpose besides delivering to a Web browser, such as for print or distribution on a CD.

Troubleshooting the MediaRich Server

Occasionally, some of the components of MediaRich fail to perform as expected for a variety of reasons. This section is designed to help the administrator diagnose and solve some of the most common issues.

Server Troubleshooting Checklist

When troubleshooting MediaRich, the following check list may help you in resolving common problems:

- Check the `ScriptErrors.log` file. This file indicates critical errors that prevent MediaScripts from executing. (If there is a non-critical error, the MediaScript may still execute, using default values in place of the erroneous commands or parameters.)
- Check the `MediaGenerator.log` file. If the `ScriptErrors.log` file does not display any script error messages, check the `MediaGenerator.log` file for system error messages. This log file shows the current state of the web filter (if used), Media Generator, and System Monitor. This can help you to solve the problem and/or provide Equilibrium Technical Support with a starting point.

Note: You can use the "LogLevel" setting in the Local Properties window of the MediaRich

Server Administration utility to modify the amount of information in the logs.

- Check the `MediaGeneratorStartup.log` file, located in the folder where MediaRich Merchandizer was installed. This file contains information about any errors encountered by the web filter at startup.
- Restart web server and the MediaGenerator and System Monitor services. Restarting these components may clear some problems.

Solving Common Problems on Windows

MediaGenerator service failing to start

When this situation occurs, no images are generated. Do the following to resolve this issue:

- If you are running IIS, ensure that IIS and the Default Web Site are both running properly. If IIS and the Default Web Site are running, check the `ScriptErrors.log` file.
- Ensure that the MediaGenerator is running. (You can start the MediaGenerator using Computer Management under **Services and Applications > Services**.)
- Check the `MediaGenerator.log` file for errors. If there is no `MediaGenerator.log` file but the MediaGenerator is running, check for error messages in the `MediaGeneratorStartup.log` file, located in the MediaRichCore directory.

File or directory not found

If a files cannot be accessed by the browser, it typically displays the “HTTP error 404 - File or directory not found” message on the page.

If you see this message when opening or using any of the demo pages on a Windows installation operating under IIS, do the following to resolve the issue:

- In the IIS Manager, under Web Service Extensions, make sure that Active Server Pages has a status of *Allowed*.

Incorrect or slow image display

You might encounter situations where images do not appear on the page, cached images appear rather than generated images, or the MediaGenerator is extremely slow. The first thing to do is check the log files for errors, but if no errors are indicated, do the following to resolve the issue:

If there is no `MediaGeneratorStartup.log` file on a Windows installation running IIS, check your ISAPI filters to ensure that the `URLToImageFilter.dll` filter is running:

1. Right-click the **My Computer** icon and choose **Manage**.
2. Select **Services and Applications > Internet Information Services**, right-click **Web Sites**, and choose **Properties**.
3. Click the **ISAPI Filters** tab.
The `URLToImageFilter.dll` filter should show a green arrow in the **Status** column.
4. Click the **Directory Security** tab and ensure that those settings are correctly configured.

5. Restart IIS and ensure that the green arrow now appears as described in Step 3.
If the green arrow still does not appear, remove and reinstall the URLToImageFilter.dll filter, then restart IIS again.
6. Ensure that IIS and the Default Web Site are running properly. If IIS and the Default Web Site are running:
 - Restart MediaGenerator and IIS. Sometimes IIS and the MediaGenerator get out of sync and cannot communicate properly. Restarting both components frequently clears up the problem.
 - Clear the cache using the ClearCacheTool

Changing from VisualIntegrity to Ghostscript

If you have a need to use Ghostscript here are the procedures:

1. Use the Administration Utility to stop the MediaGenerator.
2. Go to the /Library/Equilibrium/MediaRichCore/Bin/MediaEngine folder and remove "VisualIntegrity.pdv" file.
3. Start the MediaGenerator.

File System Permissions

MediaRich web server filter that handles media requests runs as the same user as web server - www for Apache, Internet Guest user (IUSR_xxx) for IIS. This means that the filter has the same read/write privileges as that user. When the filter starts, it writes the

.../MediaRichCore/MediaGeneratorStartup.log file in the install directory. Any startup errors are written to that file. On Windows, the identity of the Internet Guest user is recorded along with any failures to impersonate this user - if there are errors recorded in this file, then the password for the user is likely incorrect.

After restarting, check the default MediaRich Merchandizer demo web pages. If these pages fail to produce the desired result, but no errors are reported at the bottom of the

.../MediaRich/MediaGeneratorStartup.log, locate and open the
.../MediaRichCore/Shared/Logs/<computername>/MediaGenerator.log file.

At the bottom of this file, there should an entry for each component of the MediaRich Merchandizer server ("SystemMo", "MediaGen", and "Filter"). Successful startup generates the following:

```
2005/11/15 05:05:58 SystemMo A SystemMonitor ready
2005/11/15 05:05:59 MediaGen A MediaGenerator ready
2005/11/15 05:06:21 Filter A IIS Filter ready
```

If this file is missing, the service user (www or IIS anonymous user) might not have write access to the shared folder. In this case, add write access to this user for the Shared folder and all subfolders. If there is no entry for one of the three components, that component did not have write access to the shared folder. Determine the user of the service involved (IIS anonymous user for Filter, service logon user for MediaGen and SystemMo on Windows, www and mediarich on Mac or Linux), and make sure that each user has read/write access to the shared folder and all subfolders.

Other entries in the

.../MediaRichCore/Shared/Logs/<computername>/MediaGenerator.log file indicate other sources of error (such as a corrupt, missing, or expired license key).

If you see an error about a missing license key, you need to copy your license.bin file into .../MediaRichCore/Shared/Properties. You can also install the license file using the MediaRich Administration Utility.

In case of other errors, please go to the support section on equilibrium.com, which is the exclusive area for support submissions:

<http://www.equilibrium.com/support/support-overview/>

MediaRich Log Files

MediaRich server logs all errors into three log files at different times in the execution sequence.

When the web filter starts up, it writes an entry into the

.../MediaRichCore/MediaGeneratorStartup.log

identifying the user that it will attempt to impersonate. Any failure of this impersonation is also written to this file.

When each component starts up, an entry is written to the

.../MediaRichCore/Shared/Logs/<computername>/MediaGenerator.log file (if the component has permission to do so). The startup entries in these files typically appear as:

```
2005/11/15 05:05:58 SystemMo A SystemMonitor ready
2005/11/15 05:05:59 MediaGen A MediaGenerator ready
2005/11/15 05:06:21 Filter A IIS Filter ready
```

generally as:

```
Date Component Severity Message
```

In this example, components are SystemMo - A System Monitor, MediaGen - MediaRich Media Generator service and Filter - web filter.

These entries are appended to the log each time any component restarts. Additionally, all system level execution errors are written to this file. The server administrator should regularly examine this file for potential problems with the system.

Script errors

While server is running, all errors generated while compiling and executing MediaRich scripts are logged to ScriptErrors.log (in .../MediaRichCore/Shared/Logs). Refer to the *MediaRich Programmer's Guide* for detailed information about these errors.

LogLevel

The amount of information logged is determined by LogLevel parameter in Local Properties file (.../MediaRichCore/Properties/local.properties). You can edit properties and change the value of this parameter using the administrative utility (Properties menu).

MediaGeneratorStartup.log Error Messages

The following table lists error messages that you might encounter in the `MediaGeneratorStartup.log` file.

Note: Some error messages could require further investigation, beyond what can be described here.

Error message	Probable cause/solution
Cannot create directory	Web filter could not create the hostname subdirectory of the logs directory. Check file permissions.
Cannot write to <code>MediaGenerator.log</code>	Check file permissions.
LogonUser failed "[reason for failure]"	IIS filter could not log on as the IIS anonymous user. Check the username and password in the Internet Information Services Panel of the Computer management snap-in.

MediaGenerator.log Error Messages

The following table lists error messages that you might encounter in the `MediaGenerator.log` file.

Note: This is not a complete list.

Error message	Probable cause/solution
"The license key is invalid. Please contact Equilibrium for a new license key."	Check the Shared/Properties directory for a <code>license.bin</code> file. If the file is missing, add it. This error can also be caused by an incorrect "SharePath" entry in the <code>local.properties</code> file.
MediaRich exception "Could not connect to Media Generator process (Error:10061 Unable to connect socket)." while processing ...	MediaGenerator Service is stopped. Start the MediaGenerator Service.
MediaRich exception "Missing value for <MRL> at <loc>" while processing <MRL>	Check the MRL for errors.

ScriptErrors.log Error Messages

For information about error messages related to the `ScriptErrors.log` file, refer to the *MediaRich CORE Programmer's Guide*.

Index

A

adding servers 36
architecture 6
Audio
 file formats supported 64

B

batch process status 47

C

CheckDependencyTool 56
configuration 17
core platform 6

D

DefaultTimeToLive 40
DependencyCheckedFilesystems 40
DependencyTracking 40
Digimarc iKit 13
Digimarc ImageBridge 13

E

email configuration 32
email delivery 32

ExcludedDependencyFilesystems 40
execution threads 42
ExecutionThreadPriorities 38

F

FFmpeg 17
file formats
 audio/video 64
 images 60
 LibreOffice-enabled 67
file system 8
Flash video transcoding 17
FlushRequestCounter 40
fonts 45
 specifying path 38
 viewing available 45
FSNET plug-in 49
FTP 32, 51
 FSNET plug-in 49
FTP protocol 32

H

hot folders
 configuration 34
HotFolderNoActivityDelay 38
HTTP 51
HTTP file support 49
HTTP protocol 32

I

- IIS 14
 - Application Pools 22
- IIS anonymous user 76
- IIS filter 7, 76
- image
 - file formats supported 60
- ImageServerRoot 40
- installer 15
- installing
 - license file 30
 - MediaRich servers 11
- Internet Information Services (IIS) 14

L

- LibreOffice 13
 - file formats supported 67
 - installation 13
 - upgrade 22
- license files 30
- license key 13
- local properties 37
- LocalFontsPath 38
- LockTimeoutPeriod 40
- log files 8, 43, 77
 - display 44
- log in 29
- log options 32
- logging in 29
- LogLevel 38
- LogRolloverFrequency 38
- logs
 - log options 32
- LookupFilesystems 40

M

- major upgrade
 - complex 24
 - simple 23
- MaximumBatchInactiveTime 40
- MaximumExecutionTime 40
- Media Server 6
- MediaCache folder 6, 27

- MediaGenerator
 - managing 35
 - starting and stopping 35
 - viewing status 35
- MediaGenerator.log 77
- MediaGeneratorStartup.log 77
- MediaLoadMaxHeight 40
- MediaLoadMaxWidth 40
- MediaPath 38
- MediaRich Media Server 6
- MediaRich Merchandizer 6
- MediaSocketPort 38
- MediaSocketThreads 38
- Merchandizer 6
- Microsoft IIS 14
- Microsoft Office documents 13
- minor upgrade
 - complex 21
 - simple 20
- ModulesPath 38
- MRLHashTool 57

O

- office documents 13
- OutputPath 38

P

- permissions 76
- PhotoshopPath 38
- ProfileScriptPath 41
- properties files 37
- PSPluginExecutionTimeout 38

Q

- QuickTime 23

R

- removing servers 36
- ReturnHtmlErrors 41

S

- SavedPropertiesFiles 21
- ScriptErrorLifetime 41
- ScriptErrorLogging 41
- ScriptsPath 38
- server options 31
- server status 35
- servers
 - adding or removing 36
 - configuring 17
 - global properties 39
 - set-up 17
 - starting 31
 - stopping 31
 - working with log files 43
- service user 15
- services 7
- ServiceTool 58
- shared folder 7
- SharedColorProfilesPath 38
- SharedFontsPath 39
- SharedLogsPath 39
- SharedPath 39
- starting 31
- stopping 31
- SysPath 39
- SystemMonitorPort 39

T

- TempPath 39
- TimeToLiveTracking 41

U

- upgrades 19

V

- Video
 - file formats supported 64
- virtual file systems 33
- Virtual Filesystems (VFS) 33-34

W

- Windows
 - service user 15

Z

- zip file 19

