TELESTREAM

Episode 5.3 Users Guide

Note on License

The accompanying Software is licensed and may not be distributed without written permission.

Disclaimer

The contents of this document are subject to revision without notice due to continued progress in methodology, design, and manufacturing. Telestream shall have no liability for any error or damages of any kind resulting from the use of this document and/or software.

The Software may contain errors and is not designed or intended for use in on-line facilities, aircraft navigation or communications systems, air traffic control, direct life support machines, or weapons systems ("High Risk Activities") in which the failure of the Software would lead directly to death, personal injury or severe physical or environmental damage. You represent and warrant to Telestream that you will not use, distribute, or license the Software for High Risk Activities.

Export Regulations. Software, including technical data, is subject to Swedish export control laws, and its associated regulations, and may be subject to export or import regulations in other countries. You agree to comply strictly with all such regulations and acknowledge that you have the responsibility to obtain licenses to export, re-export, or import Software.

Copyright Statement

This document is part of the software product and, as such, is part of the license agreement governing the software. So are any other parts of the software product, such as packaging and distribution media.

The information in this document may be changed without prior notice and does not represent a commitment on the part of Telestream.

Trademarks and Patents

- Episode is a registered trademark of Telestream, Inc.
- UNIX is a registered trademark of UNIX System Laboratories, Inc.
- Apple is a trademark of Apple Computer, Inc., registered in the U.S. and other countries.
- QuickTime is a trademark of Apple Computer, Inc., registered in the U.S. and other countries.
- Windows Media is a trademark of Microsoft Inc., registered in the U.S. and other countries.
- RealNetworks, RealAudio, and RealVideo are either registered trademarks or trademarks of RealNetworks, Inc. in the United States and/or other countries.

All other trademarks are the property of their respective owners.

MPEG-4 AAC

"Supply of this Implementation of MPEG-4 AAC technology does not convey a license nor imply any right to use this Implementation in any finished end-user or ready-to-use final product. An independent license for such use is required."

MP3

This software contains code from LAME, `http://lame.sourceforge.net/`. "Supply of this product does not convey a license nor imply any right to distribute content created with this product in revenue-generating broadcast systems (terrestrial, satellite, cable and/or other networks.), streaming applications (via Internet, Intranets, and/or other networks), other content distribution systems (pay audio or audio-on-demand applications and the like) or on physical media (compact discs, digital versatile discs, semiconductor chips, hard drives, memory cards and the like). An independent license for such use is required. For details, please visit `http://mp3licensing.com/`."

OGG Vorbis

This software contains code that is ©2010, Xiph.Org Foundation. "THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE FOUNDATION OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,

DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE."

PCRE

PCRE is a library of functions to support regular expressions whose syntax and semantics are as close as possible to those of the Perl 5 language.

Release 7 of PCRE is distributed under the terms of the "BSD" licence, as specified below. The documentation for PCRE, supplied in the "doc" directory, is distributed under the same terms as the software itself.

The basic library functions are written in C and are freestanding. Also included in the distribution is a set of C++ wrapper functions.

The basic library functions

Written by:	Philip Hazel
Email local part:	`ph10`
Email domain:	`cam.ac.uk`

University of Cambridge Computing Service, Cambridge, England.

The C++ wrapper functions

Contributed by: Google Inc.

The "BSD" licence

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of the University of Cambridge nor the name of Google Inc. nor the names of their contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Disclaimer of Warranty on Software

You expressly acknowledge and agree that use of the Software is at your sole risk. The Software and related documentation are provided "AS IS" and without warranty of any kind and Licensor and the third party suppliers EXPRESSLY DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. NEITHER LICENSOR NOR ANY THIRD PARTY SUPPLIER WARRANT THAT THE FUNCTIONS CONTAINED IN THE SOFTWARE WILL MEET YOUR REQUIREMENTS, OR THAT THE OPERATION OF THE SOFTWARE WILL BE UNINTERRUPTED OR ERROR-FREE. FURTHERMORE, THE TERMS OF THIS DISCLAIMER AND LIMITATION OF LIABILITY BELOW DO NOT AFFECT OR PREJUDICE THE STATUTORY RIGHTS OF A CONSUMER ACQUIRING THE SOFTWARE OTHERWISE THAN IN THE COURSE OF A BUSINESS, NEITHER DO THEY LIMIT OR EXCLUDE ANY LIABILITY FOR DEATH OR PERSONAL INJURY CAUSED BY NEGLIGENCE.

Limitation of Liability

LICENSOR AND THE THIRD PARTY SUPPLIERS EXPRESSLY DISCLAIMS ALL LIABILITY FOR DAMAGES, WHATEVER THEIR CAUSE, INCLUDING DIRECT OR INDIRECT DAMAGE, SUCH AS CONSEQUENTIAL OR BUSINESS DAMAGE, AMONGST OTHERS CAUSED BY THE NON-FUNCTIONING OR MALFUNCTIONING OF THE SOFTWARE. SHOULD LICENSOR OR THE THIRD PARTY SUPPLIERS IN ANY WAY BE LIABLE FOR DAMAGES, EITHER AS PER THE TERMS OF THIS LICENSE OR OTHERWISE, THEN THIS LIABILITY WILL IN NO EVENT EXCEED THE AMOUNT PAID BY YOU FOR THE SOFTWARE. SOME JURISDICTIONS DO NOT ALLOW THE LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES SO THIS LIMITATION MAY NOT APPLY TO YOU.

Contents

Document conventions

NOTE

Paragraphs marked like this highlight items of particular importance for the proper function of the software.

TIP

Paragraphs marked like this highlight procedures that can save time or produce particularly good results.

Paragraphs marked like this warn about features which may cause loss of data or failed execution if used incorrectly.

Document references, both internal and external, are shown in italics. Example:
See chapter 2 *Before You Install.*

Literature references are given as numbers in brackets with the full reference in the Bibliography. Example:
See [2].

Directory names, file names, code examples, and prompts, are shown in plain typewriter type. Example:
The file `printer.ppd` can be found in `/etc/cups/ppd/`.

The names of interface components are given in **bold**. Example:
Adjust the time limit with the **Time limit** slider. Select **Quit** from the **Episode Encoder for Windows** drop-down menu.

Keys to be pressed on the keyboard are displayed in bold typewriter type. Example:
Press **`Return`** to select the GUI installation.

Examples of extended dialogue will include the shell `prompt>` .

Command syntax is described in Backus-Naur form.

Copy-pasting from the manual is not guaranteed to work, as the text contains formatting information which may not be accepted by the target application.

1 About **Episode Encoder for Windows**

Episode Encoder for Windows is a *transcoding* tool. This means it takes media files—video and/or audio files—and converts them from one format to another. In addition to changing the storage format, **Episode Encoder for Windows** can also perform other operations on the media, resizing, changing its speed, improving the image quality, etc. For transcodings you can either use the many predefined templates included with the software or define your own transcoding settings.

Episode Encoder Pro

Episode Encoder Pro lets you output additional file formats, mainly useful for broadcast applications. The **Episode Encoder Pro** logotype in the margin indicates features that only are available in **Episode Encoder Pro**.

Even on the standard **Episode Encoder**, you can test **Episode Encoder Pro** features in *demo mode*, limiting your output to half the length of your input, up to a maximum of 30 seconds.

This manual is organised as follows:

- The installation of **Episode Encoder for Windows** and optional extras.
- An overview of terminology and concepts used in the rest of the manual.
- A reference section, describing each user interface element in detail.

Basic familiarity with Microsoft Windows is assumed. Some knowledge of media encoding is helpful.

If you have a question not answered by this manual, please check our support pages at `http://www.telestream.net/telestream-support/episode/support.htm`.

2 Installing **Episode Encoder for Windows**

Episode Encoder for Windows requires QuickTime, version 7 or higher.

Windows XP Running under Windows XP requires Service Pack 2 or higher.

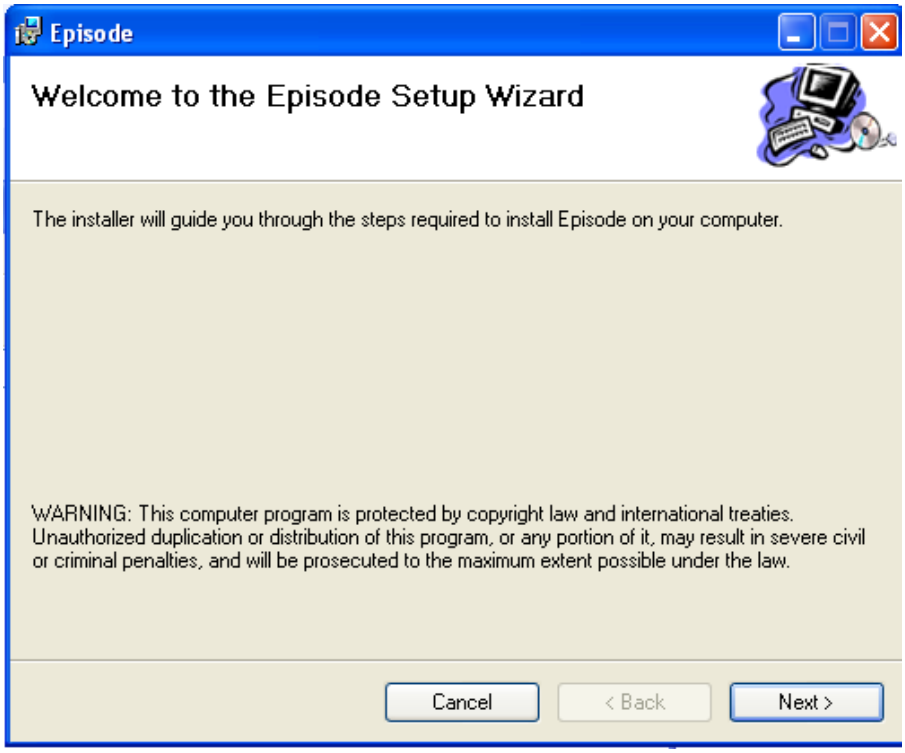
CPU	1 GHz 32-bit x86
Memory	1 GiB RAM
Hard disk	20 GiB
Graphics	1024 × 768 pixels display resolution

Windows Vista Running under Windows Vista requires Vista 32 Business.

CPU	1.5 GHz 32-bit x86
Memory	2 GiB RAM
Hard disk	40 GiB
Graphics	DirectX 9 graphics, Pixel Shader 2.0, 128 MiB graphics memory, 32 bit colour depth, 1024 × 768 pixels display resolution

The **Episode Encoder for Windows** software is delivered in the form of an installation file on CD-ROM or via electronic distribution.

Double-click the installer to start the installation process and follow the onscreen instructions. Any previous version will be uninstalled, but your settings are saved separately from the application and will still be there once you have installed the new version.



2.1 Registering **Episode Encoder for Windows**

Without a license **Episode Encoder for Windows** will run in *demo mode*. This allows you to transcode 30 seconds or half the length of your source file, whichever is shorter.

To register and unlock **Episode Encoder for Windows**, select **Edit→Preferences**, choose the **License** tab and press the **Enter Serial Number...** button to enter your serial number. See also section 4.7.4, *License*.

3 Terminology and concepts

In this chapter we will go through some common concepts in encoding and how they apply to transcoding in **Episode Encoder for Windows**. If you already are familiar with media encoding, you can skip this chapter now, it will be referred back to from the relevant parts of the rest of the manual.

3.1 File formats and codecs

Many media file formats are *wrappers*—they contain video and/or audio data encoded with one of a number of *codecs*. Files may even contain multiple media tracks, for different resolutions, bandwidths, languages, etc. A consequence is that even though a file may be in a format supported by **Episode Encoder for Windows**, the contained media may be encoded with a codec that is not supported, leading to an error during transcoding. Check appendix A, *Supported formats* for the file formats and codecs supported by **Episode Encoder for Windows**.

3.2 Colour formats

The most basic way to represent colour in digital images is to use the *RGB* colour space. In RGB each pixel has three values: red, green and blue, which are mixed to the intended colour. This is the way most computer screens store and display colour.

However, this is not how colour is represented in most video codecs, due to the way television emerged, at first with only black and white images and later with colour images. The colour format was designed to contain all the black and white information in one channel, and the colour information in two additional channels. The black and white channel is called *luma* (light), and the two colour channels are called *chroma* (colour). The separation of luma and chroma made it possible for the older television sets to still work, only picking up the black and white image, while the newer ones could benefit from the colour information.

This colour space is called *YUV*, or *YCbCr*. Y is the luma and Cb and Cr are *colour differences*, the luma subtracted from a transformed blue and red, respectively. (The green values can be computed from these.) This encoding has several advantages over RGB in terms of video compression, since most of the image information ends up in the luma channel and the chroma channels hold much less information for most video material. This, in combination with the fact that human visual perception is less sensitive to colour than to brightness, makes it possible to

sample the chroma more sparsely, thereby reducing the amount of data required to store an image.

The subsampling pattern is commonly given in the notation a:b:c, which is to be interpreted in the following way: In a block of size $a \times 2$ there are $a \times 2$ luma samples. b is the number of chroma samples along the top row and c is the number of chroma samples along the bottom row. In other words, 4:4:4 means there is a chroma sample for each luma sample—no subsampling. 4:4:0 means every second row will have full chroma samples and every second will be skipped. 4:2:2 means every second column will have full chroma samples and every second will be skipped. These two therefore have $2/3$ the number of total samples, compared to 4:4:4. 4:2:0 means every second row will have chroma samples for every second column, thus halving the number of total samples compared to 4:4:4.

The luma and chroma samples are not necessarily taken at the same positions relative to the image pixels, but that is beyond the scope of this manual. The interested reader is referred to [2, 4] for additional information.

3.3 Video scan

Video material can be rendered either a full frame at a time, *progressive scan* or with every second line of the frame at a time, *interlaced scan*.

The two half-frames in interlaced material are known as the top and the bottom fields. One of the fields is dominant and contains the majority of data. The dominant field should always be played back first. When encoding material to be burnt on a DVD and played back on a TV it is important that the dominant field is set as the first field. If you have not edited the material you are going to encode, it can be difficult to know whether the dominance lies in the top or bottom field. The normal field dominances of the more common formats are:

Format	Field dominance
DV 25	Bottom field
DVCPRO 25/50	Bottom field
DVCPRO 100 HD	Top field
IMX	Top field
Apple Intermediate Codec	Top field
Uncompressed 4:2:2	Top field

Much video material is generated from non-interlaced film material, where a single film frame may be sampled several times to generate video fields. This is known as *telecine*. For NTSC material this is typically done by taking 3 fields from one frame and 2 from the next, known as "3-2 pulldown"; for PAL "2-2 pulldown" is the normal. Making use of knowing this sequence of fields, the *cadence*, can greatly improve the quality of deinterlacing. Typically subsequent editing breaks up the cadence, but **Episode Encoder for Windows** detects and adapts to broken cadences. The interested reader is referred to [3] for an extended discussion of deinterlacing methods.

TIP

The preferred way of working with interlaced material in **Episode Encoder for Windows** is to deinterlace any interlaced source material to double-frame rate progressive material, apply video filters to that and then, if needed, reinterlace the material before output. See section 8.4, *Field Order*, section 8.5, *Frame Rate*, section 8.6, *Deinterlace*, section 8.7, *Advanced Frame Rate*, and section 8.18, *Interlace* for further information. You can also study the included settings templates.

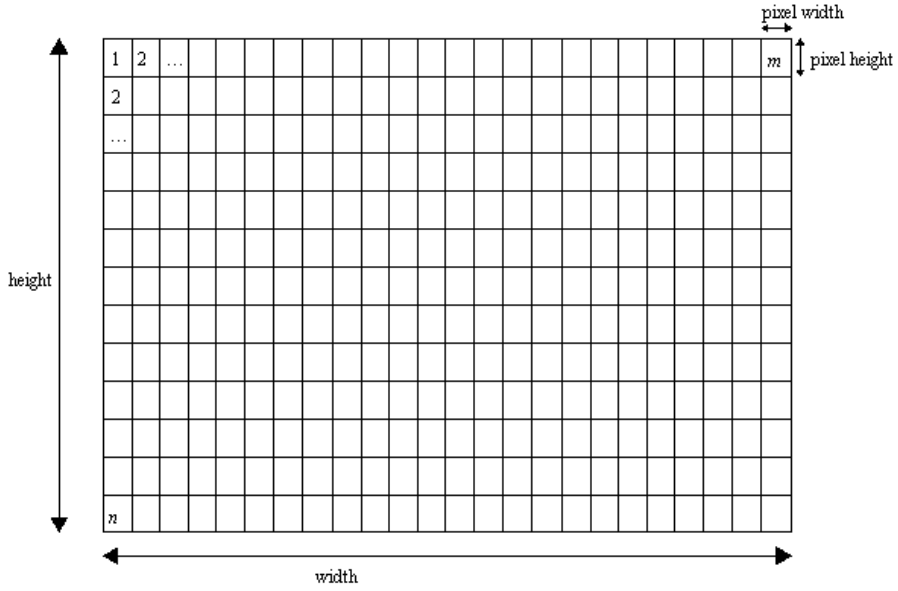
3.4 Frame types—I-, P- and B-frames

Except for raw data formats and editing formats, such as DV or Motion JPEG, most video formats do not simply consist of a sequence of frames, allowing them to be decoded independently. Since a video frame often looks a lot like its neighbouring frames, the video codec searches for differences between frames to achieve a good compression ratio (*temporal compression*). Only the differences are stored in the encoded video file. However, for the encoded stream to be decodable, independent frames, which can be decoded directly, must appear throughout the clip. These frames are called keyframes, or I-frames. To decode a frame at a certain time in the movie the decoder must therefore begin the decoding process at the nearest previous key-frame and decode to the desired frame. Keyframes spaced far apart will make the clip hard to search, but will result in a good compression ratio. Accordingly they are good for streaming material in which searching is not usually done. A compromise is to limit the distance between keyframes to some maximum distance. This is used in most encoded video material.

A frame that predicts data from a previous frame is called a P-frame ("Predictive Frame"). A frame that predicts data from both a previous and a subsequent frame is called a B-frame ("Bi-Predictive Frame"). The use of B-frames will give a somewhat better compression ratio, but is also more CPU intensive.

3.5 Picture resolution and aspect ratio

A digital video frame is a two-dimensional lattice of *pixels*, where each pixel has a given colour.



In the diagram above we have an image lattice of $m \times n$ pixels. We can call this the *pixel resolution* of the image. The *aspect ratio* of the display is *width* : *height*. While computer displays normally have an aspect ratio which is identical to $m : n$, this is often not true for TV sets. In these cases the pixels on the screen are not square, the *pixel aspect ratio* = *pixel width* : *pixel height* $\neq 1 : 1$.

For example, PAL is defined to be 720×576 pixels with a display aspect ratio of $4 : 3$. Since $720 : 576 = 5 : 4$ this means the pixel aspect ratio is $16 : 15$.

SVCD is a video format that is stored on CDs and often played on computers. SVCD stores NTSC video in 480×480 pixels with a pixel aspect ratio of $4 : 3$. The consequent display aspect ratio of $4 : 3$ requires the player software to "stretch" the pixels, interpolating along the horizontal axis to show them on 640×480 square pixels. Compressing dimensions in this way is known as *anamorphic* video.

Some codecs (D-10/IMX, DV, MPEG-2 and MPEG-4) allow the user to set a display aspect ratio for the output video in order to inform a player of the desired aspect ratio for viewing. However, this field is inconsistently used—in particular its $1 : 1$ setting does *not* mean that the display is square, but rather that the pixels have a square aspect ratio. Often this field is also labelled as "pixel aspect ratio". There is a risk that different equipment will interpret this field in different ways.

QuickTime lets you explicitly set the pixel aspect ratio if it is different from the display aspect ratio.

In **Episode Encoder for Windows**, you can use the **Resize** filter (section 8.8, *Resize*) and the codec settings to manage pixel resolution and aspect ratio.

For example, to create an SVCD video: Use the **Resize** filter and set both **Width** and **Height** to 480 in order to get the desired pixel resolution. Set the **Aspect ratio** to **4:3** in the MPEG-2 codec settings to get the desired display aspect ratio.

If the input data have a non-square aspect ratio and this is not indicated in the source file (or if you wish to override this setting) you can tell the **Resize** filter so with the **Input display aspect ratio** menu. Continuing with the example of SVCD video, we convert it to PAL output. The $m : n$ ratio is not same for SVCD

and PAL, so we select **Cut** in the **Maintain proportion with** menu and set **Input display aspect ratio** to **Assume 4:3**. In the codec we set **Aspect ratio** to **4:3** or **Same as Input**.

Finally, to convert from one pixel resolution to another, you have two options: either to scale the smallest dimension to fit the output format and cut off parts of the largest dimension, or to scale the largest dimension to fit and pad the smallest dimension with black.

Example: You start with an HD video of 1280×720 pixels and wish to encode it as a PAL video at 720×576 pixels while retaining as much of the picture as possible. You use the **Resize** filter and set the **Maintain proportion with** menu to **Letterbox (Pad)**, scaling down the width of the picture until it fits. This will shrink the vertical dimension to 324 pixels and the picture will get 126-pixel black borders along the top and bottom as shown in the picture below.

The other alternative is to select **Cut**, where the vertical dimension will be scaled to 576 pixels and the horizontal to 1024 pixels, of which 152 are cut off at both the left and right sides as shown below:

For more precise control you can use the **Initial crop** values to crop parts of the picture before scaling with either cutting or padding.

3.6 CBR, VBR and Quality Based VBR

Constant Bit Rate (CBR), Variable Bit Rate (VBR) and Quality Based VBR are coding options available in several of the video encoders. A clip encoded in CBR mode has a relatively constant bitrate throughout its duration. CBR encoding is necessary when the content will be distributed over networks or from devices that cannot handle peaks that are higher than the average bitrate. However, the use of true CBR, also called Flat Rate, is difficult since it requires every encoded video frame to be exactly equal in size. This is not good for quality. I-frames, for example, must be allowed to be larger in size for the overall quality to be good.

Different segments of a movie need different bitrates in order to maintain constant quality. The quality delivered by most modern video encoders partly depends on the amount of motion and fine detail in the material. For this reason it is a good idea to allow VBR, while keeping the average rate at the desired level. VBR is suitable for playback on devices with less limited bandwidth. Since the average rate is known, it is still possible to predict the resulting file size with good accuracy.

When quality is most important, Quality Based VBR is the best encoding mode. Using this mode you only specify the desired quality of the encoded material. For each part of the clip the encoder will use the bitrate required to reach the specified quality. The size of the resulting file cannot be predicted, since it depends on how difficult the clip is to compress. For example, for the same visual quality, a clip with a newsreader yields a small file while a clip of a football game is quite large.

3.7 VBV—Video Buffer Verifier

To control how large the variations in bitrate are allowed to be when encoding, a *Video Buffer Verifier* (VBV) is used. The buffer size determines over which time the bitrate must be kept constant. **Episode Encoder for Windows** measures the VBV in seconds. A buffer size of 0–5 seconds is considered CBR, everything over 5 seconds is considered to be VBR. During the specified VBV period of time the bitrate may vary, without limits, as long as the average rate in the region is correct. This allows the codec to use higher bitrates for difficult passages and vice versa. A larger VBV will enable the codec to encode difficult passages better, since the bitrate is allowed to peak for a longer period of time. Some specifications may give the buffer size in bits, this is converted to seconds simply by dividing by the bit rate.

Since the buffer size determines how much the bitrate may vary, it sets a constraint on how long a player must buffer before starting playback, to ensure smooth playback without need for re-buffering. In practice the size of the VBV is a trade-off. A large VBV lets the encoder vary the bitrate more freely depending on the difficulty of the current part of the material, still keeping the correct average bitrate. However, the player will have to buffer a larger portion of the clip before playback can be started safely. A small VBV forces the codec to encode at a more constant bitrate throughout the clip. This results in lower quality for difficult passages, but the buffer time for the viewer will be low.

When encoding a clip with Quality Based VBR there is no constraint on the size of the VBV, it is simply ignored. For every part of the movie, the encoder uses the bitrate required to reach the desired quality.

3.8 Frame skip probability—smooth motion vs crisp image

In order to keep the specified bitrate most video encoders vary the quality of the encoded clip. Another alternative supported by some encoders is to skip frames when the bitrate gets too high. This lets the encoder keep a higher quality for each encoded frame, but the motion of the video will not be as smooth. Depending on the material being encoded, smooth motion may be more important than crisp image and vice versa. The frame skip probability controls the tradeoff between skipping frames and lowering quality. A frame skip probability of 1 means that when the encoder has to choose between lowering the quality or skipping a frame, it will skip a frame. A probability of 0 does not mean frames will never be skipped, but that this will only happen when image quality cannot be lowered more.

Frame skip probability is also important to use when creating content for networks with extreme bandwidth limitations such as GPRS, 3G or when streaming over modem. These networks sometimes cannot handle even the slightest peaks over the specified bitrate, and frames can be skipped to avoid this. When streaming to such a device, set the priority towards sustaining the bitrate and sacrificing the frame rate if necessary. When encoding for a less bandwidth limited target platform, such as local playback on a computer, the frame skip probability value can be set lower. This is possible since the data rate from the computer's hard drive is sufficient to handle quite large bitrate peaks.

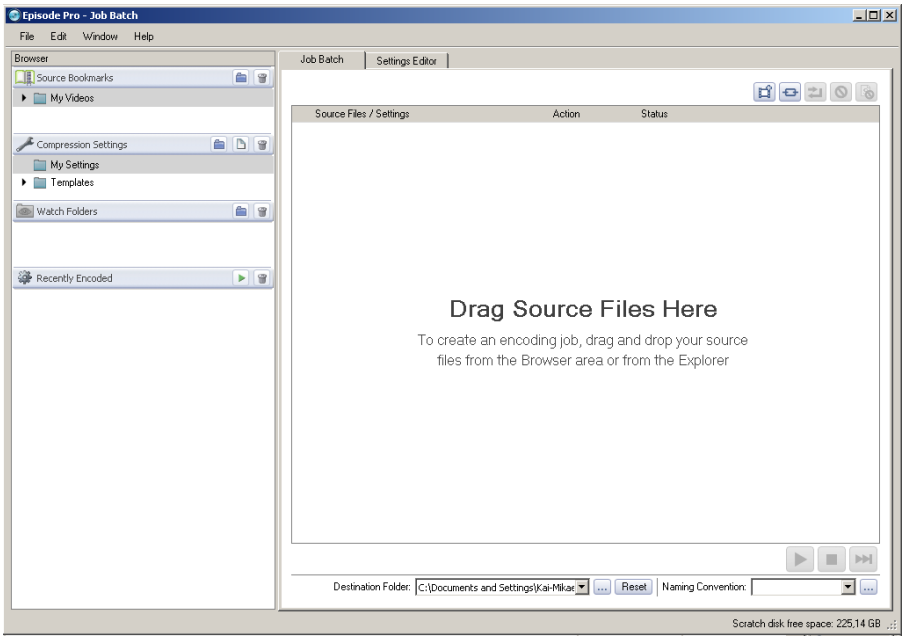
4 **Episode Encoder for Windows** in detail

The basic operation of **Episode Encoder for Windows** is simple: Designate the *source file(s)* you want to transcode, apply a set of transcoding parameters, *settings*, to it/them and perform the transcoding. In addition you can create new settings, edit existing ones, preview the results of settings and set up automatic transcoding of source files.

The following chapters will go through all **Episode Encoder for Windows** features in detail.

4.1 Interface overview

The left side of the main window is the **Browser**, which contains the **Source Bookmarks**, **Compression Settings**, **Watch Folders**, and **Recently Encoded** lists. The area on the right is taken up either by the **Job Batch** showing the files to be, or having been, encoded or the **Settings Editor** allowing encoding settings to be created or modified.



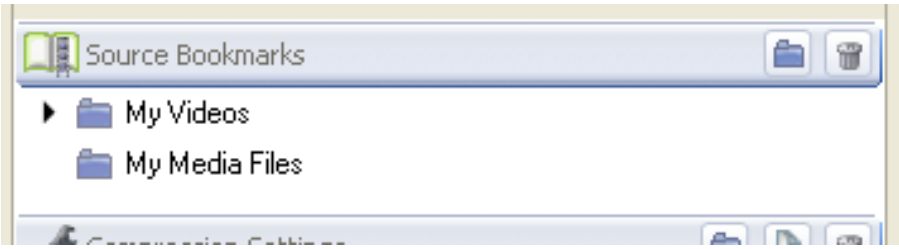
TIP

Place your mouse over any button or window and wait for two seconds without clicking—a tool tip with an explanation of that particular button or window will appear.

4.2 Source files

Source files are the files to be transcoded. All file formats and codecs that can be transcoded by **Episode Encoder for Windows** are listed in appendix A, *Supported formats*. Note that **Episode Encoder for Windows** cannot tell if a given file can be read until you actually start a transcoding.

4.2.1 Source Bookmarks

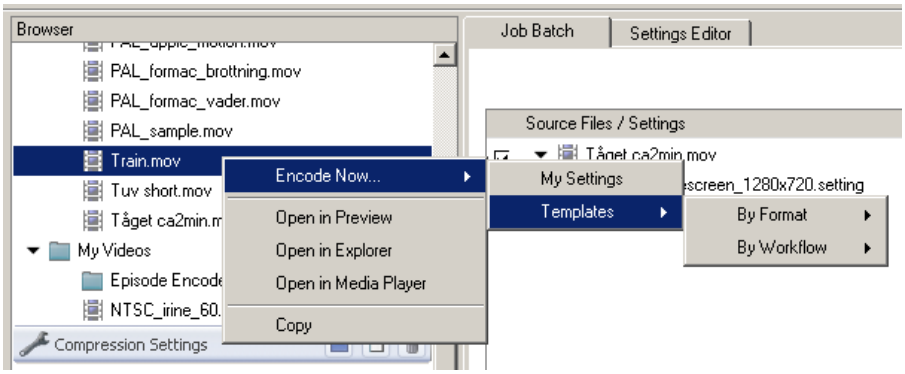


In the **Source Bookmarks** area you keep bookmarks for folders containing your source material. The `My Videos` folder in your system becomes the default bookmark folder when installing **Episode Encoder for Windows**. You can add or delete bookmarks to suit your preferences.

Click the **New Bookmark...** button ( ) to browse to and select a folder to be added to your Source Bookmarks list. In the illustration above a folder named `My Media Files` has been added as a bookmark.

Select a bookmark folder and click the **Remove** ( ) button to remove it from the browser. Note that only the reference is removed, the actual folder will not be affected.

Right-click a file in the **Source Bookmarks** area to bring up a context menu.



Encode Now… Select a setting in the cascade menu and perform a transcoding, see further section 4.5.3, *One-off transcoding*.

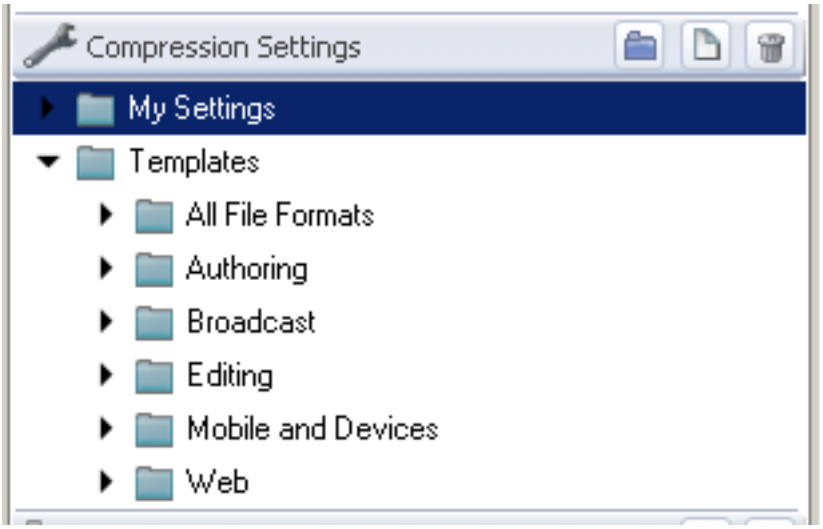
Open in Preview Show a preview of the source video; see section 4.6, *Preview*. Alternatively, click to select the source file and press ***`Enter`***.

Open in Explorer Open the folder where your source file resides in the **Explorer**.

Open in Media Player Launch the source file in **Media Player**.

Copy Copy a reference to the source file; you can then paste it to the **Job Batch**.

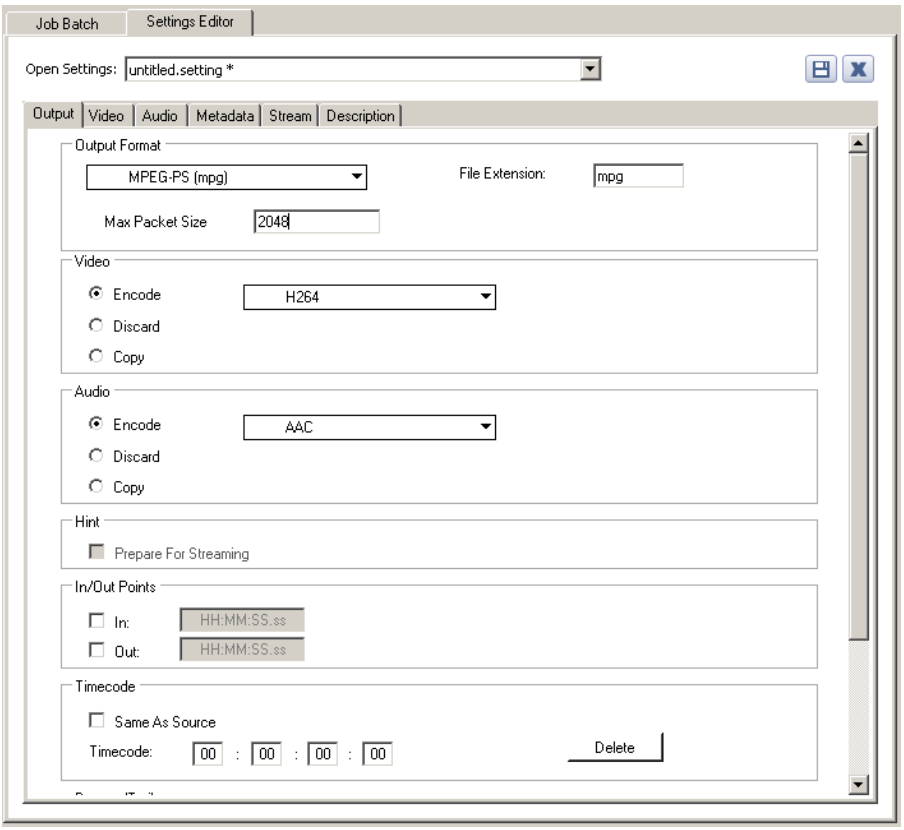
4.3 Settings



Settings are files that determine how your source files are encoded. **Episode Encoder for Windows** comes with a large number of setting templates in the `Templates` folder. They are organised in several subfolders: `All File Formats`, `Authoring`, `Broadcast`, `Editing`, `Mobile and Devices`, and `Web`. A given template may be present in several of the template folders, making it easy for you to find it, whether you are looking for it based on the file format, the device, or the application you intend to use the template for.

Use the template settings to get started with transcoding or edit them to create your own custom settings. Right-click a setting or a whole setting folder and select **Duplicate** in the context menu to make a copy to work with—this way you will always keep the original setting available. To edit a setting, double-click on the setting name; this will switch the main window to the **Settings Editor** tab showing the selected setting.

4.3.1 Settings structure



The **Settings Editor** presents a settings file in several tabs. The **Output** tab determines the output file format and related properties, including the video and/or audio codecs used. Output file formats are described in chapter 5, *Output tab—file formats*, the other file properties are described in chapter 6, *Output tab—others*.

The **Video** tab contains the parameter settings for the video codec selected in the **Output** tab, as well as all the video *filters*. A filter applies a given transformation to the video data, e g, changing the frame rate, smoothing the image, or adding a watermark to the image. Codec settings are described in chapter 7, *Video tab—codecs*, video filters are described in chapter 8, *Video tab—filters*.

The **Audio** tab correspondingly contains audio codec and filter settings. Audio codecs are described in chapter 9, *Audio tab—codecs*, audio filters in chapter 10, *Audio tab—filters*.

The **Metadata** tab contains input fields for *metadata*, data about the contents of the file to be written to the file. Metadata are described in chapter 11, *Metadata*

tab.

The **Stream** tab contains settings for files that are to be *streamed*, transmitted to a player that can play out media at the same rate it arrives. Stream settings are described in chapter 12, *Stream tab*.

The **Description** tab contains an input field to add a descriptive comment to the settings file itself. Descriptions are described in chapter 13, *Description tab*.

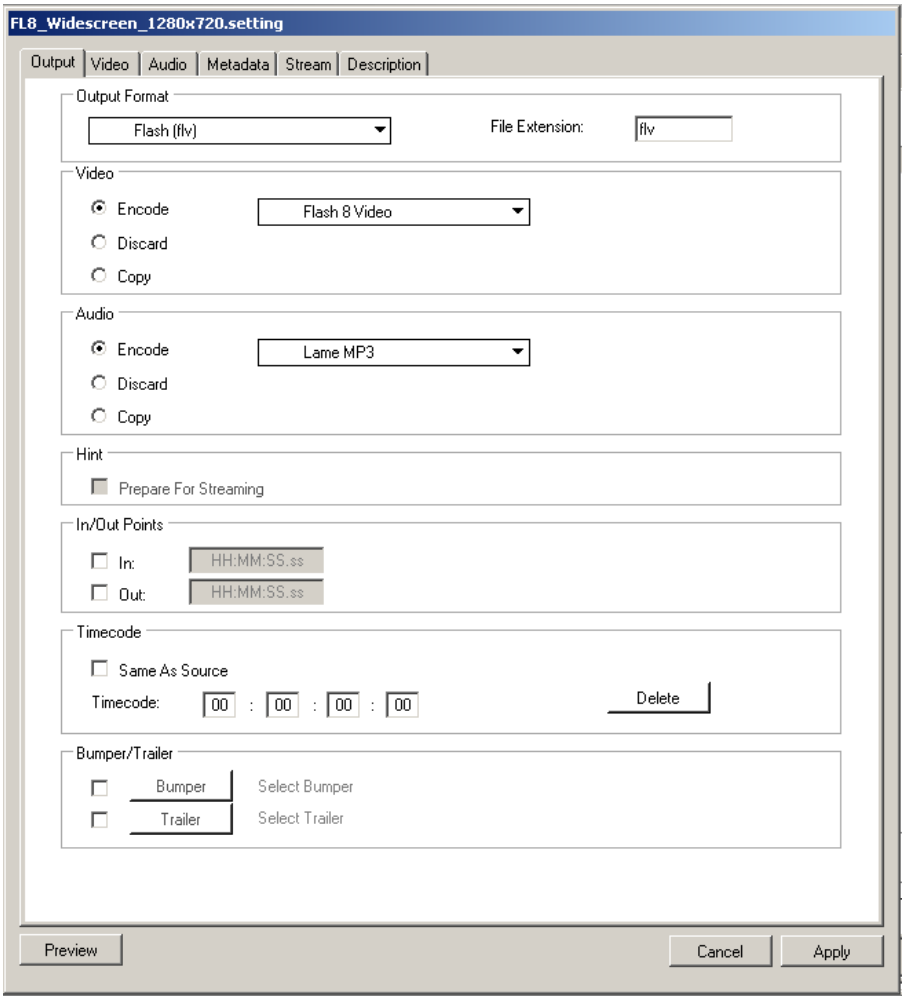
You may edit multiple settings simultaneously and switch between them with the **Open Settings** menu at the top left of the tab.

At the upper right of the tab is a row of control buttons that can be used instead of menu selections:

- Save the current setting.
- Close the current setting. If this is the last setting in the **Settings Editor**, you will be placed in the **Job Batch** after closing.

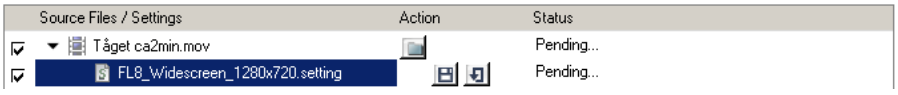
4.3.2 Editing settings in the Job Batch

When you move a setting to the **Job Batch** a *copy* of the setting is created. This means that any changes you later on make in the **Settings Editor** will not affect the setting in the **Job Batch**. However, you can still do changes to that particular setting by editing in the **Job Batch**. Double-click the setting in the **Job Batch** to open an editor window on top of the **Job Batch**.



Click **Apply** when you are satisfied with your setting changes. Your setting in the **Job Batch** is now altered and you can view the changes in the **Preview** (section 4.6, *Preview*). The original setting is not affected by the changes you have made.

If you have applied any changes to your job, two buttons appear in the **Action** column: **Save As** ( ) saves the altered setting to **Compression Settings** under a new name. **Revert to Saved Setting** ( ) reloads the original setting from **Compression Settings** if you want to return to the starting point.



4.3.3 Creating new settings

To create a new setting, select **File→New Setting**, press **`Control-N`**, or press the **New Setting** button ( ) in the **Compression Settings** area. The **Settings Editor** tab will be activated so you can edit your new setting.

TIP

If you edit a template setting to fit your needs, we suggest you save your copy under a new name and/or in a new folder so that you can install a new template pack later without losing your changes.

TIP

If your template settings have been altered or removed you can always recover the default templates with the **File→Restore Templates...** menu option.

4.3.4 Saving settings

To save a new or edited setting, press the **Save** button ( ) at the top right of the **Settings Editor**, press **`Control-S`**, select **File→Save**, press **`Control-Shift-S`**, or select **File→Save As...**.

The settings you create will be stored in the folder **My Settings**. You may create subfolders of this as you see fit.

4.3.5 Duplicating settings

To duplicate a setting, either Right-click the setting in the **Browser** and select **Duplicate** from the context menu, click the setting and select **Edit→Duplicate**, or click the setting and press **`Control-D`**.

4.3.6 Closing settings

To close a setting you have finished working with, click the **Close** button ( ) at the top right of the **Settings Editor**, or select **File→Close Current Setting**.

4.3.7 Creating new settings folders

To create a new folder in the **Compression Settings** area, click the **New Folder** button ( ). To create a sub-folder, select a folder in the **Compression Settings** and click the **New Folder** button. To rename, select the folder and then click on it once to activate the name field and type the folder name of your choice.

4.3.8 Deleting folders and settings

To delete a setting or folder, select it and click the **Delete** button ( ).

4.3.9 Multi Bit Rate (MBR) settings

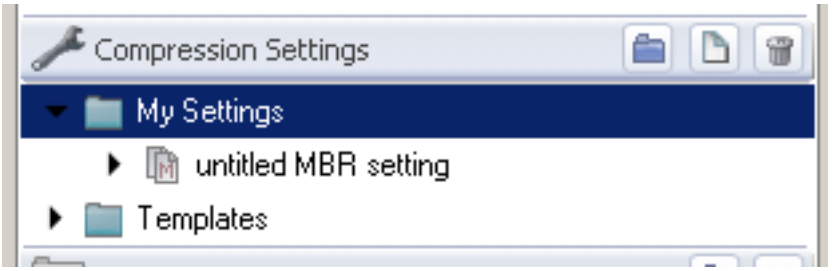
A Multi Bit Rate file is a file containing several media tracks with individual settings for bitrates, framerates, and filters. In this way you can accommodate users with low bandwidth connections as well as those with medium and high bandwidth connections in a single file. In other words, combine your modem setting, ISDN setting, and broadband setting into one MBR setting and transcode the source file once instead of three times. MBR is supported by the following formats:

- 3GPP
- Windows Media (called Intellistream files)

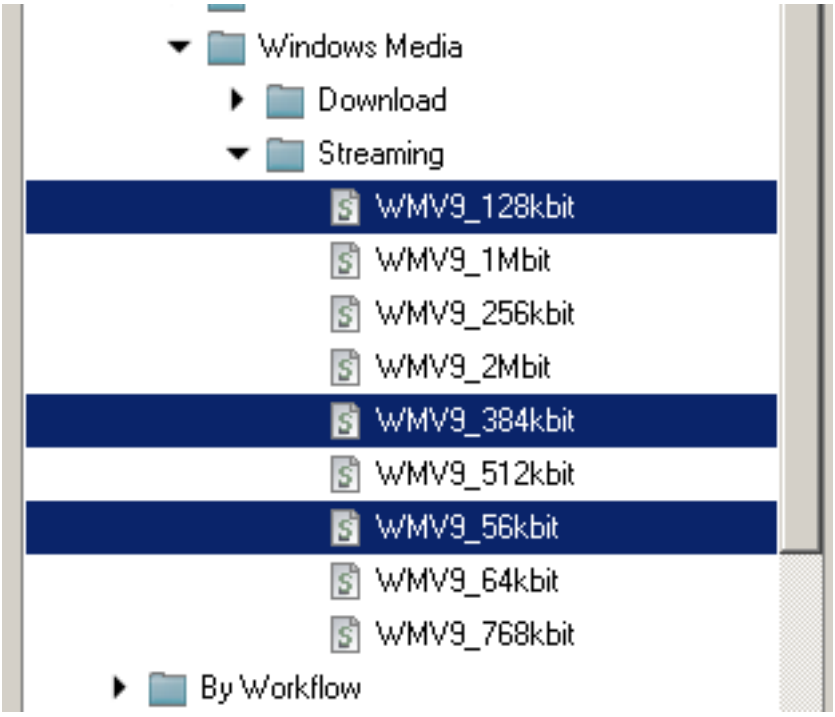
You create MBR settings either in the **Compression Settings** or in the **Job Batch**.

Creating MBR settings in Compression Settings

To create an MBR setting, select **File→New MBR Setting**. If a settings folder has been selected, the new setting will be created in that folder, otherwise it is created atthe top level of **Compression Settings**. The MBR setting can be renamed the same way as a regular setting. Click on the triangle by the MBR setting to expand it and see the individual settings it contains.



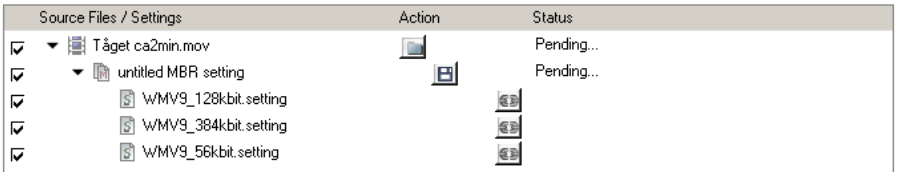
In the example below the `Templates/All File Formats/Windows Media/Streaming` folder is opened. Control-click the settings you want to add and drag them to the empty MBR setting. Only use settings that are intended for streaming. Note that all the settings in an MBR setting must be for the same output format. Windows Media Streaming Server cannot stream Variable Bit Rate-encoded files so files intended for WMSS must use Constant Bit Rate settings only (see section 3.6, *CBR, VBR and Quality Based VBR* for a discussion of Constant and Variable Bit Rate).



Once the settings of your choice have been added to the MBR setting you can start using it.

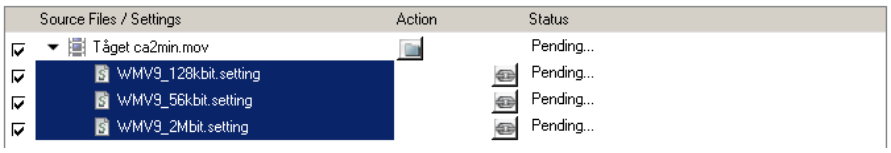
A setting is not editable if it is inside an MBR setting. To edit it, drag it outside the MBR setting, edit it, and drag it back inside the MBR setting.

To use an MBR setting, drop it on a source file in the **Job Batch**.



Creating MBR settings in the Job Batch

Drag streamable settings from **Compression Settings** and drop them on a source file in the **Job Batch**. In this example we use settings from `Templates/All File Formats/Windows Media/Streaming`.



Select the settings, Right-click and choose **Link as MBR Setting** from the context menu. Alternatively, click on one or more streamable settings and any other compatible settings will show a **Link** button in their action column, clicking this will create an MBR setting containing the clicked and the selected settings.

The expanded MBR setting in the **Job Batch** has an **Unlink** button ( ) in the action column next to the setting. Click the **Unlink** button or drag the setting outside the MBR setting to break the association. Drag a setting into the MBR setting to add it to the setting.

The difference between an MBR setting created in the **Job Batch** and one created in **Compression Settings** is that the former only exists in the **Job Batch**. Click the **Save As** button ( ) in the **Action** column to save the MBR setting to the **Compression Settings** list.

NOTE

Since they are composed of multiple settings and therefore cannot be handled as a unit by the **Settings Editor**, MBR settings that have been exported cannot be read back in with **File→Open...** even if they are in **Episode Encoder/Episode Encoder Pro** format, but they can be dragged onto the **Compression Settings** browser from the **Explorer**.

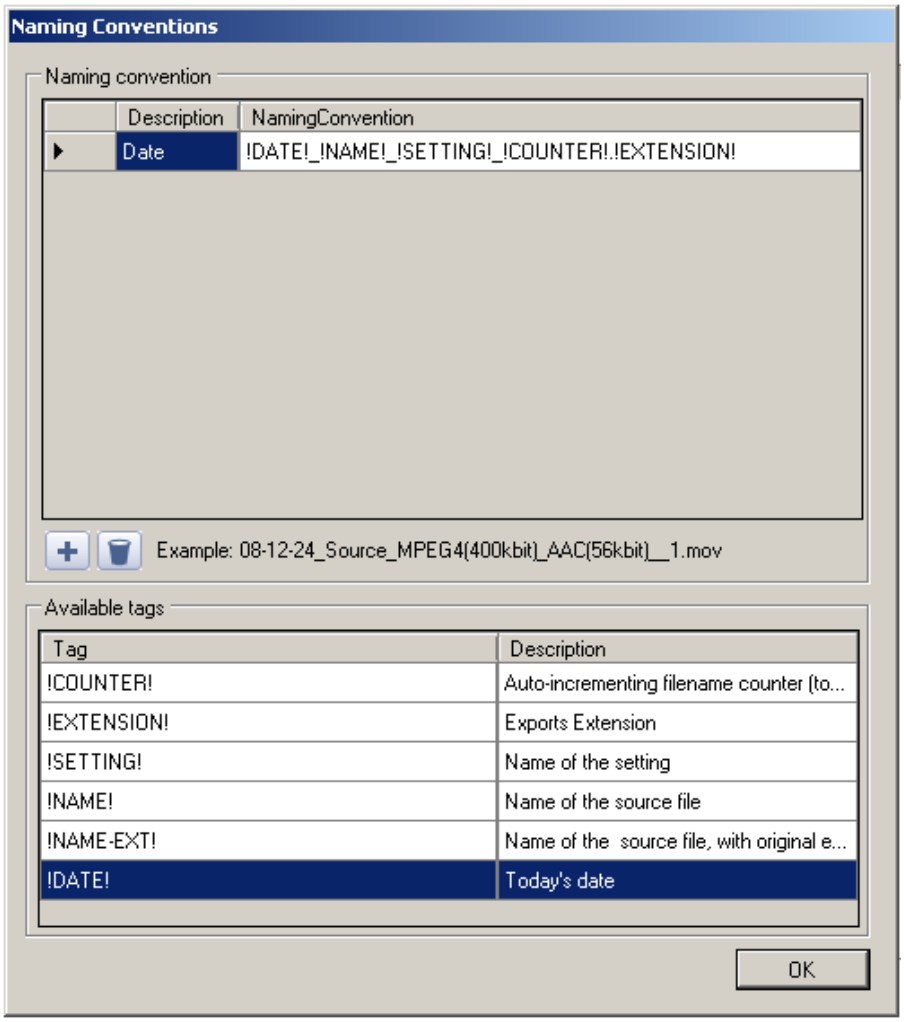
4.4 Output Options

By default, output files will be saved to `My Documents\My Videos\Episode Encoded Videos` (Windows XP) or `Videos\Episode Encoded Videos` (Windows Vista), but you can set any folder in your system to be the output destination. Select alternative destinations with the **Destination Folder** menu at the bottom of the **Job Batch** tab. Press the **...** button to create and/or specify a folder where you want your output files to be placed. The button **Reset** reverts the available destinations to the default alternatives `My Documents\My Videos\Episode Encoded Videos`, `My Documents\My Videos`, and `Desktop` (Windows XP) or `Videos\Episode Encoded Videos`, and `Desktop` (Windows Vista).

To simplify keeping track of your output files, you can define *naming conventions* that determine how the output files are named. There are two predefined naming conventions available in the **Naming Convention** menu at the bottom of the **Job Batch** tab: `Incremental filename` and `Date`.

Using `Incremental filename`, a file named `sourcefile` transcoded into QuickTime with a setting named `setting` would be output to `sourcefile_setting.mov`. The second time a file with the same name is transcoded with the same setting, the output file will be named `sourcefile_setting_1.mov` so as to prevent previously transcoded files with the same name being overwritten. `Date` forms the output filename in the same way, but prefixed with the current date.

You can create your own naming conventions. Press the **...** button next to the **Naming Convention** menu to bring up the editor.



Naming conventions are composed of strings containing `!`-delimited *tags* which are expanded in the following way:

`!SETTING!` The name of the setting.

`!NAME!` The base name (without extension) of the source file.

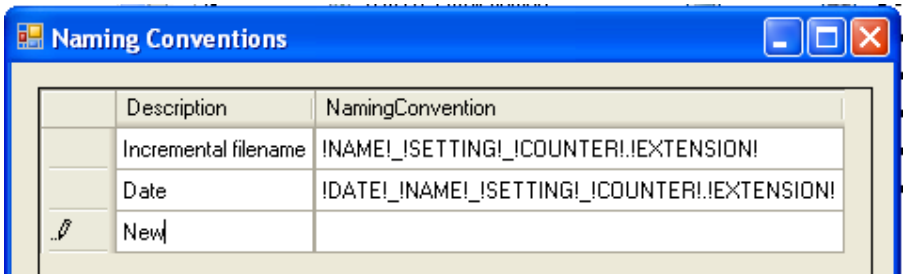
`!NAME-EXT!` The name (with extension) of the source file.

`!DATE!` The current date, in the local date format.

`!EXTENSION!` The extension set for the output file format. If you do not set the correct extension for the output file, you will not be able to play the file from the **Recently Encoded** area.

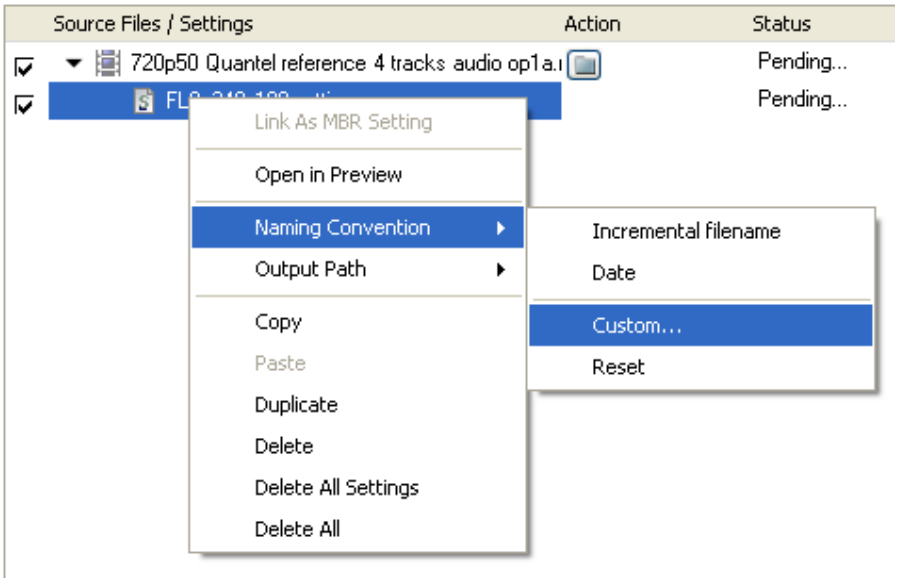
`!COUNTER!` A counter used for files which would otherwise get the same name. For the first file the field is empty, each subsequent file will be numbered consecutively, starting from `1`. If the tag is preceded by – or _, that character will not be used when the field is empty.

Click the **New Naming Convention** button ( ) to create a new naming convention. Then double-click `New` to activate the **Description** field and insert a name for your naming convention.

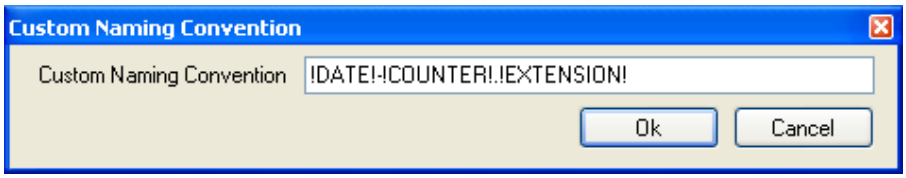


After entering a name click the corresponding **Naming Convention** to activate that field and do the adjustments you need. Close the editor window to make the new naming convention available in the drop-down menu.

Set the naming convention for individual settings by Right-clicking on a setting to bring up a context menu.



Select **Custom...** to edit the naming convention using the tags listed above.



A setting with an individual naming convention is marked with `[Name]`. The setting tooltip shows the setting name as well as the naming convention definition.

4.5 Transcoding

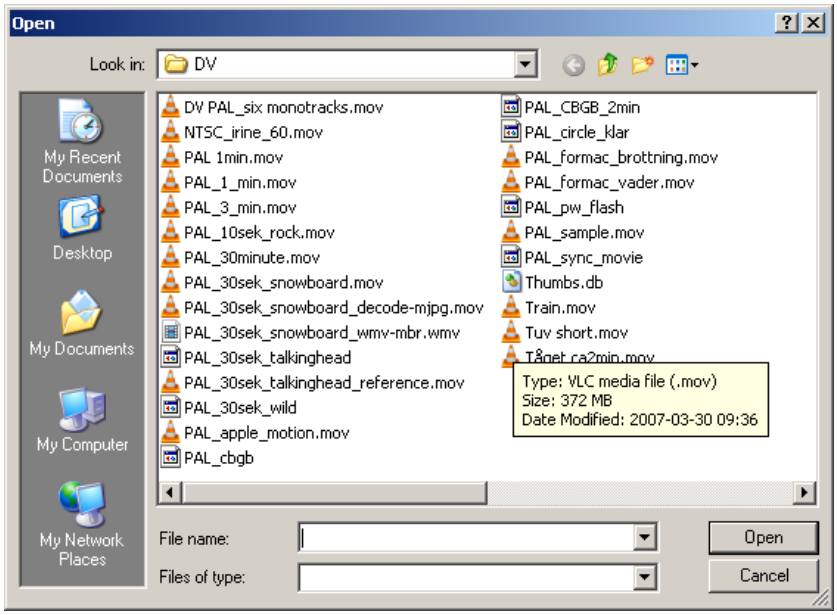
Transcoding is the process of converting your source media into a new format. This is done by applying one or several settings to your source. Transcoding in

the **Job Batch** or using watch folders lets you combine multiple source files and settings files, but you can also do quick one-off transcodings with one source file and one setting.

4.5.1 Transcoding in the Job Batch

There are several ways to add source files to the **Job Batch**:

1. Drag a source file or a whole folder with source files from the **Source Bookmarks** area to the **Job Batch**.
2. Copy one or several source file(s) in the **Source Bookmarks** and paste them to the **Job Batch**.
3. Drag a source file from the **Finder** to the **Job Batch**.
4. Click the **Add Source File(s)** button ( ) at the top right side of the **Job Batch** to bring up a file browser.
5. Right-click the **Job Batch** and select **Add Source File(s)** in the context menu to bring up a file browser.



Adding a setting to a source file in the **Job Batch** forms a *job* and completes the necessary preparations to do an transcoding. The ways to add settings to a source file are:

1. Drag a setting from the **Compression Settings** onto a source file in the **Job Batch**.
2. Copy a setting in the **Compression Settings** and paste it onto a source file in the **Job Batch**.

3. Right-click the source file in the **Job Batch** and select **Add Setting** from the context menu.

NOTE

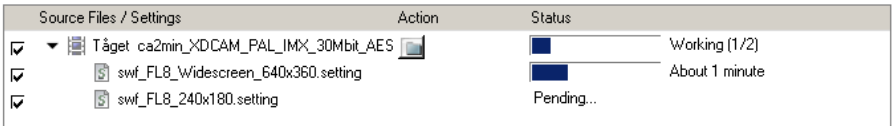
A setting is *copied* to the **Job Batch**, any subsequent edits to the original will not affect the job setting, as explained in section 4.3.2, *Editing settings in the Job Batch*.

TIP

To add multiple settings to multiple files, select the settings you wish to add to the batch and press **`Control-C`** or select **Edit→Copy** to copy them. Then select the specific source files in the **Job Batch** to which you want to paste the settings and press **`Control-V`** or select **Edit→Paste**. The copied settings are pasted onto all the chosen source files. If you want to add the selected settings to all files in in the **Job Batch**, Right-click the settings and choose **Add Setting(s) to Job Batch** from the context menu.

When you have added your settings and files to the **Job Batch** you can start transcoding.

Start transcoding by pressing the **Start Encoding** ( ) button in the lower right corner of the **Job Batch**. The first job in the **Job Batch** starts running, the **Start Encoding** button changes into a **Pause** button ( ) and a progress bar shows the progress of each track being transcoded and how long the transcoding is expected to take.



If the transcoding finishes as expected the progress bar is replaced by the message `Done`. The output file can now be accessed by clicking the **Explorer** button ( ) located in the **Action** column to the left of the **Status** column. If there was a problem with the transcoding, an error message is shown in the **Status** column. If the entire message does not fit in the field, you can keep the mouse over the message to get a tooltip with the full message.

All files in the **Job Batch** tab are preceded by a checkbox. If you do not wish to transcode a given source file, uncheck its box; if you do not wish to use a given setting, uncheck its box. If you have unchecked only some of the settings for a given source file, its checkbox will be marked with a horizontal line.

You can temporarily stop your current transcoding with the **Pause** button or press **`space`** if you need the processing power of your computer for some other application. You can even pause your job and put your computer to sleep, and then resume your transcoding at another time.

Press the **Skip Current Job** button ( ) to move to the next job in the queue; the skipped job will go to the `Stopped` state.

To stop the transcoding of all jobs press the **Stop Encoding** button ( ). The output of files transcoded so far will be left in the destination directory. Files that have only been partly transcoded will remain in the source directory. There is no way to resume transcoding of a stopped job. If you want to run the jobs again, click the **Reset Status** button ( ) at the top right of the **Job Batch**, or select **Reset Status** in the context menu for an individual setting.

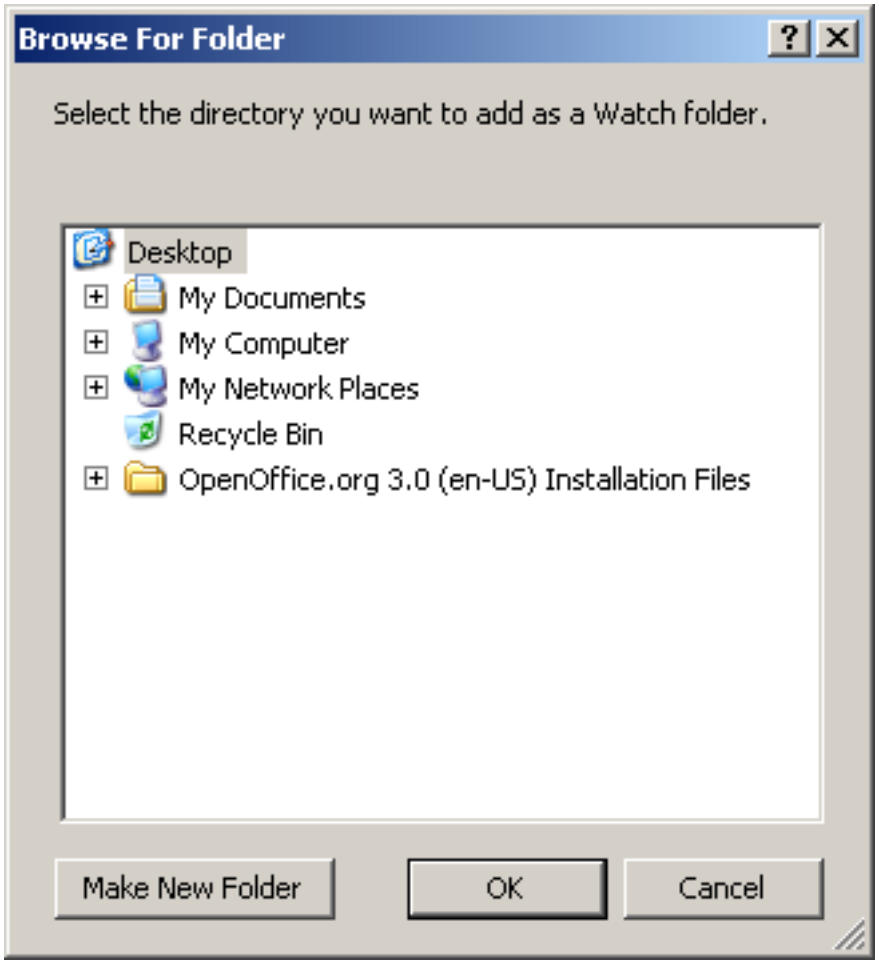
NOTE

The job queue is limited to 25 jobs in **Episode Encoder**. For larger capacity, upgrade to **Episode Encoder Pro** or use **Episode Engine**.

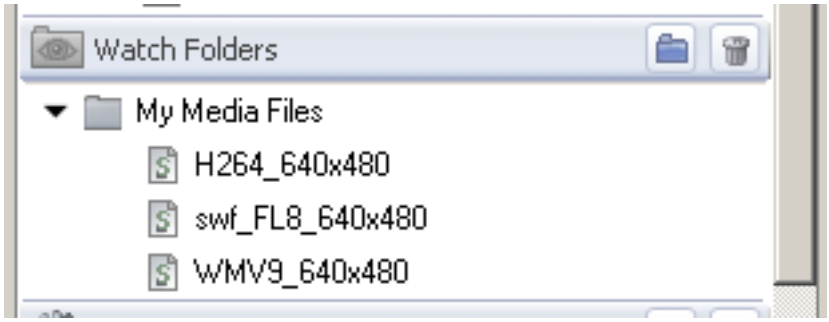
4.5.2 Transcoding with watch folders

Watch folders let you automate transcoding. You assign settings to the watch folder and when a source file is dropped into the folder, it is automatically transcoded with the settings. Note that **Episode Encoder for Windows** has to be running for transcoding to happen.

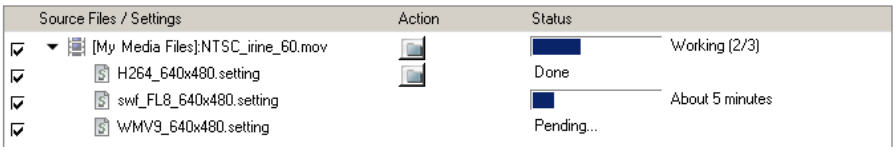
To designate a folder as watch folder, press the **New Watch Folder...** ( ) button in the **Watch Folders** browser. The output files will be named according to the **Naming Convention** currently in force in the Job Batch and be stored in a subfolder of the watch folder named `Output Files`. Source files are always removed from the watch folder and placed in a subfolder named `Processed Input Files`. You can have any number of watch folders, located anywhere in your local file system—mounted external devices may also work, but are not guaranteed to do so.



Add settings by dragging them from the **Compression Settings** browser.



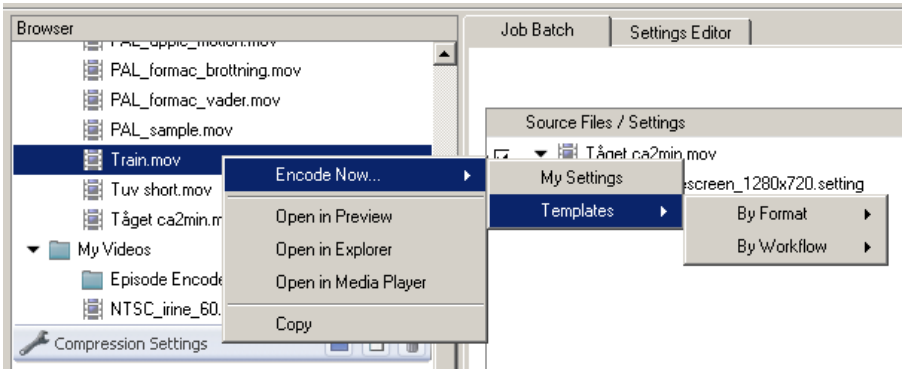
When a source file is dropped into a watch folder it and the settings are displayed in the **Job Batch**. The settings that have not yet been applied to the source can be unchecked, in which case they will be skipped and the source file kept as `Pending...`, so that you later can re-check settings and press **Start Encoding** ( ) to transcode using them. Likewise, if several files are simultaneously added to the watch folder, you can uncheck any file you don't want to transcode right then.



You can skip, pause and stop watch folder-initiated jobs just like any other job in the **Job Batch**.

4.5.3 One-off transcoding

Right-click a file in the **Source Bookmarks** list to bring up its context menu and choose a suitable setting in the submenus of **Encode Now…**. **Episode Encoder for Windows** will ask you for an output file name and the directory to store the output in. The job progress will be shown in the **Job Batch** and the output file shown in the **Recently Encoded** list (see section 4.5.6, *Recently Encoded*).

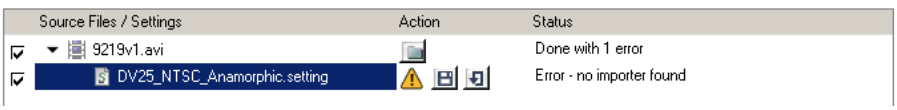


4.5.4 Control buttons

At the upper right of the **Job Batch** tab is a row of control buttons that can be used instead of menu selections:

- Add source files to the **Job Batch**.
- Connect or disconnect one or several settings to or from an MBR Setting.
- Reset the status of your jobs in the **Job Batch** tab, allowing you to start and run your job again.
- Remove all source files and settings from the **Job Batch** tab.
- Remove all settings from **Job Batch**, source files will be left untouched.

4.5.5 Action column



The **Action** column in the **Job Batch** shows icons with information about the current job. Some of the icons are buttons.

For source files, show it in the **Eplorer**. For settings, show the corresponding output file.

In/Out Points have been set in this setting.

The setting has been edited in the **Job Batch**. Click to save the setting under a new name.

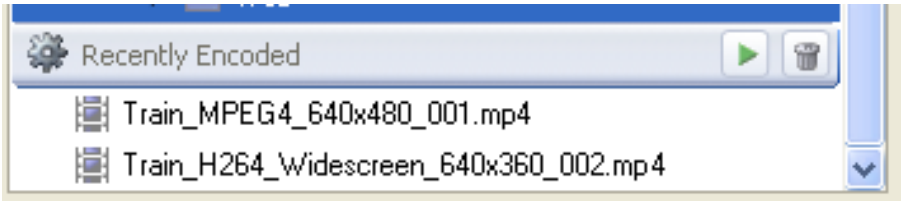
Reload the original setting from **Compression Settings**. This removes all changes made to the setting in the **Job Batch**.

Something is wrong. An error message is shown in the **Status** column.

This setting can be linked into an MBR Setting.

Unlink the setting from an MBR setting.

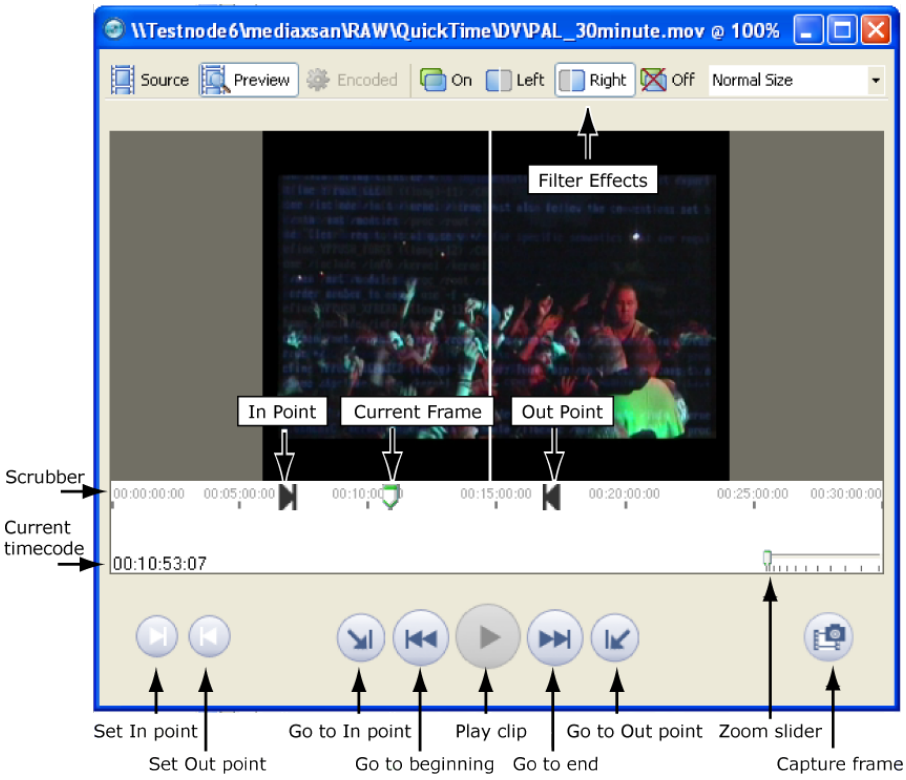
4.5.6 Recently Encoded



The **Recently Encoded** list contains the clips that have been encoded during this session. Click the **Play** button ( ) to play a selected clip, assuming you have software capable of playing files of that format. This list can be hidden by unchecking **Show Recently Encoded files** in the **Preferences** (section 4.6, *Preview*).

4.6 Preview

The **Preview** window lets you see source files and the effects of your settings on them. To open the **Preview** window, select a setting in the **Job Batch** tab or a source file in either the **Source Bookmarks** or **Job Batch** and select **Open in Preview** in the context menu, or press **`Enter`**, or double-click a source file.



The **Preview** window has three tabs, **Source**, **Preview**, and **Encoded**. If you are previewing a source file, only the first tab will be active. It lets you play the video clip with the usual video controls. There is a **scrubber** that lets you pull a marker to the point in the timeline you wish to preview.

By default, the full timeline is displayed, but for long video sequences the resolution may not be enough to let you select the precise point you want. Select a suitable scale with the **zoom slider**. The zoom will be centred around the current point.

If you select a setting for preview, you get the **Preview** tab. This shows what the video will look like if transcoded with the selected setting. In this tab you can only look at single frames of the video and the video controls are greyed out. Additionally you can turn on the display of filter effects: Pressing **On** shows the filtered view over the entire window, **Off** turns off the filtered view. Pressing **Left** or **Right** shows the filtered view in the left or right part of the window; dragging the mouse in the window moves the separator further left or right. Holding down the Ctrl key shows the filtered view as long as the key is held down.

NOTE

In order to make it easier to compare the before/after effects, the effects of the resize and deinterlace filters are always shown, regardless of whether you have turned on filtered display or not.

To save the current preview image as a file, press **`Control-Alt-C`** or click the **Capture Current Frame to Disc** button ( ). You can save the image as a BMP,

GIF, JPEG, PNG or TIFF file.

Finally, if you select a setting that has been transcoded, you can choose between the **Preview** and **Encoded** tabs. The **Encoded** tab lets you play the final result of transcoding.

4.6.1 In and out points

If you have an hour-long clip there is no need to transcode the whole clip every time to test transcodings, so to save time you can use the **Preview** to select a specific part of the source file to transcode. Select the setting in **Job Batch** and open the **Preview**. The buttons at the bottom of the **Preview** window let you set In and Out points.

- Set the in point to the current time in the scrubber. You can also press **`i`**. To remove the in point, press **`Alt`** while pressing this button, or press **`Alt-i`**.
- Set the out point to the current time in the scrubber. You can also press **`o`**. To remove the out point, press **`Alt`** while pressing this button, or press **`Alt-o`**.
- Go to in point. Brings you to the position of your in point. You can also press **`Shift-I`**.
- Go to the beginning of the clip.
- Play the clip from the current point.
- Go to the end of the clip.
- Go to out point. Brings you to the position of your out point. You can also press **`Shift-O`**.

The in/out points are indicated by and icons in the scrubber. They cannot be dragged, but are set by selecting a point in the scrubber and pressing the appropriate key/button. If you have set in/out points in the **Preview** a clock icon ( ) is shown in the **Action** column next to the setting in the **Job Batch**. If you have added several settings to a source file in the **Job Batch** and want to apply the same in/out points to all of them, then create in/out points in one of your settings, Right-click that setting to bring up the context menu and select **Copy In/Out to Siblings** to copy the in/out points to all the other settings for that source file.

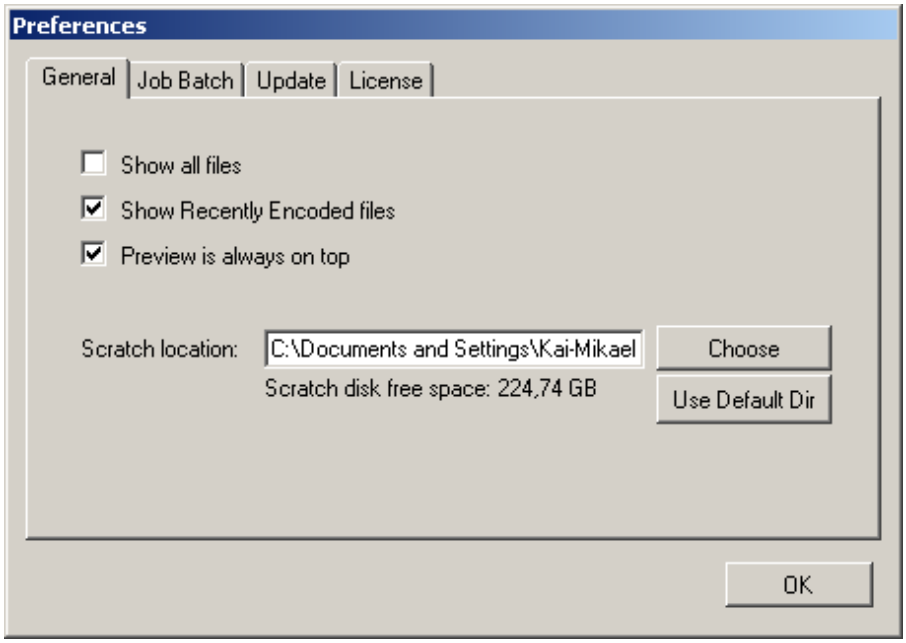
NOTE

If you set in and out points in the **Preview** they are saved only in that job in the batch and do not affect the original setting. To permanently save in/out points in a setting, open the setting in the **Settings Editor** and add the in/out point values there, see section 6.4, *In/Out Points*.

4.7 Preferences

Open the **Edit→Preferences** dialogue to adapt **Episode Encoder for Windows** to suit your requirements. The **Preferences** window is divided into tabs: **General**, **Job Batch**, **Update** and **License**.

4.7.1 General



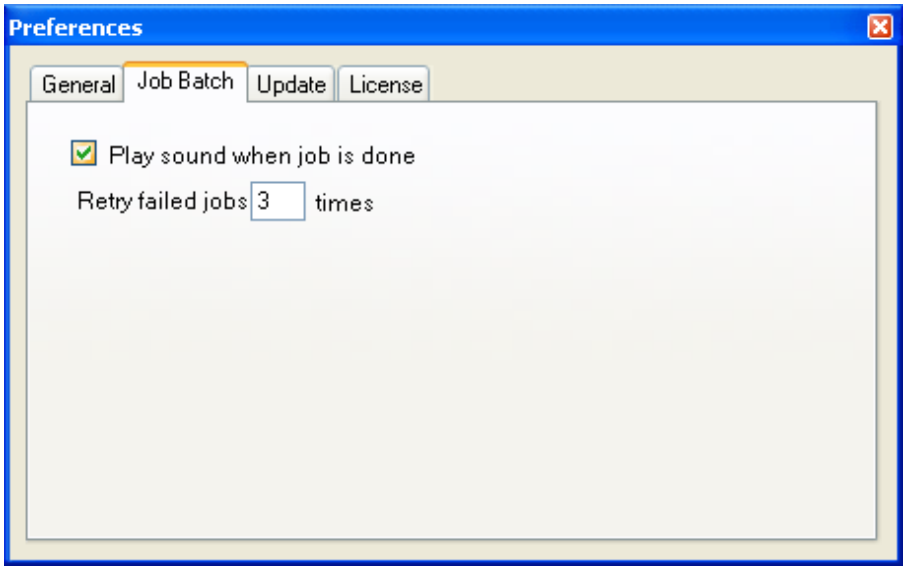
Show all files Show all files in file browsers. This is the default. If unchecked, only files recognised as media files based on their extensions will be shown in file browsers.

Show Recently Encoded files Show the list of recently transcoded files, as explained in section 4.5.6, *Recently Encoded*.

Preview is always on top The **Preview** window will always be on top, regardless of any other application in use. The default is not to put **Preview** on top.

Scratch location Set the location of the temp directory where temporary files are placed during transcoding. As these may become quite large, the available space on the disk partition containing the temp directory is displayed.

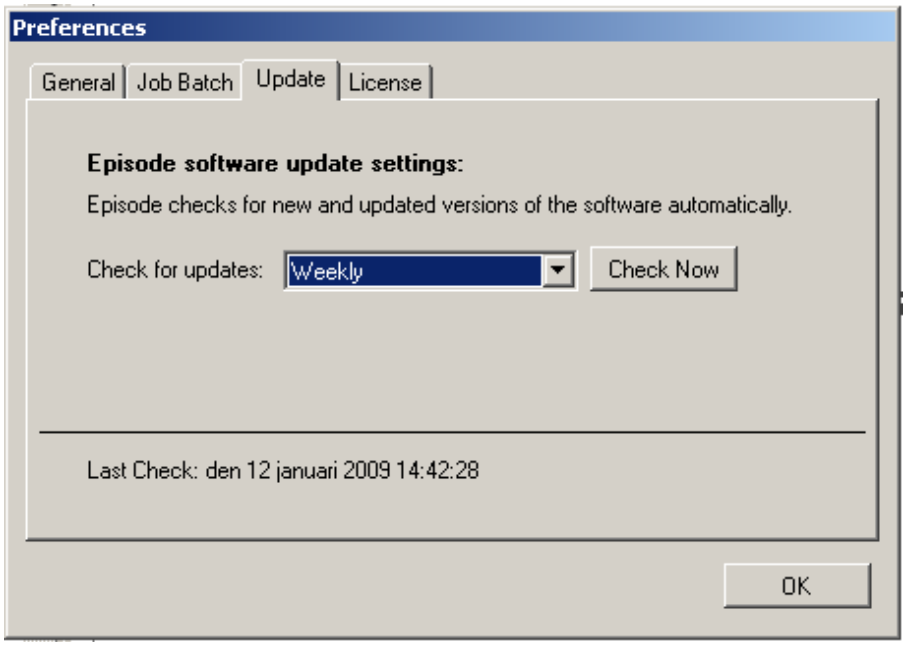
4.7.2 Job Batch



Play sound when job is done Play sound when job is done.

Retry failed jobs Set the number of times for **Episode Encoder for Windows** to retry a file that failed to transcode properly. A job failure is not necessarily caused by an error in the source file or the transcoding setting. A failure can be due to a loss of power, memory issues, intermittent network problems, etc and resolve itself on a renewed attempt.

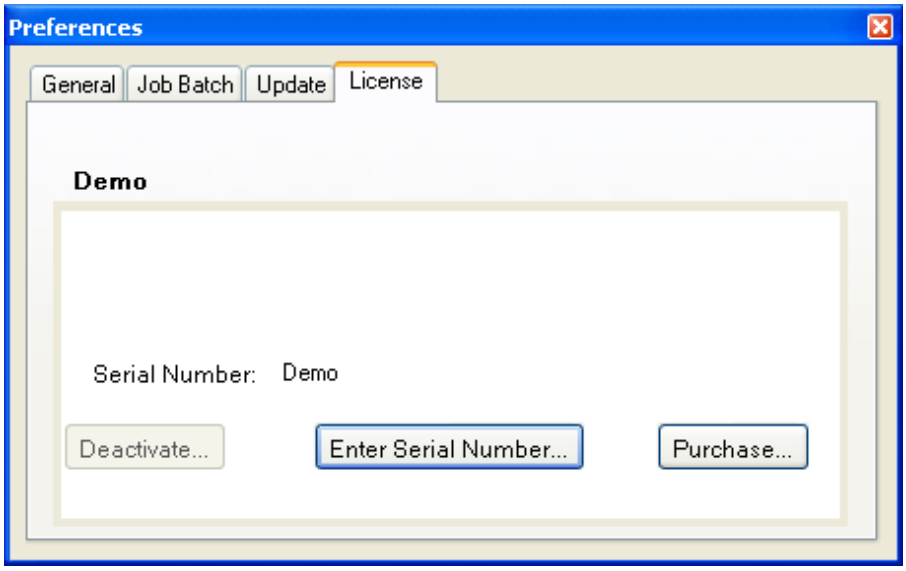
4.7.3 Update



Episode Encoder for Windows can automatically check for updated versions. Select the frequency for how often it will perform the check or perform the check manually.

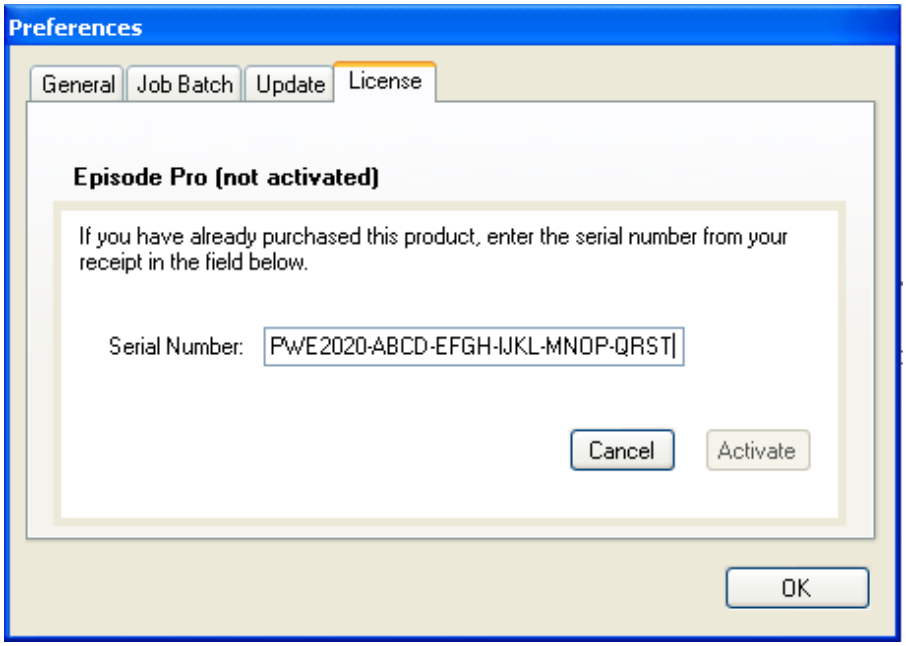
4.7.4 License

Even without a license you can do transcoding in **Episode Encoder for Windows** of half the length of the source material, up to a maximum of 30 seconds.



Press **Purchase...** to open the webshop. When the purchase is completed you will receive an email with your serial number. Press the **Enter Serial Number...** button, copy your serial number from the email message, paste into the window displayed, and press **Activate**. **Episode Encoder for Windows** is now fully functional.

When you activate your license, the **Enter Serial Number...** button changes into **Change Serial Number...**. If you later upgrade to **Episode Encoder Pro**, you enter the new serial number here.



If you need to move your license to a new computer, press **Deactivate…** to deactivate the license on the old computer. You can then install **Episode Encoder for Windows** on your new computer and enter your serial number as before.

4.8 Keyboard shortcuts

Key combination	Effect
space	Run or pause the current transcoding
Control-0	Set **Preview** window to half size
Control-1	Set **Preview** window to normal size
Control-2	Set **Preview** window to double size
Control-3	Fit **Preview** window to screen size
Control-A	Select all
Control-C	Copy the selected item
Control-Alt-C	Write current **Preview** image as file
Control-D	Duplicate the selected item
Control-N	Create a new setting
Control-O	Open a stored setting
Control-S	Save the active setting
Control-Shift-S	Save the active setting under a new name
Control-V	Paste the most recently copied item
Delete	Remove a file from the **Browser** or **Job Batch**

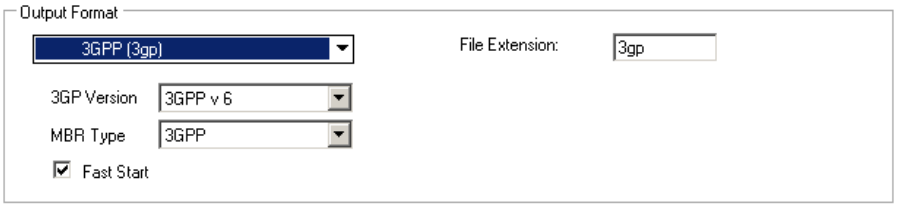
5 Output tab—file formats

The output file format is set in the **Output** tab of the **Settings Editor**. Each format has a default **File Extension**, but you can change it if needed.

Episode Encoder Pro

Some file formats are only available in **Episode Encoder Pro**. They will be marked with Pro in **Episode Encoder**. You can still use them in demo mode, encoding half the length of your input, up to a maximum of 30 seconds.

5.1 3GPP (3gp)



3GPP is a format intended for mobile phones. Flag `.3gp` files as either **3GPP v 5** or **3GPP v 6** depending on the target player. The safest option is 3GPP version 5, as there are older mobile phones that cannot handle version 6. Most version 6 phones, on the other hand, are compatible with version 5.

MBR Type is either **3GPP** and **Vidiator**. See section 4.3.9, *Multi Bit Rate (MBR) settings* for further information on Multi Bit Rate streams, see `http://www.vidiator.com/` for additional information on the Vidiator codecs.

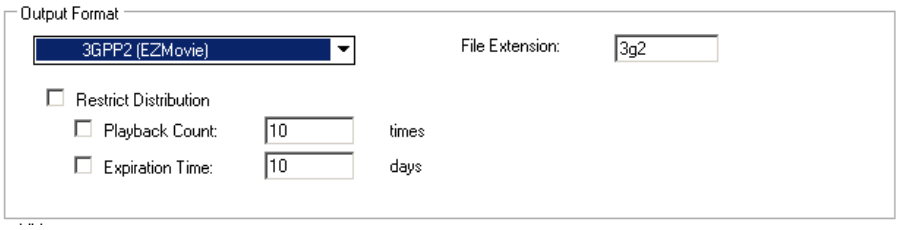
Fast start adds information that lets a player start playing the file before all of it has been downloaded (progressive download). This is an alternative to streaming.

5.2 3GPP2 (3gp2)

3GPP2 is a format for mobile phones similar, but not identical, to the 3GPP for-

mat. It has no settings.

5.3 3GPP2 (EZMovie)



Episode Encoder Pro

3GPP2 is a format for mobile phones similar, but not identical, to the 3GPP format. The EZMovie features let you **Restrict distribution** of a downloaded file by limiting the **Playback Count** to a chosen number of times and/or set the **Expiration Time** of the file to a specified number of days.

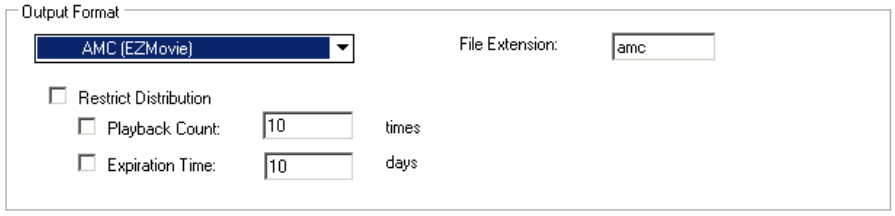
5.4 ADTS (aac)

ADTS (Audio Data Transport System) is a wrapper format for AAC-encoded audio data. It has no settings.

5.5 AIFF

AIFF (Audio Interchange File Format) is an audio-only file format. It has no settings.

5.6 AMC (EZMovie)



Episode Encoder Pro

AMC is a 3GPP variant mainly used in Japan. AMC lets you **Restrict distribution** of a downloaded file by limiting the **Playback Count** to a chosen number of times and/or set the **Expiration Time** of the file to a specified number of days.

5.7 AMR

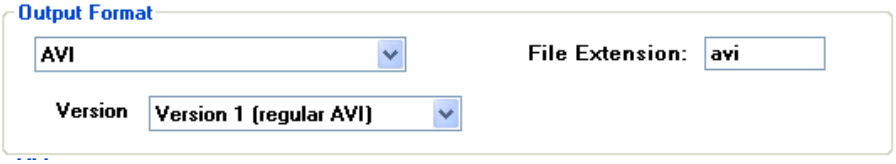
Episode Encoder Pro

AMR (Audio Multi-Rate) is an audio-only file format. It has no settings. It requires **Episode Encoder Pro**.

5.8 ATSC A/52

ATSC A/52 (Advanced Television Systems Committee standard A/52) is an audio-only file format. It has no settings.

5.9 AVI



AVI is a wrapper format that comes in two versions: **Version 1 (regular AVI)**

and **Version 2 (large files)**. Version 2 is not supported by all players but allows encoding of source files larger than 4 GiB.

5.10 DV-Stream (dv)

DV (Digital Video) is a wrapper format for its own video and audio codecs. It has no settings.

5.11 Flash (flv)

Flash Video is a wrapper format for video intended to be played by **Adobe Flash Player**. It has no settings.

5.12 Flash (swf)

SWF (Small Web Format) is a wrapper format for video, audio, interactive functions etc, intended to played in **Adobe Flash Player** or web browser plugins. **Episode Encoder for Windows** will only create video and audio tracks for SWF. It has no settings.

5.13 GXF

Episode Encoder Pro

GXF (General eXchange Format) is a data transfer format developed by Grass Valley. It has no settings.

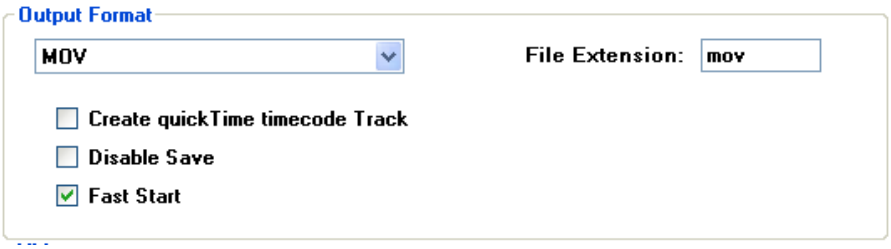
5.14 iTunes Audio (m4a)

iTunes Audio is AAC audio in an MPEG-4 wrapper. It has no settings.

5.15 iTunes Video (m4v)

iTunes Video is an MPEG-4 wrapper intended for playing in **iTunes**. It has no settings.

5.16 MOV



QuickTime is a wrapper format for many codecs.

Create QuickTime Timecode Track together with the **Timecode** option (see section 6.5, *Timecode*), creates a QuickTime timecode track. This track can then be toggled on or off in **QuickTime Player**.

Disable Save prevents any transcoding of your file to other formats.

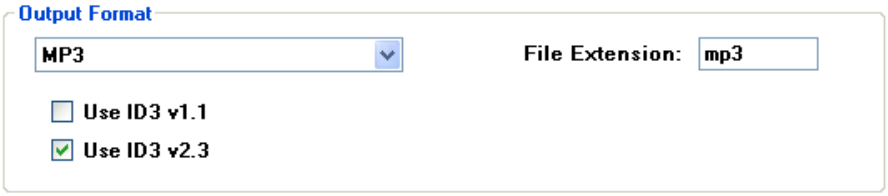
Fast start adds information that allows a player to start playing the file before all data have been downloaded (progressive download). This is an alternative to streaming.

You can use all third-party QuickTime codec plugins in your system, so you can quickly extend the encoding abilities of **Episode Encoder for Windows** by downloading and installing plugins. You access the QuickTime-specific codecs by using the **QuickTime** video and audio codecs (see section 7.13, *QuickTime* and section 9.12, *QuickTime*).

NOTE

There is a bug in **QuickTime Player** in that it fails to increment the timecode for the first B-frame in a video. The timecode will therefore seem to be out of synch, but no frames are skipped.

5.17 MP3



MP3 (MPEG-1 Audio Layer III) is an audio-only file format. **Episode Encoder for Windows** supports ID3 v1.1 and v2.3 metadata tags for MP3 files. Check **Use ID3 tag v 1.1** and **Use ID3 tag v 2.3** as needed. See chapter 11, *Metadata tab* for more information on metadata and `http://www.id3.org/` for more information on valid values for ID3 tags.

5.18 MP4

The MPEG-4 file format is the latest version of the MPEG standards. It has no settings. Note that there is also an MPEG-4 *codec*, which can be used in other formats (section 7.12, *MPEG-4*).

Episode Encoder for Windows adds ODSM (Object Descriptor Stream) and SDSM (Scene Description Stream) tracks to MPEG-4 files, as they are required by some players.

5.19 MPEG Audio (.m1a)

An MPEG Elementary stream contains a single medium, video or audio. MPEG-1 Audio is an audio-only Elementary Stream. It has no settings.

5.20 MPEG-ES (m1v)

An MPEG Elementary stream contains a single medium, video or audio. `m1v` is a video-only MPEG-1 Elementary Stream. It has no settings.

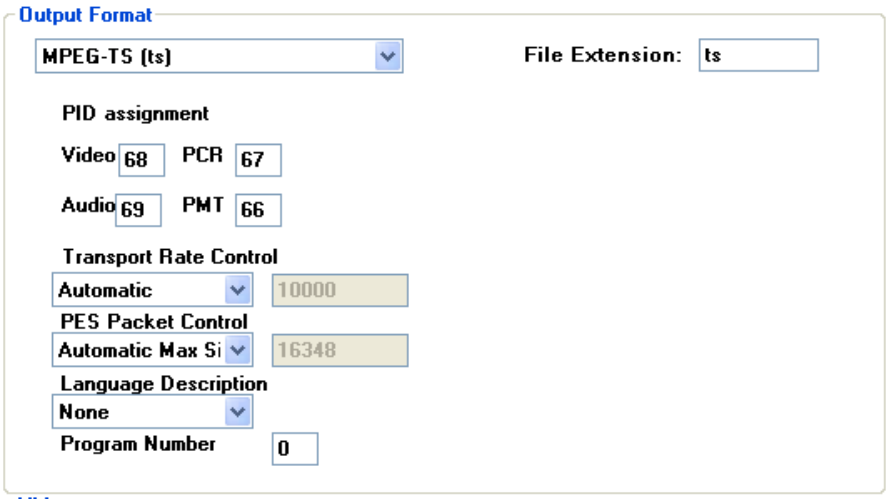
5.21 MPEG-ES (m2v)

An MPEG Elementary stream contains a single medium, video or audio. `m2v` is a video-only MPEG-2 Elementary Stream. It has no settings.

5.22 MPEG-PS

An MPEG Elementary Stream (MPEG-ES) contains only a single medium, video or audio. A Program Stream (MPEG-PS) interleaves (multiplexes) video and audio data. The **Mux Packet Size** can be set in the range 300 to 5000 bytes.

5.23 MPEG-TS



Episode Encoder Pro

An MPEG Elementary Stream contains only a single medium: video or audio. A Transport Stream (MPEG-TS) is intended for broadcast media, so it interleaves (multiplexes) video and audio data with headers that let receivers pick up an ongoing transmission.

PID assignment For the receiver to know which video or audio stream a packet belongs to, each packet contains a PID (Packet IDentifier). The default values are 68 for the **Video** PID and 69 for the **Audio** PID, but some applications may require you to use other values. A PID can be set in the range 16 to 8190, but video and audio streams should not use values below 32.

The timing of media is controlled by sending Program Clock Reference (PCR) packets. The **PCR** field specifies the PID of these packets. The PID can either be identical to that of a media track (typically the Video PID) or have a separate value. The PCR PID cannot be the same as the PMT PID.

The structure of the tracks within a program in the transport stream is described by sending Program Map Tables (PMTs) for each program. Since **Episode Encoder for Windows** creates single program transport streams only one PMT needs to be specified. The PID for these packets is set with the **PMT** field.

Transport rate control The transport rate determines the bit rate of the resulting stream and can be set to either of two values:

Automatic The automatic rate is the sum of the rates of the contained media tracks plus approximately 10% overhead for system packets. This is the lowest possible rate for the stream.

Manual The manual rate lets you specify any bit rate for the stream. If the sum of the rates of the contained media tracks and system packets is lower than the specified bit rate, the stream will be padded up to the specified bit rate. If the total rate is higher than the specified rate, the rate will be silently adjusted up to the lowest possible rate just as for **Automatic**. The bit rate can be given in either **bps**, **kbps**, or **Mbps**.

PES packet control Packetized Elementary Stream (PES) describes how the media tracks (elementary streams) are stored within the transport stream. **Episode Encoder for Windows** provides three ways for controlling the size of the PES packages:

Automatic Max Size The maximum size of the PES packets is set automatically.

Manual Max Size The maximum size of the PES packets is set manually in the range 1024–65535.

One PES per frame Each frame of video is stored in exactly one PES packet. Storing the video in this way is required in order to be compatible with some video-on-demand services.

Language Description If a language is selected, an ISO-639 (Part 2) language descriptor is added to the audio track of the stream. If **None** is selected, no language descriptor will be added to the stream.

Program number The number for the single program stream within this transport stream in the range 1–65535.

5.24 MXF Op1a

The *Material eXchange Format* is a wrapper format that may contain a number of media streams and metadata information. There are several different implementations of MXF. Op-1a corresponds to the SMPTE 378M standard. The Thomson Grass Valley K2 video server nominally uses MXF Op-1a, but adds proprietary modifications to the standard. Checking **K2 server compliant op1a** generates a K2 server-compatible output file.

5.25 MXF OpAtom

Op-Atom corresponds to the SMPTE 390M standard. It is similar to MPEG Elementary streams, in that only either a video track or an audio track can be wrapped, but not both at the same time. Avid video editing systems nominally use MXF Op-Atom, but add proprietary modifications to the standard. Checking **Avid compliant OpAtom** generates an Avid-compatible output file.

5.26 MXF XDCam

Episode Encoder Pro

The Sony XDCAM video camera has its own version of MXF files. The format has no settings.

5.27 OGG (.ogg)

OGG is an open-source wrapper format. It has no settings.

5.28 PSP (mp4)

The PlayStation Portable can play MPEG-4 files, but requires the files to be named `M4Vxxxxx.mp4`, where `xxxxx` is five decimal digits, and stored in the directory `E:\MP_ROOT\100MNV01` on the PSP. **Episode Encoder for Windows** does not enforce this, you have to rename the files yourself. The format has no settings.

6 Output tab—others

Once you have selected an output file format, there are additional controls on the **Output** to further determine the contents of the output.

6.1 Video

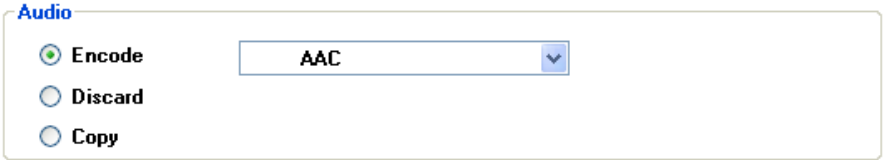
The **Video** section determines how the video track is processed.

Encode Select a video codec for the video track in the drop-down list. Video codecs that are incompatible with the chosen file format are greyed out in the list.

Discard Ignore the source video track in the encoding. No video track is encoded.

Copy Copy the video track without transcoding, if the input source format is compatible with the output format.

6.2 Audio



The **Audio** section determines how the audio track is processed.

Encode Select an audio codec for the audio track in the drop-down list. Audio codecs that are incompatible with the chosen file format are greyed out in the list.

Discard Ignore the source audio track in the encoding. No audio track is encoded.

Copy Copy the audio track without transcoding, if the input source format is compatible with the output format.

Copying audio from AVI, MPEG-4, and QuickTime files does not work—the output audio will be stored as uncompressed 32-bit audio, which cannot be used in all output formats, or, at the least, will greatly increase the size of your output file.

6.3 Hint

To stream QuickTime- and MPEG-4-based files from a streaming server you have to *hint* the file, i e packetize the tracks of the file so that a streaming server can send the information as a stream viewable in real time. If the selected video and audio codecs are streamable, the button **Prepare For Streaming** is enabled. Check the button to add packetizer(s) and prepare the file for streaming. This also activates the **Stream** tab where you set the packetizer parameters for your video and audio tracks, see chapter 12, *Stream tab*.

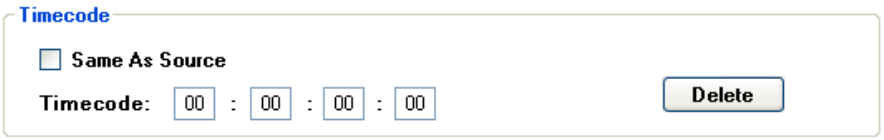
6.4 In/Out Points

To only encode a segment of your source file, specify **In/Out Points** in your setting. This can be a helpful feature if you tweak a setting and want to do several quick test encodings to evaluate filter settings etc.

The times are given as ⟨hours⟩ : ⟨minutes⟩ : ⟨seconds⟩ , ⟨hundredths⟩. Invalid times are set to `00:00:00,00`.

In/out points are not available in demo mode.

6.5 Timecode



Episode Encoder for Windows writes a timecode track to those output formats that support it: GXF, QuickTime and Windows Media. (QuickTime requires you to explicitly specify the creation of a timecode track, see section 5.16, *MOV*.)

Check **Same As Source** to transfer any timecode from the source material. If you enter a time in the **Timecode** field it is used as the starting time for a new timecode track.

For output formats that do not support a timecode track you can combine **Timecode** with the **Burn Timecode** filter (see section 8.19, *Burn Timecode*) to add the timecode directly to the picture rather than as a separate timecode track. This way you can use timecodes in *any* file format. Note that the **Burn Timecode** filter is destructive—once the timecode has been added to the picture it cannot be removed.

6.6 Bumper/trailer

Bumpers and trailers are clips added respectively before and after the main source media. Bumpers and trailers are set to the same format, size and frame rate as the output file but are not otherwise transformed, in particular they are not deinterlaced if the output is to be progressive and vice versa. Bumpers and trailers must have video and audio tracks corresponding to the video and audio tracks of the output file, i e output with both video and audio requires bumpers and trailers with both video and audio, but video-only output does not require audio tracks in the bumper and trailer.

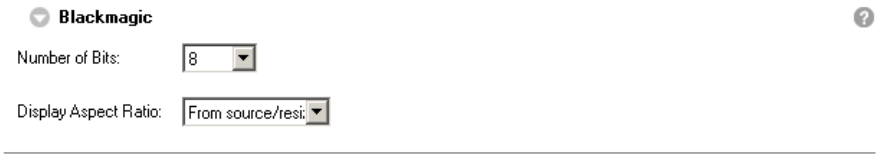
7 Video tab—codecs

Episode Encoder for Windows supports a variety of video codecs suitable for everything from very low bitrate encoding to uncompressed material. This chapter covers these codecs and their individual settings and parameters.

Episode Encoder Pro

Some of the codecs are only available in **Episode Encoder Pro**. They are marked with a Pro in **Episode Encoder**. You can still use them in demo mode.

7.1 Blackmagic



The Blackmagic codec is uncompressed video used by Decklink and Aja products. You have the option to store the video with either **8** or **10** bits per pixel and channel. Note that 10-bit encoding cannot increase the quality of 8-bit source data.

Selecting **From source/resize configuration** uses the display aspect ratio information in the source material if none has been set in the **Resize** filter (section 8.8, *Resize*).

7.2 D-10/IMX

Episode Encoder Pro

The D-10 or IMX codec is defined by the SMPTE 356M standard. It is a form of MPEG-2 4:2:2 I-frames-only video.

Bit Rate is **30 MBit/s**, **40 MBit/s**, or **50 MBit/s**.

Display Aspect Ratio is **1:1 (square pixels)**, **4:3**, **16:9**, or **2.21:1** (see section 3.5, *Picture resolution and aspect ratio*).

7.3 DV

DV is a common editing format. The DV codec does not use prediction between frames (all frames are keyframes), which makes every frame decodable separately. This is what makes the codec suitable for editing, and gives the video an even quality, even in parts of the video that are typically difficult to encode for other video encoders.

The format is restricted to the NTSC and PAL frame sizes and frame rates. Other frame sizes or frame rates cannot be encoded.

DV Type **Episode Encoder for Windows** supports the following DV formats:

DV The DV (25) format is specified by the ISO/IEC 61834 standard:

- The frame size must be 720 × 576 (PAL) or 720 × 480 (NTSC) at a frame rate of 25 and 29.97 frames per second, respectively.
- The bitrate is fixed at 25 Mbit/s.
- The image sampling structure is 4:2:0 for PAL and 4:1:1 for NTSC clips (see section 3.2, *Colour formats*).
- 32, 44.1 and 48 kHz audio sample rates are supported.

DVCPRO DVCPRO is specified by the SMPTE 314M standard:

Episode Encoder Pro

- The frame size must be 720 × 576 (PAL) or 720 × 480 (NTSC) at a rate of 25 and 29.97 frames per second, respectively.
- The bitrate is fixed at 25 Mbit/s.
- The image sampling structure is 4:1:1 for both PAL and NTSC (see section 3.2, *Colour formats*).
- Only 32 and 48 kHz audio sample rates are supported.

Episode Encoder Pro

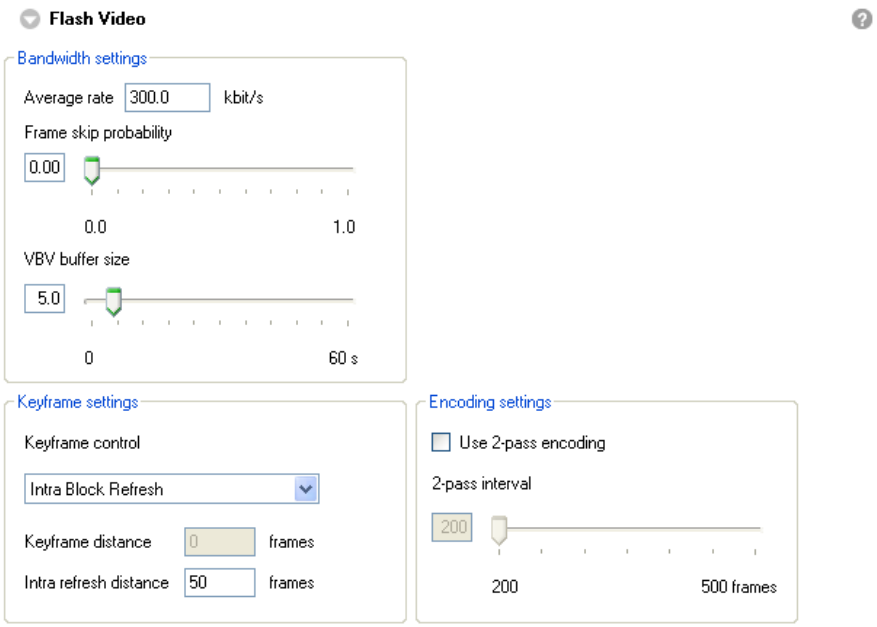
DVCPRO50 The DVCPRO50 format has a bitrate of 50 Mbit/s and an image sampling structure of 4:2:2, but is otherwise the same as the DVCPRO format.

Display aspect ratio **4:3** or **16:9** (see section 3.5, *Picture resolution and aspect ratio*).

NOTE

The **DV Type** automatically sets the audio **Sample Rate** filter (section 10.4, *Sample Rate*) to the appropriate value.

7.4 Flash Video



The Flash 7 Video format is based on H.263. It is often used for video content

on the Web and is stored in either the Small Web Format (SWF) or Flash Video (FLV) file format.

Average rate The desired bandwidth of the video track in kilobits per second. The range is 0–100 000 kbit/s.

Frame skip probability The value 0.0 is least likely to cause frame skipping, while the value 1.0 is most likely to skip frames when bitrate cannot be sustained (see section 3.8, *Frame skip probability—smooth motion vs crisp image*).

VBV buffer size The VBV buffer size is 0–60 seconds (see section 3.7, *VBV—Video Buffer Verifier*).

Keyframe control Keyframes can be set in these modes:

Keyframes Only Make all frames keyframes. This option should only be used for very high bitrates.

Forced Keyframes Only Create a keyframe after the specified number of frames.

Natural and Forced Keyframes Let the codec choose keyframe, but also ensure that there is at least one keyframe within the specified interval. If **Keyframe distance** is set to zero, keyframes are created only when a scene change is detected, making this the same as **Natural Keyframes Only**.

Natural Keyframes Only Let the codec decide when it is appropriate to insert a keyframe. The codec automatically detects a scene change and inserts a keyframe.

Intra Block Refresh If you select **Intra Block Refresh** (IBR), the **Infra refresh distance** must also be set. This differs from the other keyframe options in that the codec does not update the whole frame. The codec updates the different areas in the image in blocks instead of the whole frame. This is useful when encoding for very low bitrates and streaming, since it makes the bitrate more constant.

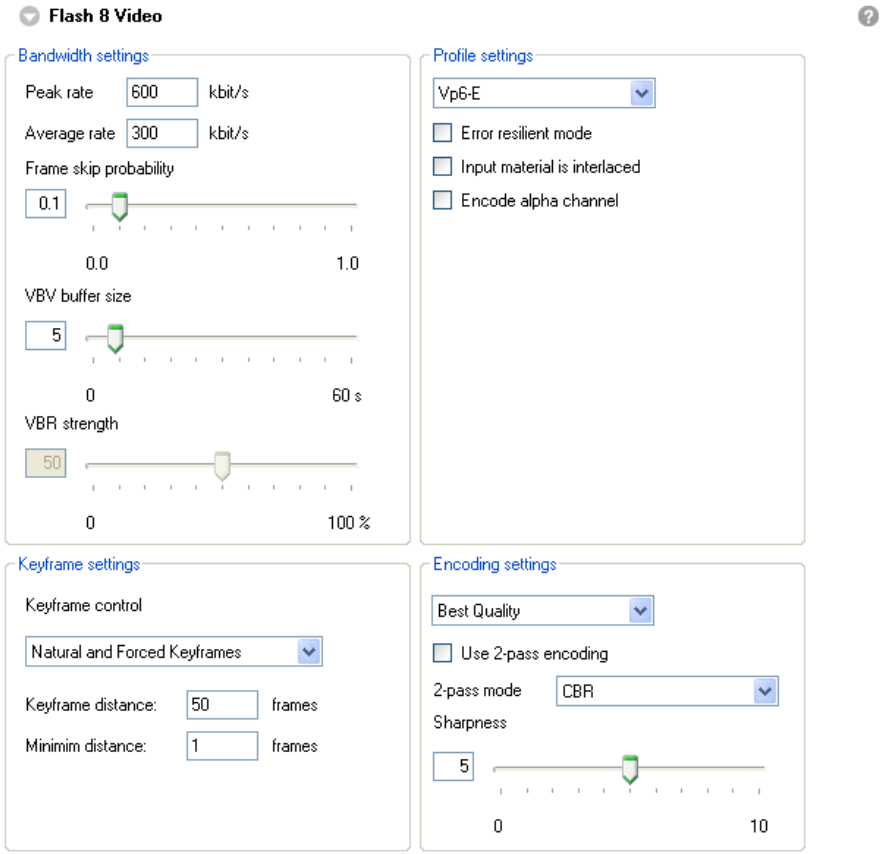
Keyframe distance The maximum distance between any keyframes regardless of scene changes. With long clips with much redundant data such as news clips (talking head), setting a maximum distance allows the stream to recover more rapidly if losing packets. The range is 0–10 000 frames.

Use 2-pass encoding Encoding is performed in two passes. In pass one, the codec analyses the frames and collects data. In the second pass it uses the collected data as the basis for how to best distribute the bits. 2-pass encoding improves quality, but slows down encoding. The **2-pass interval** value sets the number of frames the codec analyses in the first pass before encoding the second pass. Using a higher 2-pass interval increases quality, but encoding is slower. The 2-pass value is 200–500 frames.

NOTE

Flash 7 uses MP3 as audio codec. The sample rates defined and allowed in the Flash standard are 44.1, 22.05, and 11.025 kHz.

7.5 Flash 8 Video



The Flash 8 Video format is stored in either the Small Web Format (SWF) or the Flash Video (FLV) file format.

Peak rate The maximum allowed bitrate. The range is 15 kbit/s to 100 000 kbit/s.

Average rate The desired bandwidth of the video track in kilobits per second. The range is 15 kbit/s to 100 000 kbit/s.

Frame skip probability The value 0.0 is least likely to cause frame skipping, while the value 1.0 is most likely to skip frames when bitrate cannot be sustained (see section 3.8, *Frame skip probability—smooth motion vs crisp image*).

VBV buffer size The VBV buffer size 0–60 seconds. Please read section 3.7, *VBV—Video Buffer Verifier* for more information.

VBR Strength If the **2-pass mode** is set to **VBR**, set the amount of variability allowed; 0 corresponds to CBR.

Keyframe control Keyframes can be set in these modes:

Keyframes Only Make all frames keyframes. This option should only be used for very high bitrates.

Forced Keyframes Only Create a keyframe after the specified number of frames.

Natural and Forced Keyframes Let the codec choose keyframe, but also ensure that there is at least one keyframe within the specified interval. If **Keyframe distance** is set to zero, keyframes are created only when a scene change is detected, making this the same as **Natural Keyframes Only**.

Natural Keyframes Only Let the codec decide when it is appropriate to insert a keyframe. The codec automatically detects a scene change and inserts a keyframe.

Keyframe distance The maximum distance between any keyframes regardless of scene changes. With long clips with much redundant data such as news clips (talking head), setting a maximum distance allows the stream to recover more rapidly if losing packets. The range is 1 to 200 frames.

Minimum distance The minimum allowed distance between keyframes. If this setting is larger than the **Keyframe distance** setting, **Minimum distance** is set equal to **Keyframe distance**. The range is 0 to 50 frames.

Profile The profile is either **Vp6-E** or **Vp6-S**. The Vp6-S profile is easier to decode and therefore suited for larger image sizes, but does not allow 2-pass encoding.

Error resilient mode If checked, error correction codes will be added to the output. This improves quality on lossy networks, but adds approximately 5% overhead to the material.

Input material is interlaced Check this box if you know the source material is interlaced. You should not use the **Deinterlace** filter (section 8.6, *Deinterlace*) in combination with this option.

Encode alpha channel Copy any alpha (transparency) channel in the source material to the output.

Quality Set the image quality to **Normal Quality** or **Best Quality**. **Best Quality** requires more encoding time.

Use 2-pass encoding Encoding is performed in two passes. In pass one, the codec analyses the frames and collects data. In the second pass it uses the collected data as the basis for how to best distribute the bits. 2-pass encoding improves quality, but slows down encoding.

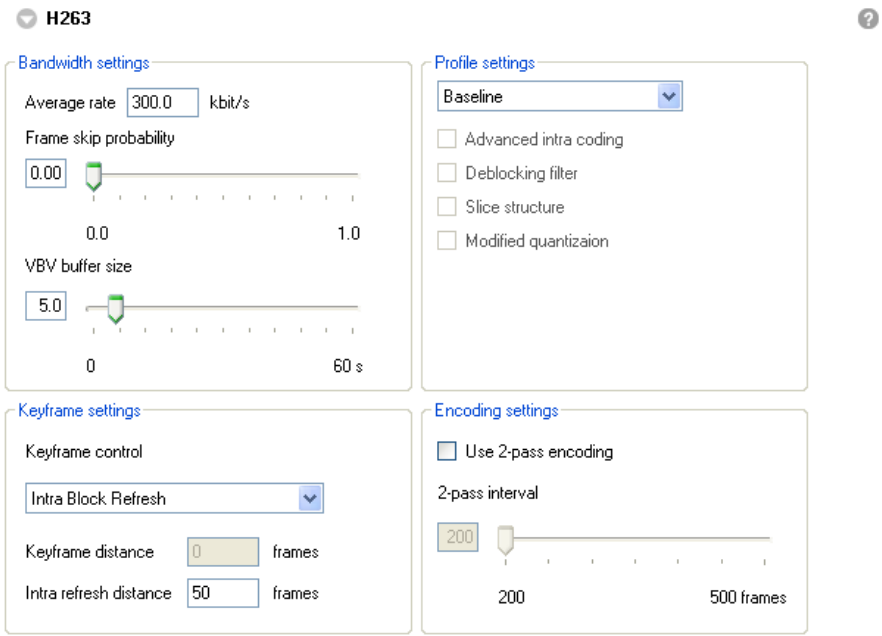
2-pass mode The 2-pass mode is **CBR**, Constant Bit Rate (default) or **VBR**, Variable Bit Rate.

Sharpness A low sharpness setting blurs the image slightly, a high sharpness setting enhances edges but may also cause image artefacts.

2-pass encoding requires intermediate storage of the first pass results on disk in raw format, which means that your available scratch disk space should be at least $1.5 \cdot \textit{output width} \cdot \textit{output height} \cdot \textit{framerate} \cdot \textit{duration}$ bytes.

Storing Flash 8 video in the SWF format restricts the possible the image frame rates depending on the audio sample rates. For audio sample rates of 11025 Hz the maximum allowed frame rate is 9 fps. For audio samples rates of 22050 Hz and 44100 Hz the maximum allowed frame rate is 38 fps. Storing in the FLV format has no such limitations.

7.6 H.263



H.263 is a video codec mainly designed for lower bitrates. The format is suitable for applications such as video conferencing and streaming to handheld devices.

Average rate The desired bandwidth of the video track in kilobits per second. The range is 0 kbit/s to 30 000 kbit/s.

Frame skip probability The value 0.0 is least likely to cause frame skipping, while the value 1.0 is most likely to skip frames when bitrate cannot be

sustained (see section 3.8, *Frame skip probability—smooth motion vs crisp image*).

VBV buffer size The VBV buffer size is 0–60 seconds (see section 3.7, *VBV—Video Buffer Verifier*).

Keyframe control Keyframes can be set in these modes:

Keyframes Only Make all frames keyframes. This option should only be used for very high bitrates.

Forced Keyframes Only Create a keyframe after the specified number of frames.

Natural and Forced Keyframes Let the codec choose keyframe, but also ensure that there is at least one keyframe within the specified interval. If **Keyframe distance** is set to zero, keyframes are only created when a scene change is detected, making this the same as **Natural Keyframes Only**.

Natural Keyframes Only Let the codec decide when it is appropriate to insert a keyframe. The codec automatically detects a scene change and inserts a keyframe.

Intra Block Refresh If you select **Intra Block Refresh** (IBR), the **Infra refresh distance** must also be set. This differs from the other keyframe options in that the codec does not update the whole frame. The codec updates the different areas in the image in blocks instead of the whole frame. This is useful when encoding for very low bitrates and streaming, since it makes the bitrate more constant.

Keyframe distance The maximum distance between any keyframes regardless of scene changes. With long clips with much redundant data such as news clips (talking head), setting a maximum distance allows the stream to recover more rapidly if losing packets. The range is 0 to 10 000 frames.

Profile The H.263 codec has two different profiles: **Baseline** and **Profile 3**. **Baseline** only supports the picture sizes QCIF (176×144) and Sub-QCIF (128×96). **Profile 3** supports all picture sizes and enables four added encoding options:

Advanced intra coding Use an advanced algorithm for the coding of intra blocks.

Deblocking filter Add a deblocking filter to prevent blocking due to hard quantization.

Slice structure Use a different method for dividing the picture into smaller units.

Modified quantization Use a different method for quantization to add flexibility and decrease computational load for the encoder.

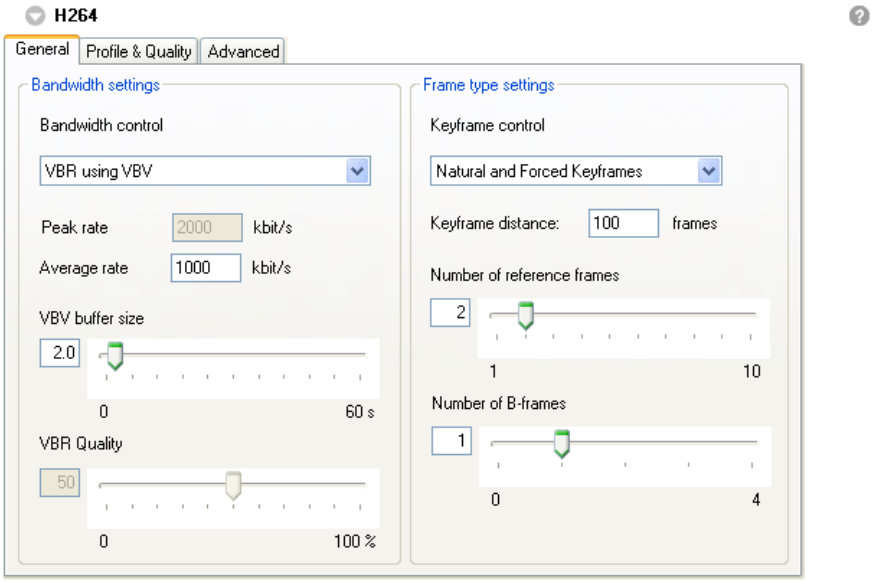
Use 2-pass encoding Encoding is performed in two passes. In pass one, the codec analyses the frames and collect data. In the second pass it uses the collected data as the basis for how to best distribute the bits. 2-pass encoding improves quality, but slows down encoding. The **2-pass interval** value

sets the number of frames the codec analyses in the first pass before encoding the second pass. Using a higher 2-pass interval increases quality, but encoding is slower. It is possible to set a 2-pass value between 200 and 500 frames.

7.7 H.264

H.264, also called AVC or MPEG-4 part 10, represents the state of the art of video compression. It uses many different techniques to achieve a good video compression ratio for bitrates ranging from very low levels for hand-held devices to high levels for HD television.

Due to the large number of controls, the H.264 settings have been split into three tabs.



Bandwidth control Determine which parameters to use to determine the bandwidth allocation: **VBR using VBV**, set the average bitrate over the period set by **VBV buffer size**. **VBR using Peak Rate**, set the average bitrate and the allowed maximum bitrate. **CBR**, set constant bitrate with **Average rate**. **Padded CBR**, set the constant bitrate with **Average rate**. If the video does not need the full bandwidth, the frames are padded with "Stuffing SEI Messages". **VBR - Quality Based** uses the **VBR Quality** alone.

Peak rate The maximum allowed bitrate. The range is 15 kbit/s to 50 000 kbit/s.

Average rate The desired bandwidth of the video track in kilobits per second. The range is 15 kbit/s to 50 000 kbit/s.

VBV buffer size The VBV buffer size is 0–60 seconds (see section 3.7, *VBV—Video Buffer Verifier*).

VBR quality The image quality for a video frame is set from 0% (most compression but lowest quality) to 100% (least compressed but highest quality).

Keyframe control Keyframes can be set in these modes:

Keyframes Only Make all frames keyframes. This option should only be used for very high bitrates.

Forced Keyframes Only Create a keyframe after the specified number of frames.

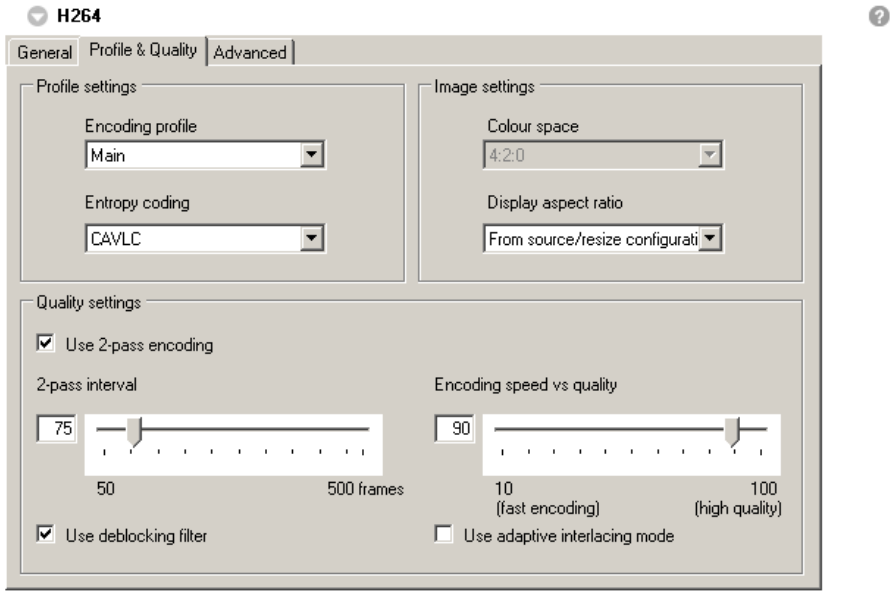
Natural and Forced Keyframes Let the codec choose keyframe, but also ensure that there is at least one keyframe within the specified interval. **Keyframe distance** is the maximum distance between any keyframes regardless of scene changes. With long clips with much redundant data such as news clips (talking head), setting a maximum distance ensures that the stream can recover more rapidly if losing packets. If the value is set to zero, keyframes are created only when a scene change is detected, making it the same as **Natural Keyframes Only**.

Natural Keyframes Only Let the codec decide when it is appropriate to insert a keyframe. The codec automatically detects a scene change and inserts a keyframe.

Keyframe distance The maximum distance between any keyframes regardless of scene changes. With long clips with much redundant data such as news clips (talking head), setting a maximum distance allows the stream to recover more rapidly if losing packets. The range is 1 frames and upwards.

Number of reference frames Set the number of reference frames that P-frames search for prediction. In general, using more than 3 reference frames improves quality only for sequences with a large amount of movement.

Number of B-frames Set the number of B-frames in a GOP from 0 to 4, a higher number giving more efficient encoding but requiring more complex processing.



Encoding profile **Baseline** is the fastest-encodable profile while **Main** may give better compression.

Episode Encoder Pro

High gives higher quality encoding, useful primarily for HDTV applications and editing.

Entropy coding Choice of entropy coding is only possible for the Main and High profiles. **CAVLC** (Context-adaptive variable-length coding) is the simpler and faster coding method and the one used by the Baseline Profile; **CABAC** (Context-adaptive binary arithmetic coding) is the more efficient method.

Colour space The High profile lets you to set the colour encoding of the output video to either **Same as Source**, **4:2:0** or **4:2:2**. The other profiles will use 4:2:0. (See section 3.2, *Colour formats*.)

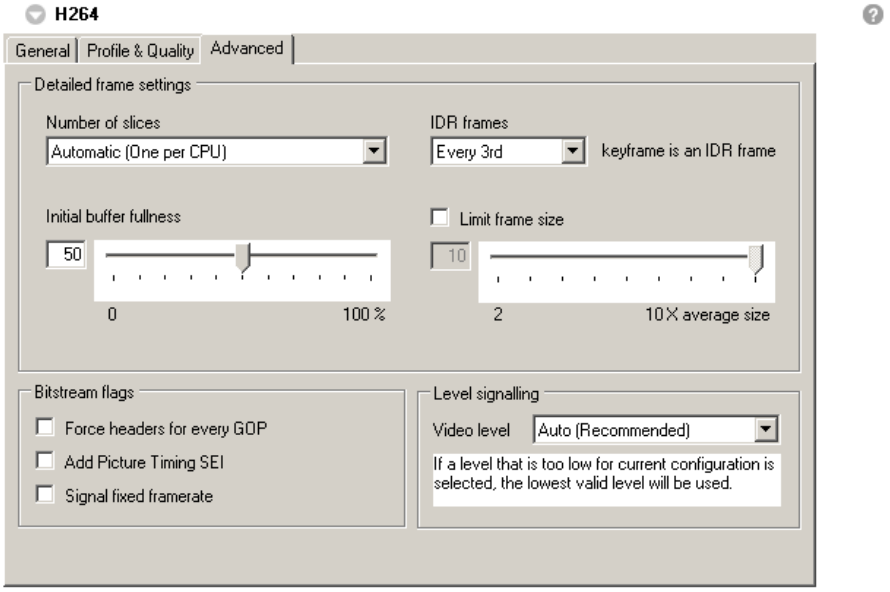
Display Aspect Ratio The **Display aspect ratio** menu makes it possible to create stretched widescreen anamorphic material. To do this you encode your clip at the usual size, for example PAL (720×576) or NTSC (720×480). Then select the desired display aspect ratio. When viewing the clip, the player will stretch the image to the selected aspect ratio. Selecting **From source/resize configuration** uses the display aspect ratio information in the source material if none has been set in the **Resize** filter (section 8.8, *Resize*).

Use 2-pass encoding Encoding is performed in two passes. In pass one, the codec analyses the frames and collects data. In the second pass it uses the collected data as the basis for how to best distribute the bits. 2-pass encoding improves quality, but slows down encoding. The **2-pass interval** value sets the number of frames the codec analyses in the first pass before encoding the second pass. Using a higher 2-pass interval increases quality, but encoding is slower. It is possible to set a 2-pass value between 50 and 500 frames.

Use de-blocking filter Smooth out block artefacts which may occur in the image when using lower bitrates. The de-blocking filter may increase image quality considerably.

Encoding speed vs quality The H.264 encoder has a wide range of encoding methods to use, which may result in a very time consuming encoding process. The **Encoding speed vs quality** setting determines the complexity of the encoding by switching on or off different tools. **Encoding speed vs quality** can be set between 10 and 100; 10 represents the fastest speed, with most of the advanced features turned off, 100 represents the most advanced coding mode, yielding the best quality, but also taking a considerably longer time. In general, values over 50 yield very small improvements in visible image quality.

Use adaptive interlacing mode Set the H.264 encoder to generate more efficient interlaced output. Increases coding time.



Number of Slices On a multi-core computer one can speed up processing by transcoding parts, *slices*, of the same frame in parallel. Using more slices may however decrease image quality somewhat as redundancies between parts of the frame cannot be fully utilised. The number of parallel slices can be set to **Automatic (One per CPU)**, **1 Slice**, **2 Slices**, **3 Slices**, or **4 Slices**.

Initial buffer fullness When encoding starts, the encoder assumes a certain level of bits in the buffer, to get an even bitrate right from the beginning. How full the buffer is assumed to be affects how large the first frames will be. Lower fullness means there are fewer bits available and the first frames will be smaller. This is good when you want to limit the size of the first frame which typically can be very large, especially if there is little motion in the material (because then it makes sense to spend a lot of bits on the quality of the first frame). Setting this value low decreases quality slightly for the first frames of the movie. The default value is 50%.

IDR frames Predictive frames (P-frames and B-frames) coming after an I-frame may look at frames before the I-frame for best compression. An IDR frame is an I-frame whose preceding frames cannot be used by predictive frames. The frequency of IDR frames can be set from **Every** I-frame to **Every 10th** I-frame. More distant IDR frames may allow more efficient compression but makes moving to arbitrary points in the video more complex.

NOTE

Unless every I-frame is an IDR frame **QuickTime Player** may show image artefacts when you scrub the timeline. Sequential playing is not affected.

Limit frame size Even if the average bitrate stays below the set limit, individual frames may become larger than a decoder can handle in real time, thus you

can set limits on how large frames can get. This may reduce image quality considerably, so you should not use this setting unless you have definite problems. The maximum value of any single frame is limited to be 2–10 times the average size of frames.

Force headers for every GOP Insert a header before every GOP, a requirement for Blu-ray material.

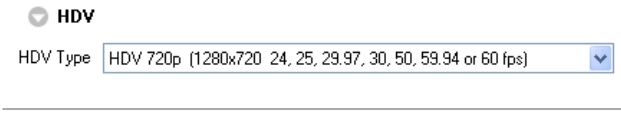
Add Picture Timing SEI Add Supplemental Enhancement Information fields indicating the global time for each frame and suitable decoder settings. They add a small overhead to the file size, but can help a decoder play out the video more efficiently.

Signal fixed framerate Indicate that the clip is to be played out with fixed framerate.

Video level The H.264 standard defines a number of *levels*, which correspond to given maximum values for macro block rates, frame sizes and bit rates. **Episode Encoder for Windows** automatically indicates the level corresponding to the lowest level matching the current values for those parameters, but if you for some reason need to signal a higher level, you can select the desired level here. If you indicate a level lower than allowed by the current parameter values, the lowest legal level is silently selected.

7.8 HDV

Episode Encoder Pro



HDV is a High Definition Video codec using MPEG-2 compression. **HDV Type** is **HDV 720p (1280x720 24, 25, 29.97, 30, 50, 59.94 or 60 fps)**, **HDV 1080i (1440x1080 25, 29.97 or 30 fps)**, or **HDV 1080p (1440x1080 24 or 25 fps)**.

7.9 MJPEG

Motion JPEG is a sequence of JPEG still images, and is suitable to use as an editing format.

Profile The available profiles are **Mjpeg A**, **Mjpeg B** and **Photo Jpeg**. The actual encoding is identical for all versions, but the headers are different.

Colour space Set to **Same as Source**, **4:2:0** or **4:2:2** (see section 3.2, *Colour formats*).

Interlacing Set the video scan to **Progressive**, **Interlaced** or **Same As Source**. If **Interlaced** is selected you can set the field dominance to either **Bottom First** or **Top First**.

Encoding quality Set the image quality for a video frame from 0 (most compression but lowest quality) to 100 (least compressed but highest quality).

7.10 MPEG-1

Video encoded with the MPEG-1 codec is accepted by most video players. This format is suitable when it is important to reach a large audience on different platforms. MPEG-1 is recommended for use with sizes up to CIF (352×288) and bitrates up to 1500 kbit/s, above these levels we suggest that you use MPEG-2 instead.

Coding type Base the encoding on keeping to a given bitrate (**Bitrate-based**) or to a certain level of quality (**Quality-based**).

Average rate Set the desired bandwidth of the video track. The range is 16 kbit/s and upwards.

Encoding quality The image quality for a video frame is set from 0 (most compression but lowest quality) to 100 (least compressed but highest quality).

GOP format Different frame types (I-, P-, and B-frames) are encoded in the same order throughout a clip. This repeating group of frame types is called a Group Of Pictures (GOP). For an explanation of frame types, please see section 3.4, *Frame types—I-, P- and B-frames*. Select one of a number of predefined GOP structures, or select **Custom...** to specify some other GOP structure with the fields **P-frames between I-frames** and **B-frames between P-frames**.

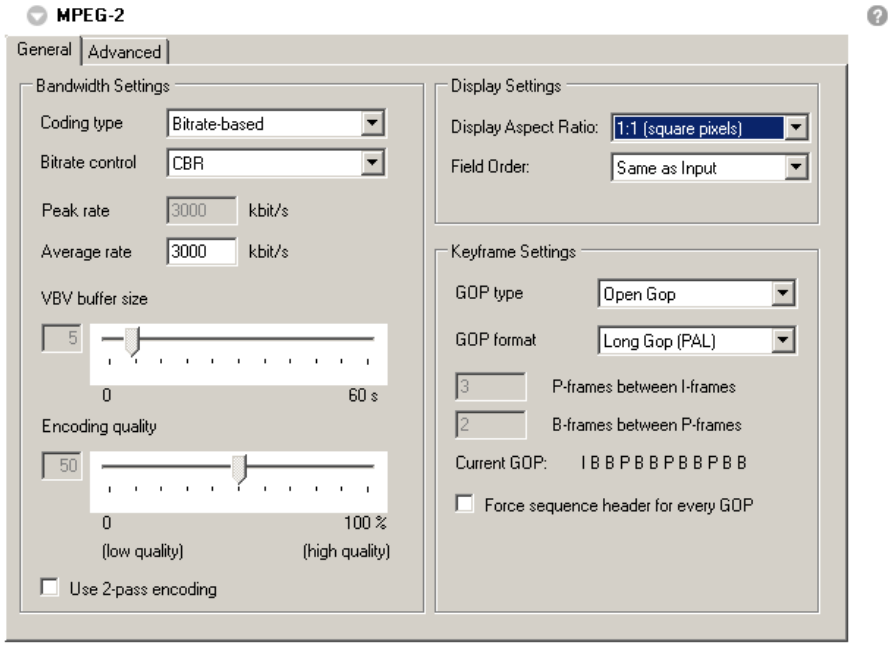
GOP type The type is **Open GOP** or **Closed GOP**. In a file using an Open GOP structure, frames are allowed to predict data from frames outside the GOP. This gives better compression but is not accepted by all applications and does not work well when using MPEG-1 as an editing format. When using Closed GOP each frame in the GOP is independent of the frames outside of the GOP. Hence all the predictive coding is done inside each GOP.

Force sequence header for every GOP Insert a sequence header before every GOP, a requirement when creating MPEG files for editing.

7.11 MPEG-2

The MPEG-2 codec is similar to the MPEG-1 codec, but is more suitable for larger frame sizes and higher bitrates. MPEG-2 is the standard format for DVDs and for digital television.

The settings are split into two tabs.



Coding type Base the encoding on keeping to a given bitrate (**Bitrate-based**) or to a certain level of quality (**Quality-based**).

Bitrate control This menu determines how the bitrate should be maintained. It has the following options:

CBR Encode with a constant bitrate. The VBV buffer size is adjusted so that the output conforms to Main Profile at Main Level. If you encode the video as I-frames only, frames are padded to reach the set bitrate. This is best suited for high bitrates, 30 Mib/s and upwards.

VBV Size This option lets you enter a buffer time value in the **VBV buffer size** field. The **VBV buffer size** field controls the variation in bitrate. A larger buffer size allows bigger variations in bitrate. For more information about the impact on bitrate by the buffer size, please read section 3.7, *VBV—Video Buffer Verifier*.

Peak Rate Control the variation in bitrate by entering a maximum bitrate value in the **Peak rate** field. This value is the highest allowed bitrate in the clip.

Peak rate The maximum allowed bitrate. The range is 16 kbit/s to 300 000 kbit/s.

Average rate The desired bandwidth of the video track in kilobits per second. The range is 16 kbit/s to 300 000 kbit/s.

VBV buffer size The VBV buffer size is 0–60 seconds (see section 3.7, *VBV—Video Buffer Verifier*).

Encoding quality Set the image quality for a video frame from 0 (most compression but lowest quality) to 100 (least compressed but highest quality).

Use 2-pass encoding Perform encoding in two passes. In pass one, the codec analyses the frames and collects data. In the second pass it uses the collected data as the basis for how to best distribute the bits. 2-pass encoding improves quality, but slows down encoding.

Display aspect ratio The **Display aspect ratio** menu makes it possible to create stretched widescreen anamorphic material. To do this you encode your clip at the usual size, for example PAL (720×576) or NTSC (720×480). Then select the desired display aspect ratio. When viewing the clip, the player will stretch the image to the selected aspect ratio. Selecting **From source/resize configuration** uses the display aspect ratio information in the source material if none has been set in the **Resize** filter (section 8.8, *Resize*).

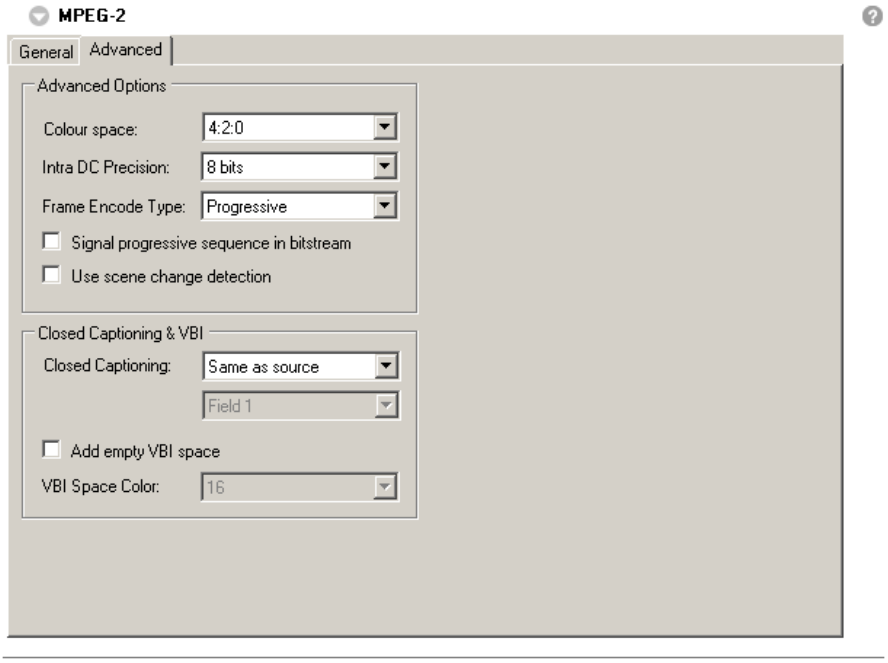
Field Order **Field Order** can be set to **Bottom First**, **Top First** or **Same as Input**. If the output is progressive, this setting is ignored by the player.

GOP type Different frame types (I-, P-, and B-frames, see section 3.4, *Frame types—I-, P- and B-frames*) are encoded in the same order throughout a clip. This repeating group of frame types is called a Group Of Pictures (GOP).

This menu lets you specify **Open GOP** or **Closed GOP**. In a file using an Open GOP structure, frames are allowed to predict data from frames outside the GOP. This gives better compression but is not accepted by all applications and does not work well when using MPEG-2 as an editing format. When using Closed GOP each frame in the GOP is independent of the frames outside of the GOP. Hence all the predictive coding is done inside each GOP.

GOP format Select one of a number of predefined GOP structures or select **Custom...** to specify some other GOP structure with the fields **P-frames between I-frames** and **B-frames between P-frames**.

Force sequence header for every GOP Insert a sequence header before every GOP, a requirement when creating MPEG files for editing.



Colour space Set the colour space to **Same as Source**, **4:2:0** or **4:2:2**. Note that not all MPEG-2 players support the 4:2:2 colour space, so if you have problems with 4:2:2-encoded video, try 4:2:0 instead. (See section 3.2, *Colour formats*.)

Intra DC Precision The DC component of the encoded signal determines the base level of each encoded block. This can be encoded with from **8 bits** to **10 bits**.

Frame Encode Type Even if the output is intended to be displayed as interlaced video, the two fields can be processed and stored as progressive frames, from which the even and odd fields are then extracted by the player. This is the normal behaviour for most MPEG-2 players. Use the option **Progressive** for this.

Processing and storing interlace fields independently may give slightly better compression, but transcoding is slower and the format is not supported by most players. Use the option **Interlaced** for this.

The **Field Order** can be set independently of the **Frame Encode Type**.

Signal progressive sequence in bitstream If **Frame Encode Type** is set to **Progressive**, a flag indicating this can be set in the output stream.

Use scene change detection Insert keyframes (I-frames) in the stream when two consecutive frames differ more than a given threshold.

Closed Captioning **Episode Encoder for Windows** supports writing closed captions in MPEG-2 video in MPEG Elementary, Program, and Transport Streams. Closed captions are imported from the MPEG user data in D-10/IMX, HDV, MPEG-2, and XDCam HD files in the following formats: EIA-608 in ATSC A/53 wrapping, EIA-708 in ATSC A/53 wrapping, and SCTE-20.

Use the **Closed Captioning** menu to select the format of the output closed captioning data. **None** does not place any closed captions in the output file.

Same as source copies any closed captions from the source to the output file. **EIA-608 + EIA-708** places both ATSC A/53-wrapped EIA-608 and EIA-708 captions in the output file. The **EIA-608** and **SCTE-20** options activate the second menu:

Field 1 Place captions in field 1 if the source contains captions in field 1.

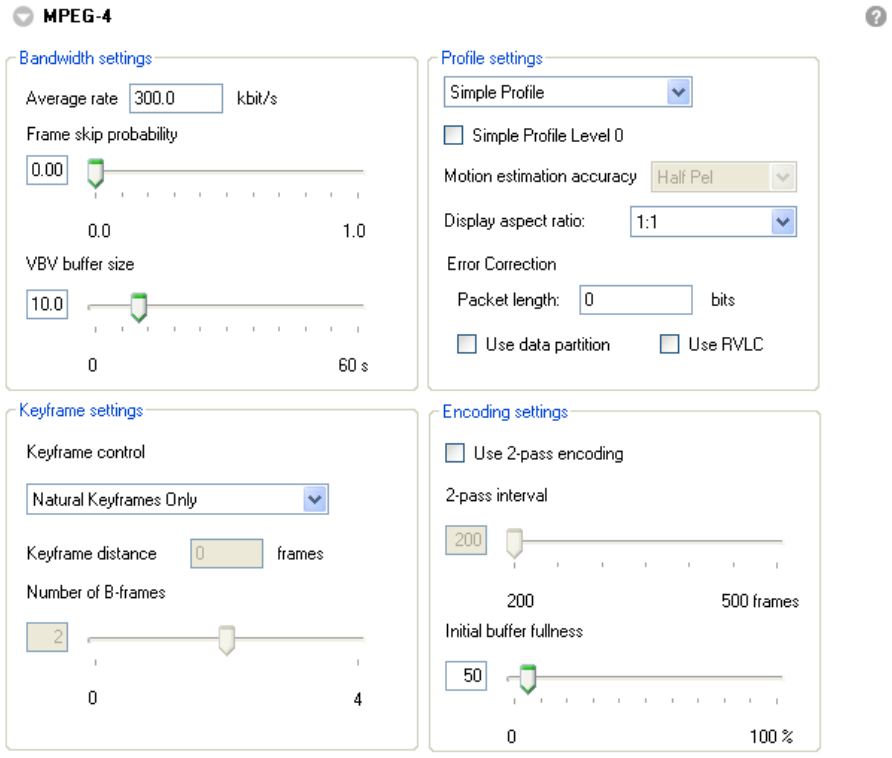
Field 1 + Field 2 Place captions in the same field as they were imported from in the source file.

Episode Encoder for Windows can also import in-band closed captions from the Vertical Blanking Interval (VBI) data using the **VBI Importer** filter (see section 8.2, *VBI Importer*).

Note that not all closed captions in the source can be converted to all formats in the output: EIA-608 in ATSC A/53 wrapping can be converted to SCTE-20; SCTE-20 to to EIA-608; VBI in-band to EIA-608 or SCTE-20.

Add empty VBI space The VBI can be used to store information such as time codes, teletext, etc. This space is not always included in digital versions of NTSC and PAL video. Check this box to pad the height of the image so that there is space for VBI lines. Images from 480 to 511 lines high are assumed to be NTSC and are padded up to 512 lines; images from 576 to 607 lines high are assumed to be PAL and are padded up to 608 lines. The **VBI Space Color** should normally be **16**, but if needed you can set it to **0**.

7.12 MPEG-4



MPEG-4 is rapidly becoming the most common format for downloadable video and audio material.

Average rate The desired bandwidth of the video track in kilobits per second. The range is 0 kbit/s to 100 000 kbit/s.

Frame skip probability The value 0.0 is least likely to cause frame skipping, while the value 1.0 is most likely to skip frames when bitrate cannot be sustained. For more information about this option, please read section 3.8, *Frame skip probability—smooth motion vs crisp image*.

VBV buffer size The VBV buffer size is 0–60 seconds (see section 3.7, *VBV—Video Buffer Verifier*).

Keyframe control Keyframes can be set in these modes:

Keyframes Only Make all frames keyframes. This option should only be used for very high bitrates.

Forced Keyframes Only Create a keyframe after the specified number of frames.

Natural and Forced Keyframes Let the codec choose keyframe, but also ensure that there is at least one keyframe within the specified interval. If **Keyframe distance** is set to zero, keyframes are created only when a scene change is detected, making this the same as **Natural Keyframes Only**.

Natural Keyframes Only Let the codec decide when it is appropriate to insert a keyframe. The codec automatically detects a scene change and inserts a keyframe.

Keyframe distance The maximum distance between any keyframes regardless of scene changes. With long clips with much redundant data such as news clips (talking head), setting a maximum distance allows the stream to recover more rapidly if losing packets. The range is 0 to 10 000 frames.

Number of B-frames Set the number of B-frames in a GOP from 0 to 4, a higher number giving more efficient encoding but requiring more complex processing. This requires the Advanced Simple Profile to be active.

Profile The **Simple Profile** is the fastest-encodable profile, the **Advanced Simple Profile** gives additional options, but is not supported by most players.

Simple Visual Profile Level 0 The Simple Visual Profile Level 0 is used in 3GPP files. If the visual bit stream in the encoded file is below 64 kbit/s it will be tagged as level 0. If the visual bit stream is between 64 kbit/s and 128 kbit/s it will be tagged as level 0B.

Motion estimation accuracy Set the active area for looking at motion estimation. **Half Pel** (1/2 pixel) and **Quarter Pel** (1/4 pixel) can be selected.

Display aspect ratio The **Display aspect ratio** menu makes it possible to create stretched widescreen anamorphic material: Encode your clip at the usual size, for example PAL (720×576) or NTSC (720×480), then select the desired display aspect ratio. When viewing the clip, the player will stretch the image to the selected aspect ratio.

Error Correction MPEG-4 contains support for error correction. The **Packet length** sets the lengths of the data packets. Large packets have less overhead, but give less error correction. The range is 0 to 163 840 bits.

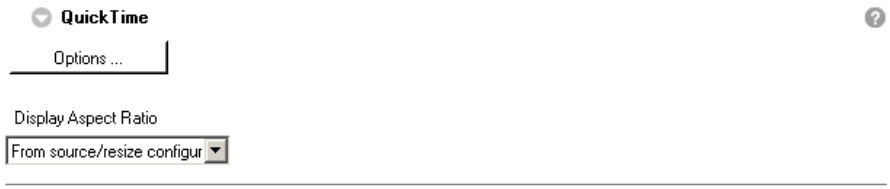
Use Data Partition Insert synchronization markers for better error correction. This is useful for very low bandwidth and error-prone networks.

Use RVLC Reversible Variable Length Codes make it possible for the player to "look back" and check previous parts of the image. If the image contains corrupt data the codec can check with previous frames for correction. Note that using RVLC increases the bandwidth requirements and therefore may result in lowered image quality. RVLC is dependent on player support.

Use 2-pass encoding Perform encoding in two passes. In pass one, the codec analyses the frames and collects data. In the second pass it uses the collected data as the basis for how to best distribute the bits. 2-pass encoding improves quality, but slows down encoding. The **2-pass interval** value sets the number of frames the codec analyses in the first pass before encoding the second pass. Using a higher 2-pass interval increases quality, but encoding is slower. The 2-pass value can be from 200 to 500 frames.

Initial buffer fullness When encoding starts, the encoder assumes a certain level of bits in the buffer, to get an even bitrate right from the beginning. How full the buffer is assumed to be affects how large the first frames will be. Lower fullness means there are fewer bits available and the first frames are smaller. This is good when you want to limit the size of the first frame which typically can be very large, especially if there is little motion in the material (because then it makes sense to spend a lot of bits on the quality of the first frame). Setting this value low decreases the quality slightly for the first frames of the movie. The default value in the decoder is 50%.

7.13 QuickTime



What QuickTime codecs you have depends on your installation, so we cannot describe them in any detail here but refer you to the codec suppliers' documentation. Press **QuickTime Movie Settings...** to display the QuickTime codec dialog. Choose the codec you wish to use from the top-most drop-down menu and enter the settings you want to use.

Set the **Display Aspect Ratio** to **Square pixels**, **4:3**, **16:9**, **2.21:1**, or **2.35:1**. Selecting **From source/resize configuration** uses the display aspect ratio information in the source material if none has been set in the **Resize** filter (section 8.8, *Resize*).

Episode Encoder for Windows overrides the frame rate specified in the QuickTime dialog, so you have to explicitly use the **Frame Rate** filter (section 8.5, *Frame Rate*).

7.14 RGB

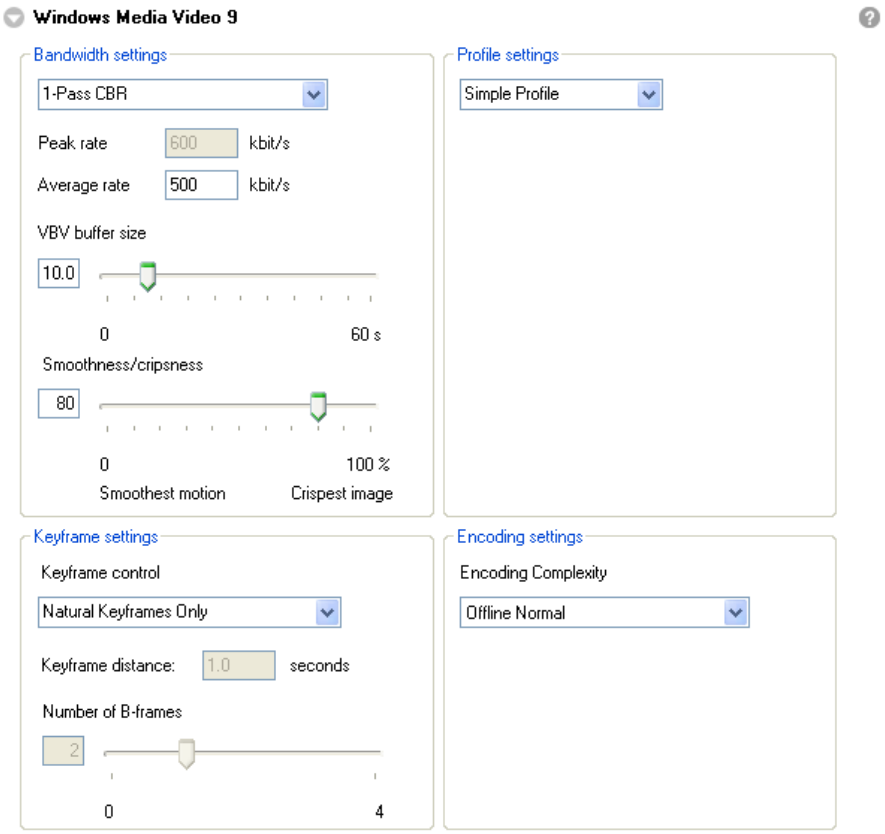
This is an uncompressed format used by QuickTime. **Bit Depth** can be **24** or **32**.

7.15 Targa Cine YUV

Episode Encoder Pro

Targa Cine YUV is an uncompressed video format used with Cinewave cards, it has no settings. The pixel values are stored with 4:2:2 subsampling.

7.16 Windows Media Video 9



The Windows Media 9 codec is used in the Windows Media (WM) format, a proprietary format playable in Windows Media Player.

Windows Media Coding Modes The **Coding Mode** drop-down menu presents you with different coding modes that are available for the Windows Media Video encoder. Depending on which coding mode you select, different settings are available in the GUI. The settings are described below, next to each coding mode.

1-pass CBR Encode the clip at the bit rate specified in the **Average rate** field. How much the rate can vary depends on the **VBV buffer size** slider. A smaller buffer allows smaller variations in bitrate, and vice versa. Please read section 3.7, *VBV—Video Buffer Verifier* for more information.

In 1-pass CBR mode the **Smoothness/crispness** slider sets the trade-off between good picture quality and smooth frame rate, i e the frame skip probability. Please read section 3.8, *Frame skip probability—smooth motion vs crisp image*.

1-pass VBR Encode the clip with variable bitrate, at the quality specified in the **Smoothness/crispness** slider. This setting is picture quality-based only and has no bitrate setting. The encoder uses whatever bitrate necessary to maintain the specified quality. Unlike 1-pass CBR

a higher quality never causes the encoder to skip frames, only to use a higher bitrate.

2-pass CBR Encode the clip with constant bit rate. The encoder analyses the source clip in the first pass and encodes in the second pass. This setting is bitrate-based, and has no picture quality setting. As with 1-pass CBR, the **VBV buffer size** slider decides how much the rate may vary. A smaller buffer will allow smaller variations in bitrate, and vice versa.

2-pass VBR Unconstrained Encode the clip with variable bit rate. The codec analyses the source clip in the first pass and encodes in the second pass. This setting is bitrate-based, but has no peak bit rate limitations.

2-pass VBR Peak Constrained Encode the clip with variable bitrate. The codec analyses the source clip in the first pass and encodes in the second pass. This setting is bitrate-based. You can specify a **Peak rate** and set the **VBV buffer size** to control the average bitrate.

2-pass encoding requires intermediate storage of the first pass results on disk in raw format, which means that your available scratch disk space should be at least $1.5 \cdot \textit{output width} \cdot \textit{output height} \cdot \textit{framerate} \cdot \textit{duration}$ bytes.

Peak rate The maximum allowed bitrate. The range is 5 kbit/s to 20 000 kbit/s.

Average rate The desired bandwidth of the video track in kilobits per second. The range is 5 kbit/s to 20 000 kbit/s.

VBV buffer size The VBV buffer size is 0–60 seconds (see section 3.7, *VBV—Video Buffer Verifier*).

Smoothness/crispness The value 0 is least likely to cause frame skipping, while the value 100 is most likely to skip frames when bitrate cannot be sustained. For more information about this option, please read section 3.8, *Frame skip probability—smooth motion vs crisp image*. Two-pass encoding does not use **Smoothness/crispness**.

Keyframe control Keyframes can be set in these modes:

Keyframes Only Make all frames keyframes. This option should only be used for very high bitrates.

Forced Keyframes Only Create a keyframe after the specified number of frames.

Natural and Forced Keyframes Let the codec choose keyframes, but also ensure that there is at least one keyframe within the specified interval. If **Keyframe distance** is set to zero, keyframes are created only when a scene change is detected, making this the same as **Natural Keyframes Only**.

Natural Keyframes Only Let the codec decide when it is appropriate to insert a keyframe. The codec automatically detects a scene change and inserts a keyframe.

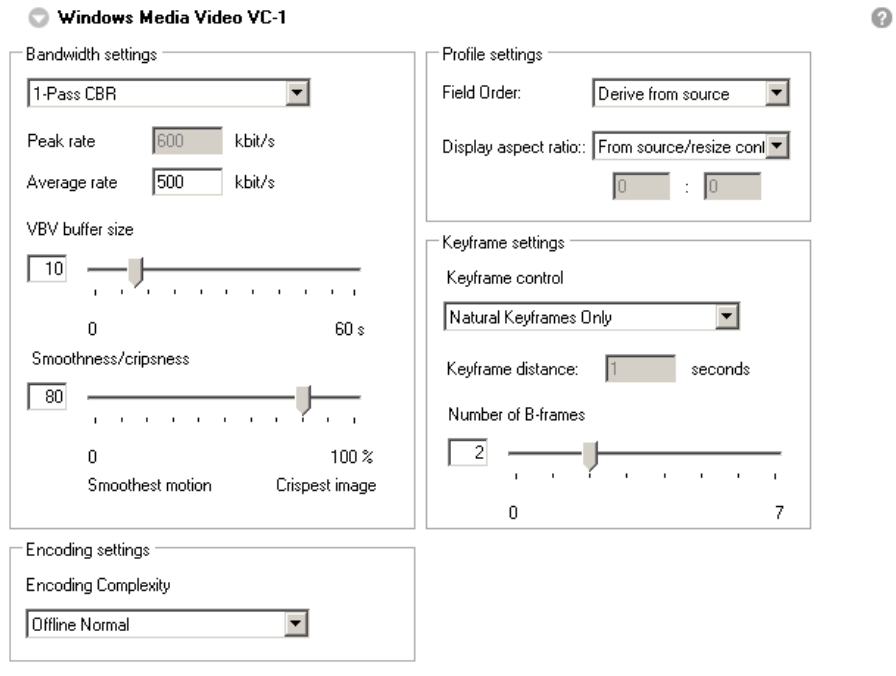
Keyframe distance The maximum distance between any keyframes regardless of scene changes. With long clips with much redundant data such as news clips (talking head), setting a maximum distance allows the stream to recover more rapidly if losing packets. The range is 0–60 s.

Number of B-frames Set the number of B-frames in a GOP from 0 to 4, a higher number giving more efficient encoding but requiring more complex processing. This requires the Main Profile to be active.

Profile The **Simple Profile** is the fastest-encodable profile, the **Main Profile** allows additional encoding options.

Encoding Complexity For streaming video you should choose **Live Fast** or **Live Normal**. The former encodes faster but gives lower quality video. For downloadable video you can choose **Offline Fast**, **Offline Normal**, **Offline Slow**, or **Offline High Quality**, which give increasingly slower, but higher quality encoding.

7.17 Windows Media Video VC-1



The VC-1 codec corresponds to the Windows Media Video 9 Advanced Profile; it is the same as the SMPTE 421M video codec standard. It offers support for interlaced content and is transport independent.

Windows Media Coding Modes The **Coding Mode** drop-down menu presents you with different coding modes that are available for the Windows Media Video encoder. Depending on which coding mode you select, different

settings are available in the GUI. The settings are described below, next to each coding mode.

1-pass CBR Encode the clip at the rate specified in the **Average rate** field. How much the rate can vary depends on the **VBV buffer size** slider. A smaller buffer allows smaller variations in bitrate, and vice versa. Please read section 3.7, *VBV—Video Buffer Verifier* for more information.

In 1-pass CBR mode the **Smoothness/crispness** slider sets the trade-off between good picture quality and smooth frame rate, i e the frame skip probability. Please read section 3.8, *Frame skip probability—smooth motion vs crisp image*.

1-pass VBR Encode the clip with variable bitrate, at the quality specified in the **Smoothness/crispness** slider. This setting is picture quality-based only and has no bitrate setting. The encoder uses whatever bitrate necessary to maintain the specified quality. Unlike 1-pass CBR a higher quality never causes the encoder to skip frames, only to use a higher bitrate.

2-pass CBR Encode the clip with constant bitrate. The encoder analyses the source clip in the first pass and encodes in the second pass. This setting is bitrate based, and has no picture quality setting. As with 1-pass CBR, the **VBV buffer size** slider decides how much the rate may vary. A smaller buffer allows smaller variations in bitrate, and vice versa.

2-pass VBR Unconstrained Encode the clip with variable bitrate. The codec analyses the source clip in the first pass and encodes in the second pass. This setting is bitrate-based, but has no peak bit rate limitations.

2-pass VBR Peak Constrained Encode the clip with variable bitrate. The codec analyses the source clip in the first pass and encodes in the second pass. This setting is bitrate based. You can specify a **Peak rate** and set the **VBV buffer size** to control the average bitrate.

2-pass encoding requires intermediate storage of the first pass results on disk in raw format, which means that your available scratch disk space should be at least $1.5 \cdot \textit{output width} \cdot \textit{output height} \cdot \textit{framerate} \cdot \textit{duration}$ bytes.

Peak rate The maximum allowed bitrate. The range is 5 kbit/s to 20 000 kbit/s.

Average rate The desired bandwidth of the video track in kilobits per second. The range is 5 kbit/s to 20 000 kbit/s.

VBV buffer size The VBV buffer size is 0–60 seconds (see section 3.7, *VBV—Video Buffer Verifier*).

Smoothness/crispness The value 0 is least likely to cause frame skipping, while the value 100 is most likely to skip frames when bitrate cannot be sustained. For more information about this option, please read section 3.8, *Frame skip probability—smooth motion vs crisp image*.

Keyframe control Keyframes can be set in these modes:

Keyframes Only Make all frames keyframes. This option should only be used for very high bitrates.

Forced Keyframes Only Create a keyframe after the specified number of frames.

Natural and Forced Keyframes Let the codec choose keyframe, but also ensure that there is at least one keyframe within the specified interval. If **Keyframe distance** is set to zero, keyframes are created only when a scene change is detected, making this the same as **Natural Keyframes Only**.

Natural Keyframes Only Let the codec decide when it is appropriate to insert a keyframe. The codec automatically detects a scene change and inserts a keyframe.

Keyframe distance The maximum distance between any keyframes regardless of scene changes. With long clips with much redundant data such as news clips (talking head), setting a maximum distance allows the stream to recover more rapidly if losing packets. The range is 0–60 s.

Number of B-frames Set the number of B-frames in a GOP from 0 to 4, a higher number giving more efficient encoding but requiring more complex processing. This requires the Main Profile to be active.

Field order Set the output be **Progressive** or interlaced with **Top field first** or **Bottom field first**; **Derive from source** will retain whatever the source file uses.

Display aspect ratio Set the display aspect ratio of the output to **1:1**, **4:3**, **11:9**, **16:9**, **5:4**, **3:2**, or **Custom...**. For a custom aspect ratio you set the desired aspect ratio in the fields below the menu. Selecting **From source/resize configuration** uses the display aspect ratio information in the source material if none has been set in the **Resize** filter (section 8.8, *Resize*).

Encoding Complexity For streaming video you should choose **Live Fast** or **Live Normal**. The former encodes faster but gives lower quality video. For downloadable video you can choose **Offline Fast**, **Offline Normal**, **Offline Slow**, or **Offline High Quality**, which give increasingly slower, but higher quality encoding.

7.18 Windows RGB

Episode Encoder Pro

Windows RGB is an uncompressed I-frames-only format. **Bit Depth** is **24** or **32** bits.

7.19 XDCam HD

Episode Encoder Pro

XDCam HD is an MPEG-2-based codec used by Sony for High-Definition video.

Field order is **Same as Input**, **Interlaced**, or **Progressive**.

Colour space is **Same as Source**, **4:2:0**, or **4:2:2**.

7.20 YCbCr

Episode Encoder Pro

YCbCr is a transformation of RGB images where Y is the luma (brightness) component of the image and Cb and Cr are the blue and red chroma (colour) components, respectively. Read more in section 3.2, *Colour formats*.

Color Space is **Same as Source**, **4:2:0**, **4:2:2**, or **4:2:2 Interleaved**. The default value is **4:2:0**.

8 Video tab—filters

Filters apply transformations to the source material. Filters may be concerned with adjusting the output format, such as the **Frame Rate** or **Resize** filters, or they may be used to improve the appearance of the image, such as the **Noise Reduction** or **Black and White Restoration** filters; some filters add information to the output file, such as the **Burn Timecode** and **Watermark** filters. You can see the effects of the filters in the **Preview** window (section 4.6, *Preview*), so you can easily check that you achieve the intended effect.

All filters in a new setting are collapsed and deactivated. You expand a filter by clicking on the triangle icon. You *activate* a filter by checking the checkbox in the top left corner. Note that even if you have changed the values in an expanded filter, the filter will not be applied to your clip unless you activate the filter. To deactivate a filter, uncheck the checkbox. To clearly indicate which filters are currently active, **Episode Encoder for Windows** moves activated filters on top of the unused filters, and deactivated filters back to the bottom. Collapsed filters display a text version of their parameter values.

The active filters are applied in the order they are shown from top to bottom. However, the *codec* settings are applied last even though they are topmost in the tab.

Most filters will operate on 8-bit video data, but the **Frame Rate**, **Deinterlace**, **Advanced Frame Rate**, **Resize**, and **Interlace** filters can operate on 10-bit video data.

Press the question mark to bring up a **Windows Help** page on the relevant filter.

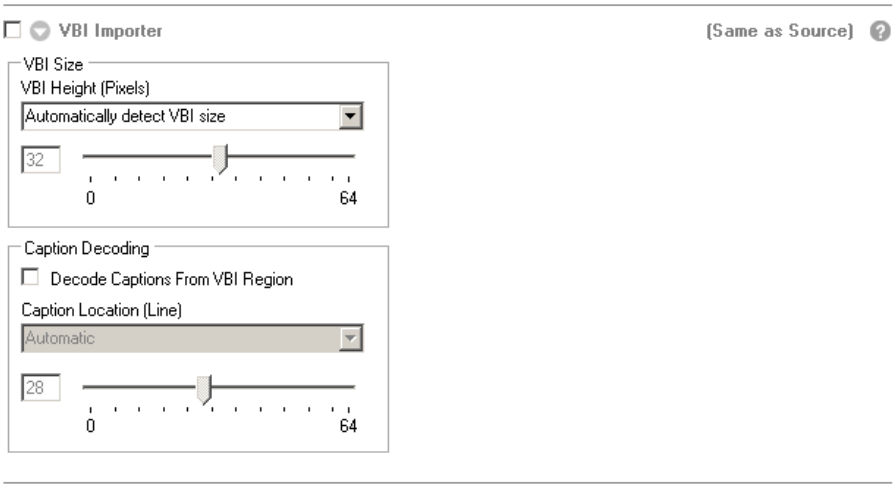
8.1 RGB 10bit to 8bit by using LUT files

This filter can operate on 10-bit video input.

Use a lookup table (LUT) to convert 10-bit source material to 8-bit. The table determines how the 10-bit RGB values are mapped to 8-bit values.

A LUT file is a text file. It has a header `LUT: 3 1024`, followed by 3×1024 lines of integers in the range 0–255. The first group of 1024 values are for R values, the next for G values and the final group for B values.

8.2 VBI Importer

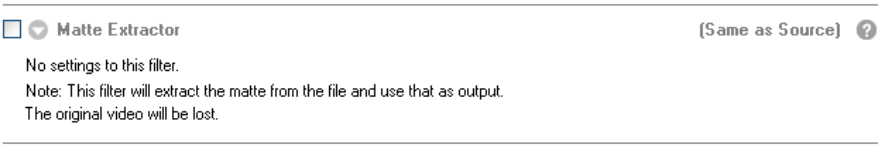


VBI Importer removes the Vertical Blanking Interval (VBI) lines from the actual video data. In this way they will not be affected by other video filters and if needed you can use the **VBI Exporter** filter (see section 8.20, *VBI Exporter*) to reinstate the VBI lines in the output.

If your source video is NTSC or PAL, select **Automatically detect VBI size** to remove the 32 top video lines. If your source has a non-standard VBI size or is neither NTSC nor PAL, select **Manually specify VBI size** to explicitly set the number of lines to remove.

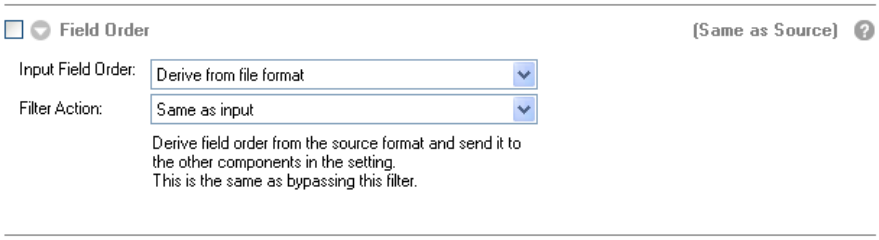
Check **Decode captions from VBI region** to import in-band closed captions from the VBI data. Determining the closed caption line number can be **Automatic** or **Manual**. **Episode Encoder for Windows** can write these closed captions to MPEG-2 output, see section 7.11, *MPEG-2*.

8.3 Matte extractor



Matte extractor extracts the alpha (transparency) channel of the source file and turns it into a grey-scale image, with completely opaque mapped to white and completely transparent mapped to black.

8.4 Field Order



As discussed in section 3.3, *Video scan* video input can be progressive or interlaced. For interlaced material you can change the field dominance.

Input Field Order Tell the subsequent filters if the incoming material is progressive or interlaced with a certain field dominance.

Derive from file format Trust the field order information in the source file to be correct.

Source has Top Field First The source is top field dominant.

Source has Bottom Field First The source is bottom field dominant.

Source is Progressive The source is progressive. This option disables the **Filter Action** menu.

Source has unknown field order Subsequent filters will have guess the field order. This option disables the **Filter Action** menu.

Filter Action You can change the dominance of interlaced material. If you use the **Deinterlace** filter, this determines the field dominance of the output.

Same as input The field dominance of the input is unchanged.

Switch to Top Field First The field dominance is changed to top dominant.

Switch to Bottom Field First The field dominance is changed to bottom dominant.

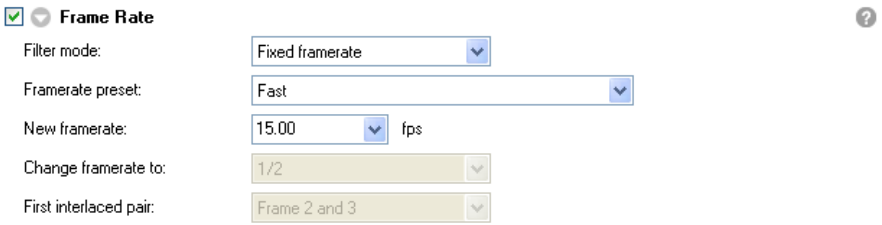
Interlaced to Progressive Deinterlace the input and create progressive output. The output frame rate from this filter is twice the input frame rate.

Since the **Interlaced to Progressive** option deinterlaces the source material, you must not also use the **Deinterlace** filter.

Further, the output frame rate from the filter is double the source frame rate, so if you also use the **Frame Rate** or **Advanced Frame Rate** filter you must take account of this.

8.5 Frame Rate

This filter can operate on 10-bit video input.



Frame Rate performs simple conversions of the frame rate of a clip. The **Advanced Frame Rate** filter (section 8.7, *Advanced Frame Rate*) performs more complex conversions, suitable for video standard conversions (e g PAL ↔ NTSC).

Filter mode **Fixed framerate** Specify the exact frame rate for the output file. Choose a predefined frame rate or enter your desired frame rate in the **New framerate** combo box. Choose a conversion algorithm in the **Framerate preset** menu.

Fractional framerate Specify the resulting frame rate as a fraction of the original frame rate in the menu **Change framerate to**. For example, if you choose **1/2** as fractional frame rate and encode a source video with 25 fps, then the resulting video will have a frame rate of 12.5 fps. The available fractions are **2x**, **1/2**, **1/3**, **1/4**, **1/5**, **1/6**, **1/10**. Note that **2x** speeds up your frame rate, this is useful when e g converting high-definition, low frame rate video to standard-definition, high frame rate video. Since this option simply selects from already existing frames without interpolating, the **Framerate preset** menu is disabled.

Upper limit Sometimes you want to want to encode at a fixed frame rate, though no higher than the source frame rate. **Upper limit** sets a maximum frame rate that will be used if the input frame rate is higher, but if the input frame rate is lower, that will used instead.

Framerate preset **Fast** does not interpolate frames but reuses frames if needed to achieve the desired frame rate. **Automatic** analyses the source video to

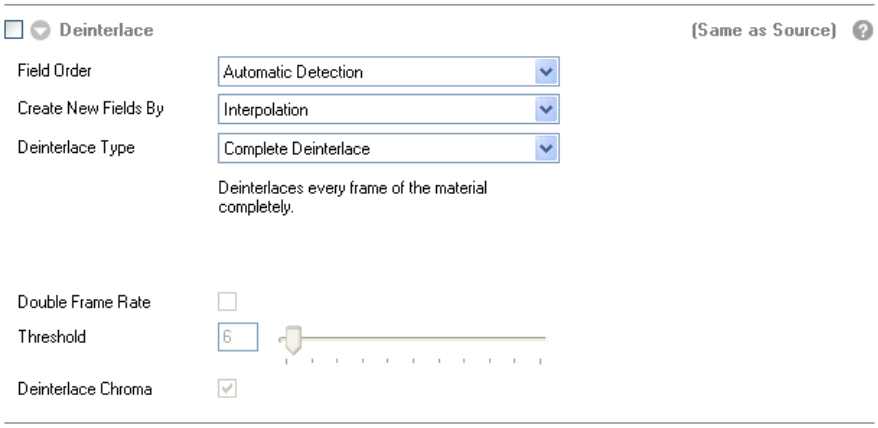
determine the best algorithm for the specific conversion. The other menu alternatives perform conversions between specific formats and inactivate the **New framerate** combo box. The available conversions are: **Telecine 24->29.97**, **Telecine 23.98->29.97**, **Inv.Telecine, Fixed cadence 29.97->23.98**, **Inv.Telecine 29.97->24**, **Inv.Telecine 29.97->23.98**, **Film => PAL 24->25**, **PAL => Film 25->24**.

As explained in section 3.3, *Video scan*, **Episode Encoder for Windows** automatically detects the cadence even when it is broken, but if you know that the cadence is fixed for the entire length of the clip, the extra processing in unnecessary and you can select **Inv.Telecine, Fixed cadence 29.97->23.98**. This enables the menu **First interlaced pair**, which lets you set which frames have been derived from the first duplicated film frame. If the first frame is a single interlaced frame, this option cannot be used.

A frame rate change that changes the duration of the video requires you to use the **Audio Speed** filter to adjust the speed of any audio track to match (see section 10.5, *Audio Speed*).

8.6 Deinterlace

This filter can operate on 10-bit video input.



As explained in section 3.3, *Video scan*, a television video frame is constructed of two interlaced fields shown after each other. Computers, mobile phones, and similar devices use progressive scan, showing a whole frame at once. Accordingly interlaced video should be deinterlaced to be shown on progressive devices.

Field Order Indicate which field is *dominant*.

Automatic Detection Let **Episode Encoder for Windows** analyse the correct field order (recommended). If the **Field Order** filter is used, the values set there are used.

Top Indicate that the input is top field dominant.

Bottom Indicate that the input is bottom field dominant.

NOTE

If the source material is, or is derived from material, in the 4:2:0 colour space (see section 3.2, *Colour formats*), the colour information is normally stored in just one of the fields. If your target is progressive, you can use the **Field Order** menu to indicate which field contains the colour (chroma) information.

Create New Fields By Create each deinterlaced frame from the two fields in one of the following ways:

Duplication Duplicate the dominant field. This process is quick but of lower quality.

Interpolation Create new pixels by linear interpolation of the nearest pixels in the dominant field only; the non-dominant field is discarded. This gives better results than duplication, but requires more processing time.

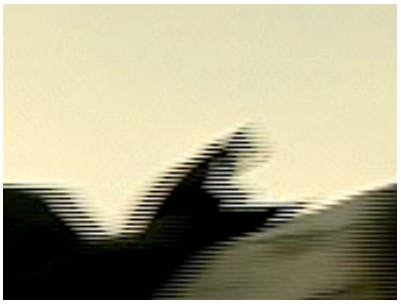
Blending Average both fields. This gives smoother motion but less sharpness than interpolation.

Smooth Blending Average both fields as for **Blending**, but lowpass filter as well. This gives a smoother image, but is slightly slower.

Edge Detecting Interpolation Interpolate, but where **Interpolation** only interpolates vertically, **Edge Detecting Interpolation** attempts to find similar elements in the frame and detect edges before interpolating. This gives more distinct diagonal edges in the output.

Edge Detecting Interpolation Heavy Interpolate as above, but with a more careful algorithm. Slower, but gives slightly better results.

Motion Compensation Analyse the motion of the objects in the video to get more information on how to best create new video frames. This gives the output a sharper look than **Blending**, while still preserving smooth motion. Motion compensation usually gives the best results, but may result in artefacts in scenes where object motion is difficult to estimate. This method is the computationally most complicated, and therefore also the slowest. It will deinterlace all frames, therefore the **Deinterlace Type** cannot be set for this method.



Unprocessed interlaced frame

Duplication

Interpolation

Edge detecting interpolation

Blending

Smooth blending

Motion compensation

Deinterlace Type

Complete Deinterlace Deinterlace the whole frame.

Deinterlace Interlaced Frames (Automatic) Deinterlace completely source frames that are determined to be interlaced. This option is suitable for material with both interlaced and progressive frames, such as telecine material.

Deinterlace Interlaced Frames (Manual) Deinterlace completely source frames that are determined to be interlaced. This option is suitable for material with both interlaced and progressive frames, such as telecine material.

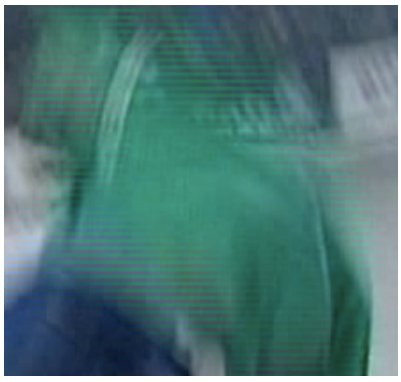
Deinterlace Moving Areas (Automatic) Deinterlace the moving parts of each video frame. This option is not suitable for material with progressive frames, such as telecine material.

Deinterlace Moving Areas (Manual) Deinterlace the moving parts of each video frame. Deinterlacing will be performed on those macroblocks where the average luminance difference between the two frames exceeds the **Threshold**. This option is not suitable for material with progressive frames, such as telecine material.

Double Frame Rate Separate interlaced frames into two consecutive frames. This doubles the frame rate, so it is necessary to apply the **Frame Rate** filter to keep the original speed. One application of this function is converting from high-definition interlaced material to standard-definition progressive material.

Threshold Set the threshold for when deinterlacing should occur. The entered value determines how large the difference can be between the pixels of the two fields before deinterlacing. If the value is set to zero the whole frame is deinterlaced.

Deinterlace Chroma In most video material the luma channel is interlaced but the Chroma channels progressive. In that case only the luma channel needs to be deinterlaced. However, in some cases the chroma channels are also interlaced and you must check **Deinterlace Chroma** to get a correct deinterlace result. In the examples below the source material is interlaced in both the luma and chroma channels. In the picture on the left only the luma has been deinterlaced. In the picture on the right both luma and chroma are deinterlaced.



Notice the deinterlacing artefacts in the picture on the left. It is not always easy to know if the chroma channels are interlaced or not. One way to find out is to open the setting with a clip in the **Preview** and step through it frame by frame. As always, we recommend you to experiment with the settings to create the best result.

NOTE

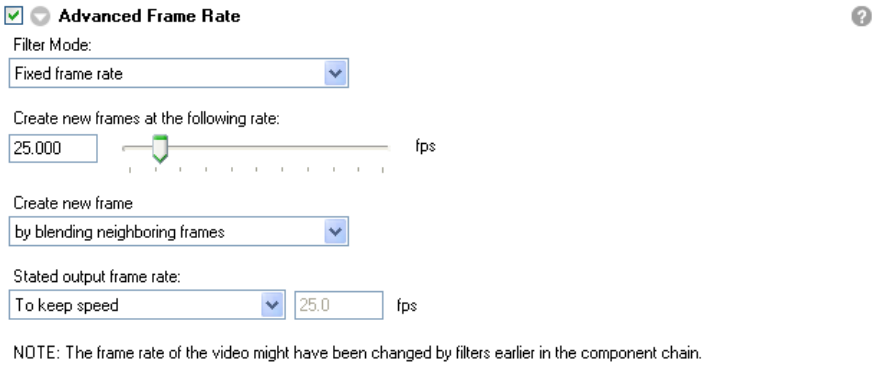
Only apply **Deinterlace** to interlaced content. If applied to non-interlaced content undesirable artefacts will appear. Note that the source material may become deinterlaced in the **Field Order** filter.

8.7 Advanced Frame Rate

This filter can operate on 10-bit video input.

The **Advanced Frame Rate** filter performs more complex frame rate conversions than are possible with the **Frame Rate** filter.

Note that a frame rate change that changes the the duration of the video requires you to use the **Audio Speed** filter to adjust the speed of any audio track to match (see section 10.5, *Audio Speed*).



Filter mode Determine what type of frame rate conversion is desired.

Fixed frame rate Enable the field **Create new frames at the following rate**, which can be set from 1 to 200 fps. New frames will be created to match the given frame rate.

Twice the framerate The output frame rate from the filter is twice the input frame rate. This can be used to create slow motion material by setting **Stated output frame rate** to **Same as input**.

Copy input frames The source frames are copied to the output, but a different frame rate can be specified with the **Stated output frame rate** menu.

Create new frame The filter modes **Fixed frame rate** and **Twice the framerate** require the generation of new video frames. This menu determines how these frames are created.

as copy of nearest neighbor Copy the source frame closest in time to the desired output frame. This is the fastest frame generation method, but may cause uneven motion, especially in pans.

by blending neighboring frames The output frame is a weighted average of the two source frames closest to it in time. This gives smoother motion, but may also cause slight blurriness.

using smart motion compensation Analyse the motion of the objects in the video to give both a sharper image and smooth motion. Scenes where the motion of objects is hard to analyse may however cause artefacts in the image. The algorithm searches in the most probable direction of motion.

using heavy motion compensation Search the entire frame to find the best match for motion. This method is very slow.

Stated output frame rate Set the output frame rate.

Same as input Set the output frame rate from the filter to be the same as the frame rate of the input material. Note that the **Field order** filter may have changed the frame rate relative to the source frame rate.

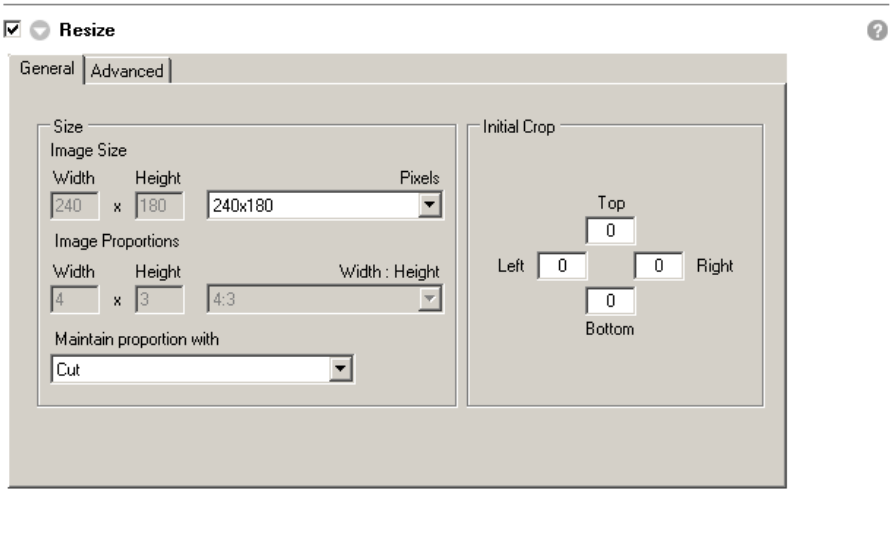
To keep speed Adjust the stated output frame rate so that one second of input material will generate one second of output material.

Set to Explicitly set the desired output frame rate from the filter.

8.8 Resize

This filter can operate on 10-bit video input.

The settings have been split into two tabs.



Size

Image size Select from a list of standard image sizes, from **QQCIF 88x72** to **HD 1920x1080**. If you select **Custom...** you can insert arbitrary values for the **Width** and **Height** of the picture.

Image proportions The **Width : Height** menu provides some image proportions commonly used in digital video to simplify size calculations when entering a new custom image size. For example, if **11 : 9** is selected and **352** is entered in the **Image size Width** field, 288 will appear in the **Height**. You can also specify custom proportions by selecting **Custom...** in the menu, and enter suitable values in the **Image proportions Width** and **Height**.

Note that the image proportions only refer to the relation between the number of pixels in the horizontal and vertical dimensions, which is not necessarily the same as the display aspect ratio, as explained in section 3.5, *Picture resolution and aspect ratio*.

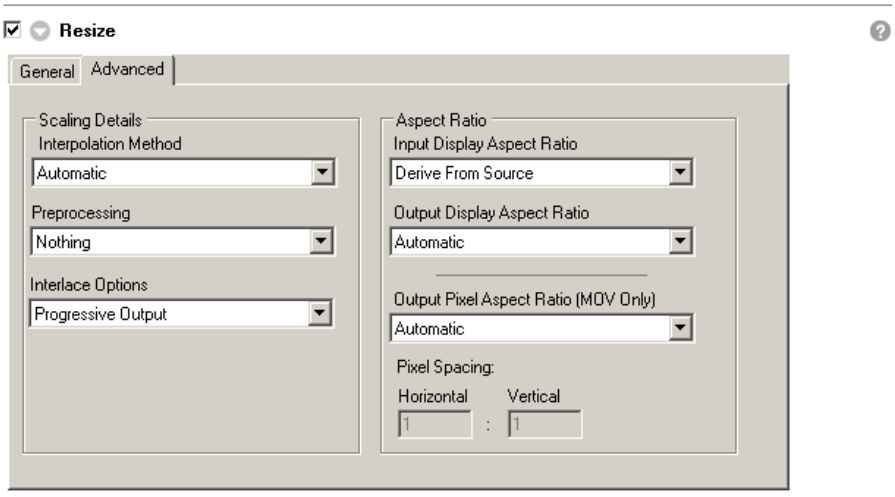
Maintain proportion with When the source material has been processed with **Initial crop** (see below), and the desired output has a different aspect ratio, the **Maintain proportion with** menu provides the following three methods for maintaining the image proportions:

Cut Keep the image proportions of the material by cropping away parts of the image. For example, if encoding from a source clip with a 16 : 9 pixel relation to a clip with 4 : 3 pixel relation, the sides of the source are cut, leaving the resulting image undistorted.

Letterbox (Pad) Pad the image with black borders to fit the destination proportions. For example, a 16 : 9 clip encoded to 4 : 3 is padded at the top and bottom of the image.

None (Distort) Stretch the output image to the desired size. This may cause distortion. The **Initial crop** (see below) will be used. When coding anamorphic MPEG-2 this is the correct option to use. Force the source material to PAL or NTSC size with this option, then select 16 : 9 display aspect ratio for playback in the MPEG-2 codec.

Initial crop Before the image is scaled to the new size a cropping operation is applied to the source material. This can be used to remove black borders or edge artefacts from the source material. The crop values set the number of pixels that are to be cut from the frame borders at the top, bottom, left and right.



Scaling details

Interpolation method Select the interpolation method to use for resizing.

Nearest Neighbor The fastest method but produces the lowest quality. It should only be used when speed is more important than quality.

Bilinear Usually best when downsizing the image.

Bicubic Usually best when upsizing the image.

Automatic Bilinear is used for downsizing and bicubic for upsizing.

Preprocessing When doing a large downscale, for example from 720×576 to 176×144, artefacts may appear with some material, especially if it contains sharp edges. Lowpass-filtering the source before downscaling can reduce the artefacts considerably. Select the preprocessing alternatives **Nothing**, **Lowpass for large downscales**, and **Always lowpass source**.

Interlace options

Progressive Output Scale the image as a whole with no regard to interlacing. This is the default mode and works well in most cases.

Only Crop/Pad to Size - No Scale Crop or pad the image to the new size, without stretching the image. This option is useful when scaling to a size which is only slightly larger or smaller in height than the source, especially if the content is interlaced.

An example is conversion from NTSC 720×486 to MPEG-2 NTSC 720×480. In this case you do not need to encode all the source lines to a format that does not use all the lines. By just cropping the unnecessary 6 lines you do not have to interpolate lines and lose quality in the process.

Scale Fields Independently Divide the image into two fields which are then scaled independently. This keeps the interlacing correct when, for example, downscaling from HD to SD material.

Automatic Select between **Progressive Output** and **Scale Fields Independently** based on the available field order information.

Aspect ratio See section 3.5, *Picture resolution and aspect ratio* for a discussion of display aspect ratios and pixel aspect ratios.

Input display aspect ratio While **Width** and **Height** are given in pixels, the pixels in the source material may not be square. Video CD (VCD) material for instance is usually encoded with 480×480 pixels, but displayed with 640×480 pixels.

Use **Input display aspect ratio** to compensate for odd frame sizes and/or anamorphic source material, by indicating the actual display aspect ratio of the source data.

Pass Through (Keep Display Aspect Ratio) Use the display aspect ratio that results from the settings in the **Size** and **Initial crop** fields as explained above for **Maintain proportion with**.

Derive From Source Use display aspect ratio information in the source file. This is available for D-10/IMX, DV, MPEG-2, and MPEG-4. For other formats, square pixels are assumed.

Example: If the source material is 480×480 pixels with its display aspect ratio field set to 4 : 3, **Image size** is set to 480×480 and **Maintain proportion with** is set to **Letterbox (Pad)**, the output will be 480×480 pixels, but with a visible area of 480×360 pixels with black margins above and below.

Assume Square Pixels Assume that the source material has square pixels and that its display aspect ratio therefore is the same as *width* : *height*.

Assume 4:3 to Assume 2.21:1 Use the selected value for the source display aspect ratio. This display aspect ratio is then used in the same manner as for **Derive From Source**.

Output pixel aspect ratio (MOV only) Set the pixel aspect ratio for QuickTime output. **NTSC ITU-R Rec. 601**, **NTSC ITU-R Rec. 601 (16:9)**, **PAL ITU-R Rec. 601**, and **PAL ITU-R Rec. 601 (16:9)** set the appropriate pixel aspect ratios for the television standards. **Square** sets a 1 : 1 pixel aspect ratio. **Automatic** determines the pixel aspect ratio from the **Image size** and **Output display aspect ratio**. **Custom** lets you explicitly set the **Horizontal** and **Vertical** sizes of pixels.

8.9 Smoothing

Smoothing acts as a blur filter and interpolates the pixels. This makes the material look smoother (but loses some contrast).

Amount The smoothing value can be set between 0 and 100. A value closer to 100 means that the smoothing is stronger.

Radius The smoothing is done with a box filter which can have a size of either **3x3** or **5x5**.

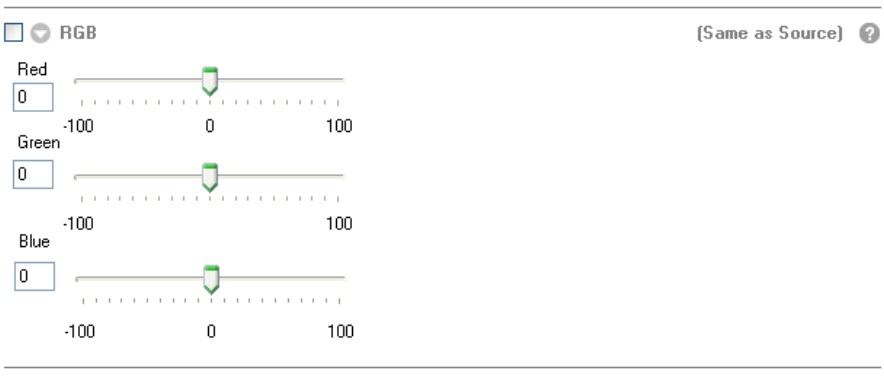
8.10 Sharpen

Sharpen enhances edges in the image, thereby creating a sharper look.

Unprocessed image

Sharpened image

8.11 RGB



RGB adjusts the colours in Red, Green and Blue colour space. Compare with the

HSV Levels which makes colour adjustments in the Hue, Saturation and Value (Brightness) colour space (section 8.16, *HSV Levels*).

8.12 Black and White Restoration

Content for television (encoded from PAL or NTSC) can sometimes look washed out, black appears dark grey and white appears as light grey. **Black and White Restoration** corrects this by setting a new Black and/or White level.

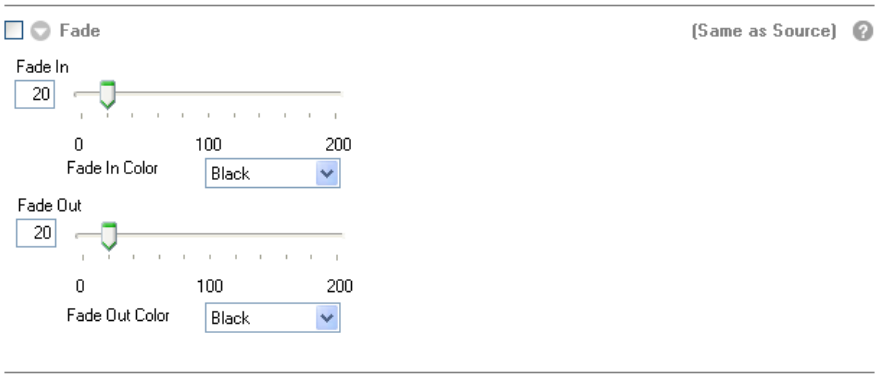
The default value for **Black** is `0`, which corresponds to no change. Example: Set the threshold value to `20` to remap luma values between 0 and 20 to black.

White works the same way as **Black**. Example: Set the threshold to `230` to remap luma values between 230 and 255 to white.

8.13 Contrast

Video encoded from PAL or NTSC can sometimes look a bit grey, or milky, when digitized. Increasing the contrast can then improve the output. Contrast adjustment makes the dark pixels darker and the lighter pixels lighter. Be careful not to increase the contrast too much as the lighter pixels have a tendency to become a "white blur". By lowering the contrast, the image will become more flat or greyish.

8.14 Fade



Fade fades the encoded clip in and/or out. The fading time can be set between 0 and 200 frames. The number of frames are counted from the beginning of the clip for a fade in and from the end for a fade out.

You can fade in/out from/to black or white.

8.15 Gamma

Gamma is probably the most common filter to use and might be the most important correction to do. **Gamma** compensates for the differences between different display technologies and devices, such as when encoding for handheld devices and terminals, targeting Mac/PC, etc.

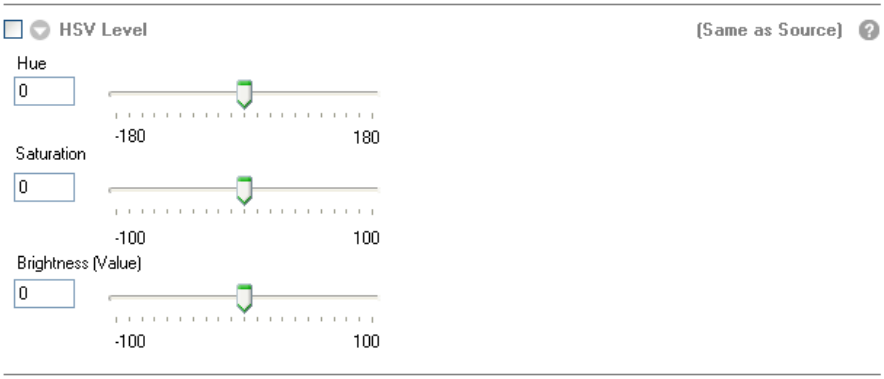
Gamma is a non-linear filter. It only affects the midrange tones but leaves the darkest and lightest parts unchanged. This is useful for darkening or lightening a picture without the risk of distorting the white areas.

Positive numbers (1 to 100) make the image lighter. Negative numbers (−1 to −100) make the image darker.

NOTE

Test the encoded file on the target platform to find out the optimal gamma correction.

8.16 HSV Levels



HSV Levels makes colour adjustments in the Hue, Saturation and Value (Brightness) colour space. Compare with the **RGB** filter which adjusts in the Red, Green, Blue colour space (section 8.11, *RGB*).

Hue changes the colour of the material. The value is measured in degrees. The value can be set between −180° and +180°. This is useful for, e g, correcting badly white-balanced material.

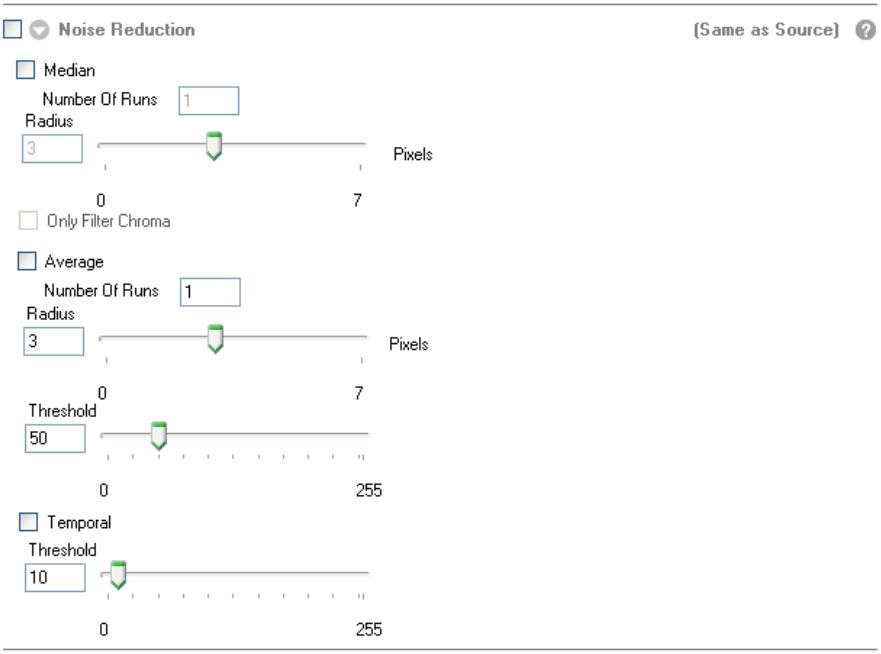
Saturation changes the intensity of the colour. Moving the slider to the right (increasing the values), intensifies the colours, and vice versa. The value can be set from −100 to 100.

Brightness makes the video darker or brighter. The value can be set from −100 to 100. It affects all pixels linearly, unlike the **Contrast** filter which makes dark pixels become darker and bright pixels brighter (section 8.13, *Contrast*).

NOTE

Test the encoded file on the target platform to find out the optimal HSV correction.

8.17 Noise Reduction



Noise Reduction uses three methods: Median, Average and Temporal. You can combine all three methods and specify how many times each method is to be applied.

Median Replace each pixel value with the median value of the pixels in the filter box. This improves the quality of images with impulse noise, by mainly affecting pixels with values very different from those of their neighbours.

Number of Runs The number of times the Median filter is applied to each frame.

Radius The "radius" is properly speaking the side of the box filter applied to the frame, so setting the radius to `3` means that the filter will be applied to the 3×3 pixels surrounding each pixel.

Only Filter Chroma Filter only the colour component of the material. Luma is left unchanged. This can be useful when encoding old VHS material, since much of the noise often resides in the chroma channel.

Average Replace each pixel value with the average value of the pixels in the filter box. This smooths the image.

Number of Runs The number of times the Average filter is applied to each frame.

Radius The side of the box filter. The larger this value, the more noise reduction will be applied, but details in the image will be lost along with the noise.

Threshold If the difference between the filtered pixel and its environment is larger than the threshold, the pixel is not filtered. This preserves text and other small objects with large contrast.

Temporal Compare each pixel in the current frame with the corresponding pixel in the previous frame. If the difference is less than the threshold value the pixel value is left unchanged, otherwise it is replaced by the pixel value in the previous frame. This smooths a series of frames over time.

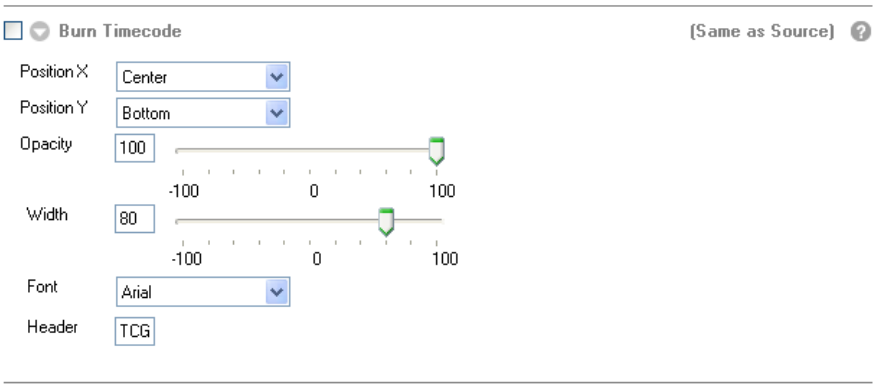
Threshold The threshold level for pixel value replacement.

8.18 Interlace

This filter can operate on 10-bit video input.

Interlace converts progressive source material to interlaced at half the input frame rate. The field order can be set to **Bottom Field First** or **Top Field First**. See section 3.3, *Video scan* for standard field orders for common formats.

8.19 Burn Timecode



Burn Timecode adds a visible timecode to the output video. The timecode is added to the video image and cannot be removed later.

You must activate the **Timecode** in the **Output** tab to use **Burn Timecode**.

The timecode text can be placed in one of nine areas in the picture with the position menus.

Position X **Left**, **Center** or **Right** of the picture.

Position Y **Top**, **Center** or **Bottom** of the picture.

Opacity Make the timecode text partially transparent. The default is fully opaque.

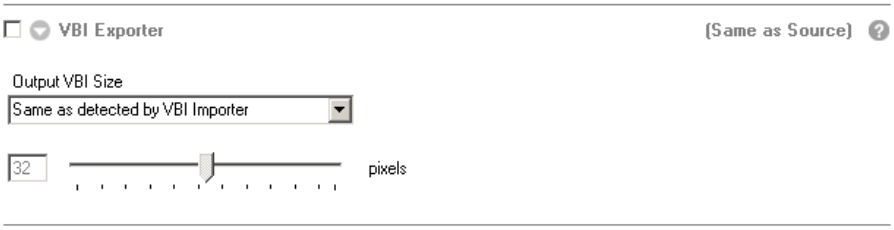
Width The width of the timecode relative to the image width.

Font The font used for the timecode.

Header A text to be inserted in front of the timecode.

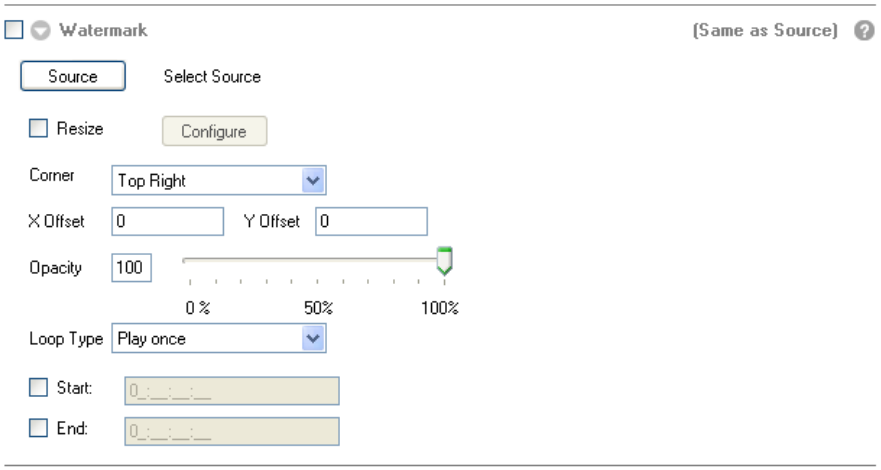
Verify that the timecode looks OK in the **Preview** before you encode.

8.20 VBI Exporter



VBI Exporter adds Vertical Blanking Interval (VBI) data to the output file. **Same as detected by VBI Importer** puts back the VBI data removed by the **VBI Importer** (see section 8.2, *VBI Importer*). **Fixed VBI height** adds as many empty VBI lines as indicated by the slider.

8.21 Watermark

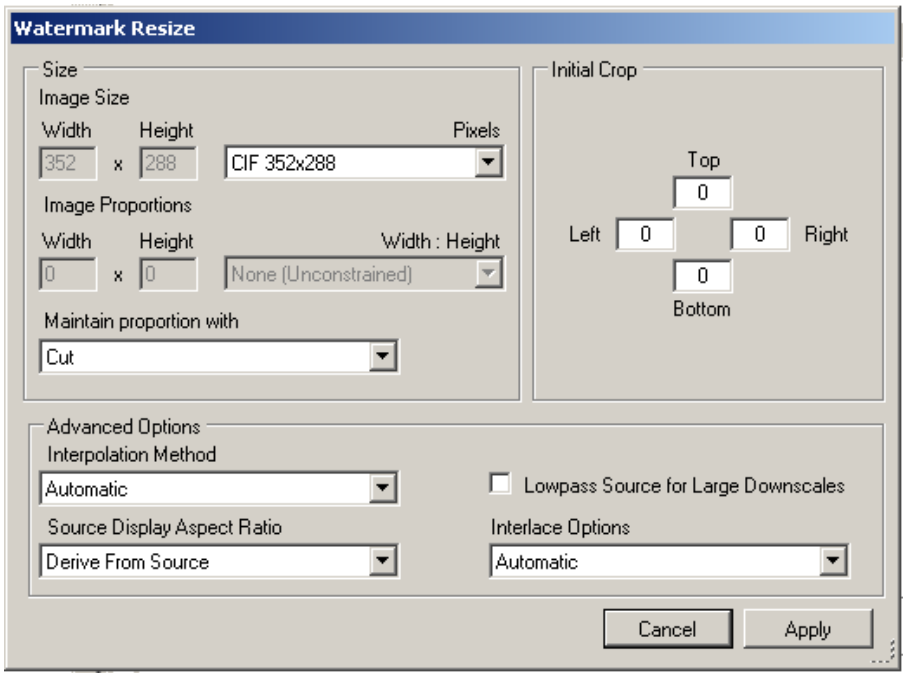


Watermarking your encoded clip is an easy way to ensure that the viewers are aware of the origin of the material that they are watching.

To add a watermark to your encoding click the **Source** button and browse to the picture file that you intend to use as your watermark. Note that you have to have enabled the filter first to be able to set the watermark source.

The path for the watermark is saved in the setting. If the watermark picture is removed, renamed or the path altered, the encoding will fail. Our advice is to have a designated watermark folder where you keep all your watermark files.

If the size of your picture file is too big to fit the image size of your output you can correct this with the **Resize** option. Example: If your watermark file is 600×600 pixels and it needs to be 50×50 pixels to fit your output size, check the **Resize** box and then click **Configure** to open the **Watermark Resize** window:



The **Watermark Resize** window gives you the same options as the **Resize** filter for video explained earlier in section 8.8, *Resize*.

Once you are satisfied with the size of your watermark you set its position. Use the **Corner** menu to choose which corner of the picture to place the watermark in: **Top Left**, **Top Right**, **Bottom Left**, **Bottom Right**. Specify the offset in pixels from the chosen corner with the **X Offset** and **Y Offset** fields.

You can set the **Opacity** of the watermark. This is not to be confused with the transparency (alpha channel) set in the image object itself. It is not possible to set a transparency mask in the **Watermark** filter. However, you can set the overall opacity when applying an already masked watermark to the video.

The **Start/End Points** values limit the time during which the watermark is shown. If the **Start** box is not checked, the watermark is shown from the beginning of the clip; if the **Stop** box is not checked, the watermark will be shown to the end of the clip. The start and end points are given as ⟨hours⟩ `:` ⟨minutes⟩ `:` ⟨seconds⟩ `,` ⟨hundreths⟩. Invalid times are set to `00:00:00,00`.

Watermark offers support for the following picture formats:

Format	Comments
BMP	16/24/32 bit RGB
GIF	
JPEG	EXIF metadata also supported
QuickTime	
Targa	16/24/32 bit RGB
TIFF	16/24/32 bit RGB

You can use animated GIF files and QuickTime movie files to create animated watermarks. If you have an animated watermark the **Loop type** menu lets you select **Play once** to play through the animation once and stop on the last frame or **Loop** to continuously loop the animation. If the watermark is static, the loop type is ignored.

Watermarks with transparent portions avoid covering more of the main image than needed. For partially transparent animated watermarks, use animated GIF files or the QuickTime Animation (Lossless, alpha channel) codec.

TIP

You can use animated watermarks to insert credit rolls, ticker tapes, subtitles and other features in your video material.

9 Audio tab—codecs

Episode Encoder for Windows supports a variety of audio codecs suitable for everything from very low bitrate encoding to uncompressed material. This chapter covers these codecs and their individual settings and parameters.

Episode Encoder Pro

Some of the codecs are only available in **Episode Encoder Pro**. They will be marked with a Pro in **Episode Encoder**. You can still use them in demo mode.

9.1 AAC

AAC (Advanced Audio Coding) is one of two audio codecs specified in the 3GPP standard (the other is AMR). It is an excellent audio codec for music.

Bit Rate You have to adapt the sample rate to your preferred bit rate (and vice versa). The **AAC Setting Guidelines** displays the following table of recommended sample rates for given bit rates:

Bitrate [kbit/s]	Mono Sample Rate [kHz]	Stereo Sample Rate [kHz]
8	8–12	Mono only
16	8–24	8–12
20	11–24	8–12
24	11–32	11–24
28	11–32	11–24
32	11–48	11–24
40	16–48	16–32
48	22–48	22–32
56	22–48	22–48
64	32–48	32–48

All higher bitrates match 32–48 kHz.

Mono sound is supported in the range 8–160 kbit/s, 2-channel stereo sound in the range 16–320 kbit/s, surround sound in the range 160–640 kbit/s, 7.1 channels surround sound in the range 224–640 kbit/s.

AAC Mode The mode can be either **Low Complexity** or **High Efficiency**.

High Efficiency AAC, also known as aacPlus, is an extension of the AAC file format using two new coding techniques: Spectral Band Replication (SBR) and Parametric Stereo. HE-AAC is only partly backwards compatible, as playback of HE-AAC files on AAC decoders is possible, but the high frequencies will not be reconstructed and only mono playback is performed if Parametric Stereo is used.

Episode Encoder Pro

In the High Efficiency mode, Spectral Band Replication is always used. SBR is a technique which copies the lower half of the audio frequencies to the higher half. A small amount of control data (about 2–4 kbit/s) is added to make sure the reconstruction of the high frequencies will be correct, or at least perceived to be correct. By doing this the AAC encoder only has to encode the lower half of the spectrum, which enables encoding at lower bitrates. SBR is recommended for source files with sample rates of 32 kHz or higher, and target bitrates of 20–80 kbit/s. (At higher bitrates, regular AAC will yield higher sound quality.) Read more about SBR in e g [1].

Episode Encoder Pro uses the Dolby AAC encoder, giving higher audio quality, in particular for low bitrates.

Keep Codec Delay For technical reasons AAC players skip a segment of data at the start of a file. This can be compensated by inserting empty data at the beginning, the "codec delay". This option should normally be turned on, but if your particular player does not handle this well, so that your audio ends up out of synch, you can try turning this feature off.

Parametric Stereo Parametric Stereo is an extension to SBR, which encodes stereo information in a very compact way (about 1–3 kbit/s). The source file is then converted to mono and encoded to AAC. Depending on the source material, Parametric Stereo can sometimes improve audio quality at very low bitrates. PS is only available for bitrates up to 56 kbit/s. As the name implies, PS can only be applied to stereo source files. HE-AAC with PS is also known as EAAC+.

9.2 AES

Episode Encoder Pro

The AES codec has been designed by the Audio Engineering Society for serial digital transmission of stereo sound. **Episode Encoder for Windows** uses the SMPTE 331M type. The **Bit depth** can be set to **16 bit** or **24 bit**.

9.3 AMR

AMR is designed for use in cellular phones and mandatory in the 3GPP standard. It is a speech codec that produces extremely low bitrates but does not work well with music.

Bit Rate AMR has a set of fixed bitrates known as *modes*: **4.75 Kbit/s (mode 0)**, **5.15 Kbits/s (mode 1)**, **5.90 Kbits/s (mode 2)**, **6.70 Kbit/s (mode 3)**, **7.40 Kbit/s (mode 4)**, **7.95 Kbit/s (mode 5)**, **10.2 Kbit/s (mode 6)**, **12.2 Kbit/s (mode 7)**.

SID If there are silent passages in the audio track(s), the SID (Silence Descriptor) option makes the AMR codec send a smaller amount of data to save bandwidth. Note that not all players support the SID option.

9.4 ATSC A/52

ATSC A/52 is used on DVDs and one of the leading formats used in movie theatres. It supports bitrates from **64 Kbit/s** to **640 Kbit/s**, but only bitrates above **80 Kbit/s** can be encoded as stereo or surround sound. Recommended bitrates for encoding without audible artefacts are **192 Kbit/s** for stereo and **448 Kbit/s** for 5.1 surround.

When you are generating material from several different sources, you should make sure that the audio levels are normalised across the sources. ATSC A/52 lets you indicate the intended average audio level so that a media player can adjust its output gain accordingly. Typically the audio level of speech is considered as the base level. Set the **Dialog Normalization** slider to the average dialogue level in "dB of full scale" in the source material as indicated by your editing tool. A value of −31 is defined as unit gain, i e 0 dB attenuation, −1 means 30 dB attenuation.

Note that setting the level does not change the audio content of the source, but just tells the media player at what gain to play the audio. If you need to adjust the audio levels within the track, use the **Volume** filter (see section 10.10, *Volume*).

9.5 BWF

The Broadcast Wave Format is based on the WAV format, extended with metadata fields. Set **Bit depth** to **16 bit** or **24 bit**.

9.6 DV audio

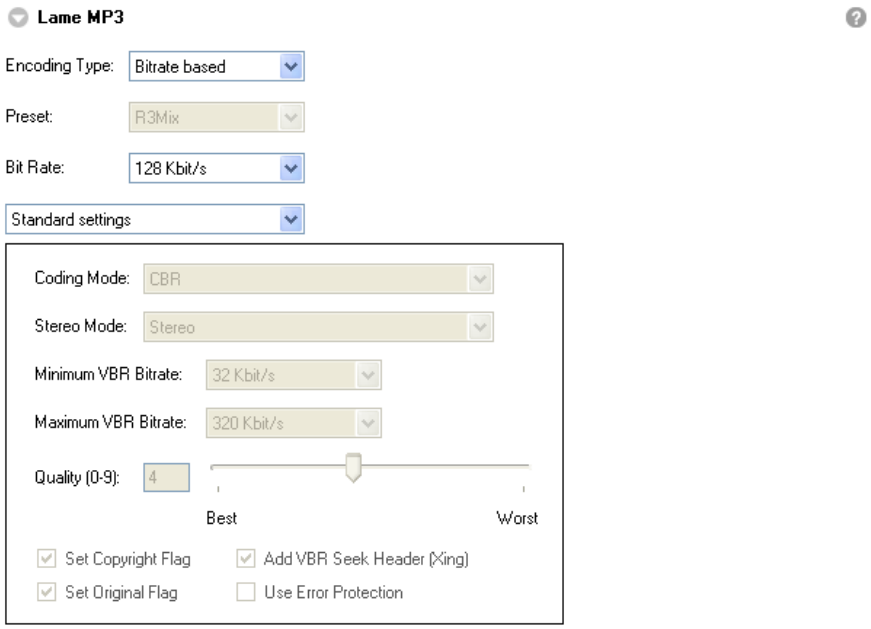
DV audio settings are dictated by the DV video settings. Set **Sample Size** to **12 bit** or **16 bit**.

9.7 EVRC

Episode Encoder Pro

EVRC is a codec intended for speech only used in the 3GPP2 standard for mobile phones. Set **Bit Rate** to **4.8 Kbit/s** or **9.6 Kbit/s**.

9.8 Lame MP3



Lame MP3 is one of the most used implementations of the MP3. See `http://lame.sourceforge.net/` for further information.

Encoding type The **Bitrate based** option lets you set the average bit rate of the data and make other adjustments, the **Lame Preset** option lets you choose one of the options in the **Preset** menu, which will then set all parameters to predefined values.

Preset

R3Mix A legacy setting used at `http://www.r3mix.net/`. It gives slightly better quality than the **Medium** setting.

Medium Acceptable audio quality for most uses.

Medium Fast A faster algorithm but should give almost as good audio quality as **Medium**.

Standard Good audio quality for normal use.

Fast Standard A faster algorithm but should give almost as good audio quality as **Standard**.

Extreme The best audio quality for high-quality equipment.

Fast Extreme A faster algorithm but should give almost as good audio quality as **Extreme**.

Insane The absolutely best audio quality, but requires very high bandwidth.

Bit Rate For bitrate-based encoding the average bitrate can be set from **16 Kbit/s** to **320 Kbit/s**.

Settings **Standard settings** gives you suitable default settings for your chosen bitrate, **Advanced settings** lets you adjust additional encoding parameters:

Coding Mode Your selected bitrate can be used for **CBR** (Constant Bit Rate) or **VBR** (Variable Bit Rate). If the latter is selected, you can set the **Minimum VBR Bitrate** and **Maximum VBR Bitrate** between **16 Kbit/s** and **320 Kbit/s**.

Stereo mode **Stereo** encodes each stereo channel separately. **MS Stereo** uses mid/side encoding, where the shared content of the stereo channels will be coded in higher resolution than the difference between them; this decreases the bandwidth requirements for low bit rates (< 128 Kib/s) and small stereo separations. **Joint Stereo** decides, frame by frame, whether to use separated stereo or MS stereo.

Quality `0` represents the *best* quality encoding, while `9` gives the lowest quality. Better quality implies slower algorithms.

Set copyright flag The material is tagged as copyrighted.

Set original flag Unless checked, the material will be tagged as a copy.

Add VBR seek header (Xing) A "Xing" header adds information to a VBR-encoded file so that a player can jump to arbitrary positions in the file.

Use error protection Activate CRC error protection. This allows reconstruction of lost packets, but requires slightly more bandwidth to encode the error protection data.

9.9 MPEG Audio

MPEG Audio includes the following three different settings:

Bit Rate The bitrate can be set from 8 to 448 Kib/s, but not all bitrates are available for both layers.

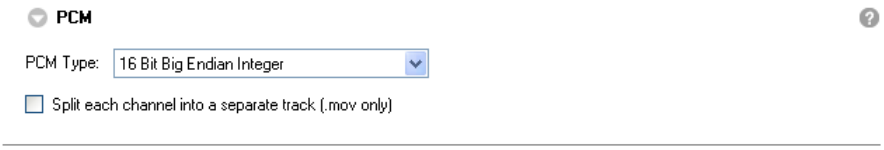
Layer The alternatives are **Layer I** and **Layer II**. (Layer III is the same as MP3.) Layer I uses a simpler encoding method and works best for higher bitrates, Layer II uses a more complex encoding method but compresses better.

Psycho Model The psycho-acoustic model is used to determine the features of the sound that are inaudible and therefore can be compressed away. **Model 1** is the simpler model, which gives somewhat worse results for less computation; **Model 2** requires more computation but gives better results.

The setting of the **Sample Rate** filter determines the encoding: sample rates from 16 to 24 kHz are encoded as MPEG-2 audio, sample rates from 32 to 48 kHz are encoded as MPEG-1.

More information about MPEG audio encoding may be found in [5].

9.10 PCM

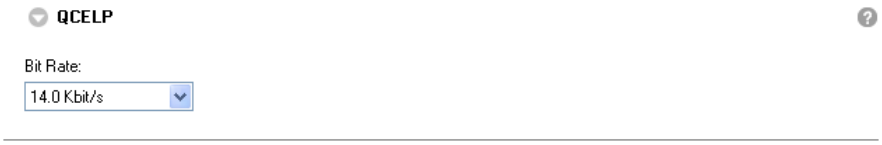


PCM is an uncompressed audio format that can be encoded in a number of ways from **8 Bit Unsigned Integer** to **32 Bit Little Endian Float**. Not all outformats support all PCM encodings, so the selected encoding may be quietly folded into an encoding supported by the active outformat.

Check **Split each channel into a separate track (.mov only)** to place the stereo and surround sound channels in separate tracks on QuickTime output. For other output formats the button setting is ignored.

9.11 QCELP

Episode Encoder Pro

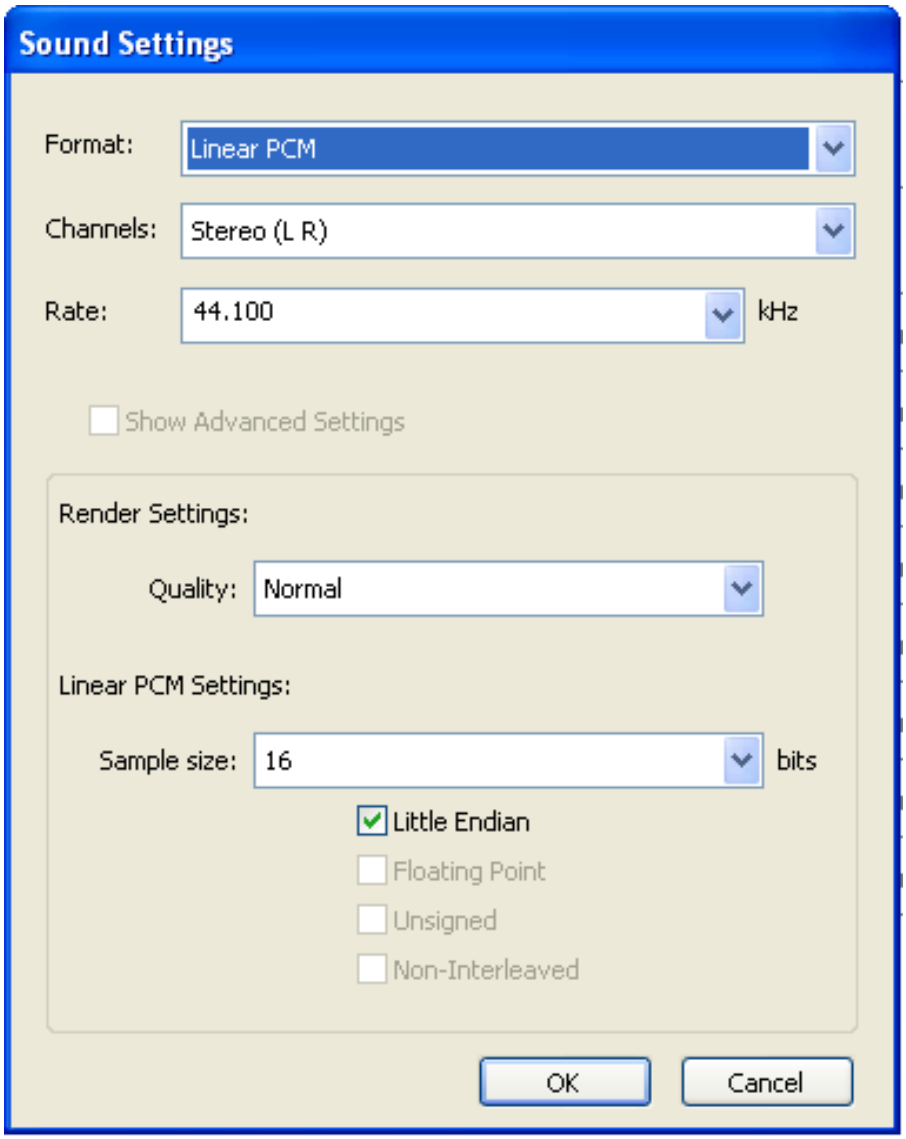


QCELP is a codec intended for speech only used in the 3GPP2 standard for mobile phones. Set **Bit Rate** to **6.80 Kbit/s** or **14.0 Kbit/s**.

9.12 QuickTime

What QuickTime codecs you have depends on your installation, so we cannot describe them in any detail here but refer you to the codec suppliers' documentation. Press **QuickTime Movie Settings...** to display the QuickTime codec

dialog. Choose the codec you wish to use from the top-most drop-down menu, and enter the settings you want to use.

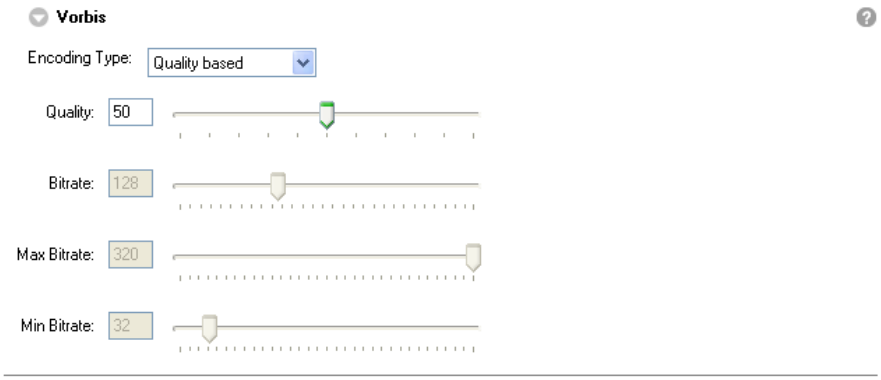


All codecs share the menu **Channels**, with alternatives **Mono** and **Stereo (L R)**, and the field **Rate** for setting the sample rate in kHz. Below these is an area with codec-specific settings. Some codecs have two versions of the specific settings, the larger version displayed when the box **Show Advanced Settings** is checked.

Note that codec settings are applied *after* all filter settings. This means that if you have set the **Channels** filter to **Mono** and your codec has been set to produce stereo, the audio track is first folded into mono, and this mono track is then replicated to make stereo tracks. The same goes if you set the **Sample Rate** filter to a lower rate than set in the codec.

You should therefore ensure that **Channels** and **Sample Rate** are unchecked `(Same as Source)` when using the QuickTime codec.

9.13 Vorbis



Encoding type Select **Quality based** to enable the **Quality** slider, **Bitrate based** enables the other three sliders.

Quality A lower quality setting requires less bandwidth, but gives worse sound reproduction.

Bitrate The target bitrate for the audio, in the range 32–320 Kib/s. The codec generates the best audio quality possible for this bitrate.

Max Bitrate The maximum allowed bitrate, in the range 32–320 Kib/s.

Min Bitrate The minimum allowed bitrate, in the range 32–320 Kib/s.

9.14 Windows Media Audio 9

The Windows Media Audio 9 codec has the following versions:

WMA 9 Standard Encode the audio in the WMA 9 Standard format. It supports the coding methods **One pass, constant bit rate (CBR)** and **One pass, variable bit rate (VBR)**.

WMA 9 Professional Encode the audio in the WMA 9 Professional format. It supports multiple channels, sample rates above 48 kHz and wider than 16

bit samples. However, it is often not available on lower-end platforms, such as mobile phones. It supports the coding methods **One pass, constant bit rate (CBR)** and **One pass, variable bit rate (VBR)**.

WMA 9 Lossless This is a non-destructive codec delivering uncompressed audio containing all of the data in the original content. The final bitrate is dependent on your original source. It supports the coding method **One pass, variable bit rate (VBR)**.

The WMA version you select determines the available menu options in the top menu in the **Coding Method** area. The selected option in combination with the selected coding method determines the available menu options in the bottom menu. **One pass, constant bit rate (CBR)** contains encoding alternatives ranked in order of their bit rate, **One pass, variable bit rate (VBR)** contains encoding alternatives ranked in order of audio quality.

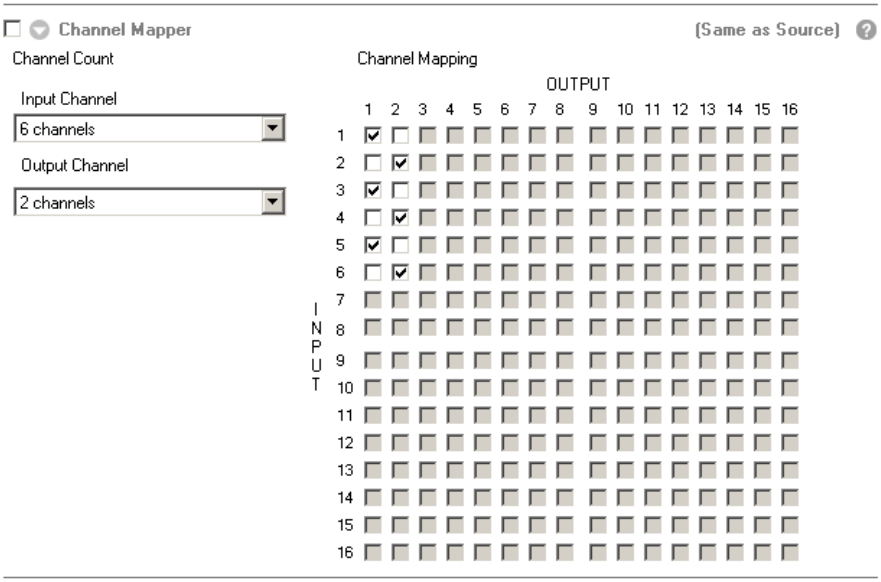
10 Audio tab—filters

You expand a filter by clicking on the triangle icon. You *activate* a filter by checking the checkbox in the top left corner. Note that even if you have changed the values in an expanded filter, the filter will not be applied to your clip unless you activate the filter. To deactivate a filter, uncheck the checkbox. To clearly indicate which filters are currently active, **Episode Encoder for Windows** moves activated filters to above the unused filters, and deactivated filters back to the bottom. Collapsed filters display a text version of their parameter values.

The active filters are applied in the order they are shown from top to bottom. However, the *codec* settings are applied last even though they are topmost in the tab.

Press the question mark to bring up a **Windows Help** page on the relevant filter.

10.1 Channel Mapper



Most of the time you need not worry about remapping sound channels from source file to output file, as this is handled automatically, but occasionally you have a source file that contains a number of mono audio channels that have to be explicitly remapped to output channels.

Set the number of input channels to be used with the **Input Channels** menu. If the

source file has more channels than indicated, only the indicated number is used, if it has fewer channels, only as many channels as are actually present in the file are used. Set the desired number of output channels with the **Output Channels** menu. **Input Channels** and **Output Channels** are limited to the maximum number of channels supported by the output format.

The **Channel Mapping** checkbox matrix maps the input channels to the outputs. In the image above we are mapping a six channel audio input to two output channels. Input channels mapped to the same output channel are mixed together. Input channels which are not checked at all are dropped.

Press **Clear** to remove all connections. Press **Set Defaults** to connect each input channel to the corresponding output channel.

For up to 8 output channels, the channels represent spatial positions according to the following table:

	1	2	3	4	5	6	7	8
Mono	C							
Stereo	L	R						
3.1	C	L	R	LFE				
4.0	L	R	LS	RS				
5.1	C	L F	R F	L	R	LFE		
7.1	C	L F	R F	L	R	LS	RS	LFE

C = Centre, L = Left, R = Right, F = Front, S = Surround, LFE = Low Frequency Effects

10.2 Channels

Channels resamples incoming audio tracks, regardless of how many they are, to **Mono**, **Stereo**, **5.0**, **5.1**, or **7.1**. Not all options are available for all codecs. Resampling from fewer input channels to more output channels is usually not meaningful as it requires additional bandwidth without improving the sound quality. For the same reason you should not combine **Channel Mapper** and **Channels**.

10.3 High Pass/Low Pass

High Pass Cut off all frequencies below the threshold value. The threshold can be set between 10 and 1000 Hz. Check the box to activate high pass filtering and enter the value of your choice.

Low Pass Cut off all frequencies above the threshold value. The threshold can be set between 1 and 20 kHz. Check the box to activate low pass filtering and enter the value of your choice.

10.4 Sample Rate

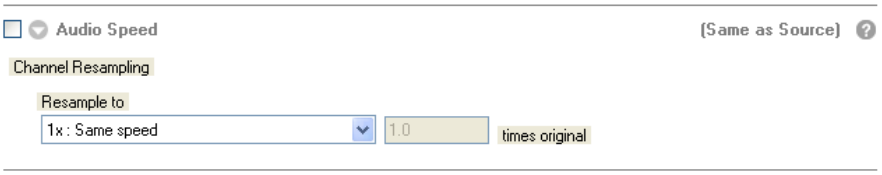
Sample Rate has a drop-down menu with the available sampling frequencies for the audio codec that you have chosen in the **Output** Tab. The available frequencies in the list vary from codec to codec. Note that not all output formats allow all sample rates supported by a given codec.

The sampling frequency is the number of samples per second in the audio track. Higher sample rates allow higher sound frequencies to be reproduced.

NOTE

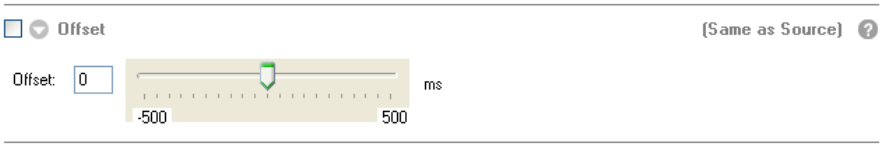
The Nyqvist frequency is the highest reproducible sound frequency. It is half the frequency at which the clip was sampled. E g, choosing 16 kHz as sampling frequency allows you to encode audible frequencies up to 8 kHz.

10.5 Audio Speed



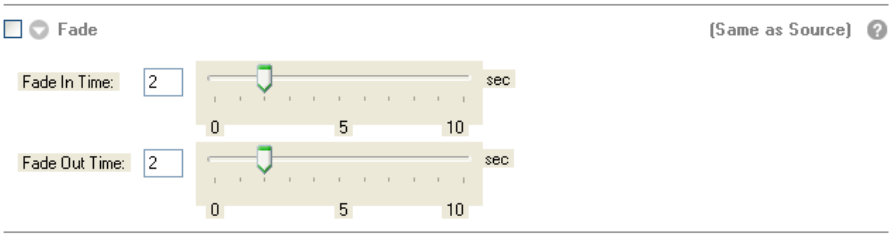
Some video frame rate conversions speed up or slow down the video speed (see section 8.5, *Frame Rate* and section 8.7, *Advanced Frame Rate*). In these cases you have to adjust the speed of the audio track to match. The **Speed change** menu contains a set of standard speed conversions: **2x : Twice the speed**, **1x : Same speed**, **1/2: Half the speed**, **1/4: A Quarter of the speed**, **To Match 24 -> 25 fps speed-up**, **To Match 23.98 -> 25 fps speed-up**, **To Match 25 -> 24 fps slow-down**, **To Match 25 -> 23.98 fps slow-down**. You can also select **Custom...** and enter any arbitrary multiplier in the **times original** field.

10.6 Offset



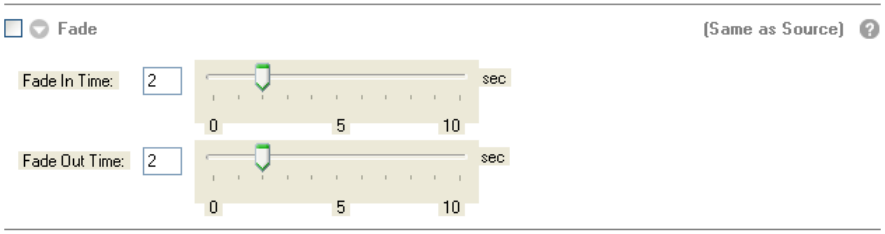
The **Offset** slider lets you add an offset to the audio track to compensate for timing differences between the audio and video tracks.

10.7 Fade



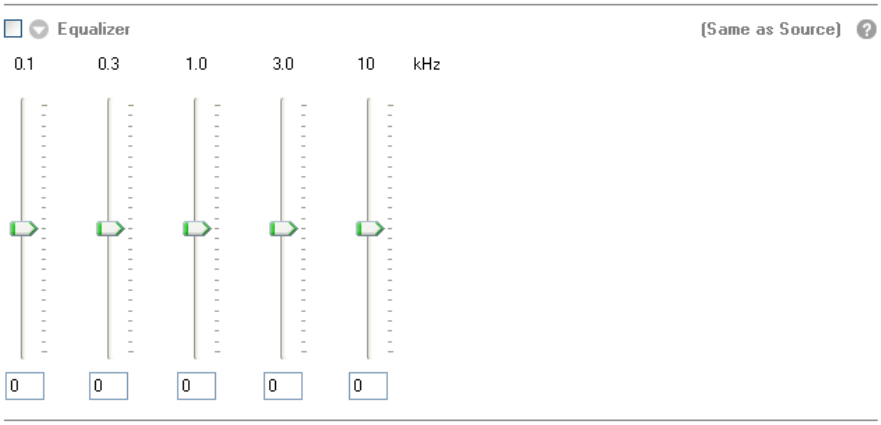
Fade fades the beginning and/or the end of the audio track. The length of the fade can be set between 0 and 10 seconds.

10.8 Balance



Balance sets the stereo panning towards the left or right audio channel. The results of using **Balance** on surround sound are undefined.

10.9 Equalizer



Equalizer is a set of five filters, each with a fixed center frequency that cannot be changed. You can control the amount of boost (peak) or cut (notch) in each frequency band: 0.1, 0.3, 1.0, 3.0 and 10 kHz.

10.10 Volume

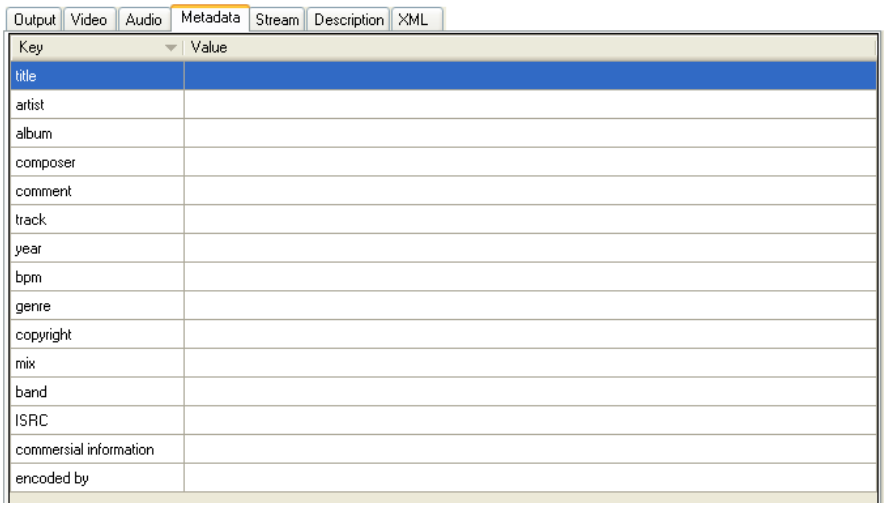
Normalize Analyse the material before encoding by looking for the loudest peak in the audio channel and then encode with this value as reference to avoid any clipping or distortion in the sound. When **Normalize** is chosen the slider shows units in percent (%). If set to 90%, this option sets the highest peak in any audio channel to be at 90% of full volume and adjusts the rest of the channels in linear correspondence.

Adjust in percent Change the volume by the value of the slider. The default value is 0% and equals no change. The value can be set from −100% to +100%.

Adjust in dB Change the volume by the value of the slider. The value can be set between −18 dB and +18 dB. The default is 0 dB and equals no change.

Normalization requires **Episode Encoder for Windows** to read and store the entire file. It is therefore easy to exceed available memory on your computer. Encoding the file in **Episode Engine Pro** using the split-and-stitch functionality minimises the memory requirements.

11 Metadata tab



Metadata is information *about* a file, such as copyright information, source file names, creation date, etc. Many media formats have data fields for metadata and **Episode Encoder for Windows** lets you set the values of any of these fields in the **Metadata** tab. Different file formats support different metadata fields, but the **Metadata** tab automatically shows what is available in the chosen output file format.

The tab contains a **Key** column and a **Value** column. The **Key** column shows all available metadata tags for the given output format. All tags are grey until you have entered a value in the corresponding **Value** field. Double-click in the **Value** column to activate the text field and enter your value. Fields for which you have not entered a value will not be added to the output file.

Note that the values are constants, you cannot enter values that depend on any properties of the source or settings files.

12 Stream tab

Streaming media are sent out in real time from a server. QuickTime- and MPEG-4-based files may contain media tracks that do not give sufficient information for a streaming server to transmit the file at the correct rate. The server therefore needs *hint tracks* that tell it how to *packetize* the media for transmission.

Accordingly, a hinted file contains up to four tracks: 1. video track, 2. hinted video track, 3. audio track, and 4. hinted audio track. The file size of a hinted file is therefore larger than the original file without containing any more information. Thus, a file should not be hinted unless it is intended to be streamed from a streaming server. Note that Apple QuickTime Streaming Server version 10.4.8 and later requires *all* streaming media to be hinted.

The correct packetizer for each media type is automatically selected when you check **Prepare For Streaming** in either the **Output** tab or the **Stream** tab.

12.1 Streamable file formats

File Format	Extension	Codecs
3GPP	`.3gp`	H.263, H.264, MPEG-4, AAC, AMR NB
3GPP2	`.3g2`	H.263, H.264, MPEG-4, AAC, AMR NB, *EVRC*, *QCELP*
MPEG-4	`.mp4`	H.264, MPEG-4, AAC
QuickTime	`.mov`	H.263, H.264, MPEG-4, AAC, AMR NB

Episode Encoder Pro

Codecs in *italics* are only available in **Episode Encoder Pro**.

NOTE

Windows Media can also produce streamable files but their streaming settings are found in the codec settings.

12.2 AAC Low Complexity

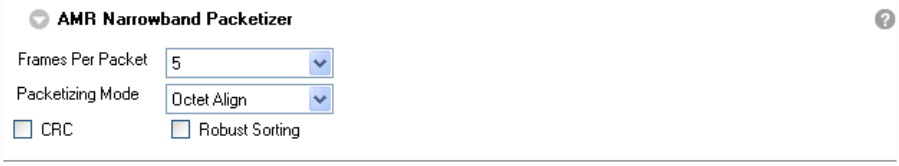
Two methods of hinting are available for AAC:

LATM (MPEG-4 and 3GPP Default) Typically for mobile phones.

Generic (ISMA and QuickTime Default) Typically for Internet use.

The **Packet Size Limit** value can be set between 0 and 1500. Setting the **Packet Size Limit** to 0 has limited utility as the stream would only consist of empty packets.

12.3 AMR NB



The options below are subject to support in the target player.

Frames per Packet The number of frames can be set from 1 to 10.

Packetizing Mode The packetizing mode can be one of two alternatives:

Octet Align Supported by all players.

Bandwidth Efficient Supported by some players.

CRC An error detection mechanism (cyclic redundancy check). Not supported by all players.

Robust Sorting Robust sorting reorders data in order to decrease the impact of transmission errors. Not supported by all players.

12.4 EVRC

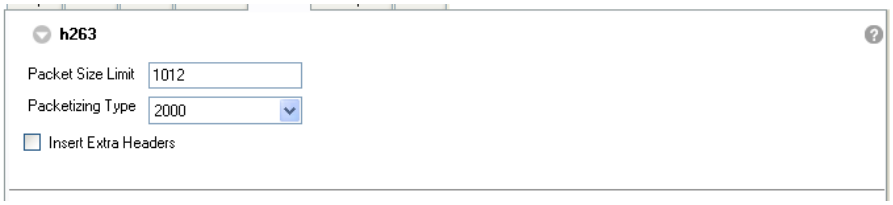
Episode Encoder Pro

Frames per packet Using a higher number of frames per packet gives better bandwidth utilisation but increases the chance that the receiver will not be able to handle all frames.

Interleave Interleaving increases the robustness against error, but also requires higher memory capacity in the recipient.

Interleave frames A higher number of interleaved frames requires more memory capacity in the recipient.

12.5 H.263



The following options are available for H.263:

Packet Size Limit The value can be set between 0 and 1500 bytes.

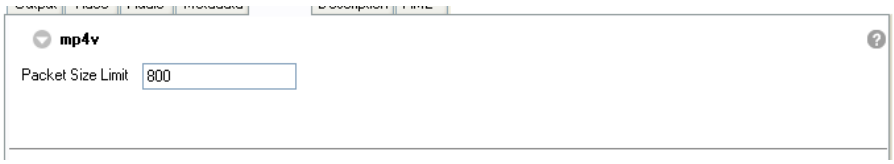
Packetizing Type The two alternatives are **1998 (QT)** and **2000**.

Insert Extra Headers Enable error correction.

12.6 H.264

Packet Size Limit can be set between 0 and 1500 bytes.

12.7 MPEG-4



Packet Size Limit can be set between 0 and 1500 bytes.

12.8 QCELP

Episode Encoder Pro

Frames per packet Using a higher number of frames per packet gives better bandwidth utilisation but increases the chance that the receiver will not be able to handle all frames.

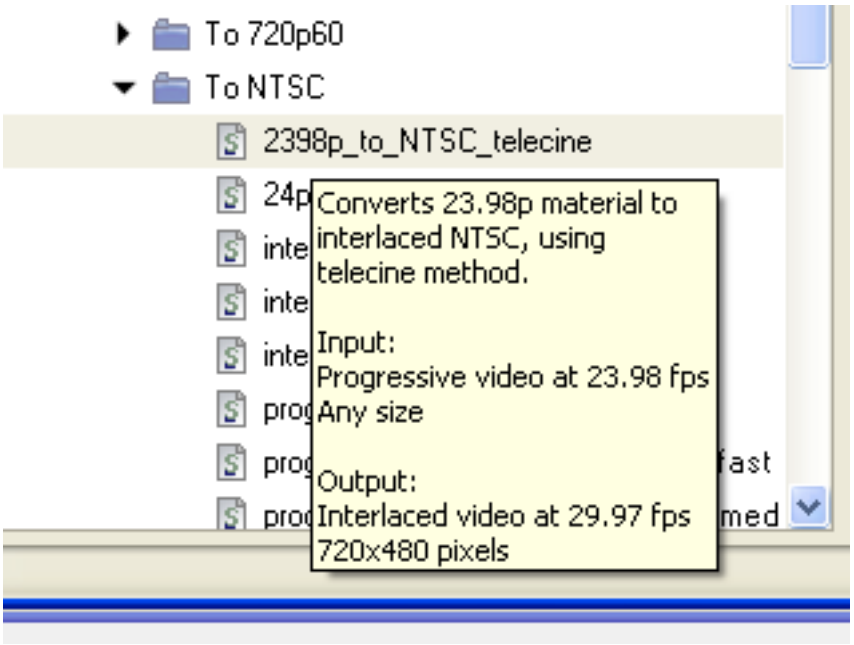
Interleave Interleaving increases the robustness against error, but also requires higher memory capacity in the recipient.

Interleave frames The number of interleaved frames, a higher number requires more memory capacity in the recipient.

13 Description tab

The description tab lets you add textual descriptions to the settings.

Enter an explanatory comment in the **Setting Description** field and save the setting. The comment is used for tooltips in the **Compression Settings** as well as for other applications using **Episode Encoder for Windows** settings.



Appendix A Supported formats

The following media formats and codecs are supported by **Episode Encoder for Windows**:

3GPP (.3gp)

The 3GPP (3rd Generation Partnership Project) video format is based on the ISO/IEC 14496 (MPEG-4) media file format and intended for mobile phones. It is defined in 3GPP TS 26.244: *Digital cellular telecommunications system (Phase 2+); Universal Mobile Telecommunications System (UMTS); LTE; Transparent end-to-end packet switched streaming service (PSS); 3GPP file format (3GP)*, see `http://www.3gpp.org/` for additional information.

Supported codecs: AAC, AMR NB, H.263, H.264, MPEG-4.

Restrictions: H.264 High Profile is input only. Multi-Bit Rate is output only.

Pro adds: H.264 High Profile support for output. HE-AAC, but for output only.

3GPP2 (.3g2)

The 3GPP2 (3rd Generation Partnership Project 2) video format is based on the ISO/IEC 14496 (MPEG-4) media file format and intended for mobile phones. It is defined in 3GPP2 C.S0050-B: *3GPP2 File Formats for Multimedia Services*, see `http://www.3gpp2.org/` for additional information.

Supported codecs: AAC, AMR NB, EVRC, H.263, H.264, MPEG-4, QCELP.

Restrictions: EVRC, H.264 High Profile, QCELP are input only.

Pro adds: EVRC, H.264 High Profile, QCELP support for output. HE-AAC, but for output only.

3GPP2 (EZMovie) (.3g2)

The 3GPP2 format can be extended with the EZMovie features developed by KDDI Corporation. Among other things, EZMovie lets a distributor limit how many times a file is played. See `http://www.au.kddi.com/ezfactory/tec/spec/ezmovie01.html` for additional information. (In Japanese.)

Supported codecs: AAC, AMR NB, EVRC, H.263, H.264, MPEG-4, QCELP.

Restrictions: EZMovie is only available in Pro and is output only.

ADTS (.aac)

Audio Data Transport Stream is a wrapper format for AAC-encoded audio files. It is defined in ISO/IEC 13818: *Information technology – Generic coding of moving pictures and associated audio information – Part 7: Advanced Audio Coding (AAC)*. See http://www.iso.org/ for additional information.

Supported codecs: AAC.

Restrictions: HE-AAC not supported for input.

AIFF (.aif)

The Audio Interchange File Format was developed by Apple. It is described in *Inside Macintosh: Sound*. See http://developer.apple.com/ for additional information.

Supported codecs: PCM.

AMC (.amc)

AMC is based on the MPEG-4 standard and has been developed by KDDI Corporation. It supports the EZMovie features. Among other things, EZMovie lets a distributor limit how many times a file is played. See http://www.au.kddi.com/ezfactory/tec/spec/ezmovie01.html for additional information. (In Japanese.)

Supported codecs: MPEG-4, QCELP

Restrictions: EZMovie files with distribution restrictions are only supported for output.

Pro adds: EZMovie only available in Pro.

AMR (.amr)

AMR (Adaptive Multi-Rate) is a mandatory audio codec in 3GPP. The file format is defined in IETF RFC 4867: *RTP Payload Format and File Storage Format for the Adaptive Multi-Rate (AMR) and Adaptive Multi-Rate Wideband (AMR-WB) Audio Codecs*. See http://www.3gpp.org/ for additional information.

Supported codecs: AMR Narrowband.

Restrictions: Input only.

Pro adds: AMR support for output.

ATSC A/52 (.a52)

ATSC A/52 is an audio format developed by Advanced Television Systems Committee, Inc. It is defined in ATSC A/52B: *Digital Audio Compression Standard*

(AC-3, E-AC-3) Revision B. See `http://www.atsc.org/` for additional information.

Supported codecs: ATSC A/52.

AVI (`.avi`)

AVI (Audio Video Interleave) is a multimedia container format developed by Microsoft. It is described in *AVI RIFF File Reference*. See `http://msdn.microsoft.com/` for additional information.

Supported codecs: DV25, DVCPRO25, DVCPRO50, MJPEG, MP3, MPEG-4 (*DivX*, *XViD*, FMP4), PCM, RGB16 (555), RGB16 (556), RGB24, RGB32, UYVY, Windows RGB, YCbCr 4:2:0, Y8, YUY2, YV16, YVU16, YVU9, YV12.

Restrictions: DivX, DVCPRO25, DVCPRO50, FMP4, MJPEG, Windows RGB, XviD, YCbCr only supported for input. **Episode Encoder for Windows** uses QuickTime to read AVI files, you can thus extend the number of codecs available by installing additional QuickTime components (popular such codecs are indicated by *italics* in the list above). However, we do not guarantee full functionality of, nor offer helpline support for any such third party components.

Pro adds: DVCPRO25, DVCPRO50, Windows RGB, YCbCr supported for output.

DV (`.dv`)

DV (Digital Video) has been developed by several producers of video cameras. DV is defined in IEC 61834: *Recording - Helical-scan digital video cassette recording system using 6,35 mm magnetic tape for consumer use (525-60, 625-50, 1125-60 and 1250-50 systems)*, DVCPRO and DVCPRO50 are defined in SMPTE 314M: *Television—Data Structure for DV-Based Audio, Data and Compressed Video—25 and 50 Mb/s*. See `http://www.iec.ch/` and `http://www.smpte.org/` for additional information.

Supported codecs: DV25, DVCPRO25, DVCPRO50.

Restrictions: DVCPRO25 and DVCPRO50 only supported for input.

Pro adds: DVCPRO25 and DVCPRO50 supported for output.

Flash (`.flv`)

The Adobe Flash video format is defined in *Video File Format Specification, Version 10*. See `http://www.adobe.com/devnet/flv/` for addition information.

Supported codecs: H.263, MP3, VP6.

Flash (`.swf`)

The Adobe Small Web Format format is a multimedia wrapper format. It is defined in *SWF File Format Specification, Version 10*. See `http://www.adobe.com/devnet/swf/` for additional information.

Supported codecs: H.263, MP3, VP6.

Restrictions: Only audio and video data are supported.

GXF (`.gxf`)

GXF (General eXchange Format) is an interchange format for storage and data transfer originally developed by Grass Valley Group. It is defined in SMPTE 360M: *General Exchange Format (GXF)*. See `http://www.smpte.org/` for additional information.

Supported codecs: MPEG-2, PCM.

Restrictions: Input only.

Pro adds: GXF supported for output.

MP3 (`.mp3`)

Properly MPEG-1 Audio Layer III. It is defined in ISO/IEC 11172-3: *Information technology – Coding of moving pictures and associated audio for digital storage media at up to about 1,5 Mbit/s – Part 3: Audio*. See `http://www.iso.ch/` for additional information.

Supported codecs: Lame MP3

MPEG Elementary Stream (`.m1a, .m1v, .m2v, .mpg`)

An MPEG Elementary stream contains a single medium, audio or video, and can in turn be contained in a Program Stream. MPEG-1 elementary streams are defined in ISO/IEC 11172-1: *Information technology – Coding of moving pictures and associated audio for digital storage media at up to about 1,5 Mbit/s – Part 1: Systems*, MPEG-2 elementary streams in ISO/IEC 13818-1: *Information technology – Generic coding of moving pictures and associated audio information: Systems*. See `http://www.iso.ch/` for additional information.

Supported codecs: AES, MPEG Audio, MPEG-1, MPEG-2

Restrictions: AES is input only.

MPEG Program Stream (`.mpg`)

An MPEG program stream contains elementary streams. Program streams are intended for reliable media such as DVD or SVCD. MPEG-1 program streams are defined in ISO/IEC 11172-1: *Information technology – Coding of moving pictures and associated audio for digital storage media at up to about 1,5 Mbit/s – Part 1:*

Systems, MPEG-2 program streams in ISO/IEC 13818-1: *Information technology – Generic coding of moving pictures and associated audio information: Systems*. See `http://www.iso.ch/` for additional information.

Supported codecs: AAC, AES, ATSC A/52, H.264, MPEG Audio, MPEG-1, MPEG-2, MPEG-4, PCM

Restrictions: AAC is output only. AES, H.264 High Profile is input only.

Pro adds: H.264 High Profile supported for output.

MPEG Transport Stream (`.m2t, .ts`)

An MPEG Transport Stream is intended for broadcast media where packets may be lost and viewers have to be able to enter a transmission in mid-stream. Elementary streams are interleaved (muxed) on the Transport Stream. MPEG-2 program streams are defined in ISO/IEC 13818-1: *Information technology – Generic coding of moving pictures and associated audio information: Systems*. See `http://www.iso.ch/` for additional information.

Supported codecs: AES, ATSC A/52, H.264, HDV, MPEG Audio, MPEG-1, MPEG-2, PCM.

Restrictions: Only supported for input.

Pro adds: Support for input and output. AAC, MP3, and MPEG-4 support for output only.

MPEG-4 (`.m4a, .m4b, .m4v, .mp4`)

MPEG-4 is intended to improve on the earlier MPEG standards. The `.m4a`, `.m4b`, and `.m4v` versions are adapted for iPods as audio, audiobook and video specialisations, respectively. PlayStation Portable can play MPEG-4 files, but requires that they be named `M4Vxxxxx.mp4`, where `xxxxx` is five decimal digits, and stored in the directory `E:\MP _ROOT\100MNV01` on the PSP. MPEG-4 is defined in ISO/IEC 14496: *Information technology – Coding of audio-visual objects*. See `http://www.iso.ch/` for additional information.

Supported codecs: AAC, H.264, MPEG-4.

Restrictions: H.264 High Profile is input only.

Pro adds: H.264 High Profile and HE-AAC supported for output.

MXF (`.mxf`)

The Material eXchange Format is a wrapper format. There are currently major interoperability problems with different implementations of MXF, so interoperability has to be tested for each case. The MXF file format is defined in SMPTE 377M: *Television - Material Exchange Format (MXF) – File Format Specification*. See `http://www.mxf.info/` for additional information.

Supported codecs: AES, BWF, D-10/IMX, DV25, DVCPRO25, DVCPRO50,

DVCPROHD, JPEG2000, MPEG-2, MPEG-4, XDCam HD. Codecs in *italics* require third party plugins. While any installed codec plugins will be used, we do not guarantee full functionality of nor offer helpline support for any such third party components.

Restrictions: All formats are input only.

Pro adds: AES, BWF, D-10/IMX, *DNxHD*, DV25, DVCPRO25, DVCPRO50, *DVCPROHD*, MPEG-2, XDCam HD supported for output.

OGG (`.ogg`)

Ogg is an open media wrapper format designed for efficient streaming and manipulation. It is defined in RFC 3533: *The Ogg Encapsulation Format Version 0*. See `http://www.Xiph.Org/` for additional information.

Supported codecs: Vorbis.

Restrictions: Output only.

QuickTime (`.mov`)

QuickTime is a multimedia framework developed by Apple. It is defined in *QuickTime File Format Specification*. See `http://developer.apple.com/documentation/QuickTime/` for additional information.

Supported codecs: AAC, AMR NB, Apple Prores, Apple Video, *Avid*, *Avid DNxHD*, *Avid DV*, *Avid Meridien*,

Restrictions: B

Pro adds: lackmagic, Cinepak, D-10/IMX, DV25, DVCPRO25, DVCPRO50, *DVCPRO100*, H.261, H.263, H.264, HDV, IMA, Mace 3:1, Mace 6:1, *Media 100*, MJPEG, MP3, MPEG-4, PCM, RGB, RGB16 (555), RGB16 (556), RGB24, RGB32, Sheer Video, Sorenson Video 1, XDCAM HD, YCbCr (YUV), UYVY, Y8, YUY2, YV16, YVU16, YVU9, YV12.AAC is output only. Apple Component, Apple Prores, Apple Video, *Avid*, Cinepak, D-10/IMX, H.261, IMA, HDV, MJPEG, Pixlet, QDesign, Sheer Video, Sorenson Video 1, Targa Cine YUV are input only. QuickTime reference files are input only. Timecodes are not supported by the native QuickTime importer—this mainly affects reference files. You can extend the number of codecs available by installing additional QuickTime components (popular such codecs are indicated by *italics* in the list above). However, we do not guarantee full functionality of, nor offer helpline support for, any such third party components.D-10/IMX, HDV, HE-AAC, MJPEG, Targa Cine YUV supported for output.

Wave (`.wav`)

The Waveform audio format was developed by Microsoft and IBM. It is described in *Multiple Channel Audio Data and WAVE Files*. See `http://www.microsoft.com/` for additional information.

Supported codecs: PCM.

Windows Media (`.wma, .wmv`)

Windows Media is a proprietary multimedia framework developed by Microsoft. See `http://www.microsoft.com/` for additional information.

Supported codecs: Intellistream, VC-1, Windows Media, Windows Media MBR, WMA Pro, WMA Standard.

Restrictions: Only the largest stream is read from Intellistream multi-bit rate files. ASF files can only be read if they have WMV9 content.

Bibliography

[1] Martin Dietz and Stefan Meltzer. CT-aacPlus—a state-of-the-art audio coding system. *EBU Technical Review*, (291), July 2002.

[2] Doug Kerr. Chrominance subsampling in digital images. *The Pumpkin*, (1), November 2005.

[3] Don Munsil and Brian Florian. DVD benchmark – part 5 – progressive scan DVD. *Secrets of Home Theater and High Fidelity*, 7(4), October 2000.

[4] Graeme Nattress. Chroma sampling: An investigation. *Los Angeles Final Cut Pro User Group*, July 2005.

[5] Davis Pan. A tutorial on MPEG/audio compression. *IEEE Multimedia*, 2(2):60–74, Summer 1995.

Index