MRWS™

MediaReactor Workstation User Guide

Table of Contents

Introduction

This manual is for MediaReactor Workstation version 5.x software from Drastic Technologies, Ltd.

Conventions

This manual assumes the following:

That the user knows how to operate a mouse and keyboard and perform the basic functions of Microsoft Windows, OS-X or Linux operating system.

That the user is familiar with the video editing software in use.

That the user has access to technicians capable of placing the device on the network and setting up any SAN systems if necessary.

The name of a control or display present on the interface will be displayed in **bold** text.

Where a portion of the manual is referred to the name of section mentioned will be displayed in italics.

Certain images in this document may have been grayed out where it is useful or necessary to place indicator marks to show specific controls or displays above a darker background.

About MediaReactor Workstation Software

MediaReactor Workstation software provides plugin components to allow various third party applications to access supported file types in real time without a transcoding step. There is also a user directed interface called MR Lite, and there are command line components.

MediaReactor Workstation is a direct plug-in version of Drastic's MediaReactor file translation system. Rather than having to use a separate translation step, the artist can work with incoming footage directly in their preferred creative software. MediaReactor Workstation is available on Windows (XP, Vista, 7 - 32/64) and MacIntosh OS-X Intel (10.5+), and Linux x64. It provides plugins for Assimilate Scratch/Lab, Final Cut Pro 7/Suite 3, Avid Media Composer, Adobe products, Autodesk Flame/Flare/Lustre, QuickTime and DirectShow. Many other applications that utilize the QuickTime and DirectShow interfaces are also supported.

Contemporary production workflows are able to access an ever increasing choice of acquisition, production, mezzanine and output devices and formats. Unfortunately, the proprietary nature of these media tools often leads to time consuming and potentially lossy translations of content, simply to allow other creative tools to access them.

MediaReactor Workstation was designed to alleviate this problem. MediaReactor Workstation provides access to the widest range of file formats of any single plugin package. It specializes in high quality conversion on the fly, on everything from ancient OMF JPEG to the latest camera acquisition formats from Weisscam, Phantom and Arri. Coupled with its incredible range of file and codec support, MediaReactor Workstation provides a group of plugins that make the content directly compatible with most major editing and creative software on Mac and Windows systems.

More detailed information on MediaReactor Workstation, including related products and links to demo downloads, can be found on the product page:

www.mediareactor.ws

Streamlined versions for Adobe, and for Avid, are also available.

If you are editing with Adobe products, MediaReactor Lite for Adobe provides a targeted set of the most commonly requested read and write formats for products such as Premiere, Prelude, Media Encoder and After Effects. Check the product page:

MediaReactor Lite for Adobe

If you are editing with Avid products, MediaReactor Lite for Avid provides a targeted set of the most commonly requested read and write formats for products such as Media Composer, NewsCutter and Symphony. Check the product page:

MediaReactor Lite for Avid

System Requirements

Recommended Environment

MediaReactor Workstation software must be installed on a system at least as powerful as the configurations listed below.

Minimum Hardware Platform
Post 2000 multi core processor capable of running the host application.

Recommended Hardware Platform
Recommend hardware from the host application vendor. Multi cores and OpenCL, Cuda and shader GPUs will be used if available, but are not absolutely required.

Demo downloads of MediaReactor Workstation software are available for the user to test their application and to confirm their workflow. Faster and more powerful hardware will provide better performance. In some cases, specific hardware will be required in order to enable resource-intensive, advanced or optional features.

Supported Operating Systems

MediaReactor Workstation provides support for the three major operating systems that have well defined plugin systems (for Linux, please contract Drastic for available options). The primary plugin interfaces on Mac OS-X, Windows 32 bit and Windows 64 are:

- Autodesk Flame/Flare/Lustre import
- Assimilate Scratch and Scratch Lab SPA plugin
- Avid Media Composer AMA access plugin
- Adobe file import and export plugins
- Final Cut Pro 7/Suite 3 real time access and rendered file export

The Final Cut Pro 7/Suite 3 plugin is OS-X specific, but Avid and Adobe are available on both platforms allowing seamless access to content with platform independence. Drastic is dedicated to supporting as many file formats as possible on all three major operating systems and supporting all major production tools that provide a plugin interface.

Mac OS-X

MediaReactor's primary plugins under OS-X are Apple's QuickTime layer and Adobe's import and export plug-ins. MediaReactor Workstation connects to QuickTime at multiple levels. It provides codecs for encoding and decoding video frames, image importers and exporters, movie importers and exporters and low level data handlers. The result is that all QuickTime enabled applications can access media as easily as a native QuickTime movie including still formats like DPX, Cineon, ARI and DNG sequences. This includes metadata, multitrack audio, multitrack video, time code, closed captioning and other tracks. All supported formats, including IPB MPEG transport streams and series of stills (that would normally require conversion) are directly available for editing in Apple, Adobe and many other third party products.

Windows 32

MediaReactor Workstation supports all four primary Windows' audio/video layers, which are: VFW (Video For Windows), DirectX/DirectShow, Apple QuickTime for Windows and Adobe import/export plugins for older Adobe products (CS5+ uses the Windows 64 version). QuickTime is the most standardized platform for media access on 32 bit Windows, and the MediaReactor Workstation connects to QuickTime on multiple levels. It provides codecs for encoding and decoding video frames, image importers, image exporters, movie importers, movie exporters and low level data handlers. The result is that all QuickTime enabled applications can access media as easily as native QuickTime movies including still formats like DPX, Cineon, ARI and DNG sequences. This includes metadata, multitrack audio, multitrack video, time code, closed captioning and other tracks. All supported formats, including IPB MPEG transport streams and series of stills (that would normally require conversion) are directly supported available for editing in Apple, Adobe and many other third party products. For some applications, DirectShow or VFW provide the easiest or only interface for MediaReactor to file import. DirectShow allows player applications like Windows Media Player to access all the available file formats seamlessly. MediaReactor Workstation 32 also supports Adobe Creative Suite applications that run under 32 bits, including CS4 and earlier.

Windows 64

The latest production tools tend to prefer operating as 64 bit applications. MediaReactor Workstation for Windows 64 provides interfaces to these applications. Assimilate Scratch and Scratch Labs are great examples of powerful programs running on Windows 64. Drastic provides the technology to Assimilate for their CSP plug in, and MediaReactor Workstation takes that interface to a whole new level. Providing a much wider set of file formats and codecs, MediaReactor Workstation allows Assimilate's software to access almost every production file type your workflow could need.

The latest Adobe Creative Suite products, CS5 and CS5.5, are also native 64 bit applications. MediaReactor Workstation provides both import and export plugins for all of Adobe's Creative Suite tools. The plugins provide all the advanced media support and even improve on Adobe formats like MOV and MXF.

The Windows 64 bit version of MediaReactor Workstation license also allows you to install and use the Windows 32 bit version, which is fully compatible and very useful, in Windows 64 (XP, Vista or 7).

Getting Set Up

Installation

Here is how to install MediaReactor Workstation. Attach a standard keyboard, mouse and VGA monitor to the system. Plug the system in and turn it on.

IMPORTANT! Log on as Administrator. If the user name and password entered do not extend Administrator privileges, MediaReactor Workstation will not install properly. If unsure as to what level of privilege the log-on provides, consult a System Administrator.

Confirm that any other programs are closed before starting this installation.

Make sure the system clock is properly set. Open the **Control Panel**, go to **Date/Time**. If the time is reset while running a duration-limited license, it may destroy the license.

For Windows: Double-click or run the executable MediaReactor Workstation installation file.

SetupMRWS_Win64_5_#_###.exe

Depending on your OS setup, you may need to allow the installer to run. Once you have done this, and selected the installer language, you will see a dialog similar to this:
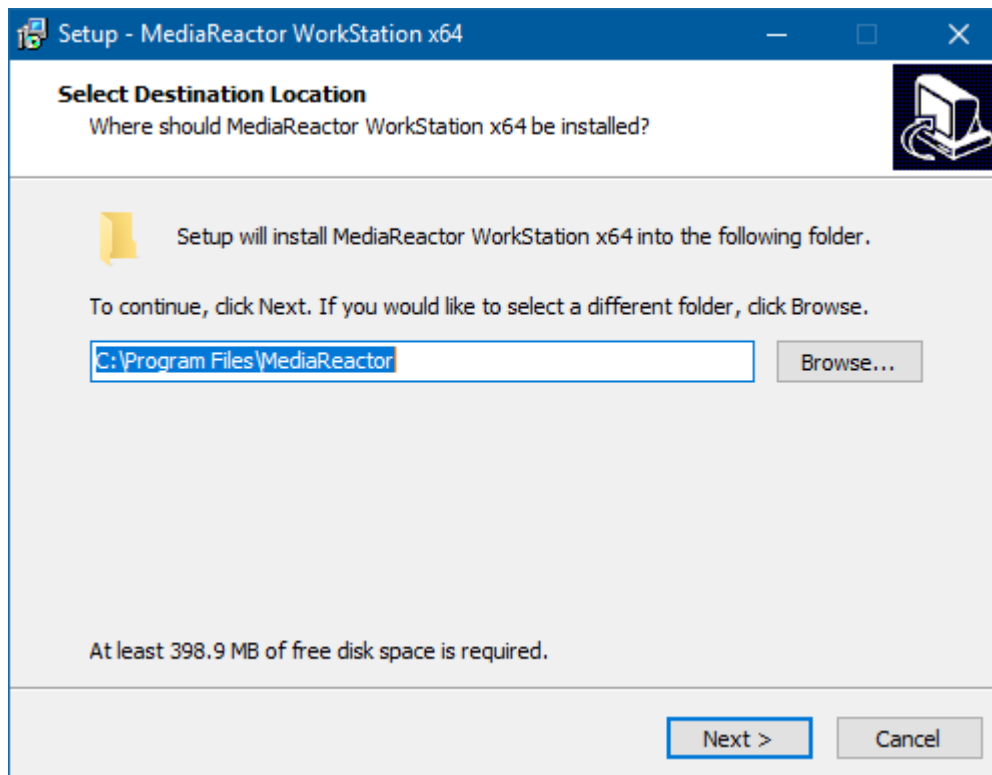
The user will be prompted to move through each step of the install by pressing the "**next**" or "**finish**" button. Carefully read the instructions that are provided.

Note: When installing on a Windows 64 bit system make sure to install both the 32 bit and 64 bit versions if you are also running 32 bit applications.

For OS-X: Double click on the OS-X PKG installer:

SetupMRWS_MacOS-X_5_#_###

This should display a dialog similar to this:
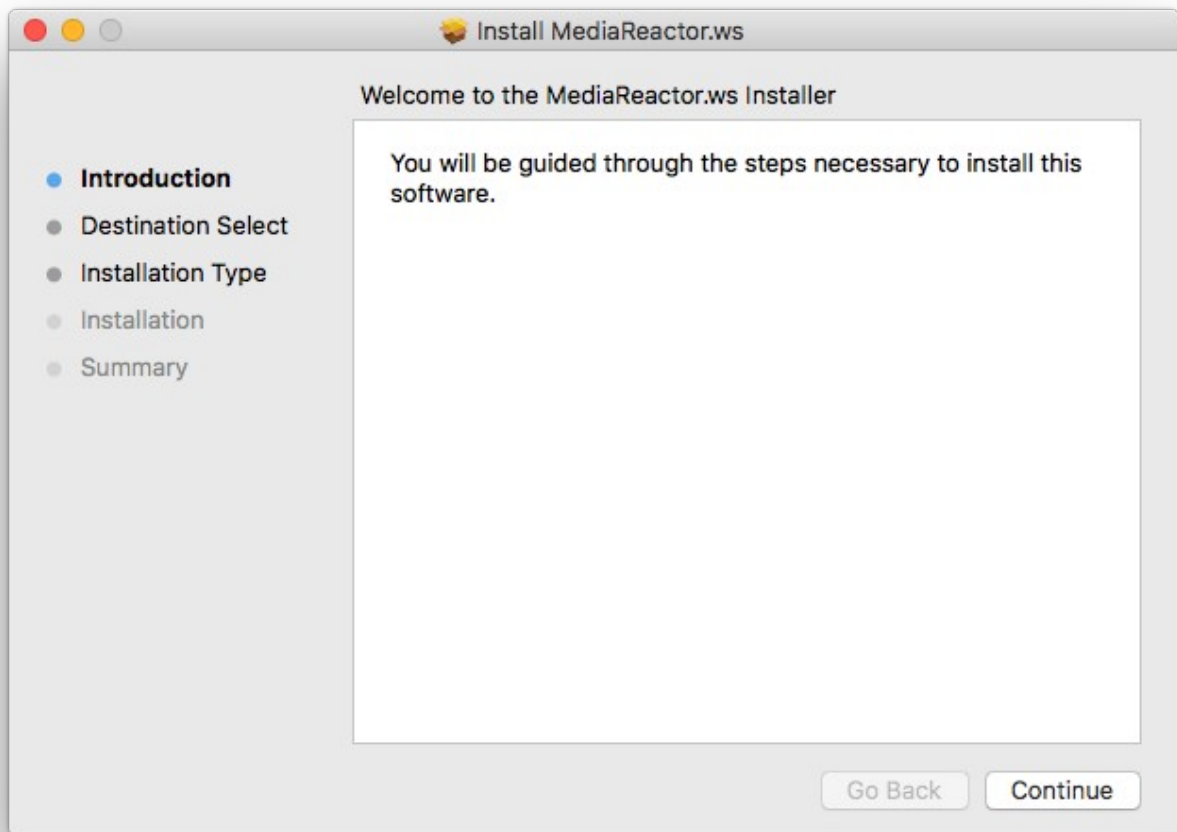
Click Continue and carefully follow the instructions to install the software.

Upon install completion, MediaReactor Workstation will prompt the user to **Restart**. Agree to **Restart**.

Where to find your files

Windows

C:\Program Files\MediaReactor\
C:\ProgramData\Drastic\
Application specific plug in directories
Links to applications are available under the Start menu, MediaReactor folder

OS-X

/Applications/MediaReactor/
/Libraries/Frameworks/
      DrasticDDR.framework
      Intel_IPP.framework
Application specific plug in directories

Linux

/opt/mediareactor/
Application specific plug in directories

License MediaReactor Workstation

Upon initial installation, Drastic software will run, albeit with specific demo limitations. To make use of the product's complete functionality the user will need to obtain a license for the software. Also, within a licensed program the user may wish to update the license, such as in cases where the application requires the addition of a licensable codec, or to add to or alter the capabilities of the existing install, to upgrade the version, or to add another Drastic software product to the system.

Here is how to license your installation of MediaReactor Workstation:

Licensing for Windows

MediaReactor Workstation installs an application called **LicenseMR**, which is included in the program folder. This may be accessed (in Windows installs) via the Start menu, Program Files|<Drastic Software>|LicenseMR). Run **LicenseMR** and you will see the following interface:
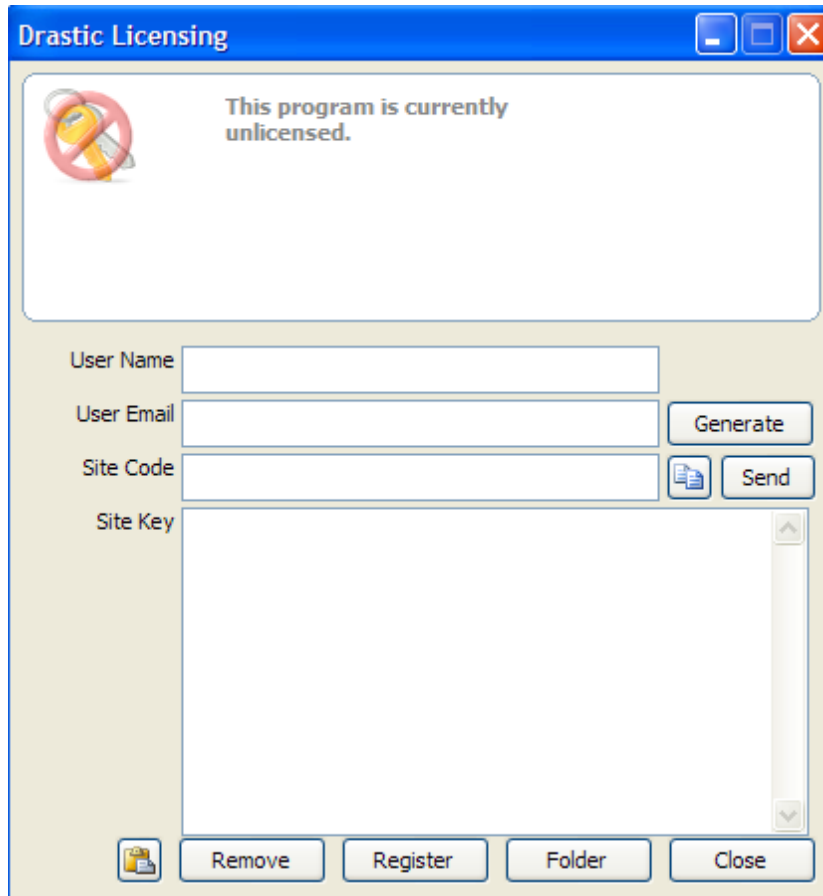
- Type the user name into the **User Name** field.
- Type a valid email address into the **User Email** field. This is the email address at which the user will receive the required site key, so make sure it can be accessed by the operator of the system upon which the Drastic software is installed.

- Press the **Generate** button. This will populate the **Site Code** field with the site code. The site code is a seemingly random string of alphanumeric characters which will be used to generate the site key.
- If the system is set up with email, the user may press the **Send** button. This opens a new email addressed to Drastic with the site code in the body of the email. If the system is not set up with email, the user may copy (either select the site code and use a standard Copy command, or press the **Copy** button which is just to the right of the **Site Code** field) and paste (use a standard Paste command) the site code into a text file, and open the text file in a system that is set up with email. In this case, create a new email addressed to authorization@drastictech.com . it would be useful to type a quick explanation regarding which software is to be licensed, and the reason for the license request (such as demo, update license, add software etc.). Send the email.
- We will reply with an email containing another seemingly random string of alphanumeric characters - this will be the site key.
- Copy the site key from the email into the buffer, and paste it into the **Site Key** field, either using a standard paste command, or by pressing the **Paste** button (the button with the icon to the left of the **Remove** button).
- Press the **Register** button, and restart the software. This will update the license status and enable all the features supported by the requested license level.
- To view the folder in which the license is located, press the **Folder** button. This opens the folder containing the license, which may be useful in cases where the user may need to move or delete the license.
- The license may be removed simply be pressing the **Remove** button.
- Once the user is finished with the licensing process, they may press the **Close** button to close **Drastic Licensing** window.

Here is an image of the Drastic licensing application in a system that has a valid license for various Drastic products. The status of the license is displayed in the top field, with a list of all software and other components that have been licensed and are available for use.
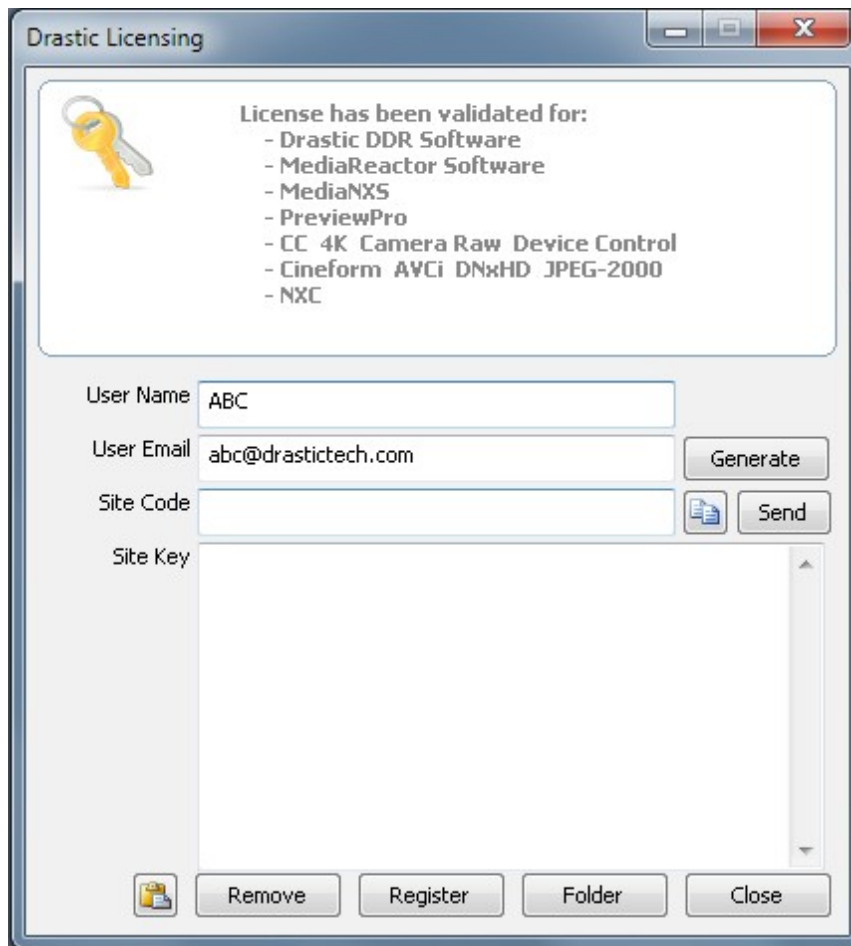
Licensing for Linux and OS-X

To access the site code using LicenseMR under Linux:

Open the command line and navigate to the opt/mediareactor directory i.e cd /opt/medi-areactor

Run the license application as a super user, i.e sudo ./licensemr4

After the application is opened, fill in your username and email and click on the Generate button. This will generate the site code which you will need to send to the authorization dept to issue a site key. Paste the key in the Site Key field and click on register to activate the license.

To access the site code using LicenseMR under OS-X:

Go to Applications and open the MediaReactor folder. In that folder will be the LicenseMR application.

After the application is opened, fill in your username and email and click on the Generate button. This will generate the site code which you will need to send to the authorization dept to issue a site key. Paste the key in the Site Key field and click on register to activate the license.

Through DDRSetup (Linux and OS-X):

There is a license app installed with MediaReactor called ddrsetup. On the command line go to its directory and type:

./ddrsetup -l -n "Your Name" -e "your.email@company.com"

In the command above, use the quotes.

Copy and paste the output and send it to authorization@drastictech.com. Drastic will reply with an email containing the Site Key.

License MediaReactor Controls and Displays

Here is a close look at the License MediaReactor interface.
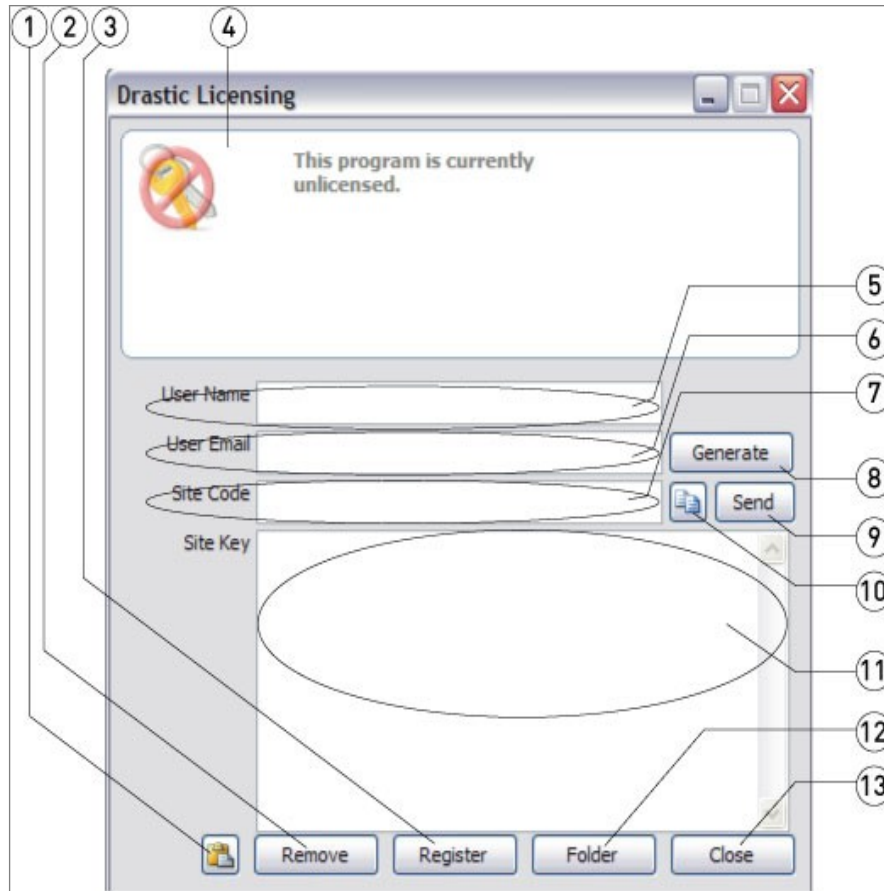
1	**Paste** button	Pressing the **Paste** button pastes the contents of the clipboard into the **Site Key** field.
2	**Remove** button	Pressing the **Remove** button deletes the existing license.
3	**Register** button	Pressing the **Register** button matches the **Site Key** with the **Site Code**, and provides a license if the combination is valid.
4	**License Status** field	Displays the current status of the license.
5	**User Name** field	In the license generation process, the user will need to enter a first and last name into this field.
6	**User Email** field	In the license generation process, the user will need to enter a valid email into this field. Note that the email provided to Drastic in this dialog will be the recipient of the **Site Key**, which is required to complete the license generation process.
7	**Site Code** field	This field displays the **Site Code**, if one has been generated.

8	**Generate** button	In the license generation process, once the user has entered a name and email, pressing the **Generate** button will populate the Site Code field.
9	**Send** button	If the system is set up with email, pressing the **Send** button creates an email containing the site code, addressed to authorization@drastictech.com.
10	**Copy** button	Pressing the **Copy** button copies the **Site Code** to the clipboard.
11	**Site Key** field	This field is where the user will paste the Site Key in order to proceed with the license generation process.
12	**Folder** button	Pressing this button opens the folder in which the license is contained.
13	**Close** button	Pressing this button closes LicenseMR.

Using MediaReactor Workstation

MediaReactor Workstation is designed to provide direct, real time editing of non editor native formats for Adobe, Avid, Apple, Assimilate, and other QuickTime or DirectShow based products.

MediaReactor Workstation installs a series of plugins which allow editors to add media (which would normally have required a transcoding step) directly into their editing package, and use the media as if it were a natively understood format.

Additionally, MediaReactor Workstation provides the MR Lite interface, for user directed transcoding. This is an application which the user can start up independently, to select media and transcode it to their preferred format.

For specific applications there are also command line components available for use.

NOTE: When outputting Panasonic P2 formats, a CONTENTS and a series of subdirectories are created to contain the file – these are not created in the selected directory.

...with Adobe Premiere CC

To bring a file into Premiere for editing, select the menu **File | Import** or right click in the Project area and select **Import**.
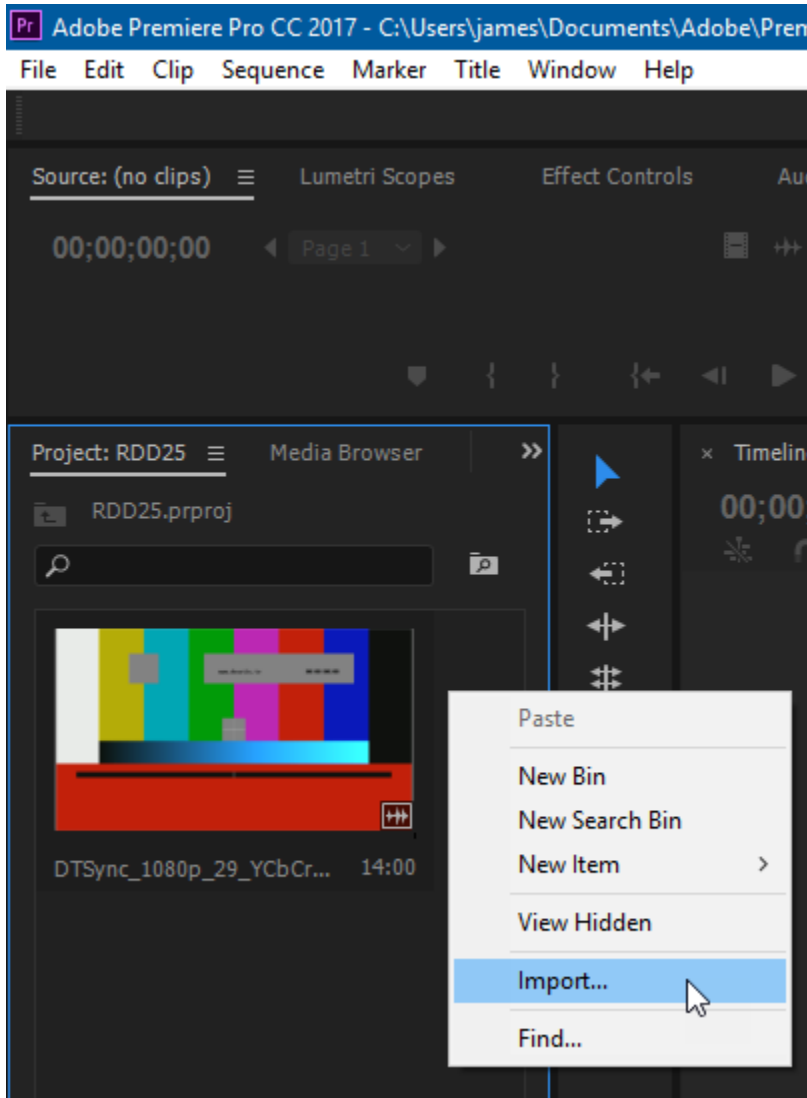
Adding files via the Media Browser area is also supported.

To export a file from Premiere, select the menu **File | Export → Media**.
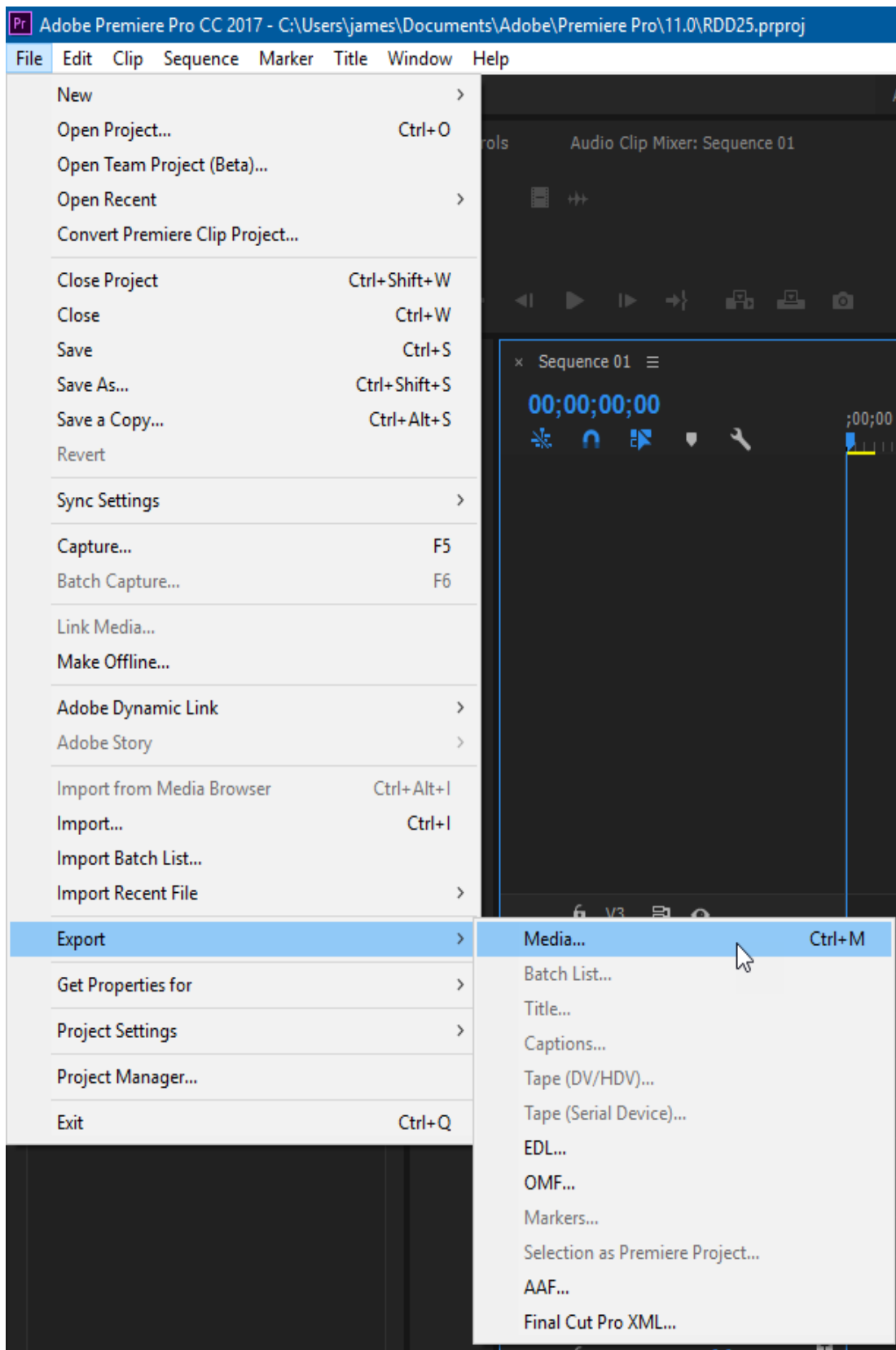
This loads the file into the dialog box.
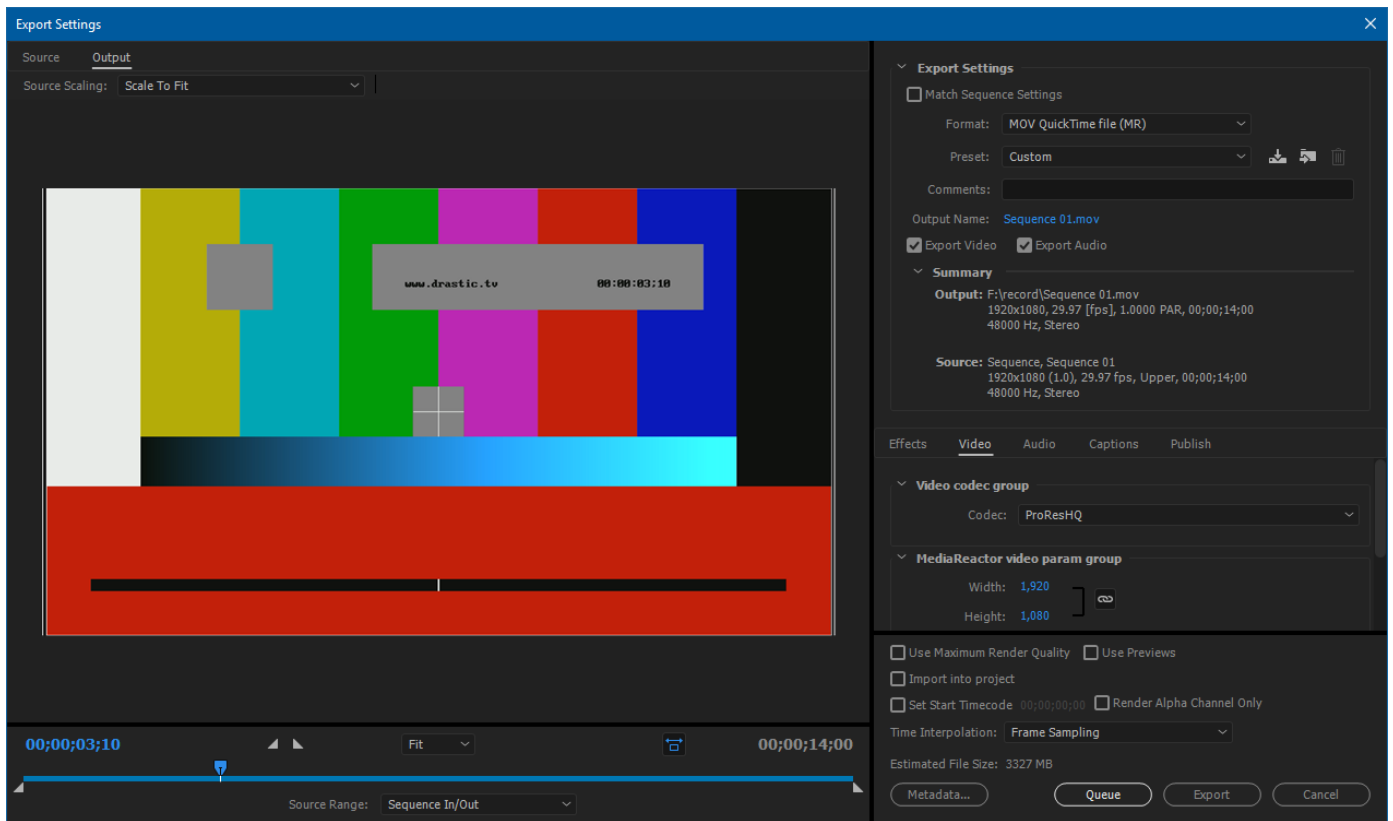
Select the Format drop down to select the type. Set the settings and click the Export button.

...with Adobe After Effects CC

To bring a file into After Effects for editing, select the menu **File | Import** or right click in the Project area and select **Import**.
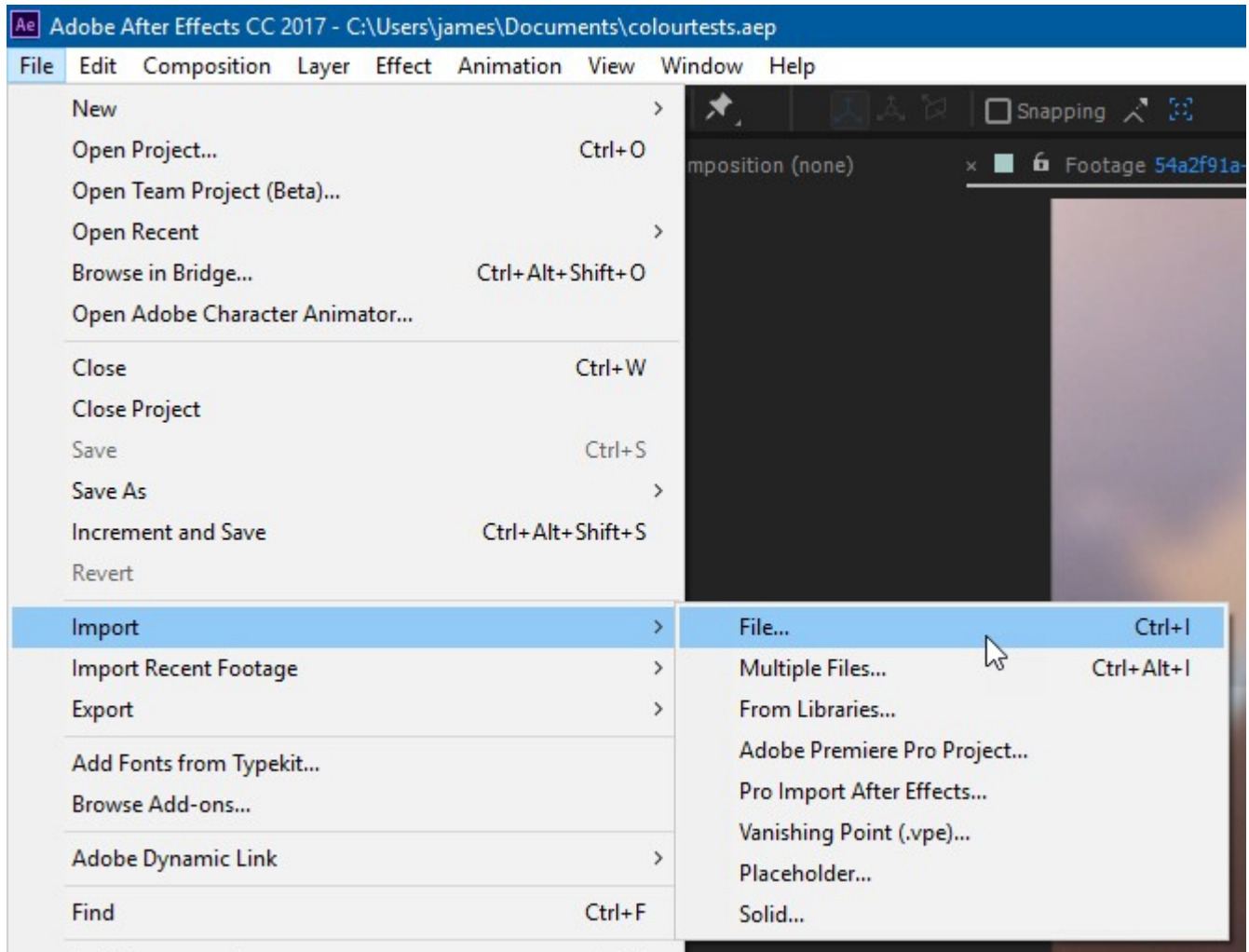
To Export a project via MediaReactor, select the File and Export menu
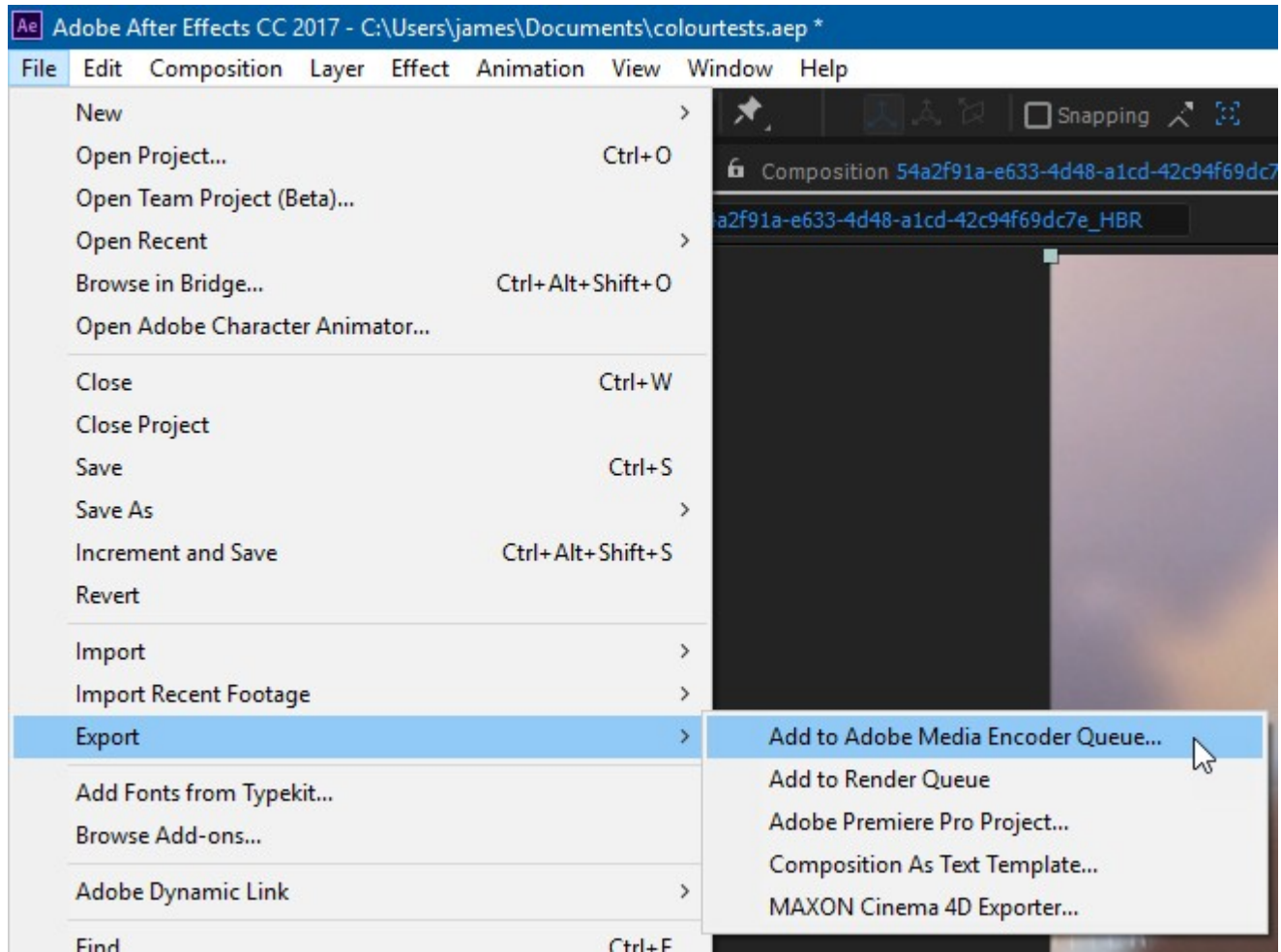
For more information on exporting, see the Premiere CC or Media Encoder CC sections of this manual.

...with Adobe Media Encoder CC

In Media Encoder, use the Media Browsers to find the file you want to convert.
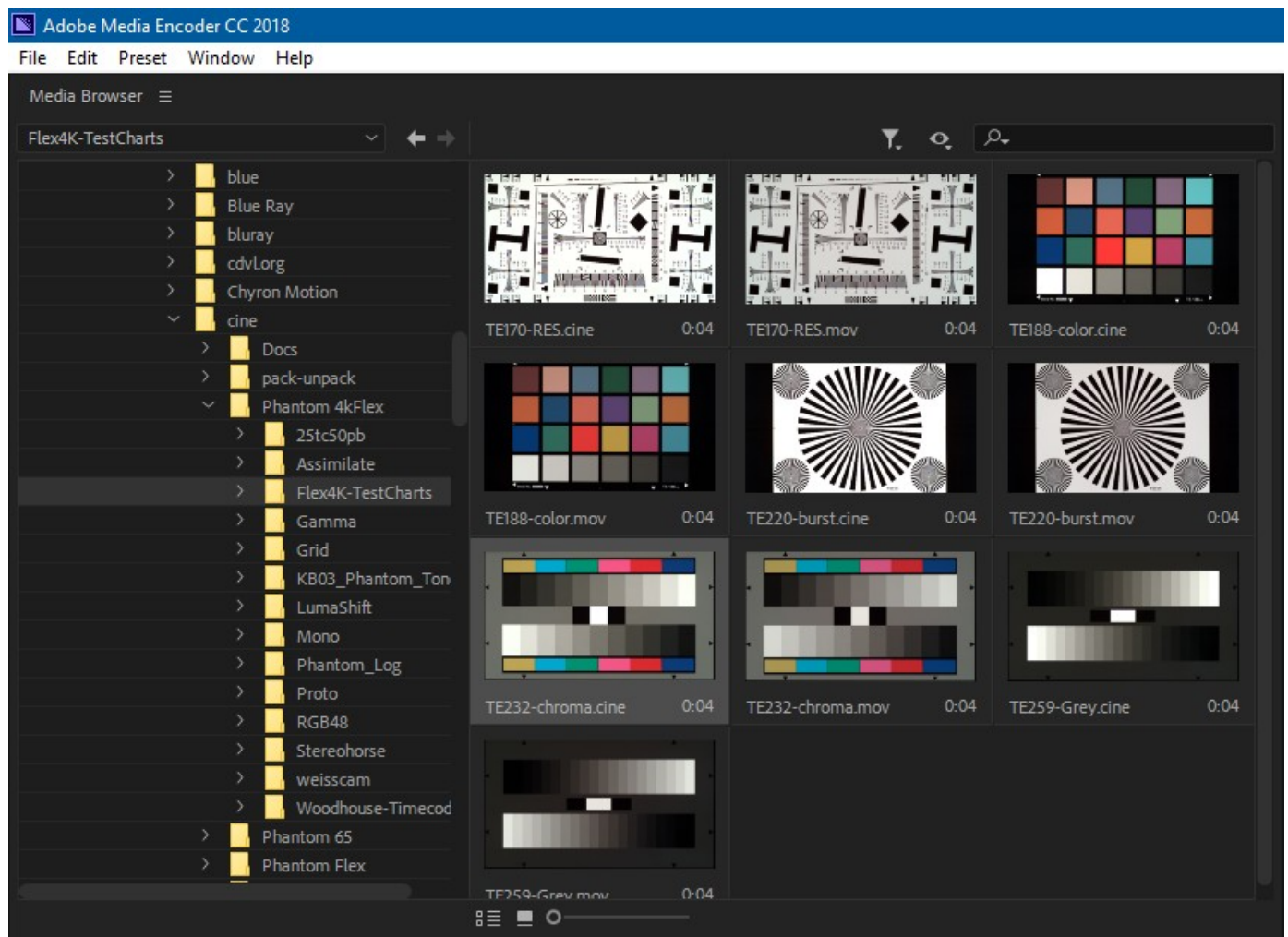
Double click on the file to add it to the Queue. Click on the first drop down to select the output file type
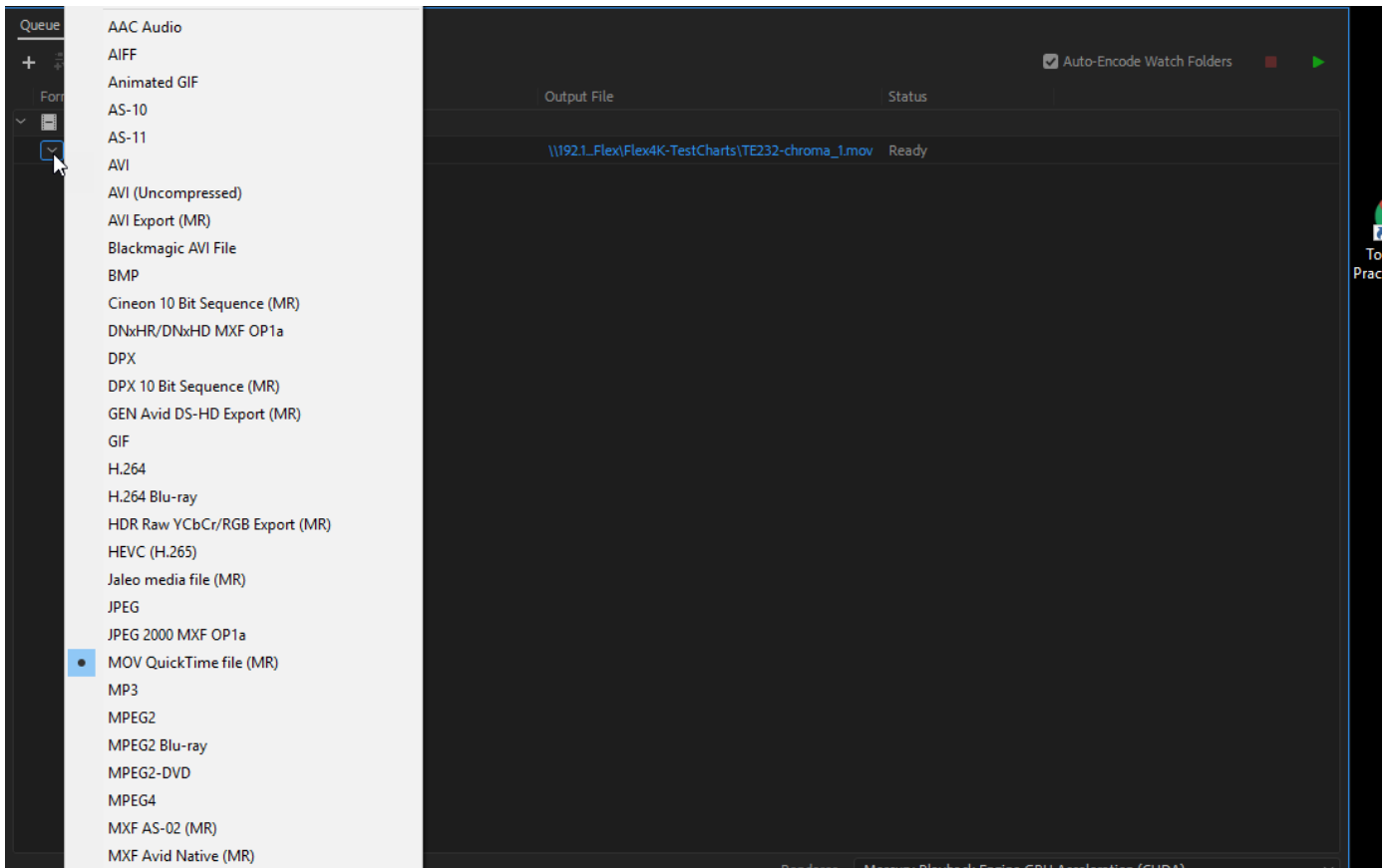
Once you have selected the output file type, clicking on the Output link will bring up the Export settings dialog.

To export a file from Premiere, select the menu **File | Export → Media**.

This loads the file into the dialog box.
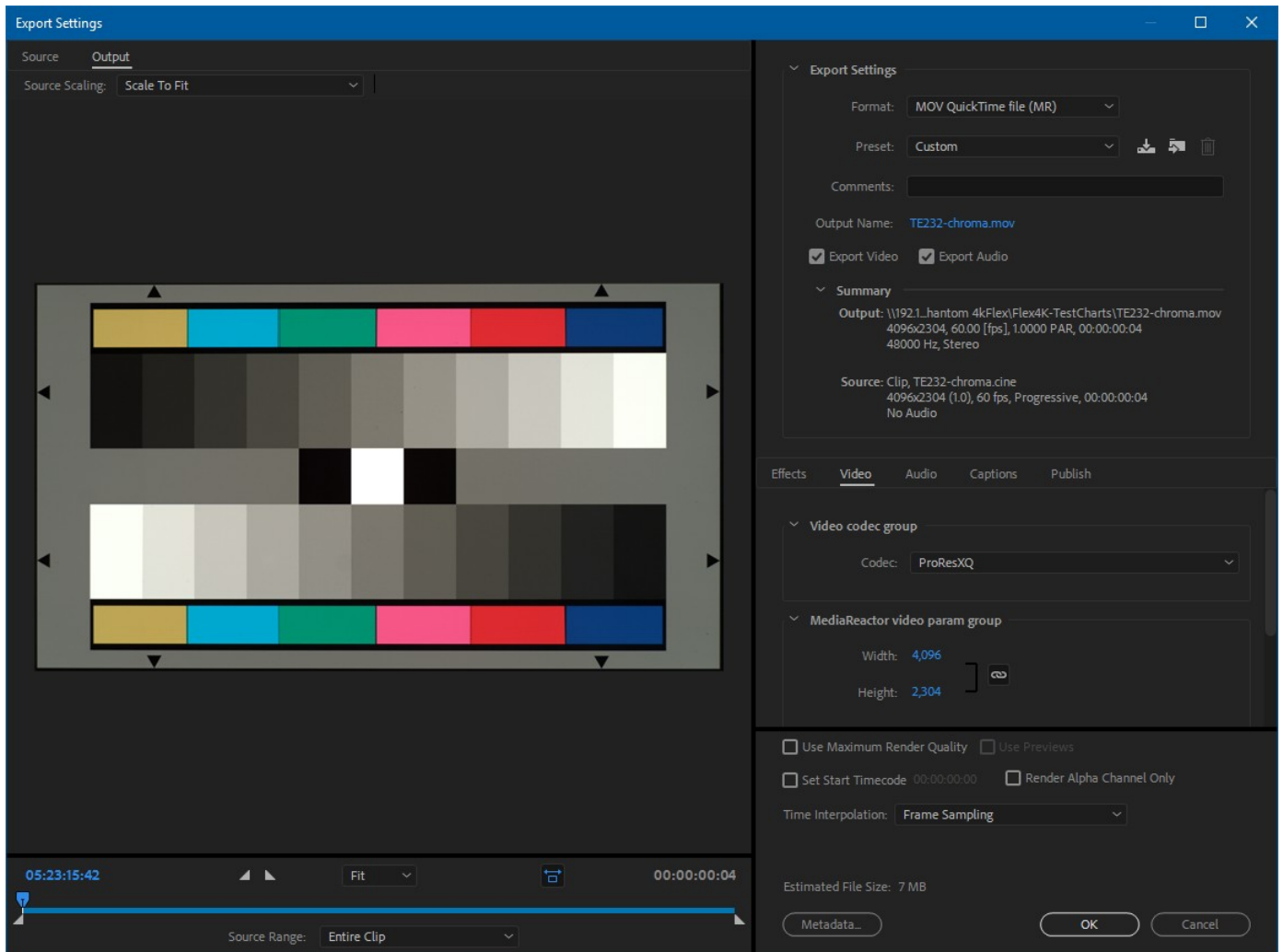
Select the Format drop down to select the type. Adjust or confirm the settings and click the Export button.

...with Adobe Prelude CC

To bring a file into Prelude for logging, select the menu **File | Ingest** or double click in the Project area.
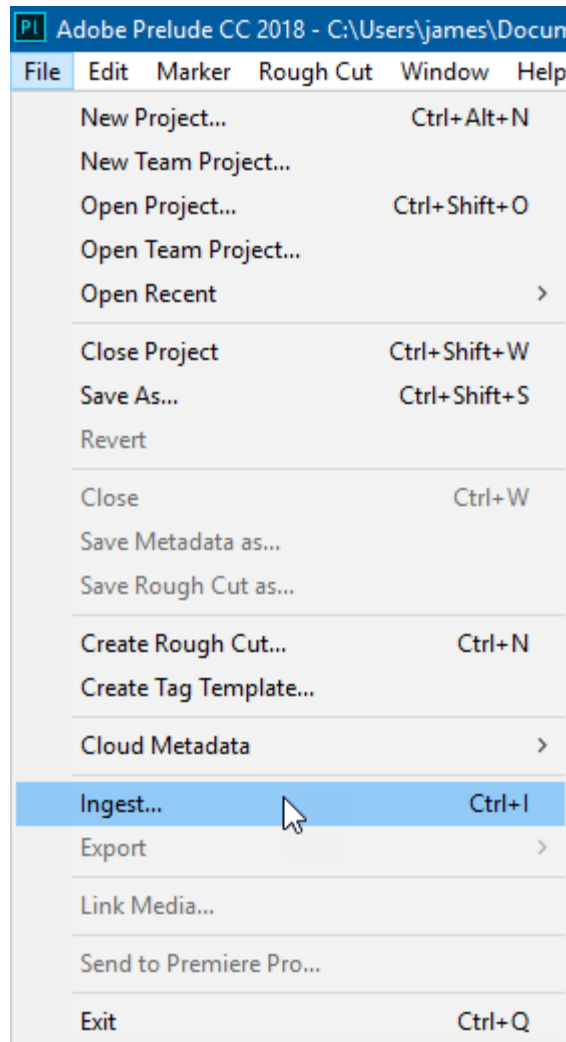
This will bring up the ingest dialog
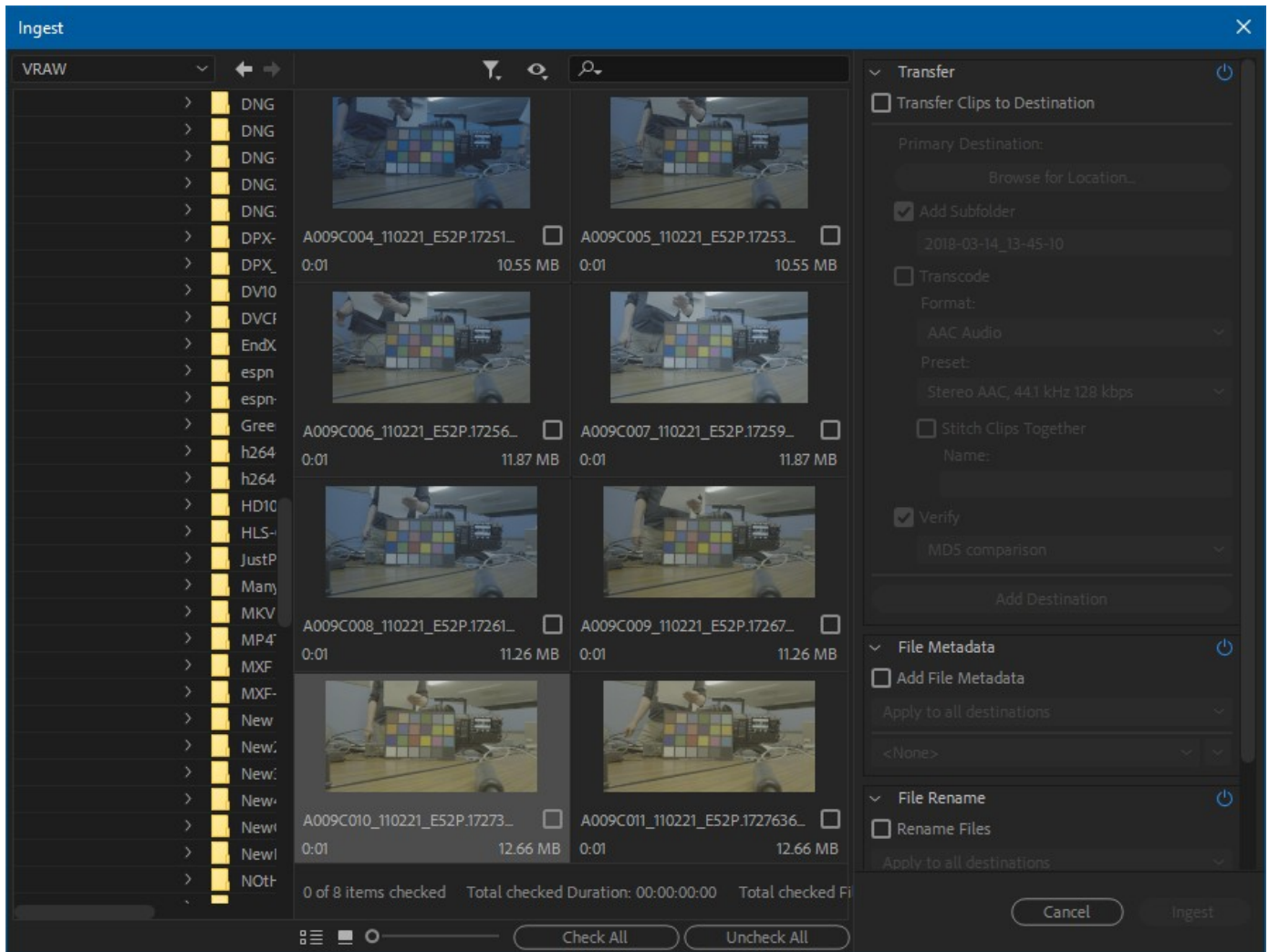
Checking one or more of the files, and clicking ingest will bring the files into Prelude
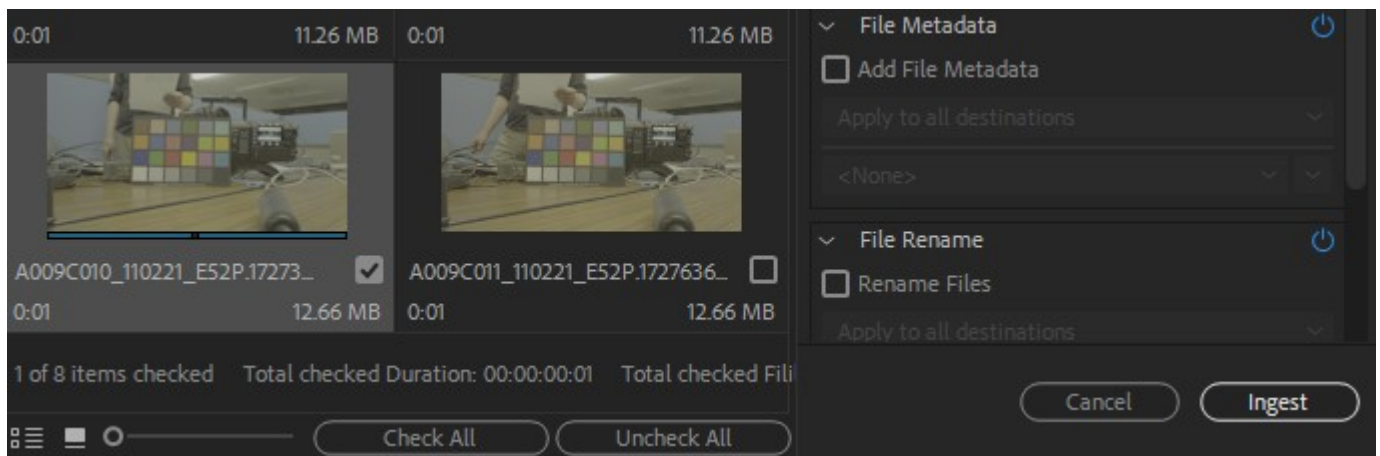
Once it is imported into the project, double clicking on its icon will bring it up in the player and timeline.
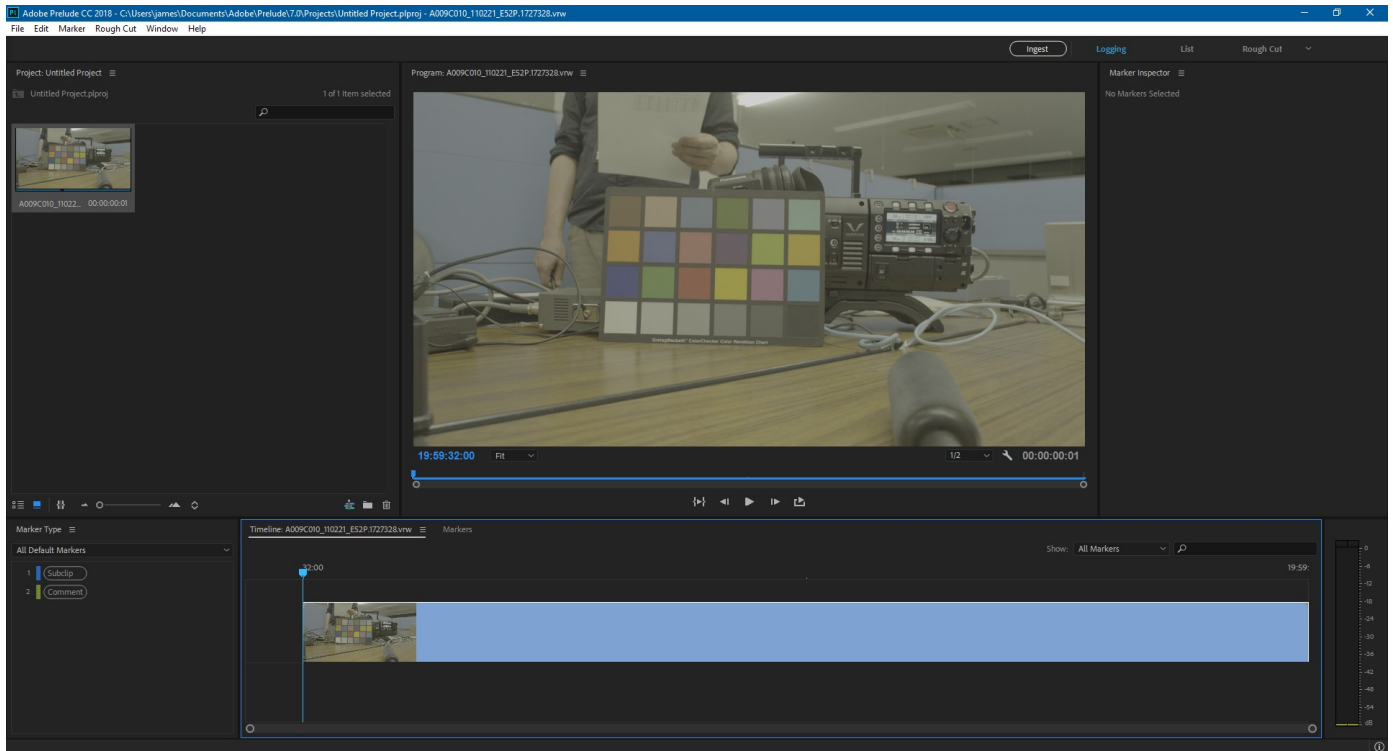
Clicking on the clip in the project will also allow you to export it as another format
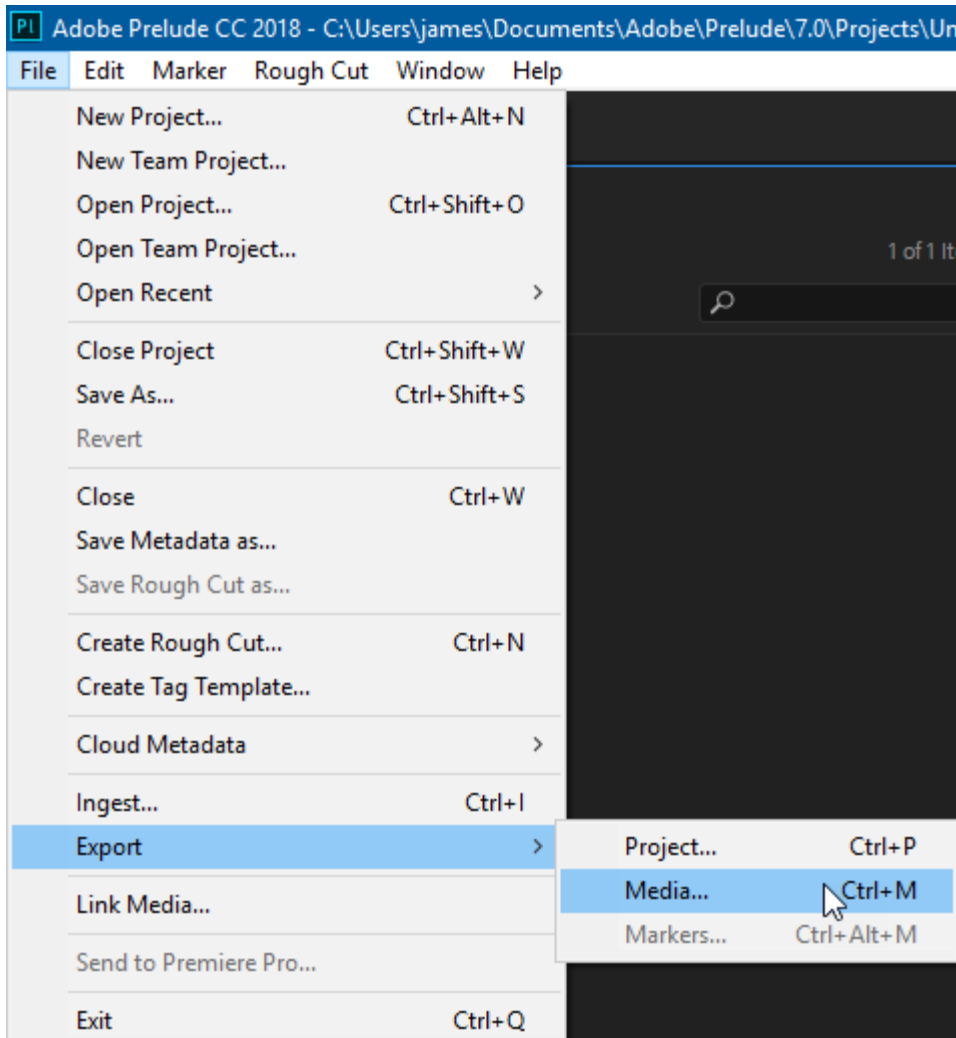
The export type can be chosen in the Export Media dialog
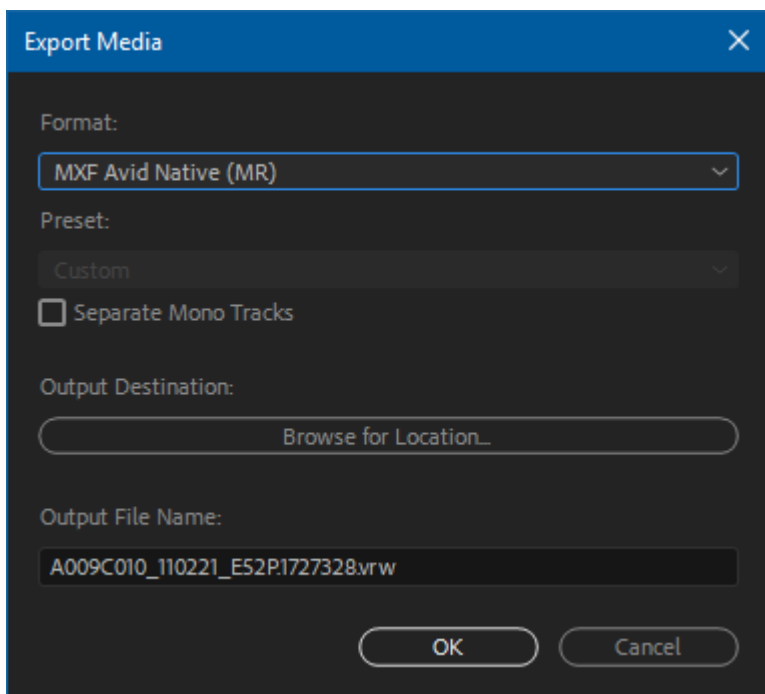
...with Adobe Premiere CS5/6

To bring a file into Premiere for editing, select the menu **File | Import** (1) or right click in the Project area and select **Import** (2).
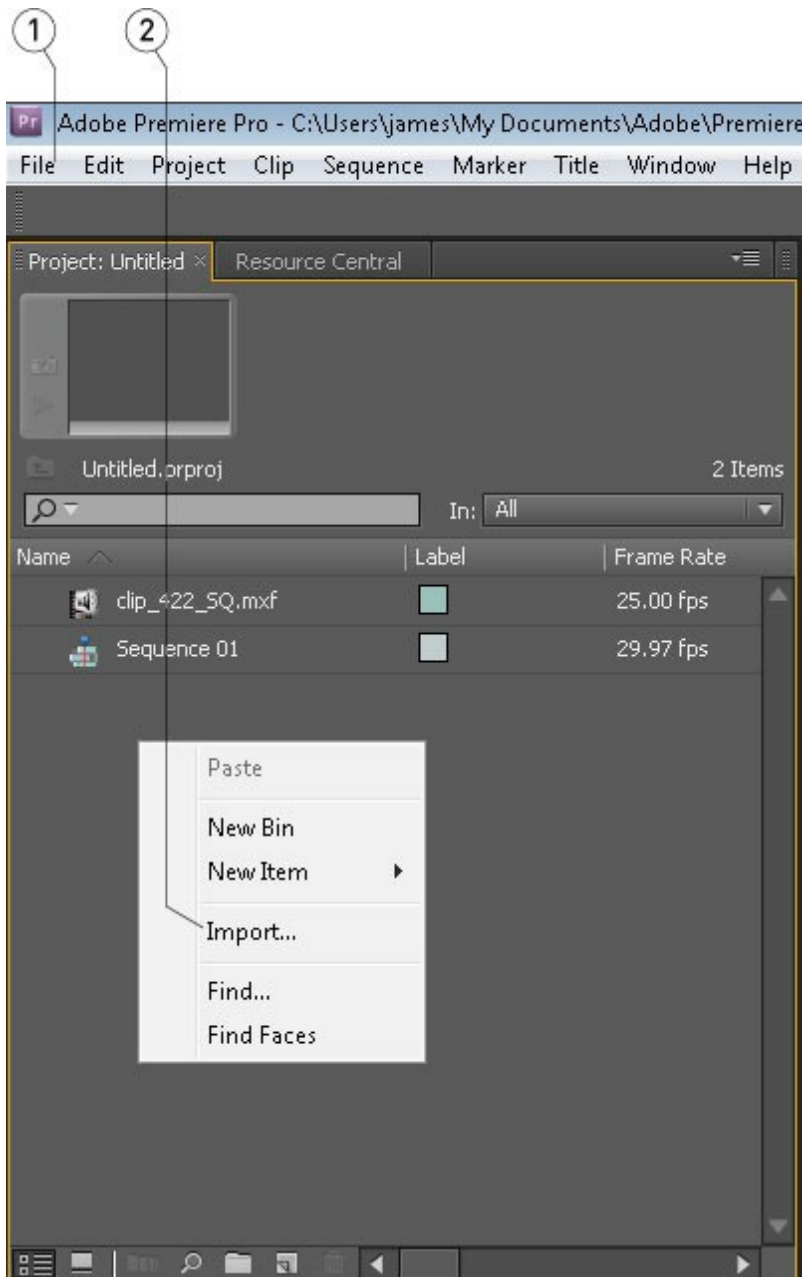
Adding files via the Media Browser area is also supported.

To export a file from Premiere, select the menu **File | Export → Media** (1).
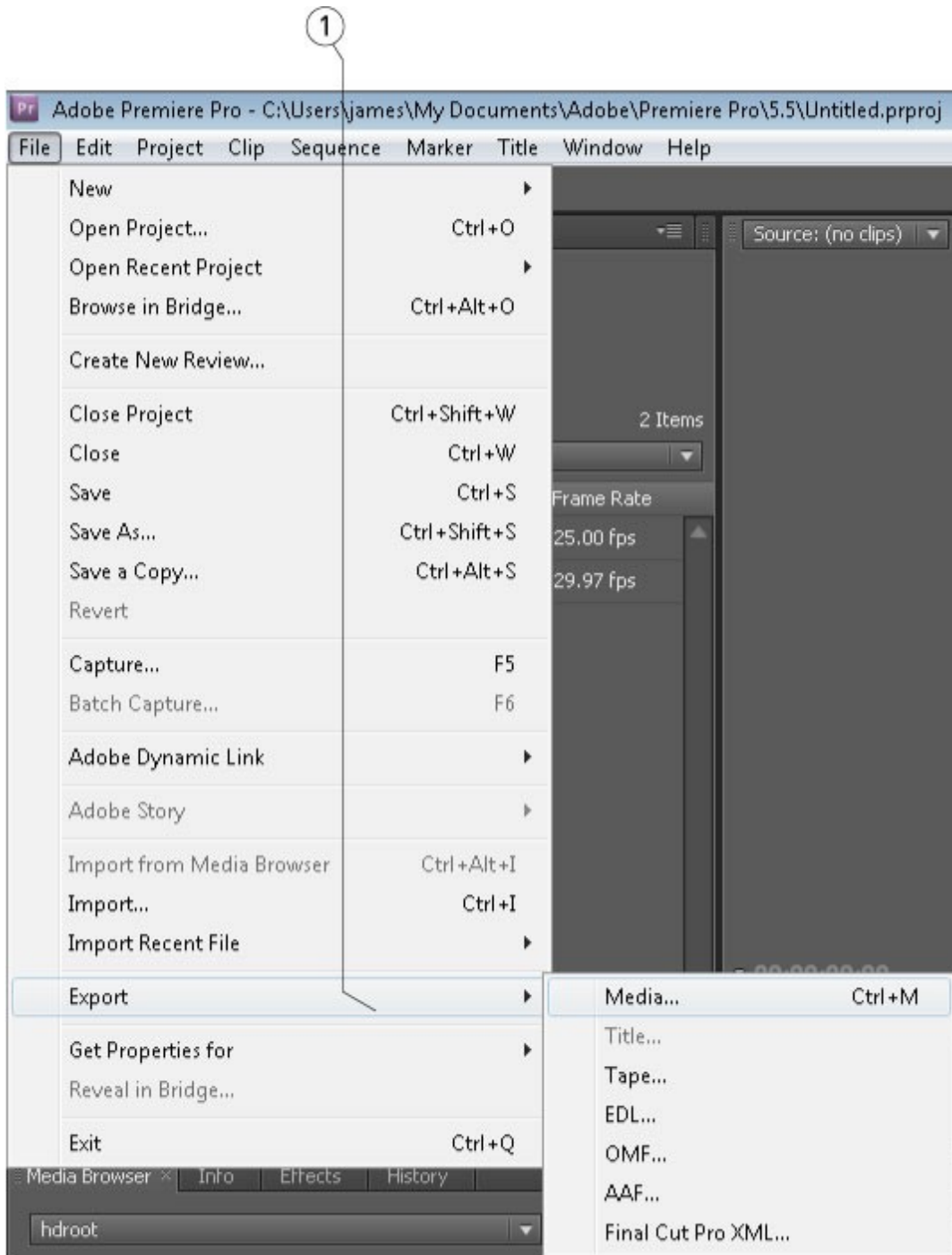
This loads the file into the dialog box.
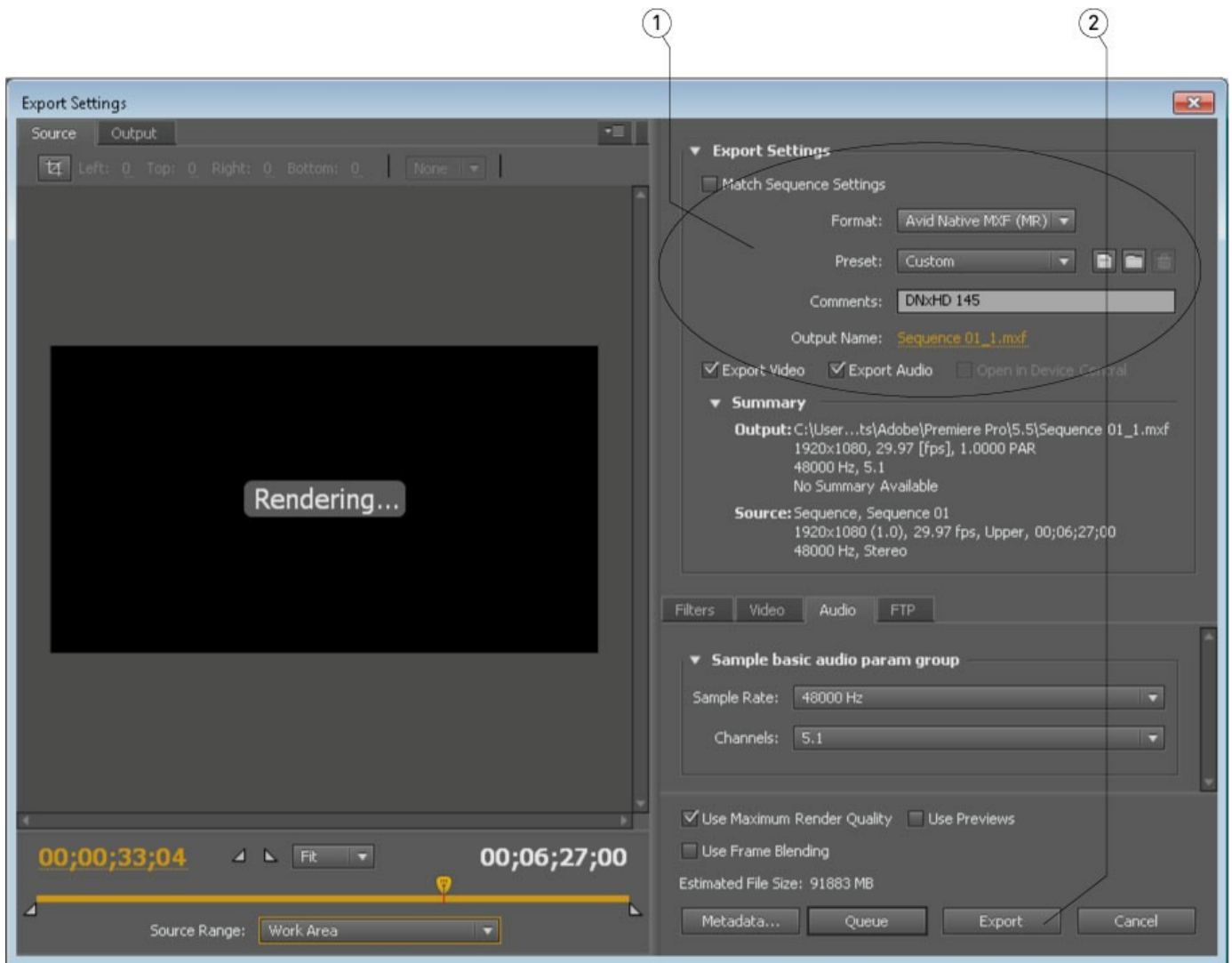
Select the **Format** drop down (1) to select the type. Set the settings and click the **Export** button (2).

NOTE: Outputting via Adobe only supports 0, 2 or 6 channels of audio. Files with more than 6 channels of audio will be limited to one of those numbers.

...with Apple Final Cut Pro (Suite 3/7)

To bring a file into FCP for editing, go to the **File** menu and select **Import** (1). Alternately, right click on the bin and select **Import->Files** (2). Browse to the file and select it.
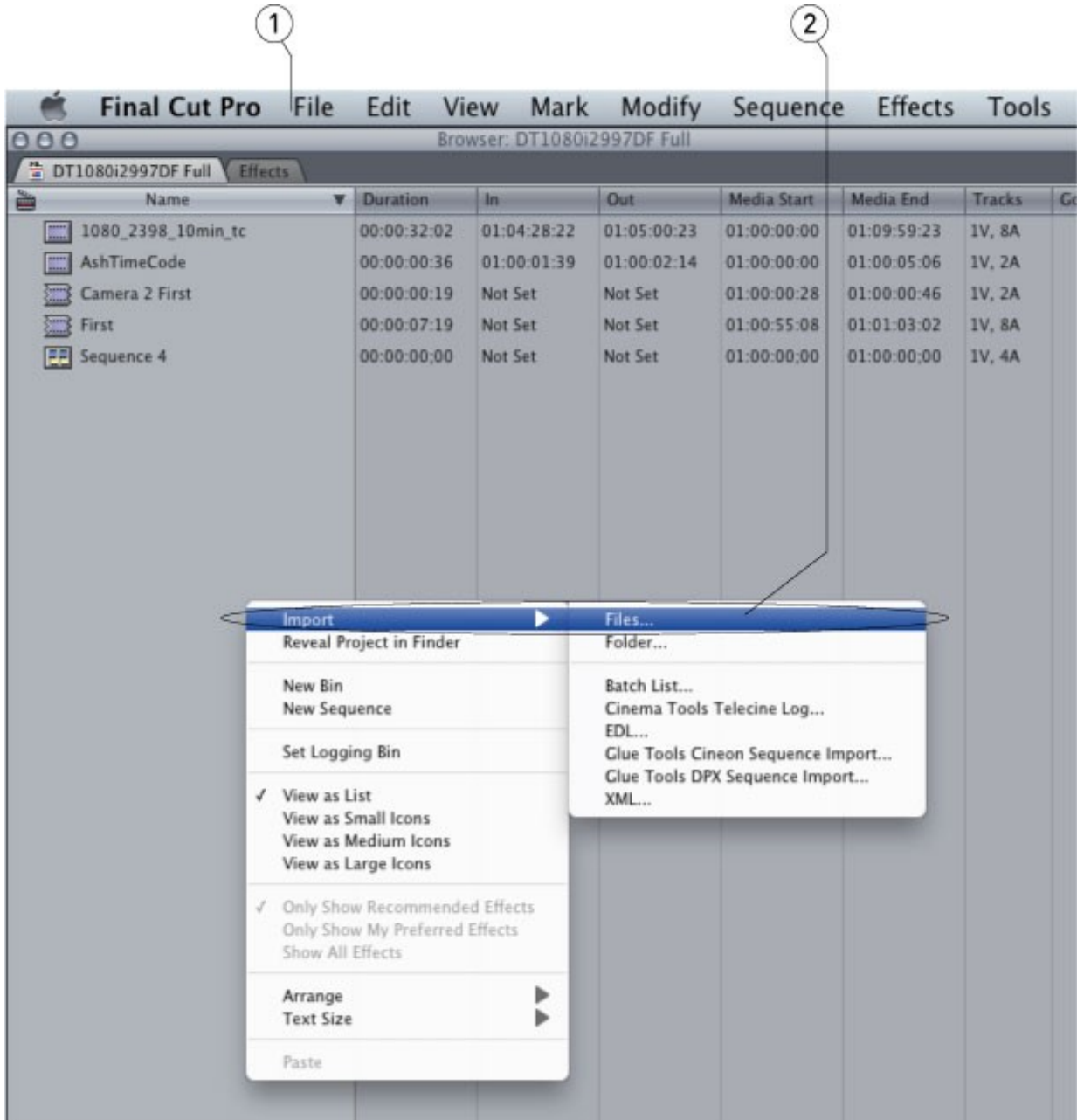
To export a file from FCP, click on the sequence you want to export. Select the **File** menu (1), then **Export->Using QuickTime Conversion**… (2).
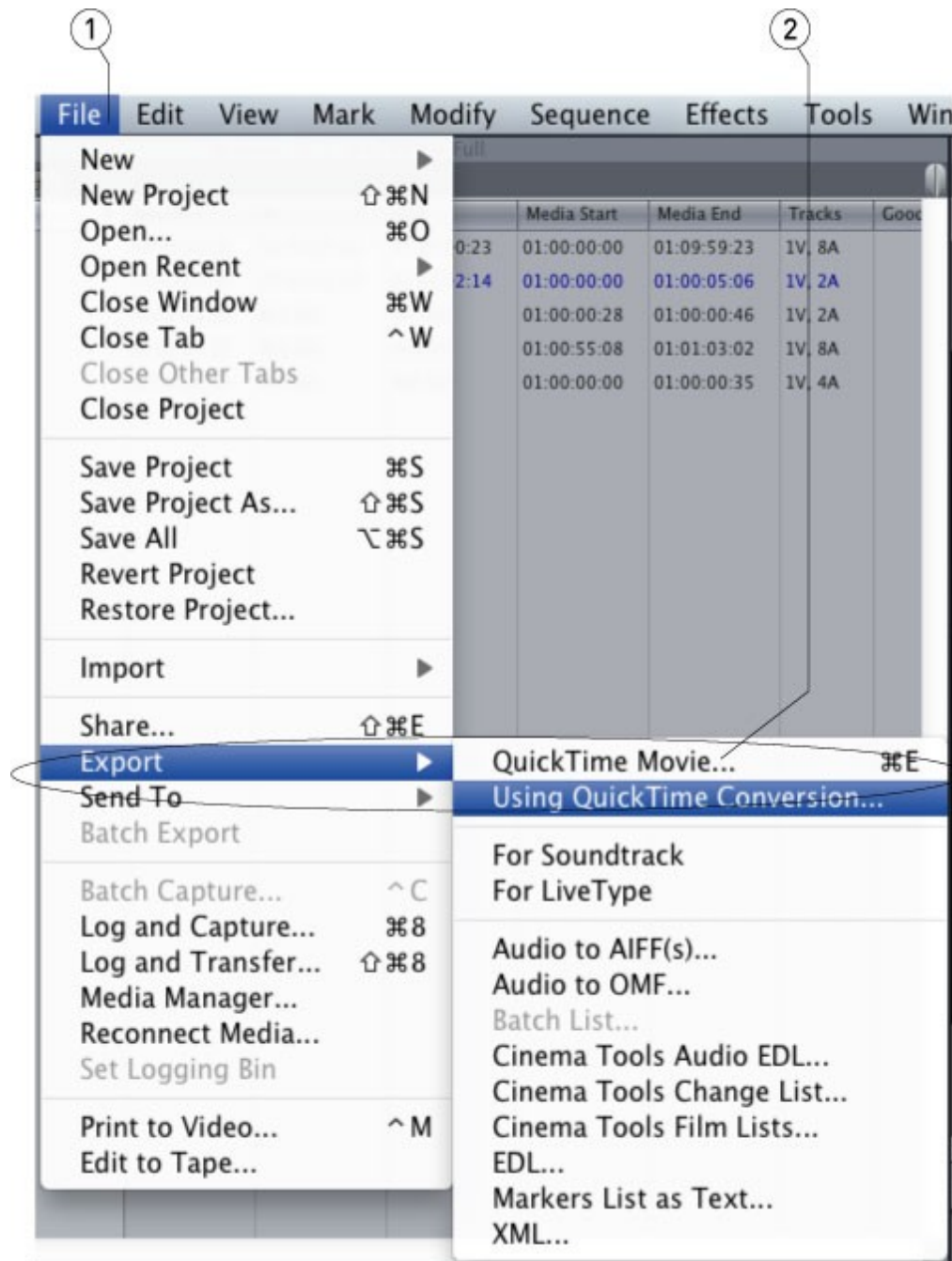
Open the **Format** list box and select the output type you want (1).
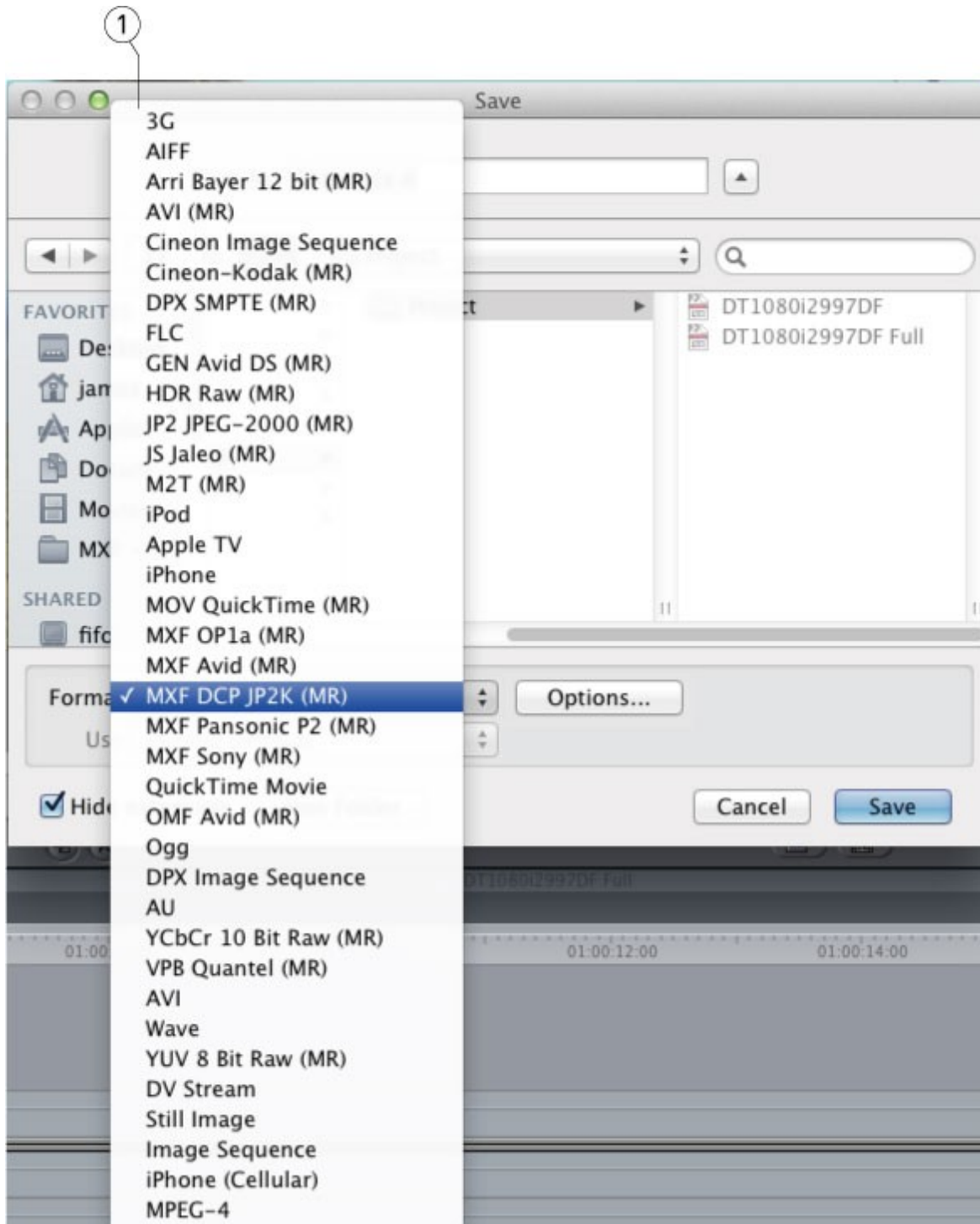
Click the **Options** button to set the options.
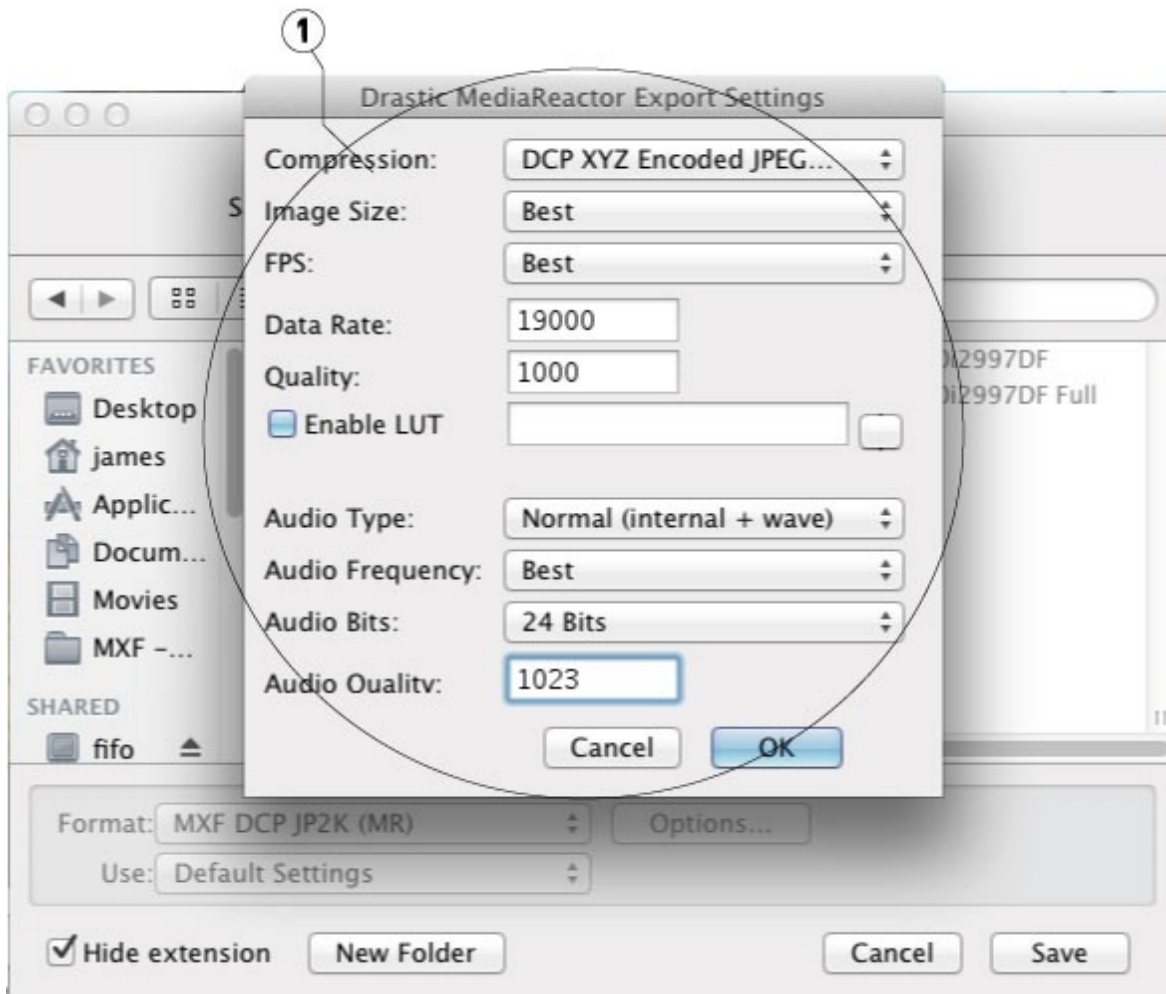
Set the name and add the correct extension (FCP will not do this for you). Click Save to export the sequence.

...with Assimilate Scratch/Lab

To bring a file into Assimilate Scratch/Lab, from the **TimeLine | Media** area, select **Load Shot**.
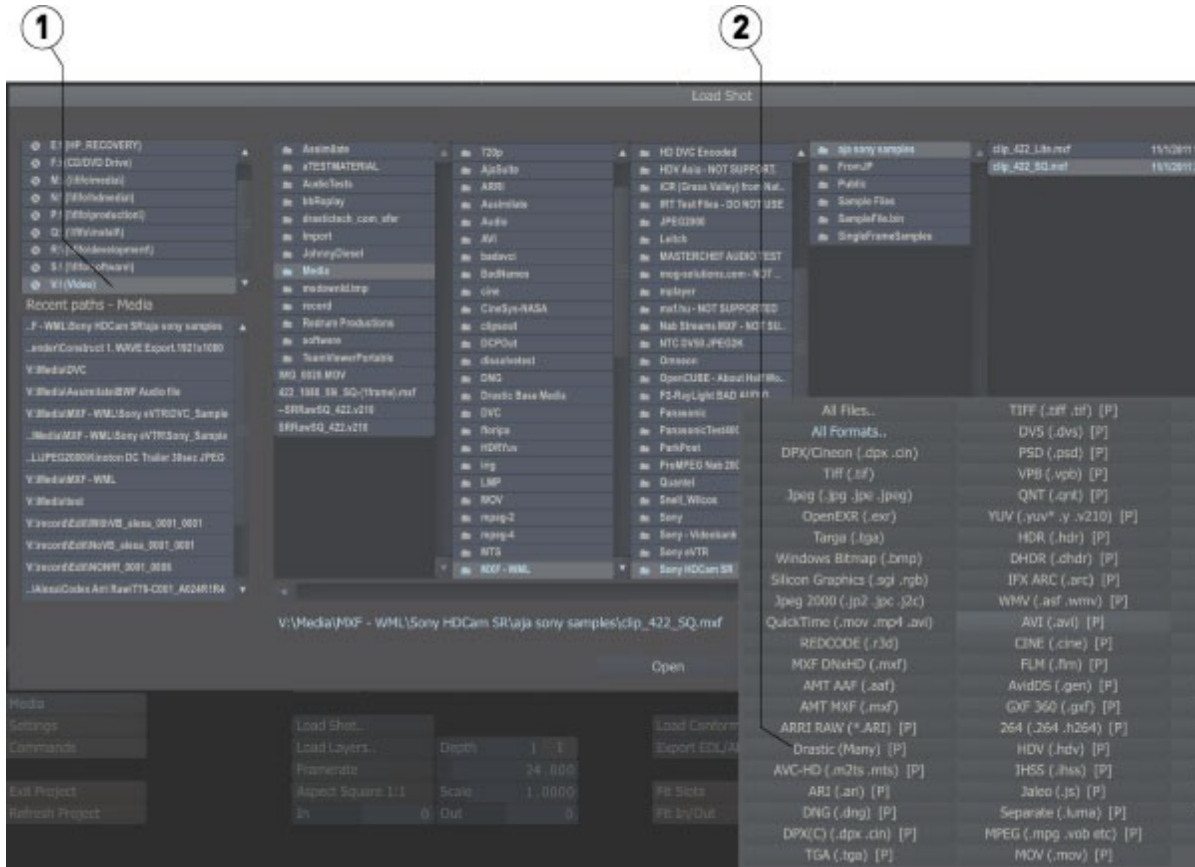
The file browser (1) will contain all the compatible types. For types that are supported in Scratch and MediaReactor Workstation (e.g. Arri, MXF), select the **Drastic Many** file filter (2) to force the MediaReactor Workstation plugin to be used.

To export a file from Assimilate Scratch/Lab, from the **Output** area, press the **Add Single Output** button (1) to add a single node.

Press the **Play** button (1) – it's the circle with an arrow in it - to go to the output page.

Click the **Insert...** button (1) to open the Export Plugins window.
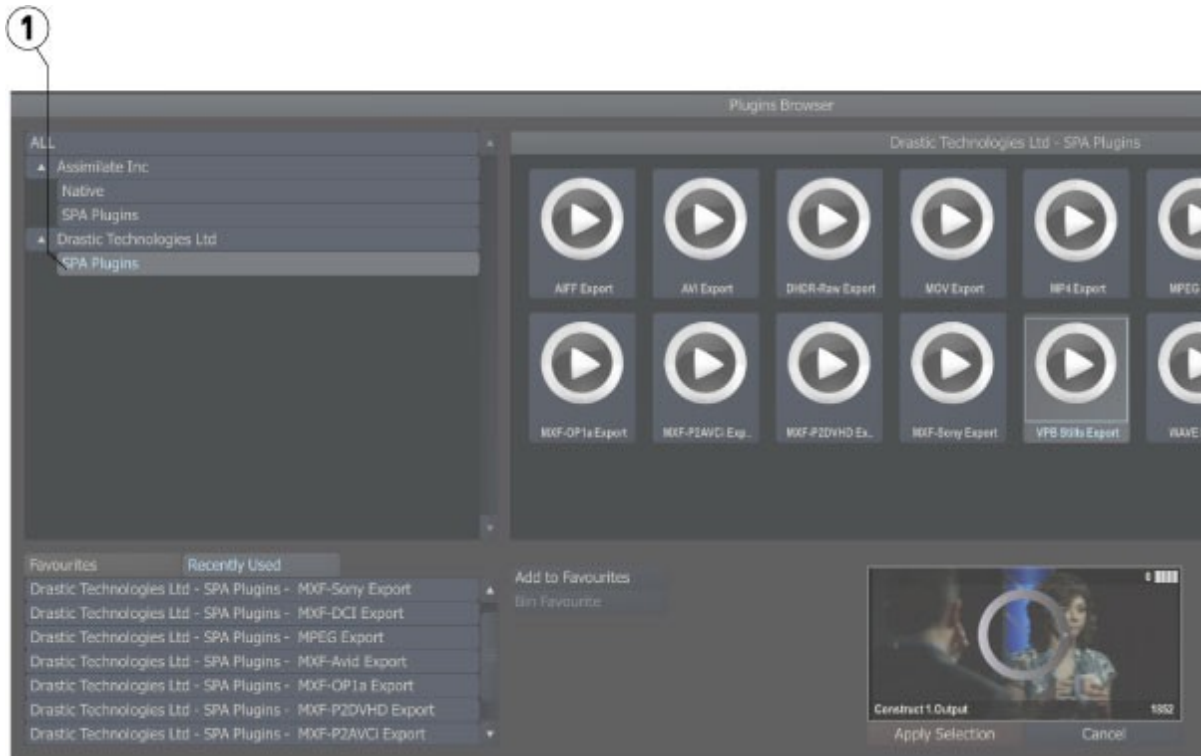
Use the **Export Plugins** window to select the output type from the **Drastic SPA Plugins** (1).
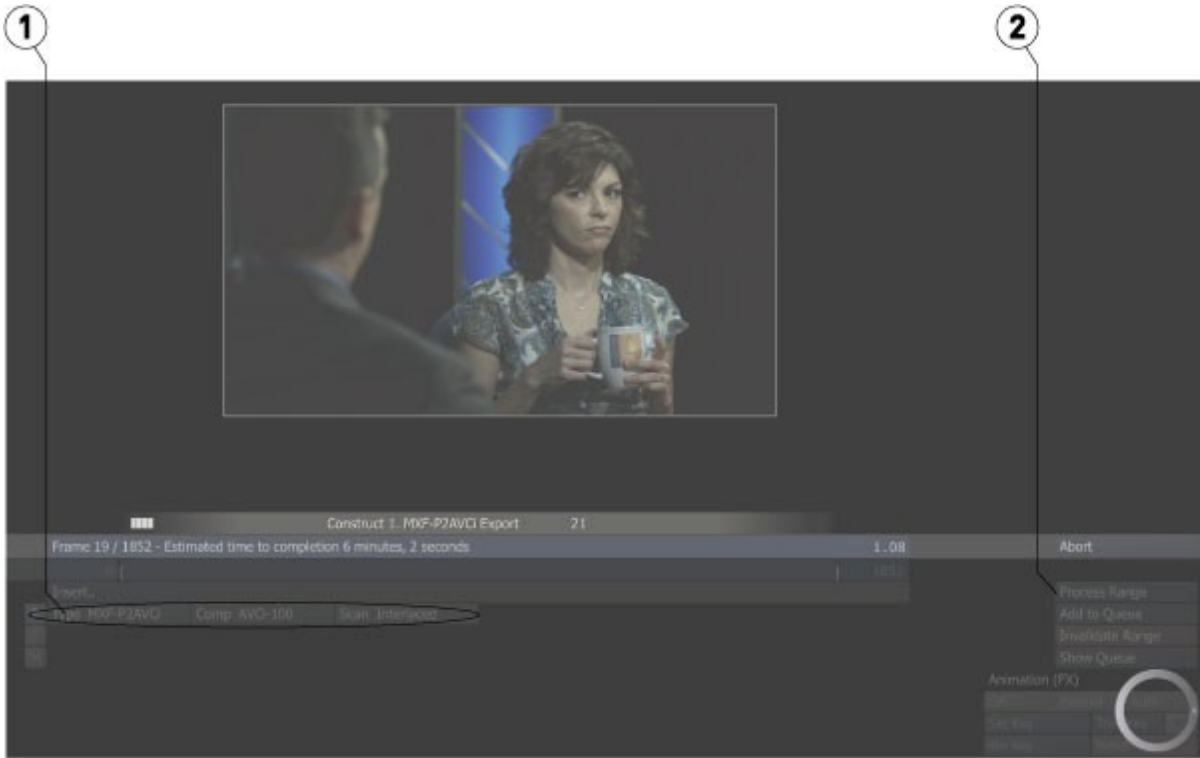
Adjust the parameters (1), and click the **Process Range** button (2) to export.

...with Autodesk Flame

Autodesk Flame users will already be familiar with how to open and work with a file in Flame. With MediaReactor properly installed and licensed, support for several new formats will be added. It is possible to confirm that the MediaReactor plugin is being used to open these files.

Open Flame, and press the MediaHub tab (1) at the bottom to view accessible media.
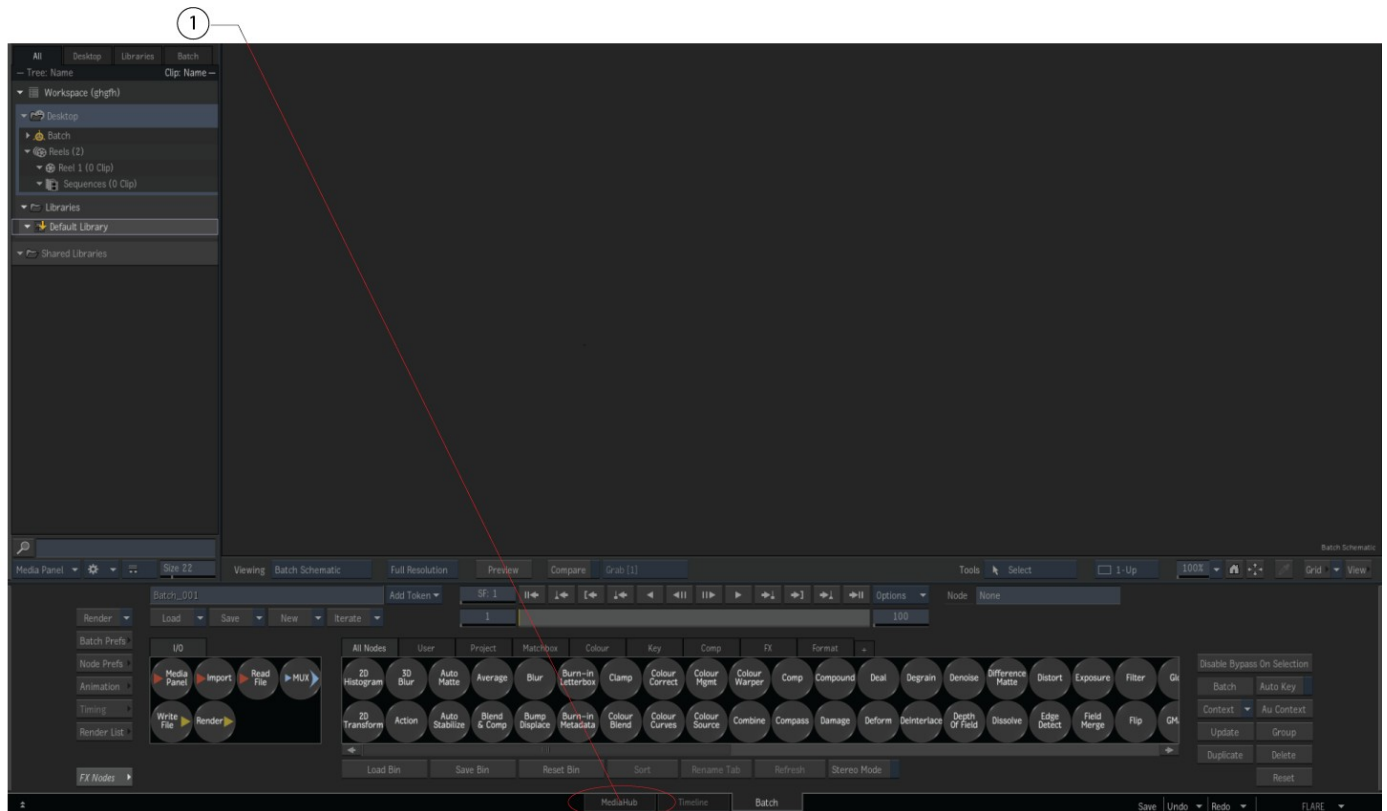
Clicking on the Media Hub displays the contents of the media folder (2).
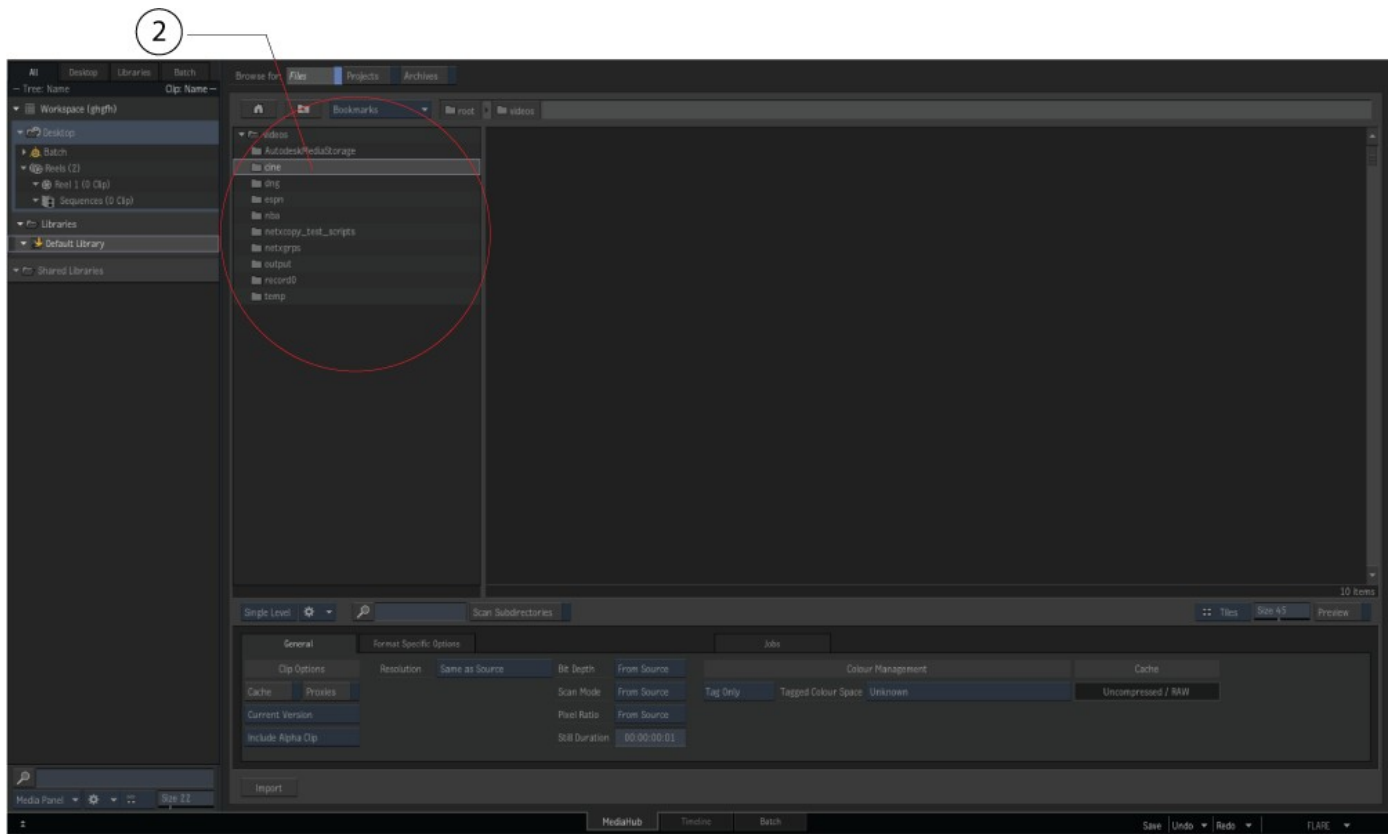
Double clicking on a folder will display the contents of the folder. In this case the Cine folder (3) has been selected, and the clips in it are displayed.
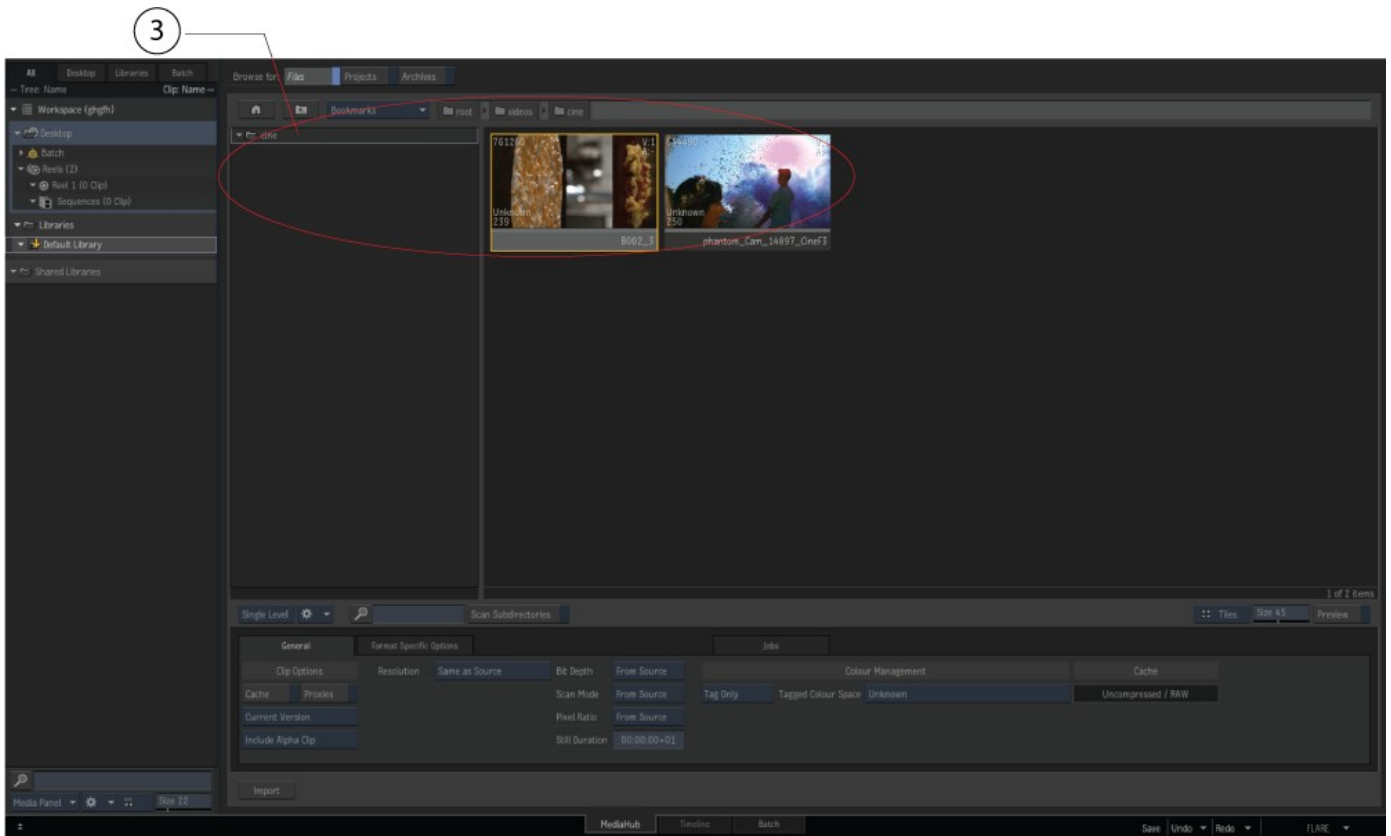
Once the clips are shown as above, a further check can be performed.

Clicking on the clip picon (4) loads the clip and its metadata into the panel on the right (5).
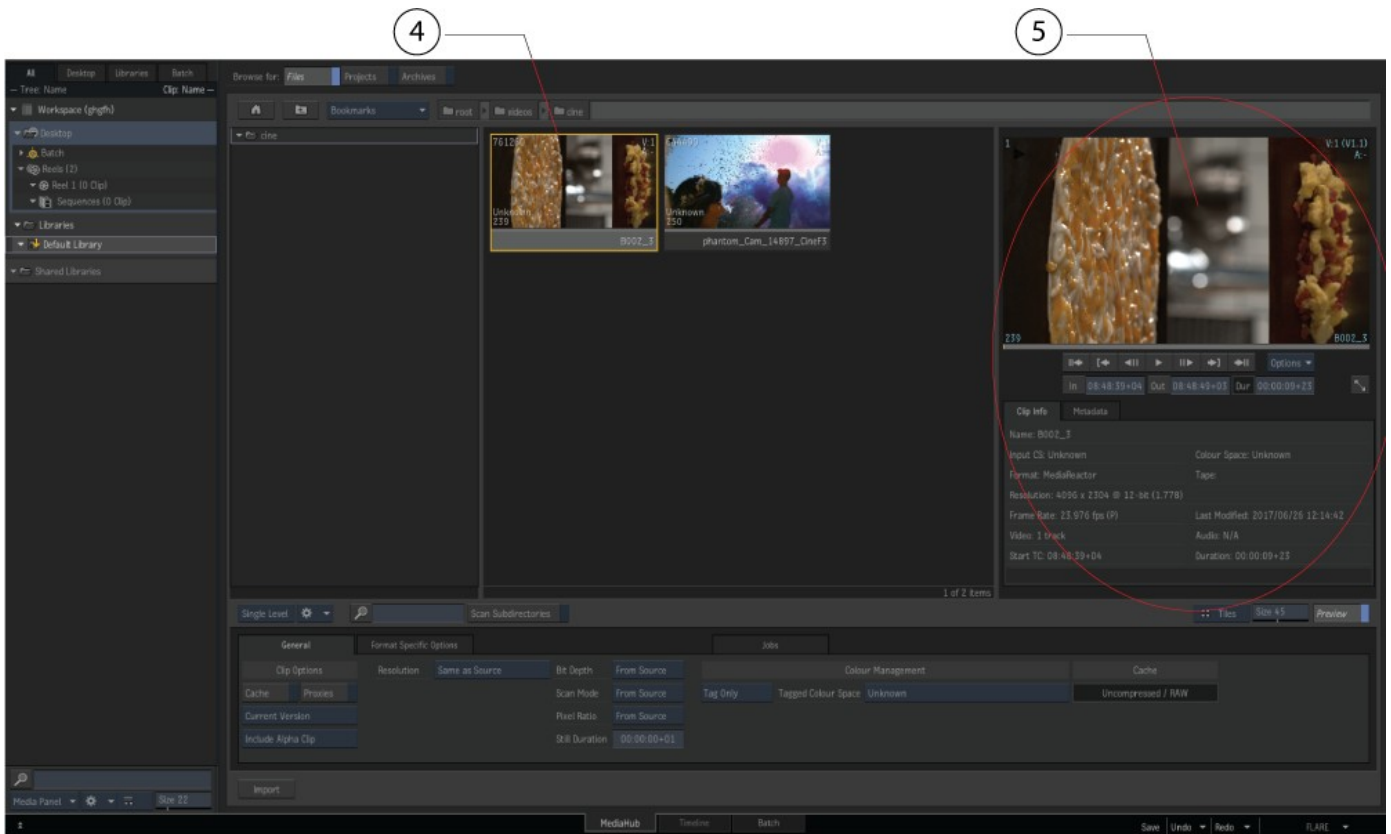
If you open a clip that requires the MediaReactor plugin, and the plugin has been properly installed and licensed, you will see in the Clip Info tab, that the Format specifies the MediaReactor plugin is being used.

Clicking on the Metadata tab (6) should also display the MediaReactor version in the Video Track Information, if the clip was loaded using the MediaReactor plugin.
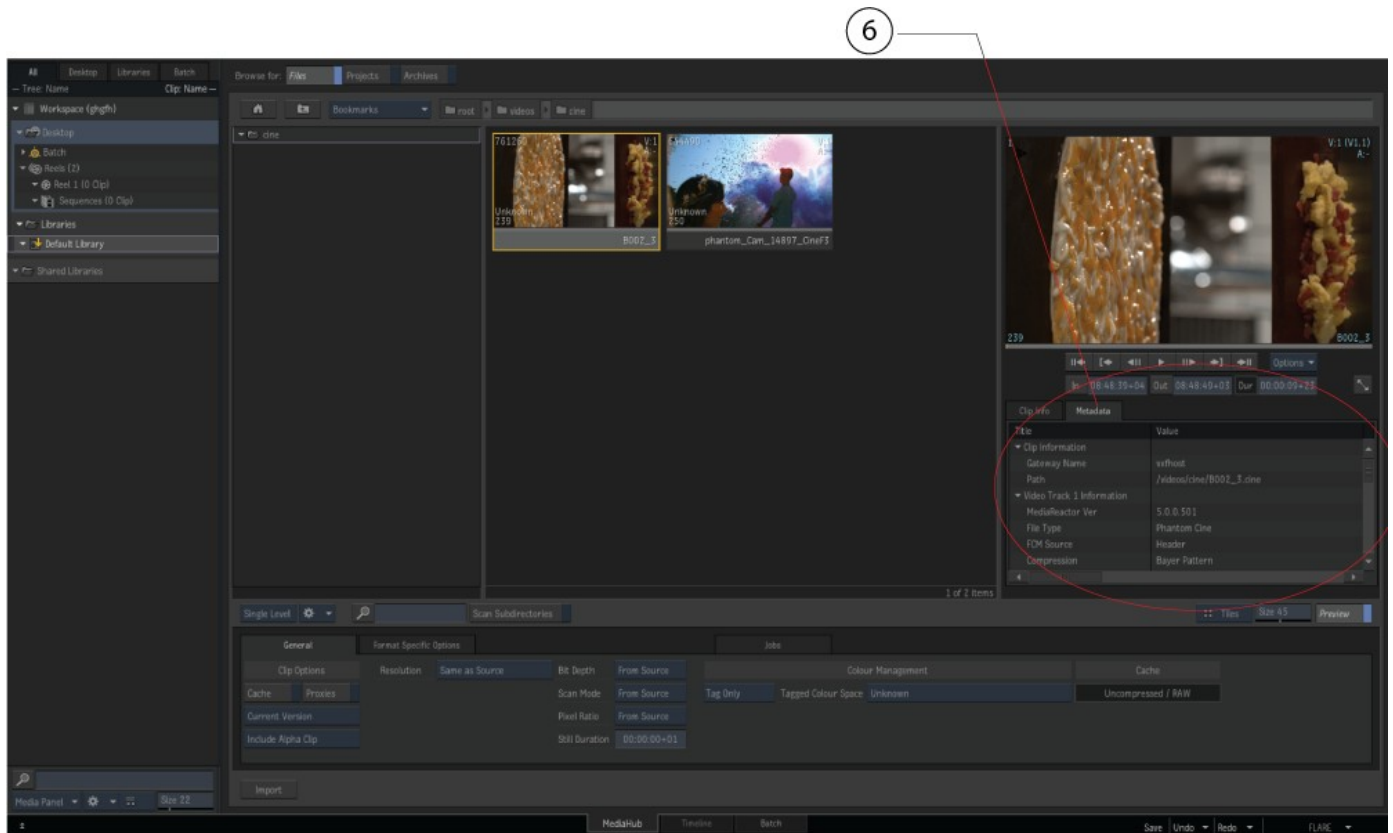
This information also shows up with the clip Batch function. Click on the Batch tab (7).
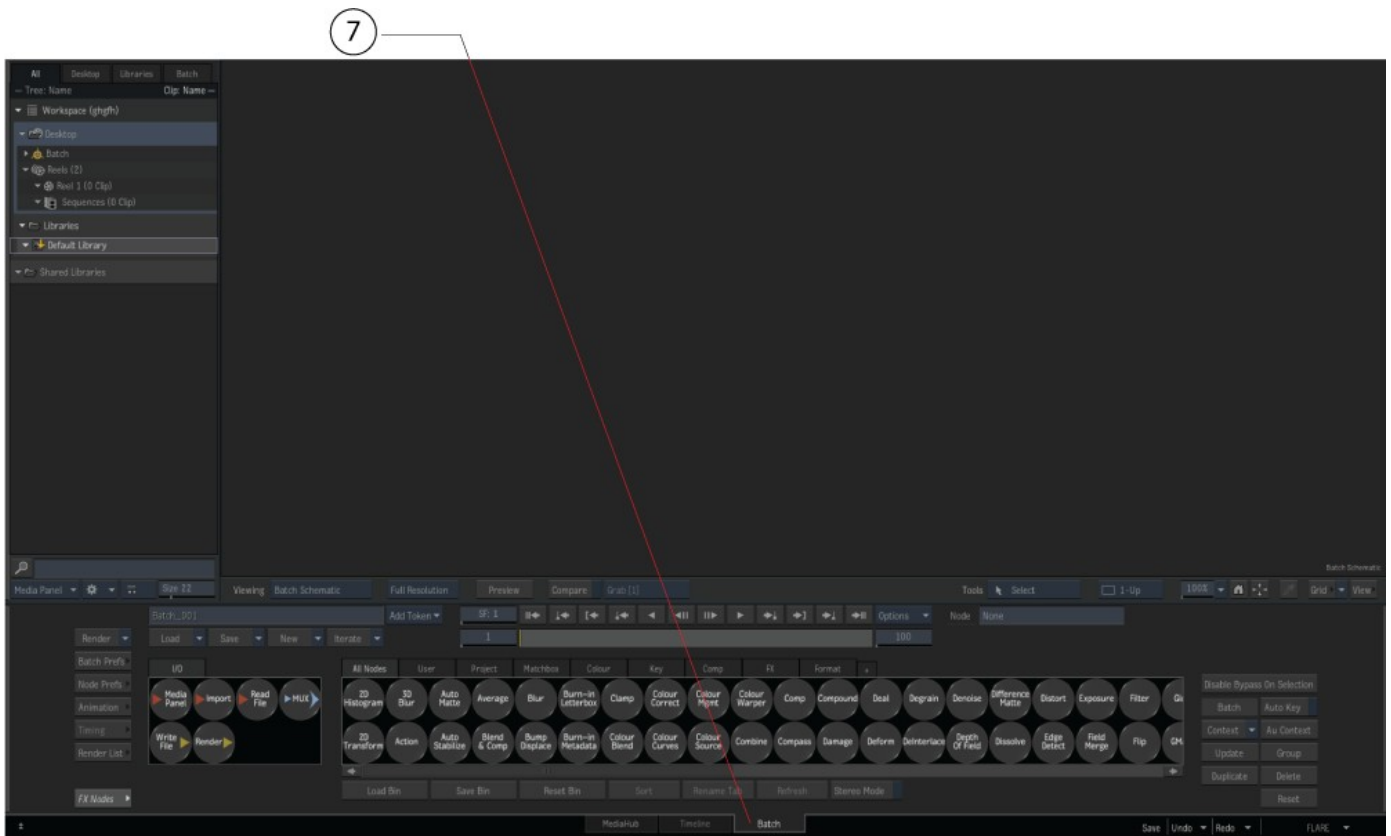
Grab the Read File button (8), and 'drag' it onto the media area (9), then release the mouse ('drop' it).
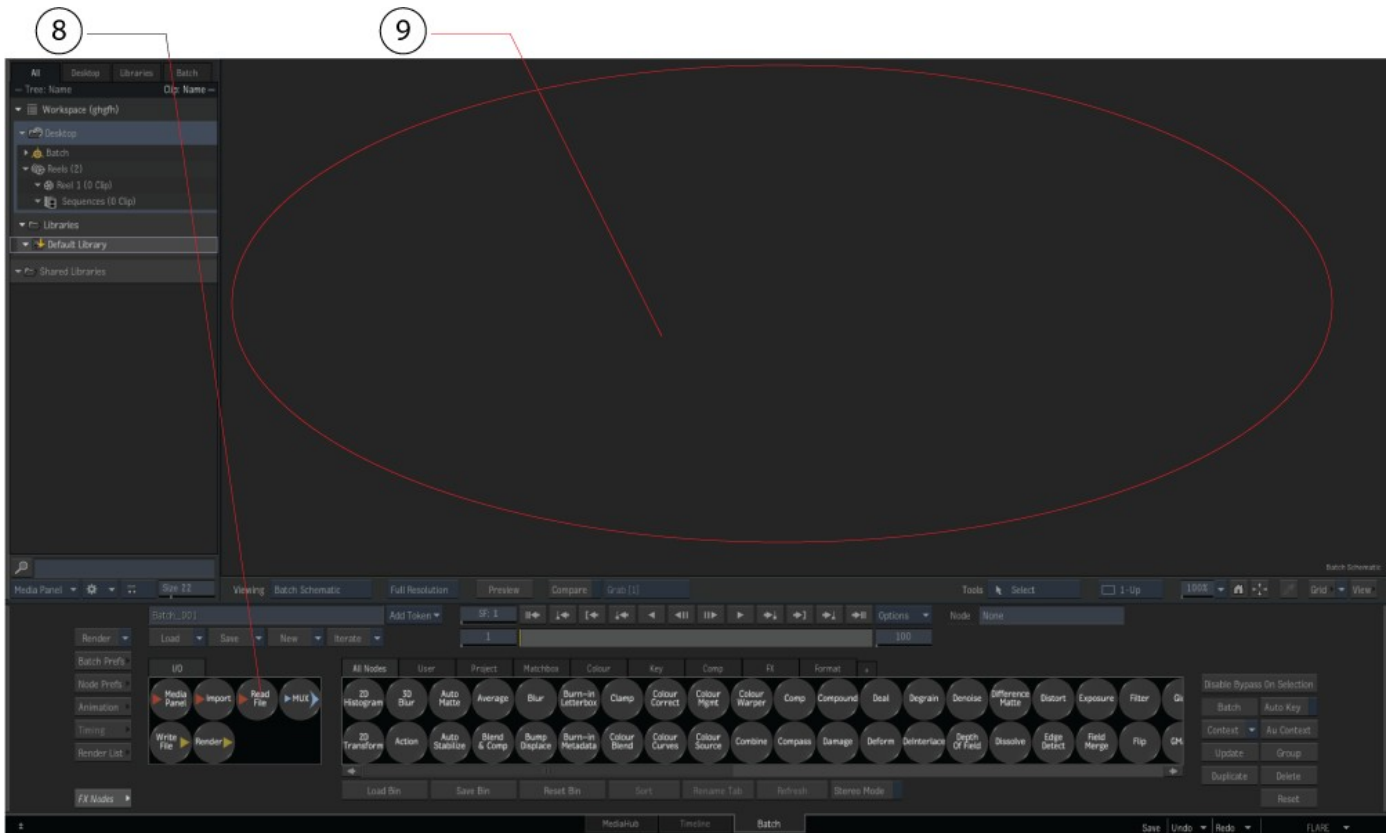
This opens the Read Files window.
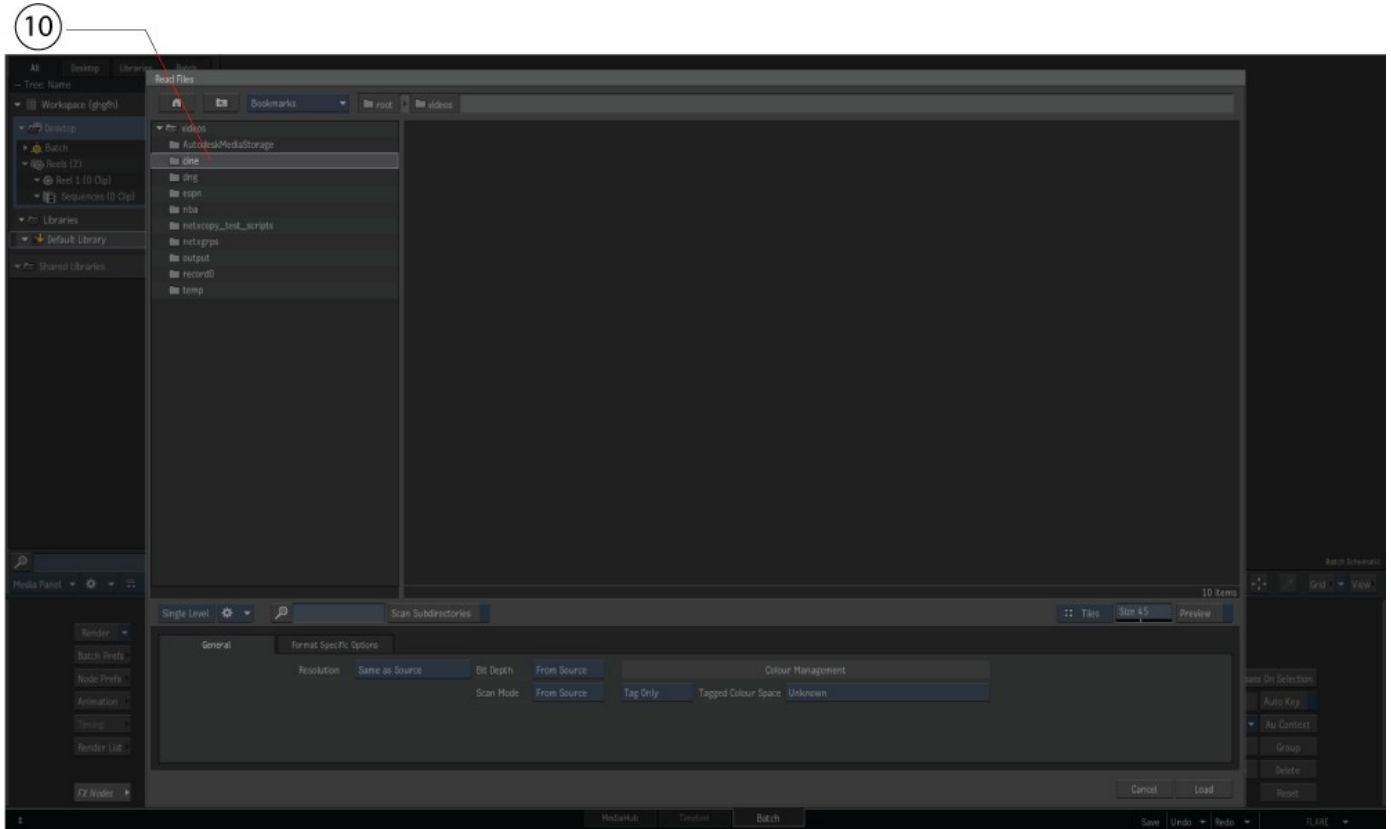
Navigate to the folder (10) and load the media as in the last example.

Click on the Viewing mode button (11).
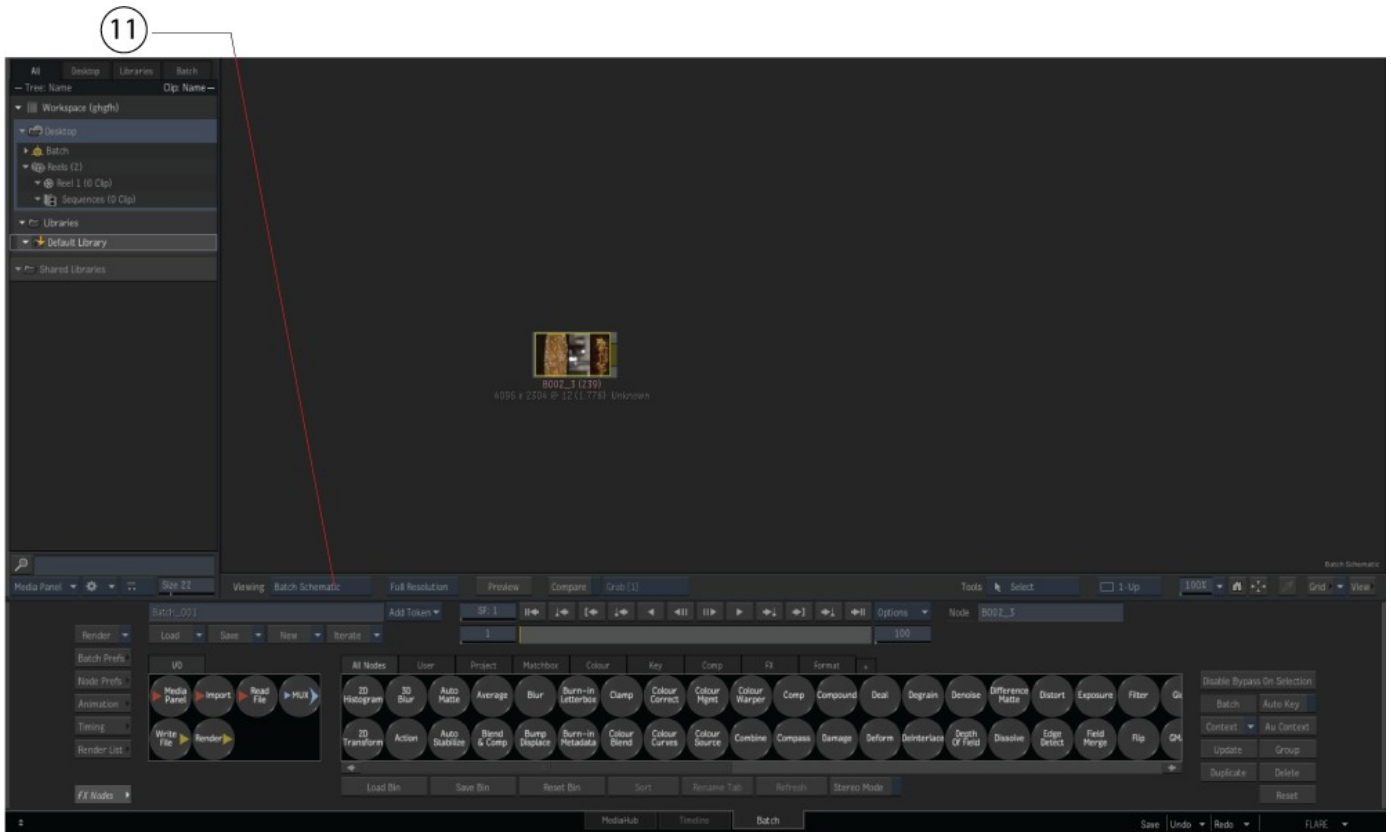
This displays the available viewing modes.

Select **Current Clip Node** (12).
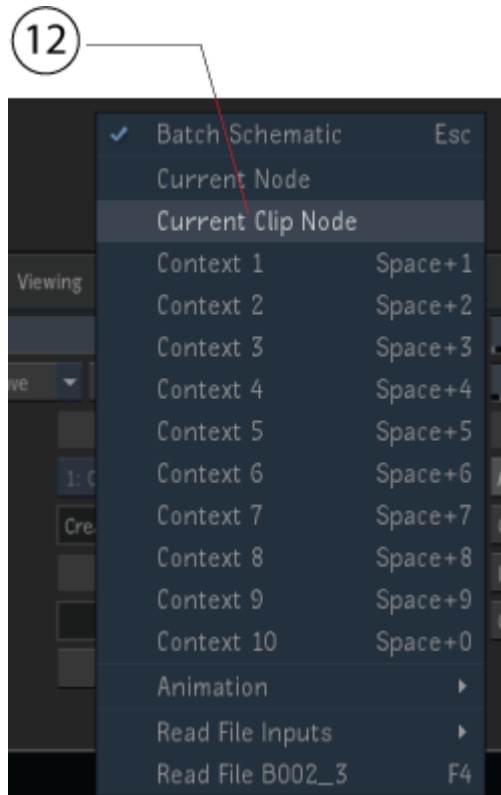
In this mode, the clip is loaded (13) and its clip info and metadata are displayed. Particularly, in the Format section of the Clip Info, MediaReactor (14) is specified.
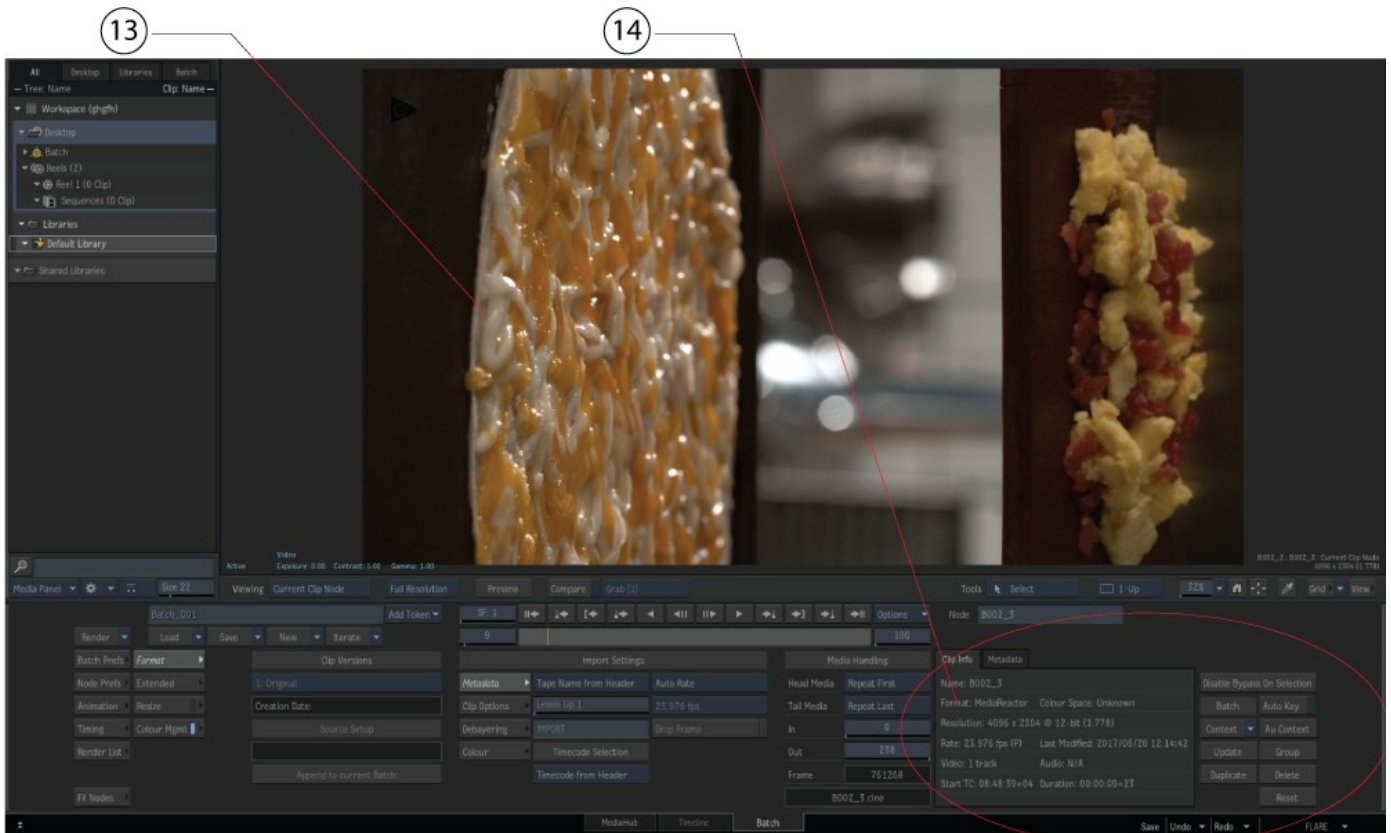
...with Avid Media Composer 6 (Symphony, NewsCutter)

To bring a file into the editor:
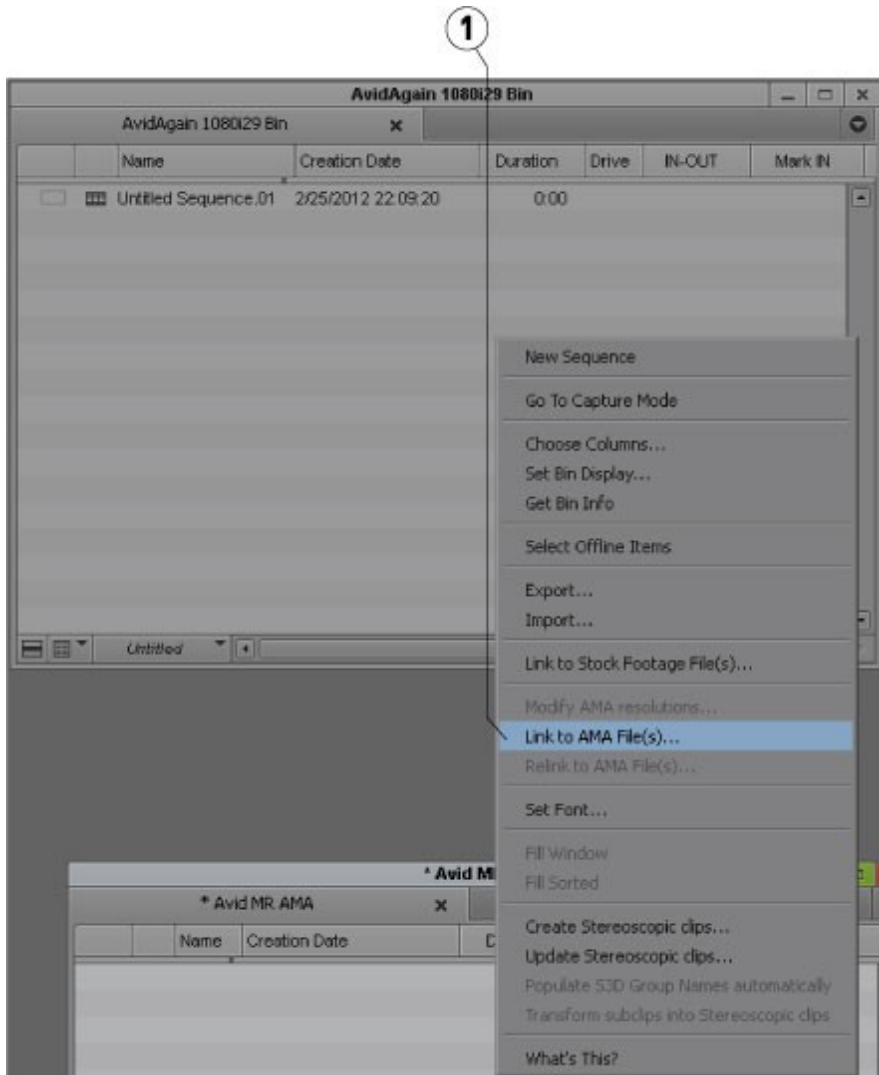
Right click on the bin and select the **Link to AMA File(s)...** menu (1) and select the file.
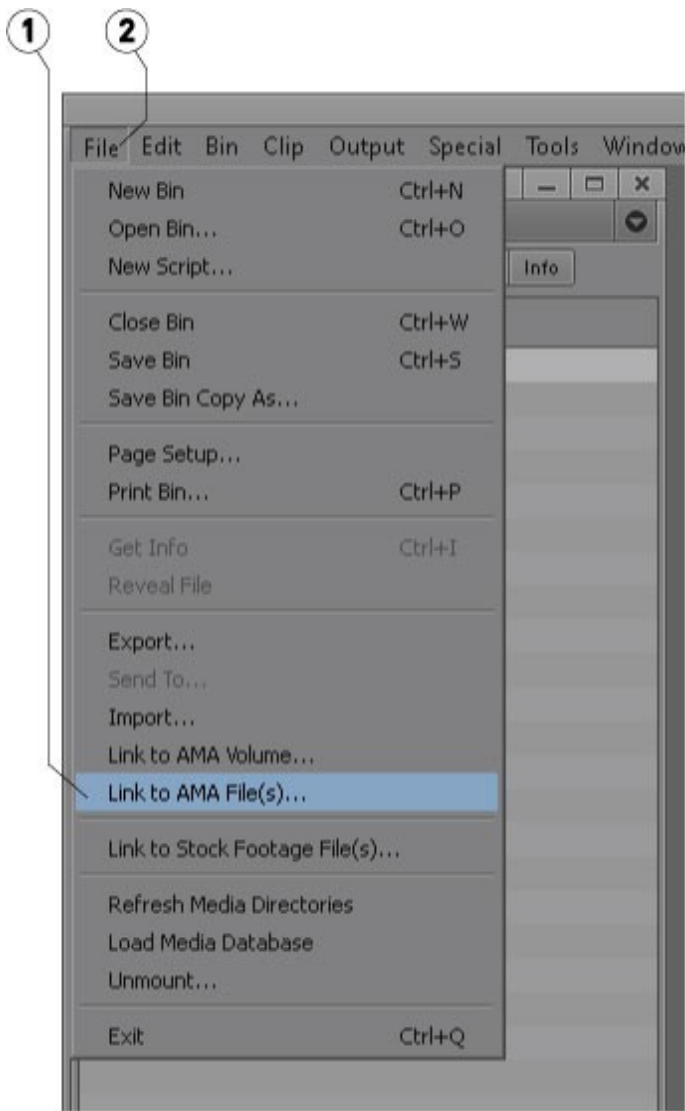
The **Link to AMA File(s)** menu (1) is also available under the **File** menu (2).

If your source files are on a network drive, you need enable access to the network drives in Media Composer. Open a **Console** window (menu **Tools | Console**) and type 'alldrives' without the quotes and hit **Return**. This only needs to be done once.

...with Apple Compressor

To convert from a MediaReactor Workstation file: Create an output profile.
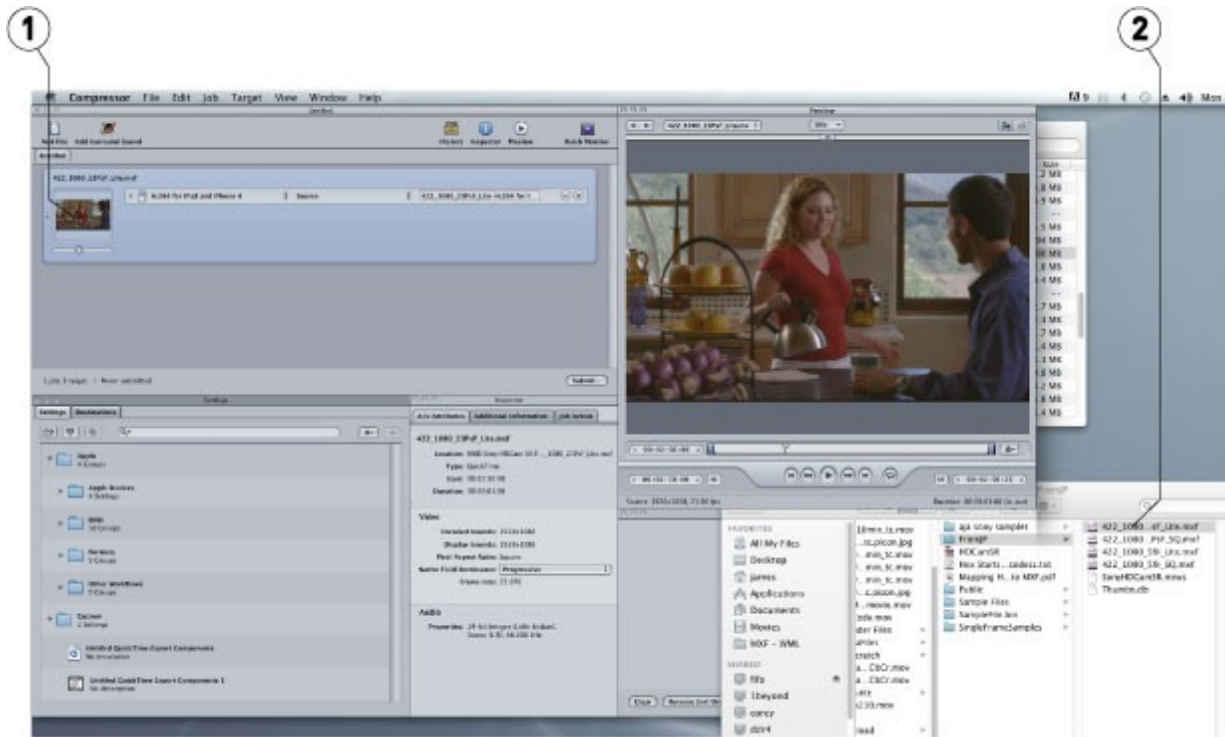
Drag the source file (2) on top of the output profile and drop it (1).

...with Apple Motion

To use a MediaReactor Workstation file in Motion, use the built in **File Browser** (1) to find and select the file.
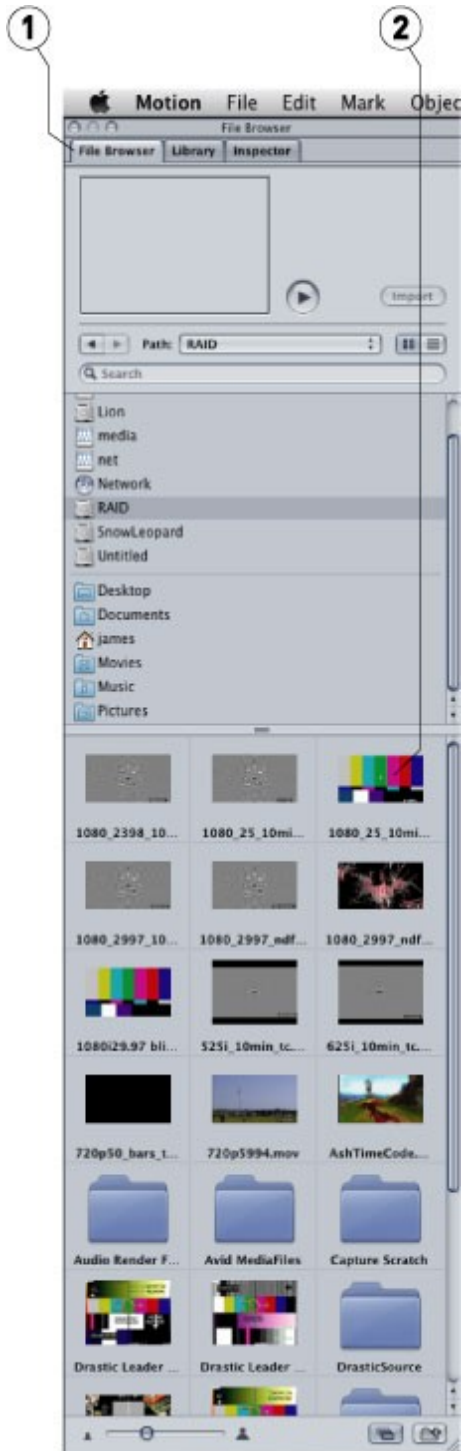
Supported files will appear as a picture of one of the frames (2) in the browser.

...with Adobe Media Encoder

To add a MediaReactor Workstation file:
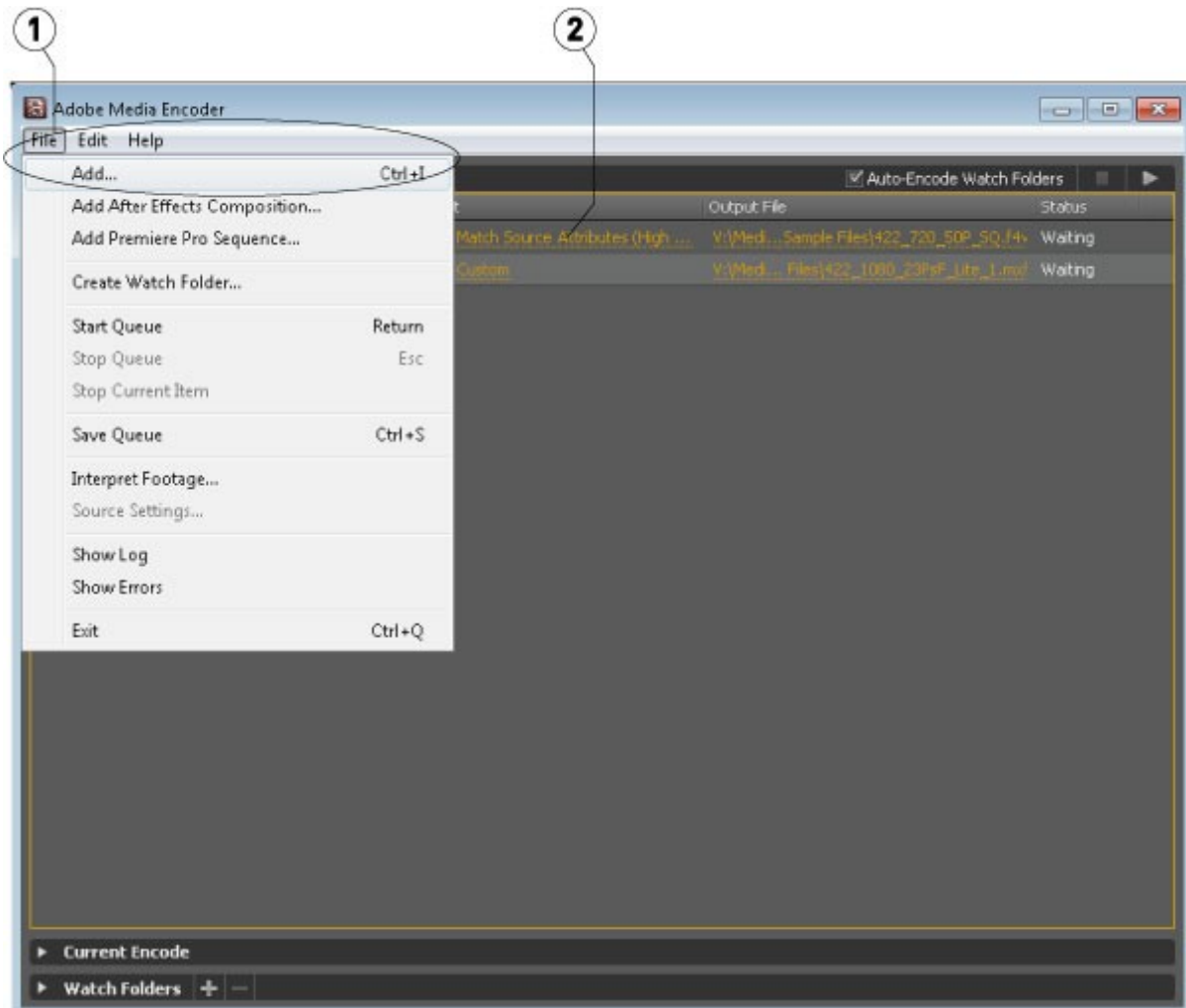
Use the menu **File | Add** (1), or drag and drop the file on the application. Click on the **Format** (2) or **Preset** column link. This opens the **Export** window.

The **Export** window opens with the selected file loaded (1). The **Export** window provides various controls (2) to allow the user to set the type of file that will be created during the export process.
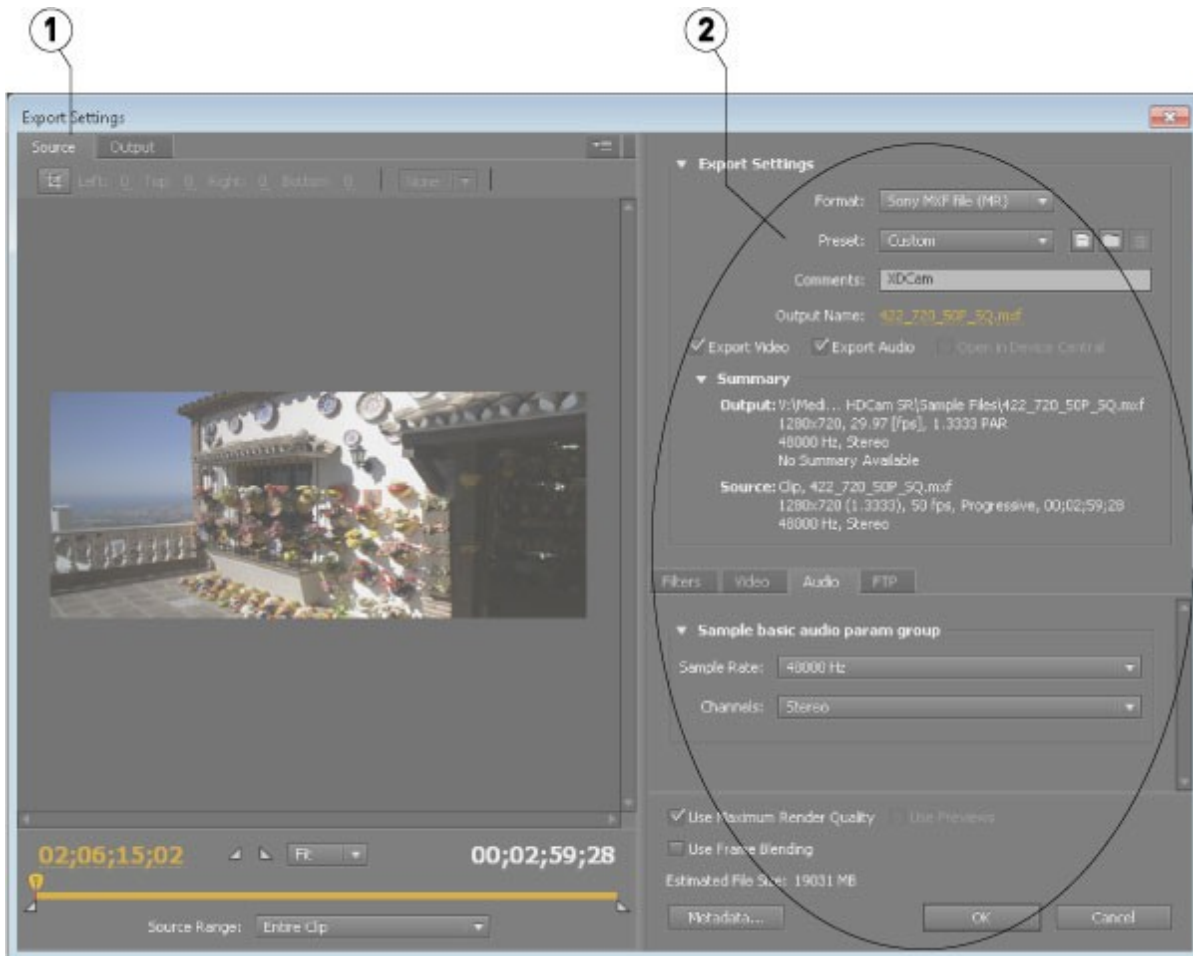
Once all the parameters have been sent the user can press the **OK** button to close the **Export Settings** window. The file will have been updated in the queue, with the parameters that will be used to create the new file displayed.

NOTE: Outputting via Adobe only supports 0, 2 or 6 channels of audio. Files with more than 6 channels of audio will be limited to one of those numbers.

...with QuickTime Player

To open a file in QuickTime Player (free or full version), select the **File | Open File** menu. In Windows it is necessary to set the file filter to all (*.*) to see non mov/avi files.

To export a file from QuickTime Player (requires full version), select the menu **File | Export...** In the export dialog, select the output type and options. Type in the file name and file extension.

When outputting a file in QuickTime, a 0k file is created in the directory which we do not use.

...with Windows Media Player

To open a file in Windows Media Player, select the **File | Open** menu, or drag and drop the file onto the Player

Live Event Editing in FCP7, Premiere and MediaComposer

MediaReactor Workstation also provides advanced edit-while-recording of live events directly in your favorite editing software, when coupled with a live event compatible recorder.

To take advantage of this huge time saver, set up your compatible recording device (Drastic DDR, MediaNXS or 3rd party systems like XOS Hurricane Replay), begin recording and load the various camera files directly into Media Composer, Premiere or Final Cut Pro 7. All material recorded at that point is immediately available for editing, and as new material is recorded it automatically becomes available on the editor. Simply edit as normal, building packages, melds, internet content or new packages during the show, rather than having to wait and work long after everyone else has gone home.

IndieCam MOV File Reader Configuration

The IndieCam cameras are small, light, HD and 2K recording RAW bayer pattern cameras. They provide the raw data via HD-SDI by embedding it into a SMPTE standard signal that can be recorded by most uncompressed recorders.

The RAW data cannot be automatically detected because the type marked by the recorder is SMPTE YCbCr 10 (v210 in general). To use these files in Assimilate Scratch and MediaReactor, the system must be set up to re-label these files.

Overriding IndieCam MOV Files' FourCC

The default fourcc (a four character code that describes the video format) in an MOV file will be 'v210'. This is the fourcc that describes the YCbCr 10 bit video that was captured from the HD-SDI output of the camera by the capture device. To have Assimilate Scratch or Lab properly interpret the raw data, it is necessary to override the fourcc with an IndieCam-specific fourcc.

To set this up, run DDRConfig from the MRWS (MediaReactor Workstation) installation:
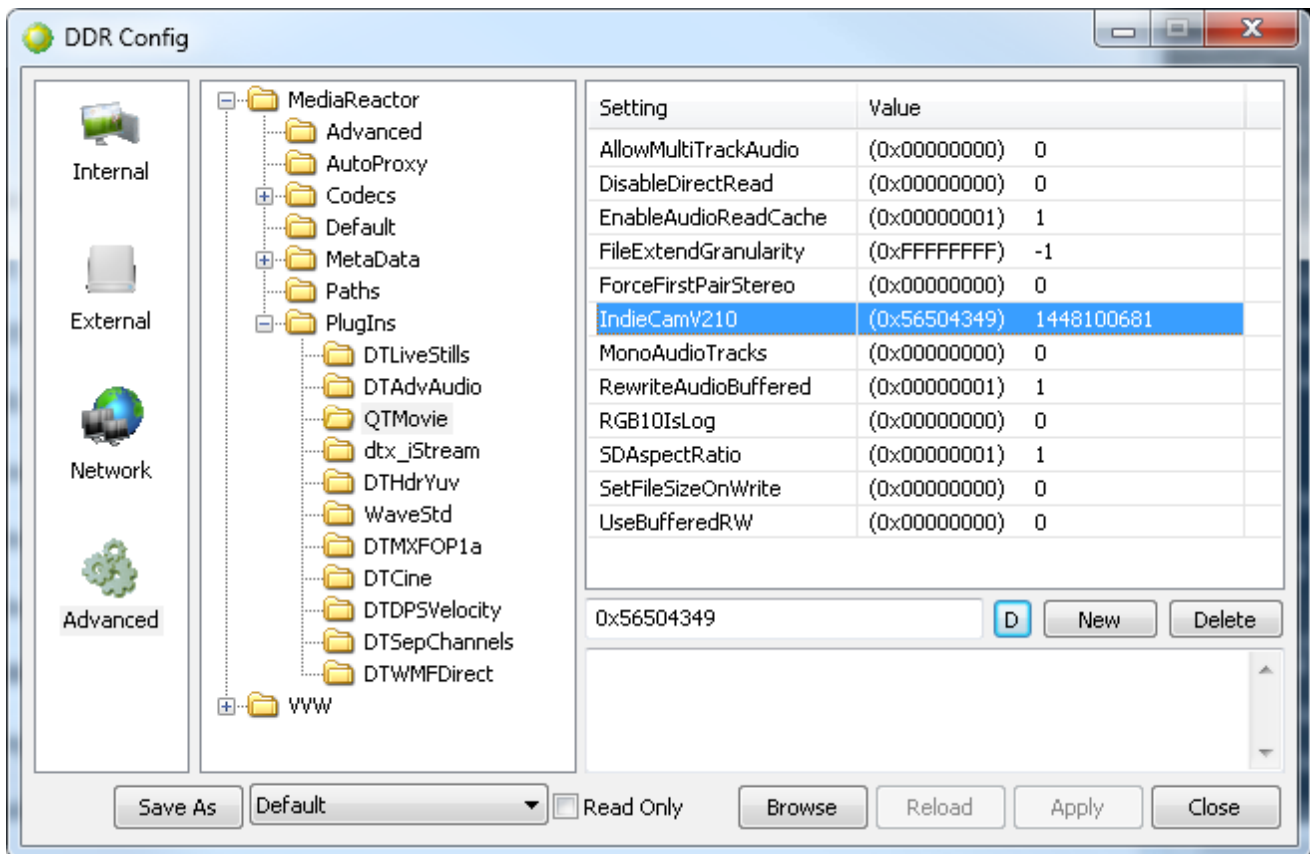
Under the MediaReactor/PlugIns/QTMovie branch there will be an IndieCamV210 entry. If this is changed to one of the supported IndieCam types, then all v210 files will be interpreted as that type of IndieCam file. The possible types are

- INDIECAM_POV ICPV - IndieCam POV GBRG
  - 0x56504349
- INDIECAM_RGGB2K IC12 - IndieCam RGGB 12bit-RAW
  - 0x32314349
- INDIECAM_2KRAW IC2K - IndieCam GBRG 2k-RAW
  - 0x4B324349
- INDIECAM_2IN1RAW IC2X - IndieCam GBRG 2in1-RAW
  - 0x58324349

Setting the IndieCamV210 entry to the hex values above (when the edit box is in hex) will cause all v210 files to be interpreted as that type of IndieCam raw file. This means, (1) you cannot use standard v210 files and, (2) you cannot mix 2K and POV files on the same time-line, as one of them will be interpreted incorrectly.

Multichannel Audio File Naming

Drastic products can support up to 16 mono (or 8 stereo pairs) of audio. There are file formats that are supported (like wave file extensible) that can contain all 16 channels in a single file, but these are not necessarily the most useful file types for production. In general, Drastic products prefer to store audio in a series of uncompressed, stereo files on disk in WAVE or AIFF format.

Here are the primary types for these uncompressed stereo files:

AIFF 16 Bit Big Endian Stereo
AIFF 20 Bit Big Endian Stereo
AIFF 24 Bit Big Endian Stereo
AIFF 32 Bit Big Endian Stereo
WAVE 16 Bit Stereo
WAVE 20 Bit Stereo
WAVE 24 Bit Stereo
WAVE 32 Bit Stereo
WAVE 16 Bin Mono (for WMV/Matrox compatibilty)

Regardless of the file type, the audio files will always have the same naming conventions. For the examples below, the '.video' is the video source file. The '.wav' could equally be '.aiff' files.

BaseName.video (BaseName.avi, BaseName000001.dpx, etc)
BaseName.wav (or BaseName.wav or BaseName.A12.wav)
BaseName.A34.wav
BaseName.A56.wav
BaseName.A78.wav
BaseName.A9A.wav (channels 9 and 10)
BaseName.ABC.wav (channels 11 and 12)
BaseName.ADE.wav (channels 13 and 14)
BaseName.AFG.wav (channels 15 and 16)

- If the video file is a series of stills, then the base name is everything before the numeric count. For example:
BaseName_000001.tga
BaseName_.wav
BaseName_.A34.wav

- The correct naming for a group of mono files is:
BaseName.video
BaseName.a1.wav
BaseName.a2.wav
BaseName.a3.wav
BaseName.a4.wav
BaseName.a5.wav
BaseName.a6.wav
BaseName.a7.wav
BaseName.a8.wav
BaseName.a9.wav
BaseName.aa.wav (channel 10)
BaseName.ab.wav (channel 11)
BaseName.ac.wav (channel 12)
BaseName.ad.wav (channel 13)
BaseName.ae.wav (channel 14)
BaseName.af.wav (channel 15)
BaseName.ag.wav (channel 16)

Stereo pairs may be missing, but mono files may not. Any missing stereo channels will be filled with silence

There are a couple of other supported naming conventions which are there purely to support third party software packages. These include:

BaseName.video
BaseName.aa.wav (one stereo pair only)

BaseName.video
BaseName_1.wav (dual mono files)
BaseName_2.wav (one stereo pair total)

BaseName.video
BaseName.1.wav (dual mono files)
BaseName.2.wav (one stereo pair total)

BaseName.video
BaseName_R.wav (dual mono files)
BaseName_L.wav (one stereo pair total)

BaseName.video
BaseName.r.wav (dual mono files)
BaseName.l.wav (one stereo pair total)

BaseName.video
BaseName.wav (these are all mono wave files)
BaseName_A2.wav
BaseName_A3.wav
BaseName_A4.wav
BaseName_A5.wav
BaseName_A6.wav
BaseName_A7.wav
BaseName_A8.wav

BaseName.video
audio12.wav
audio34.wav
audio56.wav
audio78.wav
BaseName.video
default.wav

Closed Caption Format Support

Drastic products generally support closed captions. Video I/O products can capture and play back closed caption information, depending on the capability of the software and the version. MediaReactor can convert closed caption formats along with the video and audio files.

Here is the list of closed caption formats supported by Drastic products:

- MPEG-2 'user' CC (read)
- MXF SMPTE 436 (read/write)
- MXF Avid embedded CC (read)
- MCC files side bar MacCaption/PC-Caption (read/write)
- SCC files side bar MacCaption/PC-Caption (read/write)
- MOV SD/HD CC tracks (read, and write with a special setup)

MediaReactor fully supports reading and writing the closed caption formats above. SCC and MCC are always written, along with other formats when available.

Mac OS-X Enabling legacy codecs for more output choices in QuickTime 7.4 and greater

Symptoms
With QuickTime 7.4 or later installed, some older encoders (codecs) may appear to be missing export options in applications that export QuickTime files, such as Final Cut Pro or Compressor or from the export options in QuickTime Player Pro.

Resolution
Take the following steps to enable additional encoders:
Choose Apple menu > System Preferences.
Choose View > QuickTime.
Click the Advanced button.
Select the checkbox next to Show legacy encoders.

Mac OS-X Using MediaReactor Workstation as a source reader for your own (or other) programs

MediaReactor Workstation provides a number of different ways to read supported file and compression types into your own programs. Here is a list of publicly available interfaces:

- **QuickTime Interface** - MRWS provides a series of components to QuickTime including file readers, compressors, decompressors and low level data handlers that allow any program to read all the available formats through QuickTime as though they were normal MOV files
- **DTMediaRead Interface** - This interface is a direct reader that requires an OEM agreement with Drastic.
- **MkRefMov** - MRWS includes a command line program called MkRefMov (Make Reference Movie). This program can make a reference movie (a small pointer only movie) for most supported files that can be read by any program supporting QuickTime MOV files. This includes new programs like Final Cut Pro X that don't normally support formats like MXF variants from Omneon, Pinnacle, Grass Valley and Avid.

Windows Using MediaReactor Workstation as a source reader for your own (or other) programs

MediaReactor Workstation provides a number of different ways to read supported file and compression types into your own programs. Here are a list of publicly available interfaces:

- **DirectShow** - MRWS installs a DirectShow Filter called 'Drastic MediaReactor File Source'. This can be opened with any supported file to provide audio and video to a DirectShow filter graph
- **QuickTime Interface** - MRWS provides a series of components to QuickTime including file readers, compressors, decompressors and low level data handlers that allow any program to read all the available formats through QuickTime as though they were normal MOV files

- **DTMediaRead Interface** - This interface is a direct reader that requires an OEM agreement with Drastic. See the DTMediaRead SDK page for more information.
- **MkRefMov** - MRWS includes a command line program called MkRefMov (Make Reference Movie). This program can make a reference movie (a small pointer only movie) for most supported files that can be read by any program supporting QuickTime MOV files. This includes new programs like Final Cut Pro X that don't normally support formats like MXF variants from Omneon, Pinnacle, Grass Valley and Avid.

OS Independent MkRefMov - Make Reference Movie

Usage: ./mkrefmov -i <sourcefile> [-o <targetfile>]
- -i <sourcefile> File to be referenced.
- -o [targetfile] Optional, reference file name, otherwise it will use the source name with an mov extension

NOTE: Due to limitations in QuickTime, MkRefMov will not work properly with a series of still images. If you supply one, a full MOV will be created instead.

AddTcMov - Add Time Code To An Existing QuickTime MOV

Usage: addtcmov -i <moviefile> -s <tcframe> -t <tctype> -r [reelname]
- -i <moviefile> QuickTime MOV to add tc track to
- -s <tcframe> Integer time code frame number of first frame in movie file.
- -t <tctype> Time code type
   - 24 = 1
   - NonDropFrame = 2
   - DropFrame = 4
   - Pal = 8
   - 50 = 16
   - 59.94 = 32
   - 59 special = 34 (not recommended)
   - 60 = 64
   - 23.98 = 128
   - 23.98 true = 132 (not recommended)
   - hundredths = 68
- -r <reelname> Reel name for file

mkRtIndex - Make An rtin (Real Time Index) File

Usage: mkRtIndex <sourcefile> [-v]

NOTE: parameters must be in order
- <sourcefile> File to be indexed.
- -v Optional, verify rtIndex instead of create.

MRMetaData - Extract a File's Metadata Information

Usage: mrmetadata.exe -i <sourcefile> -qx -j # -k <picon.jpg> -e # -c -n 1 -w <output.wav> -m <reference.mov>
- -i <sourcefile> File to be referenced.
- Optional parameters:
   - -q Quiet mode.

- o -x Do NOT export metadata XMLS
- o -j # Create a JPEG picon from frame #
- o -k <picon.jpg> File name for JPEG picon frame
- o -e # Extract a JPEG picon for every # frame
- o -c Extract closed captioning
- o -n # First channel of source audio for the wave file (0,2,4)
- o -w <output.wav> Extract audio to a wave file
- o -m <reference.mov> Create a QuickTime reference file

MRAnalyze - Use Standard PSNR/(MS)SSIM to Compare an Original and Converted File

mranalyze - Video/Audio analysis tool (www.drastic.tv)

Usage: mranalyze -a <source a> -b <source b> -q -p # -s # -m # -g # -f # -l # -x # -n # [-o <targetfile>]

- -a <source a> Original file.
- -b <source b> Compressed file.
- -q Quiet (no output) must be first argument.
- -p # PSNR type
- -s # SSIM type (0-Square, 1=Guassian)
- -m # MS-SSIM type
- -g # Global checks (0-none, 1-basic, 2-advanced)
- -f # First frame to check
- -l # Last frame to check
- -x # B source first frame to check
- -n # Next frame distance (def 1 = every frame)
- -o [targetfile] Optional, csv output name (or '-o i' to generate the file name)

The MR Lite Interface

The MR Lite interface is installed with MediaReactor Workstation to provide user directed transcoding functionality. The user may select a file, set up the transcoding parameters, and create a new file which is a copy of the media in the original file, but in a new file type.

MR Lite Displays and Controls

Main interface

Settings display   Profile pulldown

Configure button   Load button   Save button

Source button

Target button

Convert button   Close button

MediaReactor Lite

C:\Users\corey\AppData\Roaming\Drastic\mrdefault.mrws

Source

Target

Setting	Value	Description
File Type	2	MOV Apple QuickTime Movie file container
Codec ID	0x6D6A7062	fccName(bpjm) Description not found.
Width	720	Pixels
Height	486	Pixels
Bit Count	24	Bits
Key Frame	1	Frames
Data Rate	3200	KBytes / Sec
Video Quality	10000	Amount
Audio File Type	1	WAVE - Old Microsoft WAVE and Broadcast WAVE read...
Audio Compression	0	Default
Sample Rate	48000	Samples / Sec.
Bits/Sample	16	Bits
Audio Quality	320	Amount
Separate Audio	Disabled	
Audio Only	Disabled	
Dual Mono	Disabled	
Process Filter	Disabled	
Filter Type	0	This process has no effect on the video
Filter Option	0	
Filter Option	0	
Filter Option	0	

Convert   Close

Settings display - All of the details about the type of media file that will be created in the transcode process are displayed in this field. Each setting displays a value and a comment if available.

Profile pulldown – displays the current configuration file, and allows the user to select from any profiles that have been saved into the default location.

Configure button - Pressing this button opens the **MediaReactor Settings** window, which allows the user to view and change the settings for the file that will be created in the transcoding process.

Load button - Pressing this button opens a browser which allows the user to navigate to and load a configuration file.

Save button - Pressing the **Save** button opens a browser which allows the user to save the current configuration in a location and with the name of the user's choice.

Source button - Pressing the **Source** button opens a browser which allows the user to navigate to and select a media file to transcode. Where a file has been selected, the field to the left of this button displays the selected file's file path.

Target button - Pressing the **Target** button opens a browser which allows the user to set the location in which the transcoded file will be saved.

Convert button - Pressing the **Convert** button begins the transcode, if one has been set up.

Close button - Pressing the **Close** button closes the application.

Settings

Video
Section

Audio
Section

Filter
Section

Presets
pulldown

Accept
button

Cancel
button

MediaReactor Settings

Preset Profiles

Distribution_Quality_1

Video Settings

File Type mov - QuickTime Movie for 64 Bit

Codec ID MPEG IMX 50 Mb/s for Avid

Width 720   Height 512

Bit Depth 24   Data Rate 50000

Key Frame 1   Quality 10000

Audio Settings

☐ Separate   ☐ Audio Only   ☐ Dual Mono

File Type wave - Old Microsoft WAVE RIFF

Codec ID Default

Sample Rate 48000   Quality 320

Bits/Sample 16

Filter Settings

☑ Enable   HD->SD + Afterburn Timecode/Keycode

Value	Option
2	VITC Position
1	LTC Position
4	KEYCODE Position

☐ 2/3 Drop   ○ Add   ◉ Remove   00::0:0::00   Set

Accept   Cancel

Video section - This section contains controls to set the type of video that will be created during the transcode process.

Audio section - This section contains controls to set the type of audio that will be created during the transcode process.

Filter section - This section contains controls to set any filters that will be applied to the media created during the transcode process. There are also controls to add or remove any 2/3 drop processing for specific Telecine applications.

Presets pulldown – This pulldown menu allows the user to select between profiles that have been preset, and displays the current preset, if one has been selected.

Accept button - Pressing the **Accept** button sets any changes the user has made into the current configuration, and closes the **MediaReactor Settings** window.

Cancel button - Pressing the **Cancel** button closes the **MediaReactor Settings** window without changing the current configuration.

MediaReactor Settings – Video Section
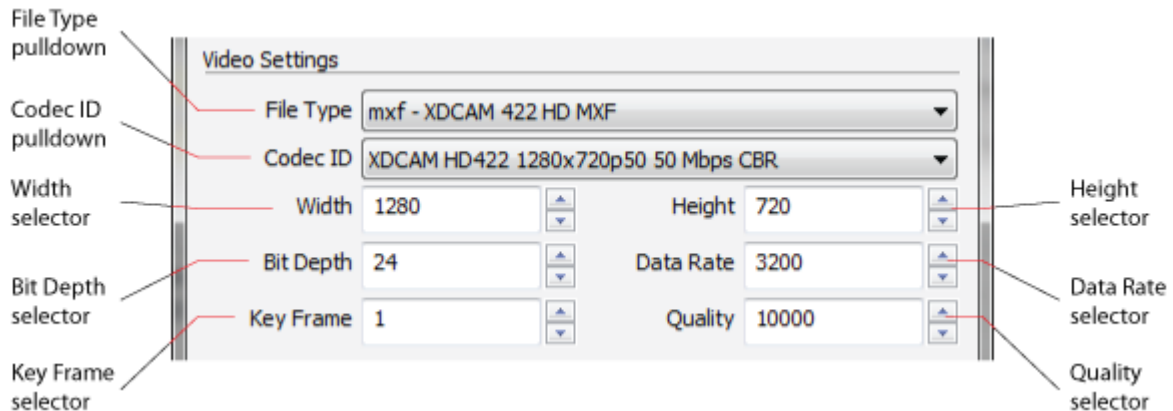
File Type pulldown - Use this pulldown menu to select between available video file types for the transcoded file.

Codec ID pulldown - Use this pulldown menu to select between available video codecs for the transcoded file.

Width selector – Where the selected video type has adjustable width parameters, clicking the up and down arrows to the right of the **Width** field allows the user to adjust the width of the transcoded video by selecting between supported values.

Bit Depth selector - Where the selected video type has adjustable bit depth parameters, clicking the up and down arrows to the right of the **Bit Depth** field allows the user to adjust the bit depth of the transcoded video by selecting between supported values.

Key Frame selector - Where the selected video type has adjustable key frame parameters, clicking the up and down arrows to the right of the **Key Frame** field allows the user to adjust the key frame for the transcoded video by selecting another frame.

Height selector - Where the selected video type has adjustable height parameters, clicking the up and down arrows to the right of the **Height** field allows the user to adjust the height of the transcoded video by selecting between supported values.

Date Rate selector - Where the selected video type has adjustable data rate parameters, clicking the up and down arrows to the right of the **Data Rate** field allows the user to adjust the data rate of the transcoded video by selecting between supported values.

Quality selector - Where the selected video type has adjustable quality parameters, clicking the up and down arrows to the right of the **Quality** field allows the user to adjust the quality of the transcoded video by selecting between supported values.

MediaReactor Settings – Audio Section
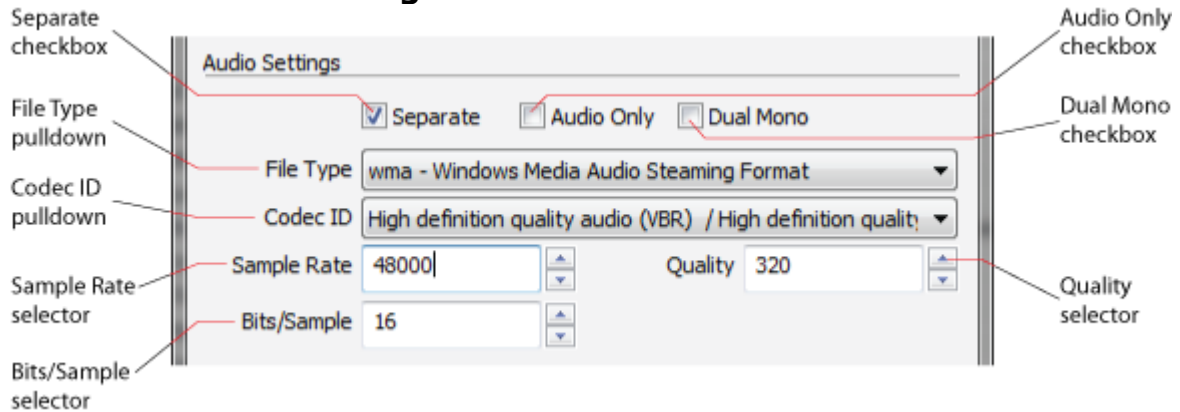
Separate checkbox - Select this checkbox to specify that the audio being created in the transcode process should be separate from the video.

File Type pulldown - Use this pulldown menu to select between available audio file types for the transcoded file.

Codec ID pulldown - Use this pulldown menu to select between available audio codecs for the transcoded file.

Sample Rate selector - Where the selected audio type has adjustable sample rate parameters, clicking the up and down arrows to the right of the **Sample Rate** field allows the user to adjust the sample rate of the audio in the transcoded video by selecting between supported values.

Bits/Sample selector - Where the selected audio type has adjustable bits per sample parameters, clicking the up and down arrows to the right of the **Bits/Sample** field allows the user to adjust the number of bits per sample for the audio in the transcoded video by selecting between supported values.

Audio Only checkbox - With the **Separate** checkbox selected, the user can select this checkbox to specify that no video file should be created during the transcode process.

Dual Mono checkbox - With the **Separate** checkbox selected, the user would be able to select this checkbox to specify that the separate audio file being created should provide two mono streams of audio.

Quality selector - Where the selected audio type has adjustable quality parameters, clicking the up and down arrows to the right of the **Quality** field allows the user to adjust the quality setting for the audio in the transcoded video by selecting between supported values.

MediaReactor Settings – Filter Section

1	**Enable** checkbox	Selecting this checkbox allows the user to apply a filter to the transcoded file.
2	**Filter** pulldown menu	Use the pulldown menu to select between available filters. Currently the following filters are available: Blend Fields, Burn-in Time Code/User bits, Deinterlace Blend, Deinterlace Interpolate, Duplicate Field 1, Duplicate Field 2, Flip Horizontal, Flip Vertical, HD->SD + Afterburn Timecode/Keycode, Invert Fields, Invert Temporal and Field Order, Invert Temporal Order.
3	**Filter Settings** field	Where a filter has values that can be adjusted, the values will be displayed in this field, and the user will be able to edit the value by clicking on the value and typing in a new value.

MediaReactor Settings – 2/3 Drop Section

1	**2/3 Drop** checkbox	Select the **2/3 Drop** checkbox to specify that either (a) a file has been created by a telecine and needs to have frames added to create a standard frame rate video, or (b) a telecine-derived video file which has had frames added needs to have those frames removed to return to a telecine frame rate.
2	**Add** button	Select the **Add** button to specify that frames will be added to create a standard frame rate video from a telecine-derived video file. Selecting this button de-selects the **Remove** button.
3	**Remove** button	Select the **Remove** button to specify that frames will be removed from a standard rate video file to create a telecine compatible video file. Selecting this button de-selects the **Add** button.
4	**Time Code** field	Where the user is removing frames from a standard rate video file created by adding frames to a telecine-derived video file, the user can set the "A" frame, which is typically an interpolated frame, by typing in the time code location of the "A" frame.

5	**Set** button	Once the time code location of the "A" frame has been entered into the **Time Code** field, pressing the **Set** button sets the time code into memory.

MR Lite Basic Operation

Here is how to select a media file, set up various parameters, and transcode the media file into another type of media file.

Transcode a File

Run **MR Lite**. **MR Lite** resides by default in the MediaReactor install folder, and it may additionally have a shortcut placed in the Start menu, within Start/Programs/MediaReactor/ MediaReactor Lite.

Select a Source File

Select the file you want to transcode:
Press the **Source** button to open the source file browser. Navigate to the location of the file you want to transcode and select it. Press the **Open** button. The browser will close and the network path of the selected file will appear in the **Source File** field.

Select a Target Location

Select a folder to save the new file in:
Press the **Target** button to open the target location browser. Navigate to the location of the folder in which you would like the transcoded file to be saved. An option is available to create a new folder. Once the location has been properly selected, press the **OK** button.

Set the Transcoding Parameters

Determine the type of file you want to create:
Press the **Configure** button. This opens the **MediaReactor Settings** window.

Video Settings

Use the **File Type** pulldown menu to select the file type.
Use the **Codec ID** pulldown menu to select the codec.
Use the **Width** and **Height** selectors to set the frame size.
Use the **Bit Depth**, **Data Rate**, and **Quality** selectors to set up transcoding to file types that have adjustable resolution settings.
Use the **Key Frame** selector to set the key frame interval for transcoding to file types that require key frames to reconstruct interpolated frames.

Audio Settings

Use the **Sample Rate**, **Quality**, and **Bits/Sample** selectors to set up transcoding to file types that support adjustable audio resolution.

The **Separate** checkbox is available for transcodes that require an audio file to be created outside of the context of the video file. Selecting the **Separate** checkbox activates the other controls in the **Audio Settings** section.

If **Separate** has been selected, selecting the **Audio Only** checkbox will ignore the video in the source file and only create an audio file during the transcode.

If **Separate** has been selected, selecting the **Dual Mono** checkbox will create two mono audio files during the transcode.

If **Separate** has been selected, use the **File Type** pulldown menu to set the file type for the separate audio file that will be created during the transcode.

If **Separate** has been selected, use the **Codec ID** pulldown menu to set the codec for the selected audio file type.

Filter Settings

Specific filter settings are additionally available for the transcoded file. One filter may be applied per transcode.

Select the **Filter** checkbox to activate the controls in the **Filter Settings** section.

Use the pulldown menu to select the **Blend Fields** filter where the transcoded file should have both fields of a frame blended together.

Use the pulldown menu to select the **Burnin – Timecode/User Bits** filter where the transcoded file should display the time code and user bits over the transcoded video. The **TC-UB Display** values may be set to 2 for both time code and user bits, or to 1 for time code only. The **X Position** and **Y Position** values set the location on the image where the time code and possibly user bits will be displayed. The **Display Scale 1-4** value sets the size of the time code and possibly user bits that will be displayed.

Use the pulldown menu to select the **Deinterlace Blend** filter where the source file is an interlaced type and the transcoded file should have these fields deinterlaced and blended. The **Edge** value may be set to determine how much blending should be applied to detected edges. The **Threshold** value may be set to determine the level at which blending will be applied.

Use the pulldown menu to select the **Deinterlace Interpolate** filter where the source file is an interlaced type and the transcoded file should have these fields deinterlaced, and interpolated. The **Edge** value may be set to determine how much interpolation should be applied to detected edges. The **Threshold** value may be set to determine the level at which interpolation will be applied.

Use the pulldown menu to select the **Duplicate Field 1** filter where the transcoded file should replace each field 2 with its associated field 1.

Use the pulldown menu to select the **Duplicate Field 2** filter where the transcoded file should replace each field 1 with its associated field 2.

Use the pulldown menu to select the **Flip Horizontal** filter where the transcoded file should be reversed left to right.

Use the pulldown menu to select the **Flip Vertical** filter where the transcoded file should be reversed top to bottom.

Use the pulldown menu to select the **HD->SD + Afterburn Timecode/Keycode** filter where the source file is an HD frame size and should be transcoded to standard definition, and additionally should have the time code and key code displayed over the image. The **VITC Position** value may be set to specify where in the sequence VITC will

be displayed, from 1 (at the top) to 4 (at the bottom). The **LTC Position** value may be set to specify where in the sequence LTC will be displayed, from 1 (at the top) to 4 (at the bottom). The **Keycode Position** value may be set to specify where in the sequence keycode will be displayed, from 1 (at the top) to 4 (at the bottom). The **Inkcode Position** value may be set to specify where in the sequence Inkcode will be displayed, from 1 (at the top) to 4 (at the bottom).

Use the pulldown menu to select the **Invert Fields** filter where the transcoded file should have each field 2 first, followed by field 1.

Use the pulldown menu to select the **Invert Temporal and Field Order** filter where the transcoded file should have each field 2 first, followed by field 1, and additionally should begin at the last frame and end at the first frame.

Use the pulldown menu to select the **Invert Temporal Order** filter where the transcoded file should begin at the last frame and end at the first frame.

2/3 Drop Settings

2/3 Drop Settings are available for telecine applications. Where a file has been created by a telecine at standard 24 fps frame rate, frames may be added to create a standard 30 fps frame rate file. Where a file has been created by a telecine and has had frames added to create a standard 30 fps frame rate file, the added frames may be removed.

Select the **2/3 Drop** checkbox to activate the controls in the **2/3 Drop Settings** section.

Select **Add** if you need to add frames to a 24 fps telecine derived file to create a standard 30 fps frame rate file.

Select **Remove** if you need to remove added frames from a telecine derived file which has had frames added. In this case you will need to set the "A" Frame, or the first frame that has been added. To accomplish this, step through the file frame by frame (starting at the first frame) until a frame appears whose quality is noticeably lower than its neighboring frames – this will be the "A" Frame. Note the time code location of this frame and enter it into the time code field. Press the **Set** button to set this as the first added frame. Since frames are added in a predictable sequence, selecting the first "A" Frame reliably specifies all of the added frames.

Accept Settings

Once all of the configuration settings have been correctly set for the required transcode, press the **Accept** button. This closes the **MediaReactor Settings** window. Looking back at the main **MR Lite** interface, the user will see their specified values within the **Configuration Settings** field.

Begin the Transcode

Press the **Convert** button to begin the transcode. A progress bar along the bottom of the window shows how the transcode is progressing. If desired, the user may stop the transcode by pressing the **Abort** button.

A checkbox is available to display the transcoded file during the transcode. Select the **Show Output** checkbox. This action displays frames from the current transcode over the **Configuration Settings** field. The current frame number and the total number of frames

are displayed at the bottom left of this field. The time code location of the current frame is displayed at the bottom right of this field.

Configuration File Options

The user may save their configuration settings as a file. This allows the user to save their most commonly used (or all of their) configurations as XML files for later use.

Save Configuration Settings

Here is how to save the current configuration settings for use on other source files. Press the **Save** button. This opens a standard browser. Use the browser to navigate to and select a folder in which you would like to save the configuration. Enter a name for the configuration. Press the **Save** button.

Load Configuration Settings

Configuration settings that were created and saved (or created elsewhere and loaded) with the current install of **MR Lite** will show up in the **Configuration Settings** pulldown menu at the top of the **MR Lite** interface. However if a configuration settings file has been created outside of this instance the user may still load it. Press the **Load** button. This opens a standard browser. Navigate to the location of the configuration settings XML file and select it. Press the **Open** button.

Close MR Lite

Once all required transcodes have been completed, the user may close **MR Lite**. Press the **Close** button. The MR Lite interface will close.

MediaReactor and MediaReactor Workstation integrate tightly with installed operating systems and applications. This section has tips and tricks on getting the most out of MediaReactor and/or MediaReactor Workstation.

Command Line Utilities

mrcmd

mrcmd - Drastic MediaReactor Command Line Interface
Build: 854
(c)opyright 1995-2018 Drastic Technologies Ltd. All rights reserved.

Usage: mrcmd [?] [-s#] [-e#] [-x#] [-y#] [-b#] [-q#] [-d#] [-k#] [-sm] [-r#] [-a#] [-c#]
[-p#] [-F<audiofiletype>] [-AV] [-f<filety
e>] [-v<codec>] [-n#] [-i#] [=P<profile>] <inputname> <ouputname>
[] = optional
Parameters:

     -s#   = start frame (def: 0)
     -e#   = end frame (def: length of input)
     -t   = allow transcode stream (def: convert stream)
     -x#   = Horizontal size >1 (def: same as input)
     -y#   = Vertical size >1 (def: same as input)
     -b#   = Bit depth 8/16/24/32 (def: optimal)
     -q#   = Video quality 0..10000 (def: highest)
     -d#   = Video data rate >=0 (def: highest)
     -k#   = Key frame distance (def: 15)
     -s   = Separate audio files (def: internal)
     -m   = Dual mono audio files (def: stereo, requires -s)
     -r#   = Audio frequency/rate (def: same as input)
     -a#   = Audio sample size 8/16 (def: same as input)
     -c#   = Audio channels 1/2/4/5/6/8 (def: same as input)
     -p#   = Audio quality 0..10000 (def: highest)
     -i#   = Output video file ID (from codeclist.txt)
     -f<filetype> = Output video file type - AVI/MOV/OMF (def: auto sense)
     -F<filetype> = Output audio file type - WAV/AIF/AU (def: auto sense, requires -s)
     -v<codec>   = Output video codec (text) - MMES/2yuv/DVSD (def: RGB
uncompressed)
     -n#   = Output video codec (numeric hex) - 88000001/1/a/f
     -A   = Disable audio output (def: depends on input)
     -V   = Disable video output (def: depends on input)
     -P<profile> = MediaReactor profile file path/name (optional)
     <inputname> = Input file path/name (required)
     <outputname> = Input file path/name (required)

To get a list of all the file type available in the file codeclist.txt, use
     mrcmd -all
To get a list of the codecs for a file type, use
     mrcmd #

mkrtindex

mkRtIndex - Make an rtin (RTIndex) file for a MOV, MXF, AVI, etc, by Drastic Technologies (www.drastic.tv)
Usage: mkRtIndex <sourcefile> [-v]
        parameters must be in order
<sourcefile> File to be indexed.
-v   Optional, verify rtIndex instead of create.

addtcmov (windows 32 only)

addtcmov - Add a time code track to a QuickTime movie file, by Drastic Technologies
(www.drastic.tv)
Usage: addtcmov -i <moviefile> -s <tcframe> -t <tctype> -r [reelname]
-i <moviefile> QuickTime MOV to add tc track to
-s <tcframe>   Integer time code frame number of first frame in movie file.
-t <tctype>   Time code type
   24   = 1
   NonDropFrame = 2
   DropFrame   = 4
   Pal   = 8
   50   = 16
   59.94   = 32
   59 special   = 34
   60   = 64
   23.98   = 128
   23.98 true   = 132
   hundredths   = 68
-r <reelname>   Reel name for file

mkrefmov (windows 32 only)

mkRefMov - Make a QuickTime reference movie file for a GXF, MXF, AVI, etc, by Drastic
Technologies (www.drastic.tv)

Usage: ./mkrefmov -i <sourcefile> [-o <targetfile>]
    -i <sourcefile> File to be referenced.
    -o [targetfile] Optional, reference file name.

mrmetadata

mrmetadata - Extract metadata information from GXF, MXF, AVI, etc, by Drastic Technologies (www.drastic.tv)
Build: 888

(c)opyright 1995-2018 Drastic Technologies Ltd. All rights reserved.

Usage: mrmetadata.exe -i <sourcefile> -qx -j # -k <picon.jpg> -e # -c -n 1 -w <output.wav> -m <reference.mov>
    -i <sourcefile> File to be referenced.
    Optional paramaters:
    -q   Quiet mode.
    -x   Do NOT export metadata XMLS
    -j #   Create a JPEG picon from frame #
    -k <picon.jpg>   Filename for JPEG picon frame
    -e #   Extract a JPEG picon for every # frame
    -c   Extract closed captioning
    -n #   First channel of source audio for the wave file (0,2,4)
    -w <output.wav>   Extract audio to a wave file
    -m <reference.mov> Create a QuickTime reference file

mranalyze

mranalyze - Video/Audio analysis tool, by Drastic Technologies (www.drastic.tv)

Usage: ./mranalyze -a <source a> -b <source b> [-o <targetfile>]
      -a <source a>   Original file.
      -b <source b>   Compressed file.
      -q   Quiet (no output) must be first argument.
      -p #   PSNR type
      -s #   SSIM type (0-Square, 1=Guassian)
      -m #   MS-SSIM type
      -g #   Global checks (0-none, 1-basic, 2-advanced)
      -o [targetfile] Optional, reference file name.

mrfix

mrfix - Change metadata in media files, by Drastic Technologies (www.drastic.tv)
Build: 888

(c)opyright 1995-2018 Drastic Technologies Ltd. All rights reserved.

Usage: mrfix -i <sourcefile> -m c [per mode parameters]
    -i <sourcefile>   File match string (e.g. *.ari)
    -q   Quiet, minimal print outs
    -t   Test, and do not alter files
    -m <mode>   What kind of file operation:
      a   Arri file metadata update
    For mode details, specify the mode (e.g. mrfix -m a ?)

Arri Metadata
    The Arri file metadata is specified in pairs, after the main parameters above. Each pair consists of the element that should be changed, and what it should be changed to. There are three types of data: U32 (a 32 bit integer number, e.g. 24), R32 (a floating point value, e.g. 1.223) and U8 (a string with a maximum size, e.g. "this is metadata"). When specifying strings, quote marks must be used if the string contains spaces. For FPS settings, they can be specified as simple frame rates (e.g. 24, 25, 30) or non integer frame rates by multiplying by 1000 (e.g. 23976, 29970).

    WhiteBalance U32
    GreenTintFactor R32
    WhiteBalanceFactorR R32
    WhiteBalanceFactorG R32
    WhiteBalanceFactorB R32
    WBAppliedInCameraFlag U32
    ExposureIndex U32
    BlackLevel U32
    WhiteLevel U32
    LookFile U8[32]
    SensorFps U32   (note: should match ProjectFPS)
    ProjectFps U32   (note: should match SensorFPS)
    CircleTake U32
    ReelName U8[8]
    SceneName U8[16]
    TakeName U8[8]
    Director U8[32]
    DoP U8[32]
    ProductionName U8[32]
    Company U8[32]
    Notes U8[256]
    SoundFileName U8[32]
    SoundRollName U8[32]
    SceneFileName U8[32]
    TakeFileName U8[32]

Info U8[32]

Example command line:

mrfix -i V:\Test*.ari -m a SensorFPS 24 ProjectFPS 24 ExposureIndex 1200 WhiteBalanceFactorR 0.901 WhiteBalanceFactorG 1.0 WhiteBalanceFactorB 1.34 LookFile testlook.look Director "James L Brooks" Info "This is a line of information

Appendix - Drastic DT3D Stereo File

The Drastic DT3D stereo file is an XML based re-director file that can merge two separate 'eye' files into a stereo pair, as well as adding audio and optional closed captioning.

DT3D Stereo File Format
Drastic Technologies Ltd
(c) Copyright 2005-2018, Drastic Technologies Ltd
All Rights Reserved

Contents:
----

1. Overview
2. File Format
3. Samples

===
====================
===
====================
===
==
====================
1. Overview
====================
====================

The .dt3d 'stereo' file is a text based XML file that specifies two separate files as the left and right eyes of a stereo pair. These files do not have to have a particular naming convention, or even be of the same file type, but their internal compression must be the same. The stereo file reader will open both video files, and an optional set of audio files, and combine them for 3D use in videoQC, Drastic DDR, MediaNXS, MediaReactor and other Drastic products.

===
====================
===
====================
===
==
==
====================
2. File Format
====================
====================

The file format is an ANSI or UTF-8 XML file using iso-8859-1 encoding. It has to use the file extension .dt3d to be recognized by the stereo reader plugin in Final Cut Pro, Premiere, Media Composer, etc. If you are only using Drastic or your own software, it can also use the .stereo extension. Extra information may encoded at any point in the file, but at minimum there must be a Stereo tag with at least two File<number> tags to specify the right and left eye files.

All stereo files start with the xml type specifier:
<?xml version="1.0" encoding="iso-8859-1" ?>
This is normally followed by a comment specifying the file's purpose and where the format document can be retrieved:
<!-- Drastic stereo file redirector -->
<!-- Documentation http://www.drastictech.com/stereofileformat.txt -->

All of the important information for the stereo reader is contained within a Stereo tag pair
<Stereo>
</Stereo>

The stereo reader is designed to handle more than two video frames (eyes) per read, but for now it is only using two. Each file is enclosed in a File<number> tag pair. The numbering next to the File should be 0 and 1 for the left and right eyes:
<File0>
 <!-- Left eye -->
</File0>
<File1>
 <!-- Right eye -->
</File1>

within these tags are sub tags that specify where the video file is and any offset required to align the two eyes. A typical file tag area will look like this:
<File0>
        <!-- Optional directory, else current dir-->
        <Dir>C:\Optional</Dir>
        <!-- First file name -->
        <Name>test_l.avi</Name>
        <!-- Offset in frames -->
        <Offset>22</Offset>
</File0>

<Dir> - Optional, contains the directory or folder in which the file resides. If it is not specified, it is assumed the file is in the same directory as the stereo file.
<Name> - Required, contains the actual file name and extension of the media file. For sequences, simply specify the first name in the sequence and the rest will be found automatically
<Offset> - Optional, if present it will be the frame in the media file that will be read as frame 0. All frames in the file will be offset by this amount. If it is not specified, it is set to 0

There is also an optional <AudioFile> tag that can be used to specify an arbitrary separate audio file to be read with the video files. This section has the same format as the File<num-

ber> sections. If it is not specified, then audio will be read from the first (File0/left eye) video file.

An MCC or SCC closed captioning file may also be specified in a <ClosedCaptions> tag, using the same <File> tag as the audio and video formats.

```
==========================================================
====================
==========================================================
====================
==========================================================
====================================
====================
3. Samples
====================
====================


-snip-------------------------------------------------------------
<?xml version="1.0" encoding="iso-8859-1" ?>
<!-- Stereo file redirector -->
<!-- http://www.drastictech.com/stereofileformat.txt -->
<Stereo>
  <!-- First file -->
  <File0>
    <!-- Optional directory, else current dir-->
    <Dir>C:\Optional</Dir>
    <!-- First file name -->
    <Name>test_l.avi</Name>
    <!-- Offset in frames -->
    <Offset>22</Offset>
  </File0>
  <File1>
    <!-- Optional directory, else current dir-->
    <Dir>C:\Optional</Dir>
    <!-- First file name -->
    <Name>test_r.avi</Name>
    <!-- Offset in frames -->
    <Offset>0</Offset>
  </File1>
  <!-- Optional audio file name. If not present then audio will be read from the first video file
-->
  <AudioFile>
    <!-- Separate audio file name -->
    <Name>test.wav</wave>
  </AudioFile>
</Stereo>
-snip-------------------------------------------------------------

-snip-------------------------------------------------------------
```

```xml
<?xml version="1.0" encoding="iso-8859-1" ?>
<!-- http://www.drastictech.com/stereofileformat.txt -->
<Stereo>
  <File0>
    <Name>test_l.avi</Name>
  </File0>
  <File1>
    <Name>test_r.avi</Name>
  </File1>
  <AudioFile>
    <Name>test.wav</wave>
  </AudioFile>
</Stereo>
```
-snip--

-snip--
```xml
<?xml version="1.0" encoding="iso-8859-1" ?>
<!-- http://www.drastictech.com/stereofileformat.txt -->
<Stereo>
  <File0>
    <Name>test_l.avi</Name>
  </File0>
  <File1>
    <Name>test_r.avi</Name>
  </File1>
</Stereo>
```
-snip--

-snip--
```xml
<?xml version="1.0" encoding="iso-8859-1" ?>
<!-- Drastic Stereo file redirector -->
<!-- Documentation:  http://www.drastictech.com/stereofileformat.txt -->
<Stereo>
  <!-- First/Left file -->
  <File0>
    <!-- First/Left file name -->
    <Name>T3D003013_Centre_fast_1080p60_l.yuv</Name>
    <!-- Offset in frames -->
    <Offset>0</Offset>
  </File0>
  <!-- Second/Right file -->
  <File1>
    <!-- Optional directory, else current dir-->
    <!-- <Dir>C:\Optional</Dir> -->
    <!-- Second/Right file name -->
    <Name>T3D003013_Centre_fast_1080p60_r.yuv</Name>
    <!-- Offset in frames -->
    <Offset>0</Offset>
  </File1>
   <!-- Base audio file name -->
```

```
  <AudioFile>
    <Name>../Audio_wav/T3a003y113_Centre_fast_act_unc.wav</Name>
  </AudioFile>
  <ClosedCaptions>
    <Name>T3D003013_Centre_fast_1080p60_r.mcc</Name>
  </ClosedCaptions>
</Stereo>
-snip-------------------------------------------------------------
```

Appendix - Drastic DHDR/HDR YUV RAW Video Header Format

Drastic software can read many file formats which it does not natively understand. Formats that have a fixed video frame size and are logically organized can be described using the industry standard HDR text file format (NOTE: Not the still image HDR). This page describes the file format.

To do this a couple of basic criteria must be met:
- The frames within the stream or series of images, must be the same size.
- The frames must be in a supported compression format (see below)
- A header (.HDR/.STILLS) file must be created to describe the video

The HDR/.STILLS must be in the following format:
HDR File Format
Drastic Technologies Ltd.
(c) Copyright 1995-2018, Drastic Technologies Ltd.
All Rights Reserved

Contents:
----

1. Overview
2. Settings
3. Calculations
4. Samples

==
====================
==
====================
==
==
=======================================
====================
1. Overview
====================
====================

This document describes header (*.dthdr/*.hdr) file format for use with the Drastic Technologies dtx_HdrYuv.d1x plug in for MediaReactor and QuickClip/VVW. This format has been in use for over 25 years in various forms as a *.hdr. By default, Drastic software now writes and reads a *.dthdr (as well as still reading the .hdr). The *.dthdr is the same format with more capabilities including better audio, split long file and time code support. Herein are outlined the most common descriptors and their meanings as they have been standardized in

our software. An updated version of this document is available at:

http://www.drastictech.com/hdrfileformat.txt

The basic concept of the HDR file is a text file that describes a simple video stream in which each frame is the same size, and its associated video and audio information. On disk, there will be 2 or more files, normally with the same starting name:

E:\TestRecord\Testing.dthdr
E:\TestRecord\Testing.yuv
E:\TestRecord\Testing.wav

In this example the yuv file contains all the captured frames (as opposed to individual num-bered yuv files), the wav file contains the matched audio and the hdr file describes them as specified below. When loading these files into QuickClip or MediaReactor, the hdr file should be chosen to make sure all files are picked up properly. The basic format of the hdr file con-sists of a series of settings, each taking two text lines. The first line is the label and is always preceded by a percentage sign (%) and the second line is the value of the setting:

% Image Size (This line describes what the next line specifies)
486 720   (This contains the values - NTSC 720 x 486)

===
===
===
==
===
====================
2. Settings
====================
====================

% Color Format
YUV420
This setting describes the format of the video frames within the file. The current possible for-mats are:
yuv422_16 - 4:2:2 Planar YCbCr in I420 order, 16 bits per component
yuv422_16le - Little endian 4:2:2 planar YCbCr in I420 order, 16 bits per component
yvu422_16 - 4:2:2 Planar YCbCr in YV12 order, 16 bits per component
yvu422_16le - Little endian 4:2:2 planar YCbCr in YV12 order, 16 bits per component
yuv422_10 - 4:2:2 Planar YCbCr in I420 order, 10 bits per component in 16 bit container
yuv422_10le - Little endian 4:2:2 planar YCbCr in I420 order, 10 bits per component in 16 bit container
yvu422_10 - 4:2:2 Planar YCbCr in YV12 order, 10 bits per component in 16 bit container
yvu422_10le - Little endian 4:2:2 planar YCbCr in YV12 order, 10 bits per component in 16 bit container
yuv422_12 - 4:2:2 Planar YCbCr in I420 order, 12 bits per component in 16 bit container

yuv422_12le - Little endian 4:2:2 planar YCbCr in I420 order, 12 bits per component in 16 bit container
yvu422_12 - 4:2:2 Planar YCbCr in YV12 order, 12 bits per component in 16 bit container
yvu422_12le - Little endian 4:2:2 planar YCbCr in YV12 order, 12 bits per component in 16 bit container
yuv422_14 - 4:2:2 Planar YCbCr in I420 order, 14 bits per component in 16 bit container
yuv422_14le - Little endian 4:2:2 planar YCbCr in I420 order, 14 bits per component in 16 bit container
yvu422_14 - 4:2:2 Planar YCbCr in YV12 order, 14 bits per component in 16 bit container
yvu422_14le - Little endian 4:2:2 planar YCbCr in YV12 order, 14 bits per component in 16 bit container
yuv444_10 - 4:4:4 Planar YCbCr in I420 order, 10 bits per component in 16 bit container
yuv444_10le - Little endian 4:4:4 planar YCbCr in I420 order, 10 bits per component in 16 bit container
yvu444_10 - 4:4:4 Planar YCbCr in YV12 order, 10 bits per component in 16 bit container
yvu444_10le - Little endian 4:4:4 planar YCbCr in YV12 order, 10 bits per component in 16 bit container
yuv444_12 - 4:4:4 Planar YCbCr in I420 order, 12 bits per component in 16 bit container
yuv444_12le - Little endian 4:4:4 planar YCbCr in I420 order, 12 bits per component in 16 bit container
yvu444_12 - 4:4:4 Planar YCbCr in YV12 order, 12 bits per component in 16 bit container
yvu444_12le - Little endian 4:4:4 planar YCbCr in YV12 order, 12 bits per component in 16 bit container
yuv444_14 - 4:4:4 Planar YCbCr in I420 order, 14 bits per component in 16 bit container
yuv444_14le - Little endian 4:4:4 planar YCbCr in I420 order, 14 bits per component in 16 bit container
yvu444_14 - 4:4:4 Planar YCbCr in YV12 order, 14 bits per component in 16 bit container
yvu444_14le - Little endian 4:4:4 planar YCbCr in YV12 order, 14 bits per component in 16 bit container
YUV420 - Planar YCbCr in IYUV/I420 order (ST/Thompson/MPEG Groups)
YV12 - Planar YCbCr in YV12 order
YUV422P - Planar YCbCr with 4:2:2 sub sampling (Sony)
YVU422P - Planar YCbCr with 4:2:2 sub sampling (chroma inverted)
V210 - Interleaved 10 Bit YCbCr v210 format (standard Quick Clip 10 Bit YCbCr)
YUV4224_10 - Interleaved 10 Bit YCbCr v210 format with alpha/key channel
YUV4224 - Interleaved 8 Bit YCbCr yuv2/UYVY format with alpha/key channel
YUV422_fields - Separate fields of 4:2:2 YCbCr (Crescent)
YUV422 - Interleaved 8 Bit YCbCr UYVY format (standard Quick Clip 8 Bit YCbCr) alias uyvy422
yuv422_16 - Interleaved 16 Bit YCbCr UYVY format
yuv422_16le - Little endian interleaved 16 Bit YCbCr UYVY format
UYVY422 - YUV 4:2:2 interleaved 8 bit packed as U Y V Y U Y V Y ...
YUY2_16 - YUV 4:2:2 interleaved 16 bit packed as Y U Y V Y U Y V ...
YUY2_16le - Little endian YUV 4:2:2 interleaved 16 bit packed as Y U Y V Y U Y V ...
YUY2 - YUV 4:2:2 interleaved 8 bit packed as Y U Y V Y U Y V ...

DPXRGBLEFILL - DPX 10 bit RGB, little endian, filled
DPXRGBLE - DPX 10 bit RGB, little endian, padded
DPXRGBFILL - DPX 10 bit RGB, big endian, filled
DPXRGB - DPX 10 bit RGB, big endian, padded

DPXABGRLEFILL - DPX 10 bit ABGR, little endian, filled
DPXABGRLE - DPX 10 bit ABGR, little endian, padded
DPXABGRFILL - DPX 10 bit ABGR, big endian, filled
DPXABGR - DPX 10 bit ABGR, big endian, padded

PRGB - 8 bit x 3 Planar RGB
PRGBA - 8 bit x 4 Planar RGBA
PBGR - 8 bit x 3 Planar BGR
PABGR - 8 bit x 4 Planar ABGR
PBGRA - 8 bit x 4 Planar BGRA
PARGB - 8 bit x 4 Planar ARGB

RGBA - 32 Bit Interleaved RGB (TIFF)
ARGB - 32 Bit Interleaved RGB (Mac)
BGRA - 32 Bit Interleaved RGB (Windows BMP/TGA)
BGR - 24 Bit Interleaved RGB (Windows BMP/TGA)
RGB - 24 Bit Interleaved RGB (TIFF)
TIFF24 - 24 Bit Interleaved RGB TIFF ordering
TIFF32 - 32 Bit TIFF (same as RGBA)
FULLDUAL - 10 bit dual frame YCbCr (stereo)
Grey - 8 bit grey/gray plane of video data alias Gray
DVSD - DVSD 'dv/dif' stream 4:2:0 or 4:1:1 8 bit SD
DV25 - DV25 'dv/dif' stream 4:2:0 8 bit SD
DV50 - DVCPro 50 stream 4:2:2 8 bit SD alias avdv
DV100 - DVCPro HD/DV-100 stream 4:2:2 8 bit HD alias avd1
AVCI100 - Panasonic AVCi 100
DNXHD - Avid DNxHD stream alsias av10, avd0~9
IMX30 - Sony IMX MPEG 30 Mbit stream 4:2:2 8 bit
IMX40 - Sony IMX MPEG 40 Mbit stream 4:2:2 8 bit
IMX50 - Sony IMX MPEG 50 Mbit stream 4:2:2 8 bit

STEREO8 - Dual 8 bit YCbCr interleaved streams (one after another)
STEREO10 - Dual 10 bit YCbCr interleaved streams (one after another)
STEREO30 - Dual 10 bit RGB DPX interleaved streams (one after another)
STEREO32 - Dual 8 bit BGRA (windows order) interleaved streams (one after another)
stereo12bggr - Dual 12 bit raw bayer (one after another)
stereo12rggb - Dual 12 bit raw bayer (one after another)
stereo12gbrg - Dual 12 bit raw bayer (one after another)
stereo12grbg - Dual 12 bit raw bayer (one after another)

bggrbe12 - Bayer pattern, 12 bits per component packed big endian
rggbbe12 - Bayer pattern, 12 bits per component packed big endian
gbrgbe12 - Bayer pattern, 12 bits per component packed big endian
grbgbe12 - Bayer pattern, 12 bits per component packed big endian
bggr12 - Bayer pattern, 12 bits per component packed little endian
rggb12 - Bayer pattern, 12 bits per component packed little endian
gbrg12 - Bayer pattern, 12 bits per component packed little endian
grbg12 - Bayer pattern, 12 bits per component packed little endian

ARRIBAYERDLRAW12 - ARRI dual link raw 12 bit bayer packed into YCbCr 10

*** DEPRECATED **
RGB_10 - 30 Bit DPX/Cineon Ordering
RGB_10LE - 30 Bit little endian DPX/Cineon ordering

*** NOT AVAILABLE **
n/a YUV411 - Interleaved YCbCr DV Order
n/a YUV411P - Planar YCbCr DV Order
n/a YUV410 - Interleaved YCbCr 4:1:0
n/a YUV410P - Planar YCbCr 4:1:0
n/a YUVA444 - Interleaved 8 bit YCbCrAlpha (Adobe)
n/a UYV444 - YUV 4:4:4 packed as U Y V   U Y V ...
n/a MJPEG - Motion JPEG

% Image Size (NbRows,NbCols)
486 720
The vertical and horizontal size of the video frame. Can be any size, but the following sizes
are normal: 480x640, 480x704, 480x720, 486x720, 512x720, 576x704, 576x720, 608x720,
720x1280, 1080x1920, 1088x1920 (please note, the 'x' between the height and width is not
supported. It must be a space in the hdr file)

% Number of Fields per Image (optional)
2
Number of separate images that make up a field. For most images this would be 1 indicating
1 full size plane with progressive or interleaved lines. If there are two separate images that
need to be interleaved, this would be set to 2. This value does not affect the Image Size
above, that is always the absolute number of pixels vertically and horizontally. Please note:
this is the correct interpretation of this value, previously we ignored this and recommended
2 where it should have been 1.

% Number of Image
1800
The number of individual frames in a file. If this value is larger than the actual number of
frames available, it will be corrected, so if you are unsure set it to a very large value (e.g.
4000000)

% Frames per Second
29
The frame rate of the video frames. This may be any value, but some values have special
meanings:
23   - Means 23.98
24   - Standard film
25   - PAL/25p/50i
29   - Means 29.97 - NTSC/29.97p/59.94i
30   - NTSC NDF/30p/60i
50   - 50p

59   - Means 59.94p (for 720p, 1080p, etc)
60   - Means 60p (for 720p, 1080p, etc)

% Header Offset (optional)
0
How far into the file the series of video frames starts

% Video Alignment (optional)
512
Alignment of each frame within the stream. If the frames are back to back this value will be 1. It is normal to pad each frame to align it to a disk sector for higher speed reads and writes. This value is normally either 0 (which indicates 1), 512 (windows), 4096 (sgi), 16384 (js)

% Video Line Pitch (optional)
1440
The number of bytes in one line of video. Especially important for YCbCr 10 bit when frame size is 1280x720 or 2048x1080

% Video Offset (optional)
0
How far into each frame, from the calculated start, the actual video starts

----
% Video Name (optional)
ThisFile.YUV
This is the name of the video file/stream associated with this header. Normally the video file has the same name as the hdr file, but with a .yuv extension. This allows the video name to be overridden to any file name.

% Audio Name (optional)
ThisFile.Wav
This is the name of an associated audio file. Normally QuickClip/MediaReactor looks for side bar wave or aiff files with the same name as the header file. This allows the audio file to be specified.

% Audio Format
Stereo

% Audio Frequency
48000

% Audio Block Align
4

% Audio Bits Per Sample
24

% Timecode (optional)
01:00:00:00
This is the time code value of the first frame. The time is assumed to run contiguously from this point.

% Userbits (optional)
BAADF00D
This is the user bits returned for this stream. It is assumed it is the same for all video frames.

% Start Frame (optional)
100
This specifies the first video frame of the stream. By default, this is 0, but in a circular file the last frame written may not be the last frame of the file. In the case above, if start frame is 100, the last frame written was 99. The reader will read from 100->end of file and then 0->99 automatically.

% Max Segment (optional)
1296000
This specifies how long one segment of a multi segment record will be. Defaults to 12 hours (above is 12 hours in frames, NDF 30).

Please note, comments and other information may exist in the file, but if the file is re-written by QuickClip or VVW, then that information will be lost.

==
====================
==
=====================
==
==
==
==================
3. Calculations
==================
==================

Total Frame Pixels = <% Image Size (NbRows,)> * <% Image Size (,NbCols)>

(1)Frame Size = <size of video file> / <% Number of Image>

(2)Frame Size =
YUV420   (<totalframepixels> + (<totalframepixels> / 2))
YUV411P   (<totalframepixels> + (<totalframepixels> / 2))
YUV422P   (<totalframepixels> * 2)
YUV411   (<totalframepixels> + (<totalframepixels> / 2))Order
YUV422   (<totalframepixels> * 2)
YUV422_10 (<totalframepixels> * 16 / 6)
YUVA444   (<totalframepixels> * 4)
RGBA   (<totalframepixels> * 4)
RGB   (<totalframepixels> * 3)
RGB_10   (<totalframepixels> * 4)
RGB_10LE (<totalframepixels> * 4)
*NOTE: 720x1280 in YUV422_10 is a special case. Please use a (1) calc for this

(1) Aligned Frame Size = <size of video file> / <% Number of Image>

(2) Aligned Frame Size = (((<Frame Size> + <% Video Alignment> - 1) / <% Video Align-
ment>) * <% Video Alignment>
*NOTE: Above math MUST be integer to work properly

Seek to a frame offset = <required frame number> * <Aligned Frame Size>

==
====================
==
====================
==
==
==
==================
4.   Samples
==================
==================

calendar_01.hdr
-snip---
% Color Format (YUV420, YUV422, YUV422P, YUV422_10, RGB, RGBA)
YUV422P % Planar 422
% Image Size (NbRows,NbCols) (576 720, 486 720, 720 1280, 1080 1920)
576 720 % PAL Size
% Number of Fields per Image
1   % Both fields in single frame group
% Number of Image
12   % If greater then actual, reader will correct
% Frame per second
25   % PAL rate

%% Video Alignment
%512 % no alignment
%% Video Name
% optional e:\calendar_01.yuv
% Audio Format
%Stereo % none, should be wav or aiff
% timecode
00:00:00;00 D
% userbits
FFFFFFFF
% Start Frame
0
-snip---

table_tennis.hdr (mpeg test stream)
-snip---
% Color Format
YUV420
% Image Size (NbRows,NbCols)
288 352
% Number of Fields per Image
1
% Number of Image
250
% Frame per second
25
-snip---

Contact Technical Support

Drastic Technologies Ltd.
523 The Queensway, Suite 102
Toronto, Ontario
M8Y 1J7, CANADA
Phone: (416) 255-5636
Fax: (416) 255-8780
E-Mail: techsupport@drastictech.com
WEB: www.drastic.tv