Special Features Manual for Cinelerra 5.1

(Last update 12/31/2016)

Table of Contents

1. Introduction
  1.1 How to Build Cinelerra 5.1 from Developer's Git Repository
    1.1.1 Notable Options and Caveats
    1.1.2 Just Install It! Binaries Already Built
  1.2 The $HOME/.bcast5 Directory
2. FFMPEG Early Probe Explanation
3. How to Create FFMPEG Options Files
  3.1 File Naming Conventions
  3.2 Option File format / Content
  3.3 Complete Options File Example
  3.4 The FFMPEG Image2 Streams
  3.5 Ffmpeg Common H.264 Rendering
4. Viewing & Modifying Ffmpeg Format Options inside Cinelerra
5. FFMPEG Items of Note
6. Problem Resolution Help
  6.1 What to Include in Problem Reports
  6.2 Crash Dumps for Analysis
7. Speed/Fade Automation Usage and Auto Gang
8. How to Create a Blu-ray or Standard DVD
  8.1 Output Terminal Messages from Creating DVDs
  8.2 Helpful Hints for Problems/Debugging DVDs Creation
  8.3 Additional Features for Creating Blu-ray/standard DVDs
    8.3.1 Subtitles
    8.3.2 Dvd Interlaced Chroma
    8.3.3 Mpeg Utilities
    8.3.4 HDV-sourced Video Recording on a Blu-ray Disc Without Re-encoding
    8.3.5 *(Future)* Ultra HD Media Creation
  8.4 Complete Example of Video Source with 4:3 Aspect Ratio Being Transcribed to 16:9 and Creating a DVD to be Displayed on a Digital TV
9. Recording Web Media or Television and Playback
  9.1 Record Web Media by real-time Audio Capture/Screen Capture
  9.2 Digital Video Broadcasting (DVB) Capture Configuration
    9.2.1 Overview
    9.2.2 Initial Setup for Digital Video Broadcast Capture
    9.2.3 Watching TV using Cinelerra
    9.2.4 Recording TV to Capture at some Future Time via Batch
    9.2.5 Recording Immediately to Capture a current TV program
    9.2.6 Record “on the fly” Table of Contents (TOC)
    9.2.7 Watching a Previously Recorded TV program
    9.2.8 Dual Screen Mode
    9.2.9 Remote Control for DVB
    9.2.10 Android Remote Control for DVB
    9.2.11 How to Setup a Remote Control Device to work with Cinelerra

9.3 The commercial DB… What is that? A TDB – Traveling Data Base
10. Themes User Interface
11. Resizable Fonts/Icons Support
12. Vicons – aka Video Icons
13. Toolset Visibility Support
14. Menu Bar Shell Commands
15. Last Play Position Memory
16. Audio H/W Gain Support
17. Valgrind Support Level
18. Subtitle Track and Subtitle Script Support
19. CFLAGS has -Wall
20. Multi-Viewer Window Support
21. Overlay Modes - Alpha Blending with PorterDuff and Operators
22. Titler Font Kerning
23. Edit Length
24. Align Edits, Reverse Edits, and Shuffle Edits
24.1 Align Edits
24.2 Reverse Edits
24.3 Shuffle Edits
25. Delete Plugins to save Resources Window Space or whatever
26. Program Selection Support
27. Realtime Priority Audio
28. Info Asset Details
29. Multi-Pane Support
30. Nested Assets
31. Keyframe Edit Params
32. Color Coded Keyframe Curves/Different Operational Characteristics
33. Playback 5.1→2 Driver Downmix
34. Multi-screen / Playback Configuration
35. “Free” Dragging Plugins
36. Simple Animation – Filescene Using Festival
37. Some Plugins – New or Newly Documented
38. Cinx and a “Bit” of Confusion
39. Lossless Rendering
40. Other note-worthy minor Enhancements
40.1 Replace-type Insertion Strategies
40.2 Sort within Sort in File Load Window
40.3 Playback Speed Automation Support
40.4 Keyframe Reticle
40.5 Updatable Icon Image Support
40.5.1 Details on where to put Plugin Icons
40.5.2 Example of new Plugin Icon Testing
40.6 Masking in Composer Window Key Difference
40.7 Plugin – SVG via Inkscape
40.8 Hide Keyframe Type
40.9 Button Operation Preference
40.10 Project SMP cpus Count

Appendix A – Crash Handling Tips
Appendix B – Developer's Section
  B.1 How Builds Really Work and Even More Options
  B.2 Thirdparty Parallel Build
  B.3 Translation Assistance for Program Messages

1. Introduction

The CV Cinelerra Manual covers “The Secrets of Cinelerra” and can be found at:

https://cinelerra-cv.org/docs/cinelerra_cv_manual_en.html

This manual covers features or areas of Cinelerra that are different or new in the 5.1 version from the standard CV Community version. Many of the new features were created by a lot of different people to include the original author and were merged in from various sources and each person is to be commended for their efforts. The content of this manual also came from various people/places/email and thanks to all.

1.1 How to Build Cinelerra 5.1 from Developer's Git Repository

These are generic build instructions for building Cinelerra 5.1. Known to work on ubuntu, mint, suse, fedora, debian, centos, arch, and 32-bit slackware. It has not been tested on every single possible distro yet so you might expect to have to make some minor changes. It has failed miserably on FreeBSD. At least once, a Gentoo version was successfully created. A 32-bit Debian version has been built by wrapping the appropriate stuff around the ubuntu binary.

Alternatively, there are some pre-built 5.1 dynamic or static binaries which are updated on a fairly regular basis (as long as code changes have been made) available at link below. See Section 1.2.

https://cinelerra-cv.org/five

There are 2 kinds of builds, the default system-build and a single-user build. A system build has results which are installed to the system. The majority of the files are installed in the standard system paths, but some customization is possible. The single user build allows for running completely out of a local user directory so it doesn't affect the system.

We recommend the single-user version when possible. It makes it very easy to install a new version without having to delete the older version in case you want it for backup – once you are happy with the new version, all you have to do is delete the entire old directory path. Also, if you install a new Operating System version and if you have Cinelerra on separate disk space that is preserved, you won't have to reinstall Cinelerra. In addition for purposes of having the ability to interrupt or to see any possible error messages, if you start the application from a terminal window command line you will have more control to catch problems. The system builds can be useful in a university lab setting where there are possibly multiple users, or multiple versions.

There are two notable differences between “standard” views of Cinelerra and this implementation for the system builds. Both of these can be configured during installation. These differences make it possible to have several different versions installed without having them “walk” on each other.

1) application name can be set during installation and defaults to: “cin”
2) the home configuration directory can also be set and defaults to: “$HOME/.bcast5”

To do a system build, you should read the README that is at the top level after you get the source.

1) You need at least 2.5GB of disk storage to operate a build. Need to have “git” installed.
2) Obviously in order to install into the system, you must run as root.
3) The "git" step has to download a bunch of stuff (approx 100MB) so allow time.
4) Run the following commands (this takes awhile):

```
cd /<build_path>/            # this is where you need the 2.5GB of disk space
git clone --depth 1 "git://git.cinelerra-cv.org/goodguy/cinelerra.git" cinelerra5
cd cinelerra5/cinelerra-5.1 # toplevel directory
```

NOTE: if your system has never had Cinelerra 5.1 installed, you will have to make sure you have all of the compilers and libraries necessary. So on the very first build you should run:

```
./bld_scripts/bld_prepare.sh <os>    # where <os> represents the Operating System of
   centos, fedora, suse, leap, ubuntu, debian.
./autogen.sh
./configure --prefix=/usr     # optional parameters can be added here
make 2>&1 | tee log           # make and log the build
```

5) Check for obvious build errors:

```
grep "\*\*\*.*error" -ai log
```

If this reports build errors, email the log which is listed below to cinelerra@lists.cinelerra-cv.org :
/<build_path>/cinelerra5/cinelerra-5.1/log

6) If there are no build errors, finally just run:

```
make install
```

7) If it all worked, you are all setup. Just click on the “blue lady” cinelerra desktop icon.

To do a single-user build, read the README that is at the top level after you get the source.

1) You need at least 2.5GB of disk storage to operate a build. Need to have “git” installed.
2) Recommend you build and run as root, just to avoid permission issues initially.
3) The "git" step has to download a bunch of stuff (approx 100MB) so allow time.
4) Run the following commands (this takes awhile):

```
cd /<build_path>/            # this is where you need the 2.5GB of disk space
git clone --depth 1 "git://git.cinelerra-cv.org/goodguy/cinelerra.git" cinelerra5
cd cinelerra5/cinelerra-5.1
```

NOTE: if your system has never had Cinelerra 5.1 installed, you will have to make sure all of the compilers and libraries necessary are installed. So on the very first build you should run as root:

```
./bld_scripts/bld_prepare.sh <os>    # where <os> represents the Operating System of
   centos, fedora, suse, leap, ubuntu, debian.
./autogen.sh
./configure --with-single-user       # the “with-single-user” parameter makes it so
make 2>&1 | tee log                  # make and log build (check for errors before proceeding)
make install
```

Then just start the application by keying in: ./cin in the bin subdirectory OR add a desktop icon by using the appropriate directory to copy the files to, run as root, and edit to correct the directory path.

cd /cinelerra_directory_path
cp -a image/cin.{svg,xpm} /usr/share/pixmaps/.
cp -a image/cin.desktop /usr/share/applications/cin.desktop
change the “Exec=cin” line to be “Exec=<your_directory_path>/bin/cin”

The preceding directions for doing a single-user build has been meticulously followed to build and run on a newly installed ubuntu 15 system WITHOUT BEING ROOT except for the bld_prepare.sh and creating the desktop icon.

1.1.1 Notable Options and Caveats

These procedures and the 5.1 Cinelerra software have all been run as “root” on home laptops/desktops. This provides the best chance to ensure all works correctly and also allows for handling errors, other problems and potential crashes with the most success. Included in this section are some of the build variations easily available for normal builds.

To see the full list of features use: ./configure –help
The default build is a system build which uses: ./configure –without-single-user
In the single-user build, the target directory is always “cin”. Because this is also the developer build, constant names are used throughout. However, you can rename files after the install is complete.

If your distro/operating system has issues with the default install to /usr/local, you might have to change the location to /usr for a system build. Then you will have to use:
./configure --prefix=/usr

If you wish to change the default directory for a system build you will have to add the destination directory path on the “make install” line. For example:
make install DESTDIR=<your selected target directory path>

The application name can be set during installation, but defaults to cin so that the 5.1 build can coexist with other Cinelerra builds if necessary. To override the default cin name, use:
./configure --with-exec-name=cinelerra

The home configuration directory can also be set, but defaults to $$HOME/.bcast5. For example:
./configure –with-config-dir=/myusername/.bcast5
NOTE: when you specify parameters to the configure program, it will create a make file as a consequence. Since in a make file, the $ is a special character, it must be escaped so in order to represent a $ as part of an input parameter, it has to be stuttered. That is, you will need $$ (2 dollar signs) to represent a single dollar sign.

It may be necessary on some distros which have missing or incomplete up-to-date libraries, to build cinelerra without Ladspa. To do so, use:

./configure --prefix=/usr --without-ladspa-build

Note that the with-ladspa-dir is the ladspa search path, and exists even if the ladspa build is not selected. This gives you the ability to specify an alternate ladspa system path by utilizing the LADSPA_PATH environment variable (that is, the default ladspa build is deselected).

To build a 10-bit version for x264 and x265, it is a sad truth that the x264 and x265 libraries require "special" builds to encode 10/12 bit output, and so a separate version of cinelerra is needed to create these kinds of codec output. To build these:

```
./configure --with-exec-name=cinx --enable-x264_hidepth --enable-x265_hidepth
```

Unfortunately, these libraries can only be configured to encode either 8-bit or 10/12-bit but not both compiled simultaneously. The 10-bit compiled Cinelerra can read 8-bit or 10-bit but can only render to 10-bit. Newer video equipment may take advantage of the 10-bit color depth for improved quality.

Note for 32-bit 14.2 Slackware, before running the configure you will need the following exports:

```
export ac_cv_header_xmmintrin_h=no
export FFMPEG_EXTRA_CFG=" --disable-vdpau"
```

1.2 Just Install It! Binaries Already Built

If you prefer to not have to take the time to build Cinelerra 5.1 yourself, alternatively there are some pre-built 5.1 dynamic or static binaries for various versions of ubuntu, mint, suse, fedora, debian, centos, and arch-linux. There is a 32-bit i686 ubuntu version and a 32-bit slackware version available also. These are updated on a fairly regular basis as long as significant code changes have been made. They are in subdirectories of:

https://cinelerra-cv.org/five/tars
https://cinelerra-cv.org/five/pkgs

The **"tars"** directory contains single-user static builds for different distros. This is the recommended usage of 5.1 because all of the files will exist in a single directory. To install the single user builds, download the designated tarball from the ./tars subdirectory and unpack as indicated below:

```
cd /path
mkdir cin
cd cin
tar -xJf /src/path/cinelerra-5.1-*.txz    # for the *, substitute your distro tarball name
```

Another 32-bit version for Debian can be downloaded from:
http://bandshed.net/packages/cinelerra-5/
with the debian target installed at /opt/cinhv-5/ . However, it may be an older version that does not contain the latest modifications.

Do NOT download the LEAP 10bit version unless you use h264/h265 (they can't render 8bit h26x).

The **"pkgs"** directory contains the standard packaged application for various distros. This will install a dynamic system version for users who prefer to have the binaries in the system area and for multi-user systems. In addition, performing the package install checks the md5sum in the file md5sum.txt to ensure the channel correctly transmits the package. There is a README.pkgs file in the "pkgs" directory with instructions so you can "cut and paste" and avoid typos; it is also shown next.

README.pkgs:
Depending on the distro, use the instructions below and select the appropriate
setup operations to install, update or remove cinelerra 5.1.
To upgrade, refresh repo, then replace "install" with "update", or whatever.

FEDORA
dnf install cinelerra --repofrompath cincv,https://cinelerra-cv.org/five/pkgs/fedora/
##dnf erase cinelerra

CENTOS
first create the file /etc/yum.repos.d/cincv, with the following contents:
[cincv]
name=cincv
baseurl=https://cinelerra-cv.org/five/pkgs/centos
gpgcheck=0
# end of cincv
yum install cinelerra
##yum erase cinelerra

UBUNTU/DEBIAN, replace ub15 with your distro id: ub14,ub15,debian,mint (see note below for mint)
Some ubuntu apt downloads register status as working 0% constantly while running the package
download, like ubuntu 14. It may take a few minutes for this step so be patient.
apt install software-properties-common apt-transport-https
apt-add-repository https://cinelerra-cv.org/five/pkgs/ub15
apt update
apt install cin
#to update a previous install
apt update
apt upgrade cin
##apt remove cin

MINT note: apt-add-repository did not work for me, I had to use the gui version:
Administration->Software Sources->Addtional Repositories->Add a new repository
deb http://cinelerra-cv.org/five/pkgs/mint trusty main
#then: apt install cin (or apt remove cin)

UBUNTU 16 - This has been known to work, but things change quickly:
apt install software-properties-common apt-transport-https
apt-add-repository https://cinelerra-cv.org/five/pkgs/ub16
VIP - for the first install, the above line adds cinelerra to /etc/apt/sources.list but...
Version 16 of Ubuntu is more strict for licensing so you will have to edit
the file /etc/apt/sources.list to add [trusted=yes] after deb and before https...cin...
For example the line should be: deb [trusted=yes] https://cinelerra-cv.org/five/pkgs/ub16 xenial main
Also, on the install you will get an error message that you can either ignore as cinelerra
will run anyway, or else (the first time only) on the commnand line keyin:
echo > /etc/sysctl.d/50-cin.conf "kernel.shmmax=0x7fffffff"
apt update
apt install cin

```
#to update a previous install (ignore any i386 errors as only 64 bit version available):
apt update
apt upgrade cin
##apt remove cin

# SUSE/LEAP, replace leap with your distro id: suse, leap
zypper ar -f https://cinelerra-cv.org/five/pkgs/leap/ cincv
zypper install -r cincv cinelerra
##zypper remove cinelerra
#to update a previous install (assuming you enabled autorefresh as above)
zypper refresh cinelerra
zypper up cinelerra # or cinelerra10bit for 10 bit

# ARCH linux
# first edit the file /etc/pacman.conf, to include the following:
[cincv]
SigLevel = Optional TrustAll
Server = https://cinelerra-cv.org/five/pkgs/arch
# end of cincv
pacman -Sy
pacman -S cin
##pacman -R cin
```

1.3 The $HOME/.bcast5 Directory

A difference in Cinelerra 5.1 is the name of the user's default settings and saved files directory. It is called .bcast5 as opposed to .bcast as a subdirectory in $HOME. Initially when cinelerra is launched there is an empty project and there are program default settings, and from then on the .bcast5 directory will contain the settings that were created before the last closing. If you need to revert to the default settings, delete the .bcast5 directory contents and restart Cinelerra. Or you may want to rename it temporarily if you think you might want it back later.

2. FFMPEG Early Probe Explanation

When you open media, a series of libraries and codec functions are used to "probe" the data, to see if it can guess the type of file format and codec parameters needed to correctly decode the file. If ffmpeg probes early - “Try FFMpeg first” is in effect for the button - it will usually find some way to try to decode just about any contemporary media file. But there are some times that the built in codecs are actually a better choice. A lot of this may fall into the category of personal preference. For example, some may prefer the mpeg library in the cinelerra code over the ffmpeg code because it has more decoding capability and seems to be more robust when the media is damaged. In that case you will want the *FF* button to read “Try FFMpeg last” in the upper right hand corner of the main window.

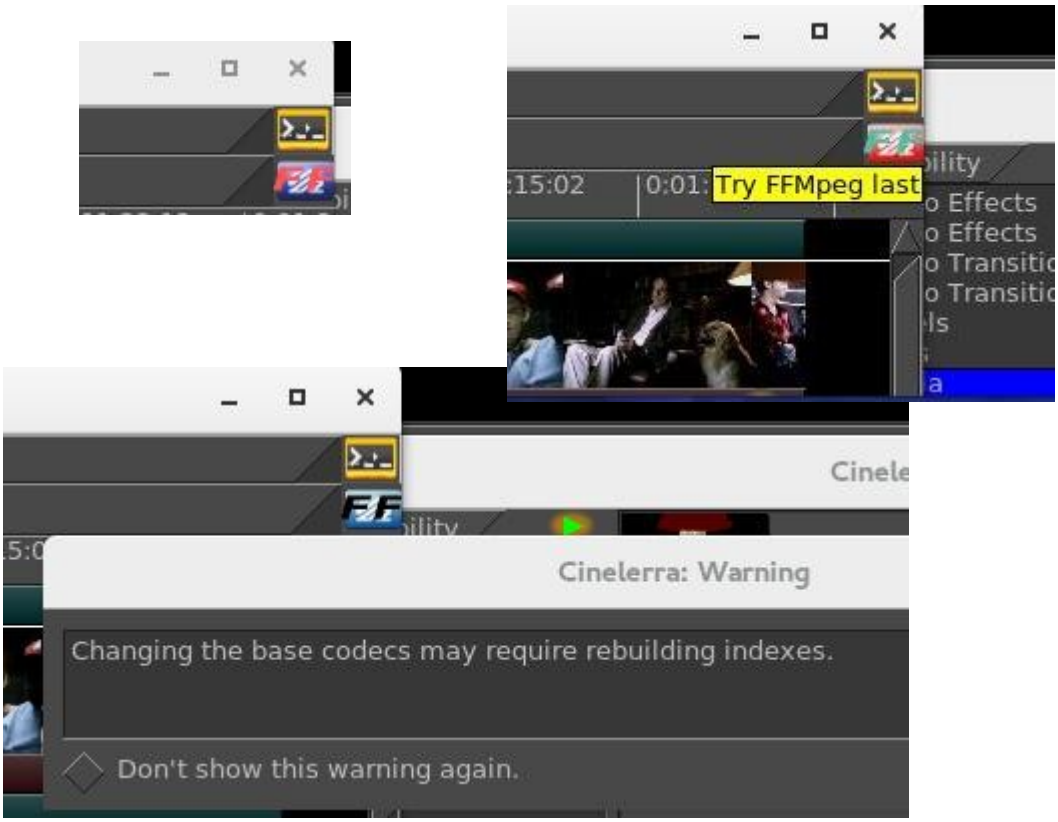
So, if ffmpeg probes early, you will never get to use the built in libraries, and if you want to skip over buggy old libraries, use ffmpeg early probe enabled so that the newest code will be tried first.

When the icon is red, ffmpeg probes early is enabled and you will see it reads “Try FFMpeg first” when

moving over the *FF* button in the upper right hand corner of the screen. When the icon is black, ffmpeg probes early is disabled so that ffmpeg probes late and it reads “Try FFMpeg last”. The initial default state of the icon is off, that is, ffmpeg probes late. This is the original code behavior before ffmpeg code was added. Suggestion is to leave it on except in a few special cases where it may be better to have early probes disabled. When you mouse over the main menu ff toggle button, the text displays ffmpeg's first/last probe position.

The ffmpeg early probe state is saved between sessions. It is important to note that the various file indexes may need to be rebuilt if you change which codec is being used to decode the file. There is a warning popup to remind you when you change the default ffmpeg early probe state (unless you have checked the box to no longer show you the warning).

Screenshots below show (1) reddish colored *FF* in upper right hand corner of main screen indicating that ffmpeg early probes is enabled; (2) “Try FFMpeg last” indicator message for ffmpeg early probes enabled (note that the color is different because you highlighted the icon); and (3) black colored *FF* indicates ffmpeg will be used last and you are changing the behavior so that Cinelerra warns you accordingly.



3. How to Create FFMPEG Options Files

This section describes how the FFMpeg options files work for decoding and encoding and goes into great detail. It will make more sense if you look at cinelerra's ffmpeg config directory and the cinelerra menus at the same time. It is meant to include everything necessary for complete understanding. You

will be able to personalize your own options files without knowing all of the information included below if you know the basics. The word encoding is used interchangeably with the word rendering.

The possible combinations for ffmpeg options files are literally combinatorial – that is a lot (factorial!). The allowed media file format / codec choices are much more flexible than you might realize. When the ffmpeg design was initially added, some parameter files which describe the choices which the program uses had to be created. There are way too many to enumerate in the deliverable Cinelerra package. Some quite detailed information for how ffmpeg options work is given here and hopefully, enough basics for simple understanding. It may all seem complicated at first, but will become obvious.

3.1 File naming convention

In Cinelerra's ffmpeg configuration directory you will see files as listed and described below. File type and extension names are the key for Cinelerra's use of ffmpeg. Basically the .opts file extension represents options; .dfl represents defaults; and all the rest are media types. For example one media type is quicktime so that *.qt file names would be the “quicktime” choices. In the file names below, “ext” refers to a set of files with file names matching the *.ext file extension. And “typ” refers to a type of format / codec combination used, i.e, the media type.

In the ffmpeg configuration directory there are a series of options files used when encoding or decoding audio or video. They are read in the order from top to bottom and only the files needed for the current operation are added to the active configuration.

ffmpeg/ffmpeg.opts = global ffmpeg options, always used
ffmpeg/decode.opts = global decoder options, used when opening existing files for decoding
ffmpeg/encode.opts = global encoder options, used when creating new files for encoding
ffmpeg/audio/audio.opts = audio encoder options, used when creating audio streams
ffmpeg/video/video.opts = video encoder options, used when creating video streams
ffmpeg/plugin.opts = parameters for ffmpeg filters as audio/video plugins

Decoder options:

Normally, only ffmpeg.opts and decode.opts are used when reading/decoding files, but may be specialized if a <path>/media.opts exists for a given <path>/media.ext file. For example, if you want to only fail on fatal errors and to always use the video filter, edgedetect, when working with your media file “dreaming.y4m”, then create a file “dreaming.opts” in the same directory with the contents of “loglevel=fatal” on the first line and “video_filter=edgedetect” on the next. These specialized settings will override the defaults. The fatal loglevel is especially handy for lesser quality media.

Encoder Options:

Within the audio /video subdirectories of the first level ffmpeg directory, the typ.ext files are for encoder (rendering) setups.

ffmpeg/audio = directory of audio encoder settings
    audio.opts = options used by all audio encoders
    typ1.ext, typ2.ext, … = are all *.ext type choices for encoding audio
    ext.dfl = contains the default selection used when ext is first selected

ffmpeg/video = directory of video encoder settings
video.opts = options used by all video encoders
typ1.ext, typ2.ext, … = are all *.ext type choices for encoding video
ext.dfl = contains the default selection used when ext is first selected

ffmpeg/format = presets needed to initialize audio / video formats

3.2 Option File Format / Content

For the option files a specific format must be followed in creating the file content.

In typ.ext encoder parameter files, the first line is defined as:

```
    muxer codec
or  muxer codec | bitstream filter [ bitstream filter options ]
```

where the | represents piping the codec data through the bitstream filter.

The rest of the lines in the file should look as follows:

```
    # in column one is a comment
    id1   value1
or  id2 = value2
```

Only one equals sign is allowed and it is just for readability. There may be any number of id / value pair lines in a media definition, including zero. A typical line might be:

```
    bitrate 4000000
or  bitrate = 5000000
```

There are 4 special id's recognized by cinelerra which cause special processing. They are:

duration	(overrides the probe duration when opening media for decoding)
video_filter	(adds an video stream filter, eg. edge detect, … at the stream level)
audio_filter	(adds an audio stream filter, eg. echo, … at the stream level)
loglevel	(sets the library logging level, as quiet, panic, … verbose, debug)

All other id's should be in the ffmpeg documentation, and correspond to the global, muxer, and codec option names and values used by ffmpeg. For example to set the aspect ratio to 4:3, use:

```
    aspect 4:3
```

Below shows an example: "decode.opts" which is used when the ffmpeg decoder is initialized.

```
    # apply at init decode
    loglevel=fatal
    formatprobesize=5000000
```

scan_all_pmts=1
threads=auto

The encoder options you see in the cinelerra menus depend on the files in these directories, NOT THE CODE. If you add files, you will get to use more variety.

In the "cinelerra" directory, which contains the ffmpeg configuration folder, there are the choices the program uses. When you open an ffmpeg format popup dialog, the listbox contains all of the codec types which are identified by the file.ext extensions. Decoding has only a few options, since the ffmpeg file probes determine most of the options by looking at the media being opened, but encoding media requires a lot of setup. Below are some of the folders and files used to determine the configurations used by ffmpeg to decode and encode files.

These extensions create audio / video media classes:

dvd m2ts mkv mp3 mp4 mpeg qt pro

which become the choices in the render pulldown menu

So if you want to create a "mov" codec class, add two new files to the ffmpeg configuration directory:

audio/aud.mov and video/vid.mov

Now you will see this as what you can choose in the rendering choices for ffmpeg.

Inside the file you will see that the first line is special. It is the muxer and codec. For example:

h264 libx265

The contents may be something like: <path>/video/vid.mov =
mp4 libx265
bitrate 4000000

This will code an mp4 formatted file using the lib264 codec encoder.

For audio and video together, t**he mux format must agree** between the aud.mov and vid.mov files when they are to be used together. The stream muxer must be the same for the all the streams in the file being written.

for example: <path>/audio/aud.mov =
mp4 pcm_mulaw

This will create mp4 media using audio format pcm_mulaw coding.

Both the audio and the video are using mp4 mux format, and so there will be 2 streams:

1) x265 video
2) pcm_mulaw audio

When the menu popup is created, there may be many choices for that class type, so you may want defaults. That can be specified as:

audio/<class>.dfl and video/<class>.dfl
so: audio/mov.dft =
aud.mov
and: video/mov.dft =
vid.mov

The above will be the default choice when the menu opens.

When you see problems in using the new options files you have created and put into place, add the following line to ffmpeg/encoder.opts:

loglevel=verbose

sometimes that will be enough to see what has caused a failure, or even catch unexpected results.

Now there is an EXCEPTION to all of the above because of a conflict between ffmpeg and the x264 person making the detect ion of default ffmpeg settings terminate with an error. If you get this error, you must workaround this termination by including parameters that don't match 5 or more of the normal expected values. So you just have to change a few parameters to avoid the probe detection. Here is an example where you will notice the x264opts line tweaking values to throw off the detection/error termination code.

<path>/ffmpegvideo/test.mp4 =
mp4 libx264
preset=slow
x264opts keyint=25:min-keyint=4:qpmin=3:qpmax=33:qp_step=4:merange=8
crf 20

Another EXCEPTION is for some forms of aac, which is the reason had to add bitstream filters. If you think that you are in this situation, use:

<path>/ffmpeg/audio/test.mp4 =

mp4 libfdk_aac | aac_adtstoasc

For more examples, look around the ffmpeg dir for examples which may be close to what you are trying to use, and see if the parameters look usable.

This is quite complicated, but that is because ffmpeg has a lot of parameters and history. Good results are not that hard to create. Initially you should mostly use the defaults. If you send any new options files to cinelerra@lists.cinelerra-cv.org , it may be given consideration to being added to the baseline for the possibility for future deliverables.

To get a listing of the current ffmpeg supported formats and codecs that can be made to work with cinelerra, provided there are option files added, run the following commands. This should be done from the <build>directory substituting the location of <build> where you have installed cinelerra on

your system and the ffmpeg may be a later version than 3.1.2 as used below. Then look at the output created in /tmp/ff-formats.txt and codecs.txt.

/<build>/cinelerra-5.1/thirdparty/ffmpeg-3.1.2/ffmpeg -formats > /tmp/ff-formats.txt
/<build>/cinelerra-5.1/thirdparty/ffmpeg-3.1.2/ffmpeg -codecs > /tmp/ff-codecs.txt

3.3 Complete Options File Example

For illustrative purposes, here is an example of the options files that need to be added for using the ffmpeg ProRes 422 format. This makes it possible to transcode to h264.mov with FFmpeg retaining 10-bit yuv422p from the source to the target output video.

Add the file named "./ffmpeg/audio/acc256k.pro" which contains the following lines:

```
mov libfdk_aac | acc_adtstoasc
strict -2
b 256000
```

(Note that in the example above, even though the bitrate is set here to 256000, it can be overridden by the render menu settings).

Add the file named "./ffmpeg/audio/pro.dfl" which contains the following lines:

```
acc256k.pro
```

Add the file named "./ffmpeg/video/med422p10.pro" which contains the following lines:

```
mov prores
preset medium
pix_fmt yuv422p10le
```

Add the file named "./ffmpeg/video/pro.dfl" which contains the following lines:

```
med422p10.pro
```

Then to use and to get 10 bit depth and preserve depth from decode to encode:

1) load media
2) use settings->format to set the frame rate, sample rate/channels, aspect ratio,
   color model = rgb_float or rgba_float if blending
3) press Shift-R and select FFMpeg format type "pro"
4) select target path
5) check ok, and watch for messages in the terminal window

3.4 The FFMPEG Image2 Streams

Another feature gained from using ffmpeg in cinelerra takes advantage of what is being referred to as the "%d trick". This trick uses the ffmpeg muxer image2 and a filename template to create a series of image files of a given kind. A specific example is described below.

To encode a series of 48 bit tiff output image files, add a file to the cinelerra data ffmpeg/video subdirectory as in:

filename: .../ffmpeg/video/tiff.dfl containing the line:
tiff48.tif

Then create an ffmpeg video encoder parameters file in the same directory:

filename: .../ffmpeg/video/tiff48.tiff containing the next 2 lines:
image2 tiff
pixel_format=rgb48

This will define a new ffmpeg encoder format which is a video image file format that uses the tiff codec for encoding, and a pixel_format of rgb48 (or a similar equivalent such as rgb48le). Next load up your project and set up for a Render using File→Render in the usual way. Now the tricky part; the output file name should contain a %d which will be the frame number used in the image output filename as in: Select a file to render to = /tmp/tiff_images/img%03d.tiff. You will get multiple files as output – one for each frame!

The resulting directory of images can be opened for reading by simply opening the template path. As in: File→Load Files = /tmp/tiff_images/img%03d.tiff. You will notice a file named as same as the template, which has been automatically created, is empty,is needed, and has to remain with the set.

3.5 Ffmpeg Common H.264 Rendering

Because H.264 is so widely used, the method in 5.1 Cinelerra is outlined below. These setup steps make it easy to just get started.

File->Render
File Format->FFMPEG + mp4
Video Wrench->Preset->h264.mp4 + bitrate: 6000000 (or whatever) +OK
Audio Wrench->Preset->h265.mp4 + bitrate: 224000 (or whatever) + OK
Set your target path in: Render=>Select a file to render to
Set your timeline in: Render=>Render range + click Project
Set your insertion strategy: Replace project (or whatever)
Press OK to start rendering.

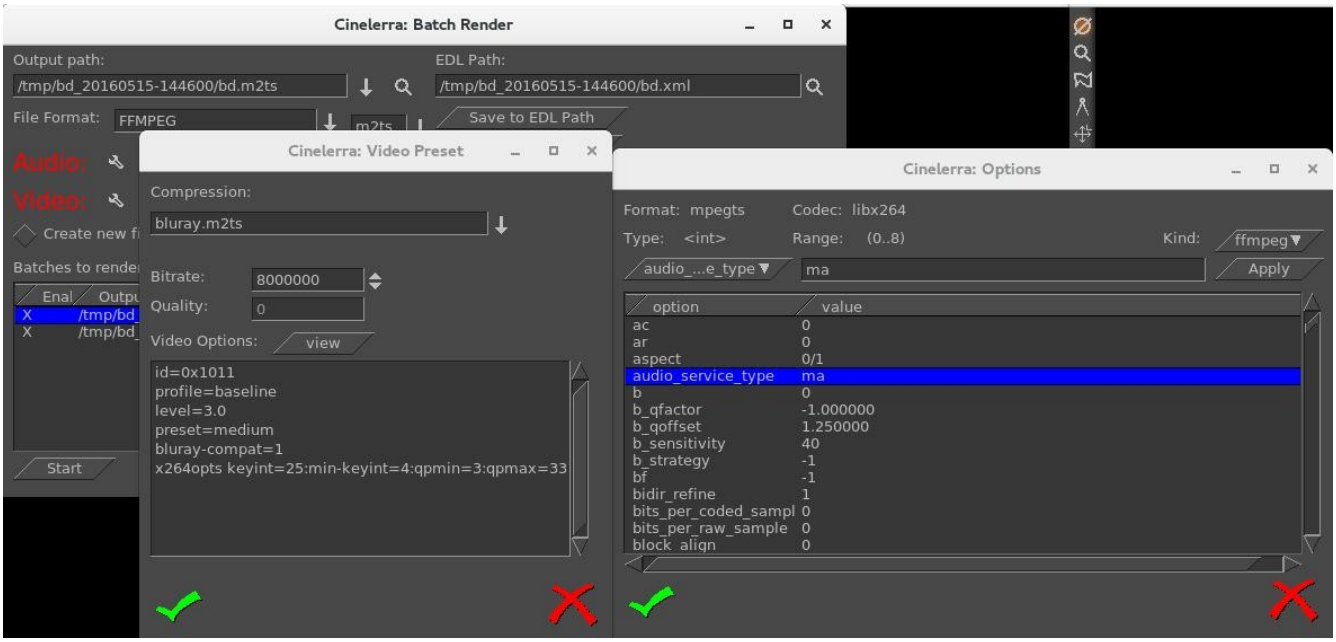
4. Viewing & Modifying Ffmpeg Format Options inside Cinelerra

There are thousands of options for using ffmpeg. Now it is possible to “view” the available options for a particular video and audio choice by using the “wrench icon” and then clicking on the “view” box. Ffmpeg has to be the selected file format for this feature to be visible. It makes it a lot easier since only the applicable options show up as opposed to everything that ffmpeg can do. These options are just “Hints” and some may be missing due to the way that ffmpeg options are coded – Cinelerra shows the option data ffmpeg has exposed.

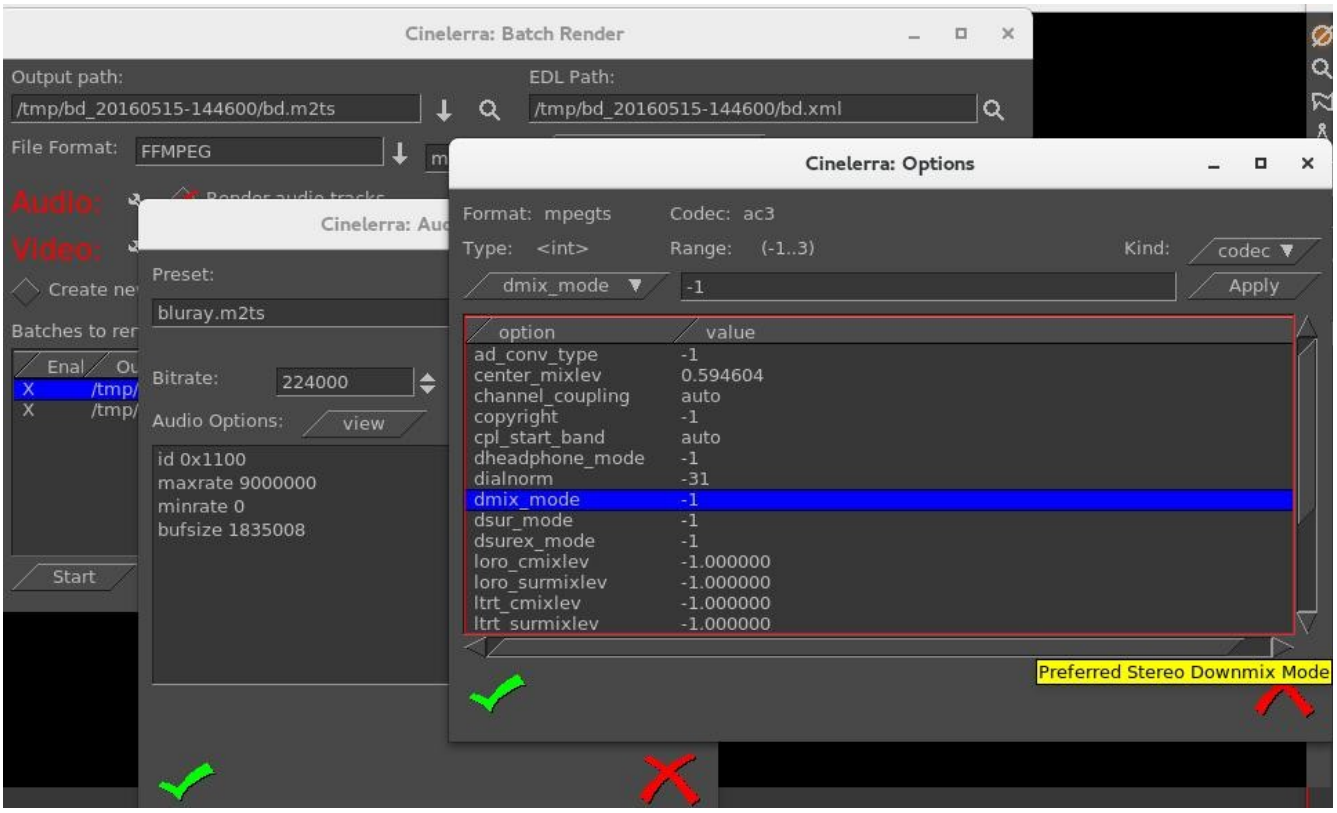
As an example, instead of reading the entire 264 library information, you only have to look at the shown available options. Both the video and the audio are browsable. The options visible in the Audio/Video Preset textbox are the final values which are used when rendering once you have checked OK. For assistance in choosing the options you want, use the view popup to see the objects that go with the selected format tool, highlight the option, modify the parameter value in the box at the top of the “Options” window based on what you want, and then click apply. Updates modify parameter values or new parameters will be appended at the bottom. Note that when you highlight an option a tooltip will show up when available in the lower right hand corner which describes the option. Also note that the Format and Codec types are shown on the top line of the Options window.

Parameters exist in 3 layers: ffmpeg, codec, and an interface layer. You can apply parameters to each layer. The top 2 layers are accessed with the Kind popup menu. The ffmpeg layer is topmost, and is selected as Kind: ffmpeg. It can specify many of the more common parameters, such as the bitrate, quality, and so on. The middle layer is selected as Kind: codec. These options can specialize your choices, and frequently includes presets and profiles useful for coding well known parameter sets, like profile=high422, preset=medium, or tune=film, etc. The interface layer may or may not be available. It is usually accessible only by an “opts” parameter, like x264opts=key=value:key=value:... These options are passed directly to the low level codec library.

The 2 screenshots next show “ffmpeg” for as the Kind: for Video first and then “codec” Audio second. Note the x264opts in the Video Preset window immediately below.



Note the yellow tooltip in the lower right hand corner describing the highlighted option. Also note the allowed “Range” values above the box provided for keyins.



5. FFMPEG Items of Note

Quality Option when rendering: Ffmpeg responds variably to the quality option in the render option but seems to respond well to bitrate. The subranges used by quality even seem to vary somewhat depending on how old the codec is. Some use 0 to 35, some use 0 to 500 or so. The quality is supposed to cause the codec to output data until the noise level is below a limit determined by the quality setting. Your specific results may vary.

Previous Changes when rendering: With ffmpeg there are 2 cases that the defaults will be used. The first time when you have nothing set up and any other time when you reset the render File Format in the Render Menu. Otherwise with ffmpeg if you change a video compression type for the render (for example h265.mp4 to h265.mp4), the settings will be from the previous session settings.

Outstanding Issues with ffmpeg: There are some problems that need to be addressed by the ffmpeg developer group that adversely affect Cinelerra. These are stated below with the hopes that that group will fix them as time permits.

1) Make all the default parameters operational. When they are not, the Cinelerra plugins can't be initialized since the initial state of the filter is not operational. If that is not possible, then provide a set of nominal parameters for each plugin, so that they can be used as the plugins initial default state.
2) Make the filter config function project the new parameter data into the filter function at any

point during filter operation. This is so that continuous updates can be done as the plugin operates.
3) Improve seek codec restarts. The past predictor must be reset or reconstructed after a seek. The only documented way to seek is open/seek/play. Reopening the format layer is very expensive.

6. Problem Resolution Help

You can report potential problems, bugs, and crashes to the Cinelerra-CV website via email address: cinelerra@lists.cinelerra-cv.org . At a minimum, instead of starting cinelerra from the icon, run from a terminal window so that if there are error messages related to the problem, they can be captured on the screen and passed along. The command to run from a terminal window is: <directory_path of where you installed cinelerra>/bin/cin – for example if installed at directory /mnt0/build5/cinelerra-5.1, you would execute the following command to start the program /mnt0/build5/cinelerra-5.1/bin/cin.

6.1 What to Include in Problem Reports

For the best help, if you have a reproducible problem the following list of descriptive items makes it possible to figure out what the problem is and how to fix it. It may be a usage/setup mistake or a real bug which a programmer might like to fix rather quickly. Although it is not always necessary to provide this much information, it eliminates the “try this” or “did you do this?” and a lot of “guessing” rather than heading straight for a solution.

Basically we **need to see what you see** with the input, output, and session file:

1) need a detailed description of the problem;
2) supply a small representative sample of the original input that exhibits the issue;
3) if possible, also provide the rendered output, again using that representative sample;
4) save a session file used with that same sample which will contain a lot of setup parameters;
5) to make sure that the same rendering setup is used, it may be necessary to send an additional session file at the definition point just before rendering starts.

It is better to upload any files to a datahostfile site as sometimes they can be larger than the CV mailing list would like.

How to create a session file: from the menubar pulldown File→Save_as… and choose a filename, for example /tmp/beforerender.xml . Do this after you have the situation setup so can see values in use.

How to create a representative small sample:
- open the input media, setup as you normally would to start the project render;
- select 5-15 seconds of media using the edit mode ptr1 button drag highlighting on the timeline;
- start the render dialog (Shift R), set the filename path, and set “Render range” to “Selection”;
- recommended “Insertion strategy” is “Replace current project”; then press the checkmark OK.
(The result will “load replace” the current project for review.)

6.2 Crash Dumps for Analysis

If you get a SEGV signal crash and can explain what steps you took, a /tmp/cinelerra_<pid>.dmp file is very useful for analysis. You can also use Ctrl-C in the controlling terminal window to force an INTR signal interrupt when you think the program is hung up. It is best to upload the .dmp, input files and other files to datafilehost or similar temporary site as they may be quite large. The .dmp filename looks like /tmp/cinelerra_<pid>.dmp. Email all helpful information on the crash, location of uploaded files, and list of setup steps that illustrate the problem to cinelerra@lists.cinelerra-cv.org .

1) Use the latest version of Cinelerra as you want to make sure problem hasn't been fixed.
2) Be sure gdb is installed (usually installed but if not, easily downloaded/installed by anyone)
3) Run as root (this enables using gdb to create a /tmp/cinelerra*.dmp).
4) Be sure Settings->Preferences->Performance->trap SEGV/INTR are checked.
5) Invoke the error. This should create /tmp/cinelerra-<pid>.dmp .

7. Speed / Fade Automation Usage and Auto Gang

Speed automation resamples the data at a higher or lower playback rate. Speed automation can operate on all tracks of the same type, either audio or video, with a single click; or all tracks, both audio and video, with a double click. The curves to be affected must be both armed and ganged in the track patchbays to be included in the operation. You can see the advantage of having different media types ganged the same so that they will remain synchronized. The double click feature also works in the same manner for Fade Automation.

Steps to demonstrate Auto Gang of Speed on all of the audio AND video tracks are listed below:

1) Use the pulldown menu “View” on the main canvas and ensure that there is a checkmark left of the Speed selection.
2) Double click and hold button 1 on any point of the Speed line on any track of the main track canvas and drag the handle where you want it.
3) Note how all video and audio boxes move together simultaneously for synchronization.

Clicking on a speed keyframe handle will restart the drag operation on tracks of the same type. Double clicking will select all tracks which have handles in the exact same cursor position.

8. How to Create Blu-ray and Standard DVDs

This section describes how to create a blu-ray DVD, or *BD* referring to a *B*lu-ray *D*VD, and a regular DVD, or *SD* referring to a *S*tandard *D*VD. The DVDs (plural usage of DVD means either BD or SD) created are unencrypted, unlike commercially available movie DVDs. This Cinelerra version conceivably can create different variations of DVD/Blu-ray media but for the casual user the most standard usages are readily usable and will be described here.

Some preliminary information follows. For NTSC, SD media is almost always 720x480 interlaced (the format in the United States, US). For PAL, SD media is almost always 720x576 interlaced (Europe, EU, and most of the world). An SD can conceivably be created with a lower resolution – for example 352x240 MPEG-1 – but it is not useful. Aspect ratio for either NTSC or PAL can be 4:3 or 16:9.

Frames per seconds is usually 29.97 for NTSC and 25 fps for PAL. The standard SD dvd generally uses the MPEG-2 program stream with a filename extension of m2v. BD blu-ray media is not normally interlaced, but you can leave it interlaced. Blu-ray uses the MPEG transport stream which contains 1 or more program streams with a default filename extension of m2ts.

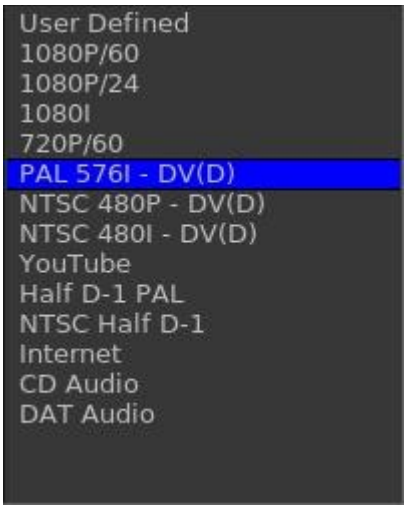
Requirements for creating DVDs is the hardware device to write the media on and, obviously, the blank media for either BD or SD. When generating SD media, you will have to install dvdauthor and for BD media, install udftools if they are not on your system. Also, keep in mind that **to mount filesystems for creating files and to burn DVDs, you will have to be root since you have to have privileges**, unless special permissions have been provided for a non-root user. It is also highly recommended to run the cinelerra startup from a terminal window, instead of the icon, in order to see informative messages of how to actually write the output (at prompt: keyin <install_directory_path/bin/cin).

A warning here – writing blu-ray BDs and regular SDs can take a large amount of clock time. Keep in mind that a blu-ray can contain 25 hours of viewing, so would take multiple hours to just write one.

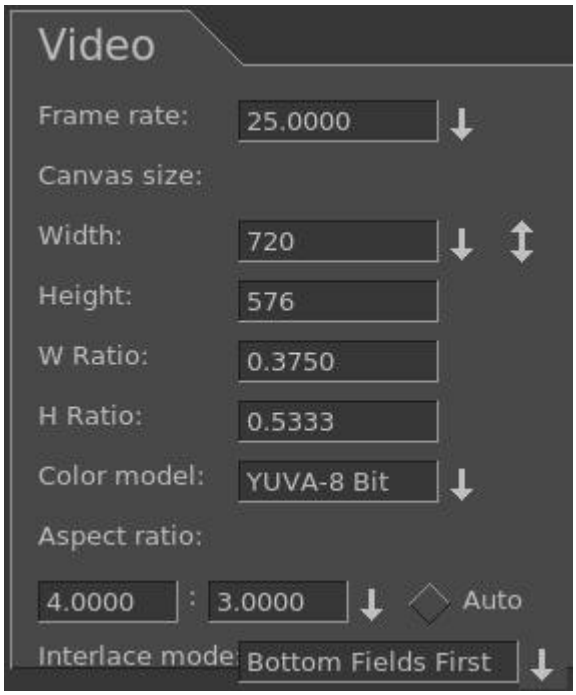
The max disk space needed is a little over 100GB for 50GB Double Layer (DL) media or 50GB for a single layer BD blu-ray. A standard SD of 4.7GB needs about 10GB disk space. You probably can get by with much less if the render for blu-ray is less than 25 hours of media. You will need twice as much disk space as the media holds to ensure you have sufficient space for working and copying.

The most important thing you need to know about in order to get started is "Format" and "Scale" in the "Create dvd or bd" window. Format settings shown in the Set Format window are set accordingly to an algorithm. Basically, it will take whatever you say in the asset format. It matches that against the known Presets available so that is what will be shown. If the asset format doesn't match any of the Presets default formats, then that will be shown as the "User Defined" format. In addition when you load media, the format is initially set to “same as source” so matches the source input and if that matches a known preset, then that is what is shown. The PAL versus NTSC only comes into play when there is no known correct matching format when you attempt to create a DVD Render batch job. It is applied when you click OK. For example, if you load up a YouTube video, it will not match any known format and will choose PAL or NTSC based on time zone.

Screenshot of Presets in Set Format window



When choose PAL, values get changed in window to reflect PAL -->



A quick set of **basic steps to create DVDs** is immediately below and usually just using the defaults will get you something. However there is a serious issue with interaction between the Operating System and bdwrite when creating a BD/blu-ray that requires automount to be turned off. Refer to the details and more specific explanations below the following steps for how to do this.

0) If not logged in as root, you will get an error message in order to avoid doing a lot of work and then failing out because root is required for automount and to write on DVD hardware.
1) Load your input source media via: File→Load files .
2) Choose PAL or NTSC for SD/dvd or 1080P/24 for blu-ray in Settings→Format .
3) For blu-ray, choose BD Render or for PAL/NTSC, choose DVD Render in File→ .
4) Designate a "work path" with sufficient disk space and then Chk-OK .
5) When the Batch Render window comes up, click on Start and the batch jobs will run.
6) Read the final messages echoed to the screen to see the command for burning… OR
7) Use the provided directory name to: cd /<target directory>bd_(or dvd_)<date-time> .
8) Load your media, format if needed, note device name to substitute for <bd> or <dvd>.
9) If rewritable blu-ray, use dd if=./bd.udfs of=/dev/<bd> bs=2048000
   If write-once blu-ray, use growisofs -dvd-compat -Z /dev/<bd>=./bd.udfs
   If any DVD media, use growisofs -dvd-compat -Z /dev/<dvd> -dvd-video ./iso

Any problems encountered will require that you read more information in this section to include specific details, helpful hints, and problem resolution.

Details and specific explanations to create blu-ray or regular DVD are provided here. It is very advantageous to startup cinelerra from the command line prompt instead of the icon. Also, please be root or your hard work will be lost when the automount is issued and fails for bluray udfs mounting.

The general design of the DVD/blu-ray generation operations is to first render media using batch rendering and then terminate Cinelerra to start a script which creates the target device filesystem data. These scripts are the dvd.sh and bd.sh scripts written into the target directory. For DVD, the general plan is to write a directory <target>/iso with the dvd filesystem via dvdauthor and then generate an iso9660 filesystem and write it to a dvd via growisofs.

For blu-ray, the filesystem generation is slightly harder. First, it creates an empty filesystem image <target>/bd.udfs using mkudffs which makes a big hole for the filesystem data. The hole is made just a little bigger than the data written by bdwrite so that you don't have to write an entire 25GB or 50GB disc even if no data exists. This empty filesystem is loopback mounted to make it writable, and the linux kernel manages the filesystem image. The bdwrite program applies the blu-ray structure to the UDF filesystem by creating the needed BDMV blu-ray filesystem, which the kernel stores onto the image file bd.udfs. When udfs is unmounted, the kernel finalizes the disk image on bd.udfs. The bd.udfs image can be written directly to a blu-ray disk via dd or growisofs.

NOTE of IMPORTANCE: there is a serious situation with the interaction between the Operating System (OS) and bdwrite when creating blu-ray, that requires automount to be turned off. The blu-ray automatic script unmounts the blu-ray/UDF filesystem but the system has not finalized the directories so the OS creates a new loop file device and the data is loaded and cached for use by the new loop but it is stale. Consequences is that not all of the data is written where it should be. The solution is for the OS not to mount this second mount so we have to make sure it doesn't. There are 2 methods to fix this. The first and easiest is by using the following command to disable automount:

```
gsettings set org.gnome.desktop.media-handling automount false
```

This can be reversed when you have completed the blu-ray generation via:

```
gsettings set org.gnome.desktop.media-handling automount true
```

A different and more complicated method you can use to turn off automount is to download and install the dconf-editor. Automount is a system parameter and only needs to be done once unless you do not want automounts to always be disabled.

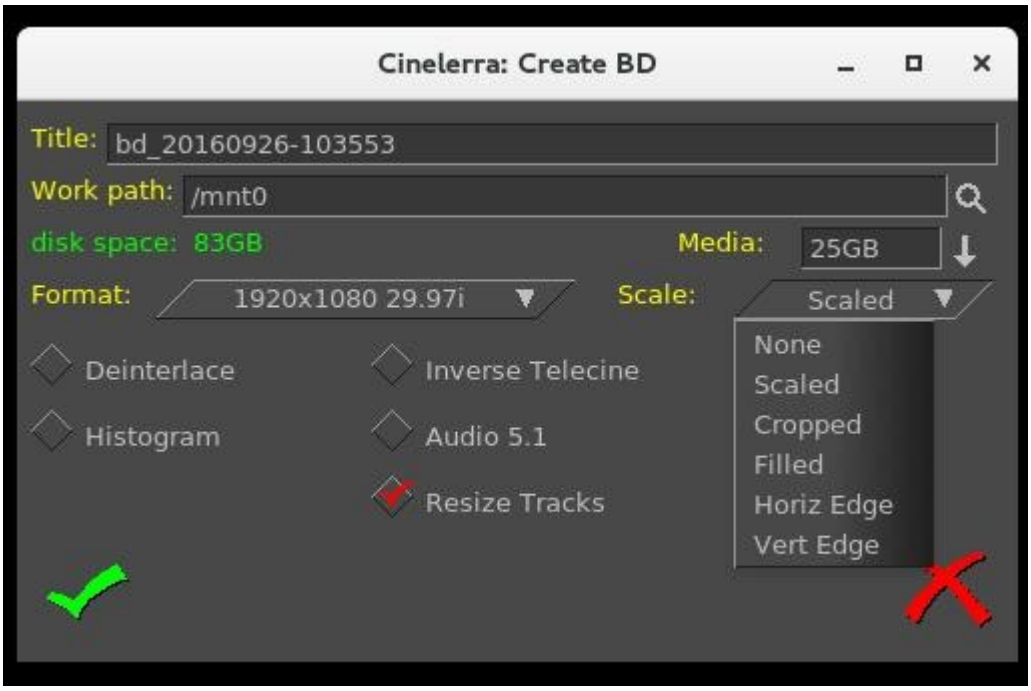
1) run: dconf-editor
2) select: org→gnome→desktop→applications->media-handling
3) uncheck: automount
4) close dconf-editor window

Immediately below are the detailed steps with explanations for creating SD or BD media.

1) Construct a session with the desired presentation:
   Format frame rates choices are 29.97 or 25 for SD, based on the user's timezone, with NTSC 29.97/US or PAL 25 /EU. For BD, the media input will be analyzed to automatically pick the default format or if unknown, the user's timezone will be used to default to 1920/29.97i for US or 1920/25i for EU. Be sure to set the rendering parameters in the settings -> format menu.
   Choose audio stereo or 5.1, again depending on your media.
   Target Geometry will be 720x480 (US) or 720x576 (EU) for SD.

2) From the main window, select file→BD Render or select file→DVD Render. Then:
   In the "Title" textbox there will automatically be a unique name based on the current day and time but you can change that to anything you want that is NOT already created.
   Choose a "Work path" Target Directory in the work path textbox. Another big warning here is to use a path that is NOT /tmp because a lot of linux distros wipe out /tmp on a regular basis via cron, crash, or reboot. Furthermore, they tend to use ram based filesystems, which will be way too small. You will lose your hard work so use disk space that is permanent. An icon in the form of a magnifying glass is next to the work path which shows you file names/paths.
   Disk space message will be green if you have at least 100GB/BD or 10GB/SD available space (the disk space message will be red to warn you if you have less than these amounts).

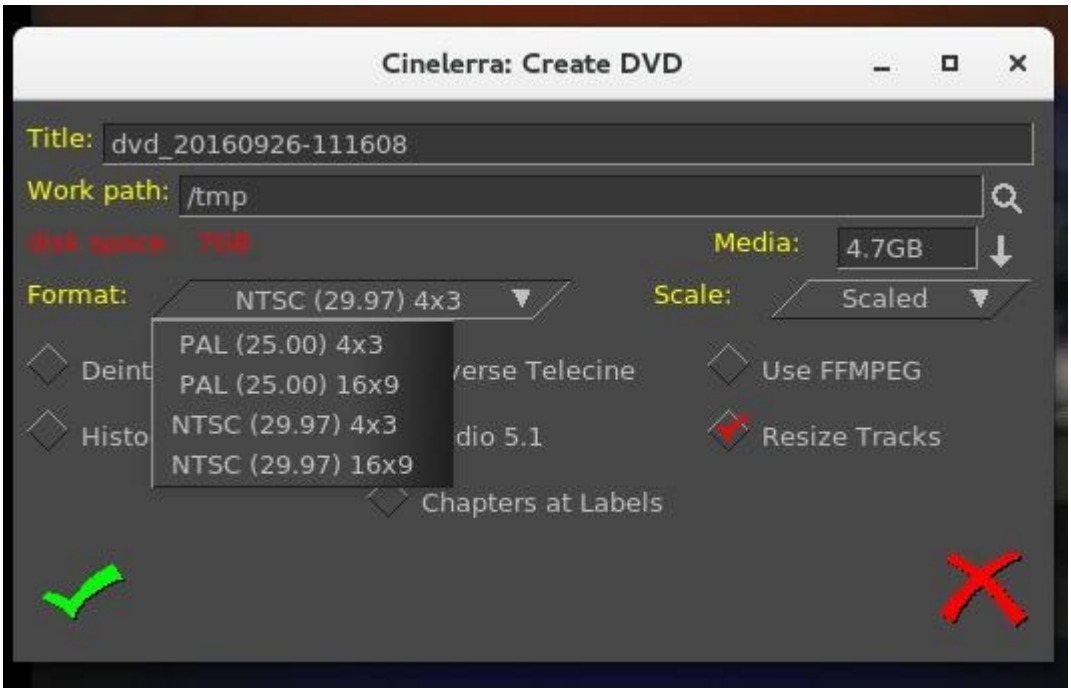
Screenshot showing sufficient available disk space in green for blu-ray creation is below.



← Note the Scaled options in the pull down menu which will be described later.

Screenshot showing insufficient available disk space in red letters for SD dvd creation is below.

Note the pull --> down displaying the PAL and the NTSC formats.



(Continuation of Step 2)

Select desired features, check / un-check as appropriate.

Click OK check button. It is **very important to realize** that when you check OK, the EDL will be saved and that will be used for batch job rendering. If you bring up the Batch Render and then change some parameters, they will not take effect UNLESS you remember to either check "Save to EDL Path" or "Use Current EDL" in the Batch Render window. You will get a reminder automatically if "warn if jobs/session mismatched" is checked.

Explanation of the choice boxes as seen in the previous 2 screenshots for both SD and BD menus is given below. Many of them are plugins which allow you to further manipulate the settings for best results. They are just suggestions set by the program automatically based on your input media, and can be reset to suit your needs. These are listed in the next 4 points.

1) If the media does not match the DVD target geometry, and the **scale** plugin is not already in use, then the scale plugin is applied with scaling set to fit the media dimensions to the DVD format target geometry.
2) If the video height is at least twice the DVD height and the input media is interlaced, then the **deinterlace** plugin is applied with odd line sampling.
3) **Audio 5.1** will automatically be set to the wide-audio if you have 6 tracks of audio.
4) To allow video data to be accessible and overlay properly, the track buffers are "resized" to the largest track frame size in use (**Resize Tracks**). The theory behind this is to make sure to have enough memory to cover the entire presentation for transcoding.

All of the current choice boxes are further defined immediately following.

Deinterlace – remove the interlace. Interlacing is a video scanning system in which alternating lines are transmitted so that half a picture is displayed each time the scanning beam moves down the screen.

You lose a lot and the quality is bad when you view interlacing on a progressive TV. You might not really want to use deinterlace, because if you deinterlace non-interlaced media, it will look awful.

Scale – alter the spatial mapping of an image to increase or reduce the size; modifies the picture. When some programs scale from 4:3 to 16:9 they will automatically cut off the appropriate section of the image for you. It is necessary to keep in mind, that square pixels is the true end goal of scaling, not the aspect ratio which could result in squished or stretched output. More information about scaling will be provided on a subsequent page with usage of the "Scale Ratio" plugin.

Histogram – remaps the color space. The color space ranges from 0-255 for 8-bit color values. You can use this tool to remap the color space to use the entire space or for stretching the contrast. Also, it lets you perform global color-correction on the image. You can use this to correct for color screens that are 'too blue', or for color Televisions that produce 'brownish' output, or whatever. In addition to color-correction, you can use the RGB modification tool to add color to images that didn't have color to begin with. For instance, you can "pseudo-color" greyscale media.

Inverse Telecine - the reverse of 3:2 pulldown where frames, which were duplicated to create 60-fields/second video from 24-frames/second film, are removed. MPEG-2 video encoders usually apply an inverse Telecine process to convert 60-fields/second video into 24-frames/second encoded video. The encoder adds information enabling the decoder to recreate the 60-fields/second display rate. Telecine, i.e. 3:2 pulldown, is used to transfer film to video. That's where the 3:2 ratio comes in. To ensure that there will consistently be 60 frames per second, the first frame is displayed on the TV screen 3 times and the second frame is displayed 2 times. The following frame is repeated 3 times, the next one 2 times, etc. throughout the film. For inverse telecine, you show 2 of the film frames for 3 of output frames. Only check "Inverse Telecine" if you have film or something that is 24fps and want to project to 30fps (most likely this will never be necessary).

Audio 5.1 – 6 channel surround sound. For most home systems, uses five full bandwidth channels and one low-frequency effects channel. Could automatically get set as explained previously.

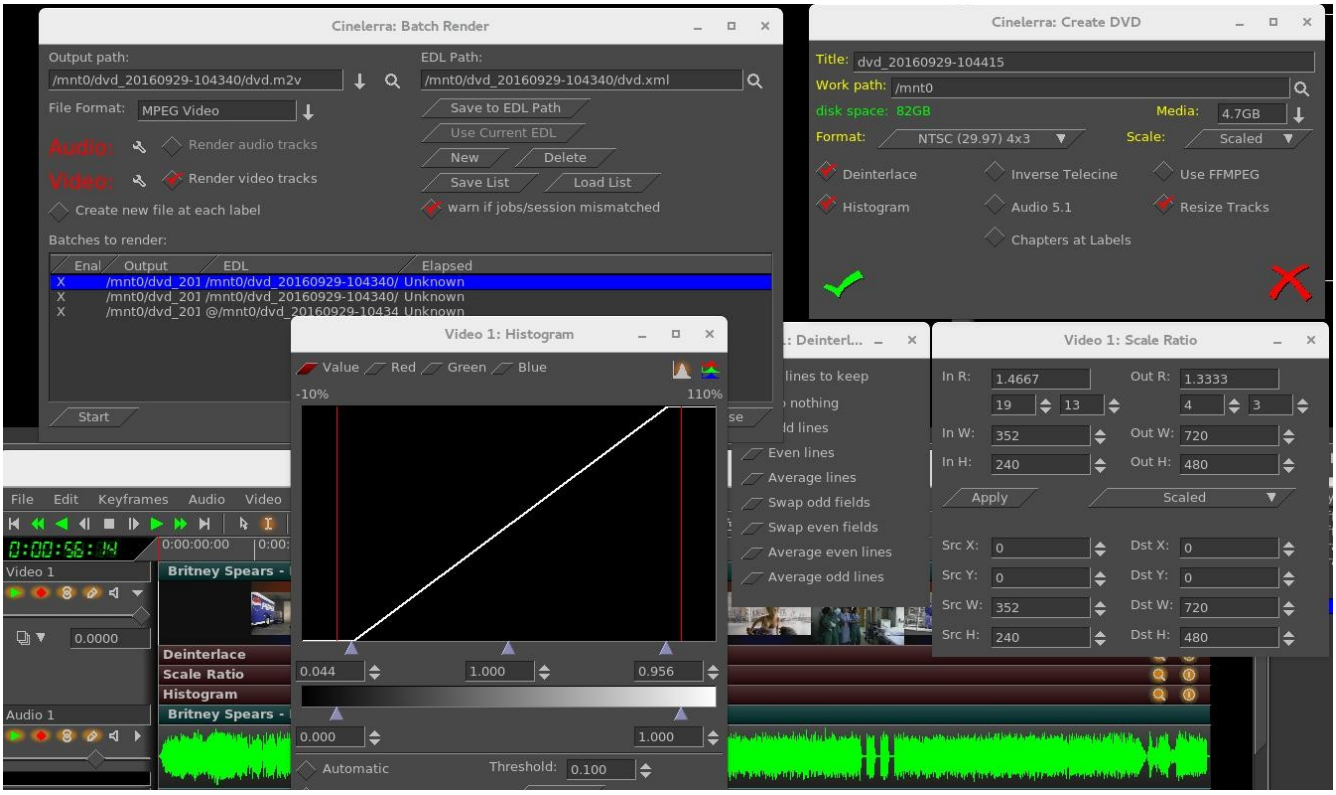
Aspect Ratio – aspect ratio may be automatically set to 4:3 or 16:9. Aspect ratio would better be defined as the size of the display, monitor, or TV which will be used to view the output. If you measure your old TV, which supposedly is 4:3 and your latest digital TV, which is supposedly 16:9, you will see that those ratios aren't always correct anyway. Then measure your laptop monitor, your desktop monitor, and your neighbor's, and lo and behold, the ratios don't fit either of the purported "standard" aspect ratio. Maintaining square pixels via scaling is more important in the long run.

Use FFMPEG – this is user's choice; it is recommended and faster but more difficult to modify due to numerous options. For blu-ray, ffmpeg must be used and is not an available option.

Resize Tracks – change track width and height as explained previously. The size is adjusted to the largest frame size needed.

Chapters at Labels – without this checked, chapters markers are automatically inserted every 10 minutes. The chapter labels can then be "skipped to" when playing the DVD. If instead, you want to put labels in at opportune times, you will have to run dvdauthor outside of Cinelerra and mark the chapter labels yourself by hand. In that case, you should checkbox "Chapters at Labels" so that the automatic 10-minute labels are not created. This checkbox is not currently available for blu-ray.

The screenshot below shows on the upper right side the “Create DVD” window with Deinterlace, Histogram and Resize Tracks checked. Also Scale is set to “Scaled”. Once the green checkmark is clicked, the “Batch Render” window comes up and in the main window you will see the 3 plugins below the video track on the main window. By clicking on the magnifying glass that appears on the rightmost side, the controls for each will popup and you can make any necessary adjustments. Note the numerous choices for Deinterlace; the Value, Red, Green, and Blue for color adjustments in the Histogram window; and Scale Ratio popup menu for numerical settings control.



The Scale parameter gives you a lot of flexibility. A default based on your input media is provided for you but possible choices are None, Scaled, Cropped, Filled, Horiz Edge, and Vert Edge. You will have the opportunity to manipulate the desired results in the “Scale Ratio” window. Values for W (width), H (height), and X/Y coordinates are the number of pixels. For example, if video is 720 x 432, that is obviously 720 pixels by 432 pixels and this would be the values for Dst W and for Dst H. So if you have some media that is off center you can crop by changing the SRC Y value AND then change DST X/Y to non-zero. It will become the output origin. To see what it does, change them from 0.0 to 400.0 and you will see big changes in the compositor window.

For example, if you have the Cropped choice for the Scale, you will want to manipulate the ScaleRatio plugin (the magnifying glass on the main window video track) which brings up the Scale window. For cropped top instead of crop both top and bottom, modify the "Src Y". As you change the Y scale, you will see the cropping take place in the Compositor.

Scaling options are provided in order to preserve image aspect ratio. To determine which scaling option to use, it is important to correctly identify your source/destination video aspect ratios. Next is a short explanation of possible options.

None – do not do any scaling. The destination output matches the source input. There is no resizing.

Scaled – do not use uniform scaling on X and Y. Just make it fit and you will might end up with squishing/stretching in one direction or the other in order to do that. This happens when the input aspect ratio is different than that displayed on the output.

Cropped – removes outer edges of a source image in order to fit the image on the output display. This is done in order to maintain uniform geometry scale on the destination display without being affected by the particular factor of original media aspect ratio. Since cropping omits image area, and the areas which you wish to view may not be viewed when the image is centered, you can pan the image source using the Src X/Y to modify which area is in view. You can also think of this as "scaled up".

Filled – the entire output display will be filled with image content but in order to do so, some of the image may have had to be cut. Any mismatch between the two will be filled with black. Centering of the image can be modified by using the Dest X/Y variables in the Scale Ratio controls. You can also think of this as "scaled down".

Horiz Edge – this indicates preserving of the Horizontal edge.
Vert Edge – this option preserves the Vertical edge.

Horizontal and Vertical are duplicates or restatement of same functionality as Cropped or Filled but are provided as options to accommodate different ways of thinking. In any case, you can choose which outer edges of the image to crop by using the "Show Controls" of the Scale Ratio plugin. For example, you can ensure that no "action" is lost by displaying the center of the screen only or making sure that any textual information on the bottom is not lost by cutting only the top off.

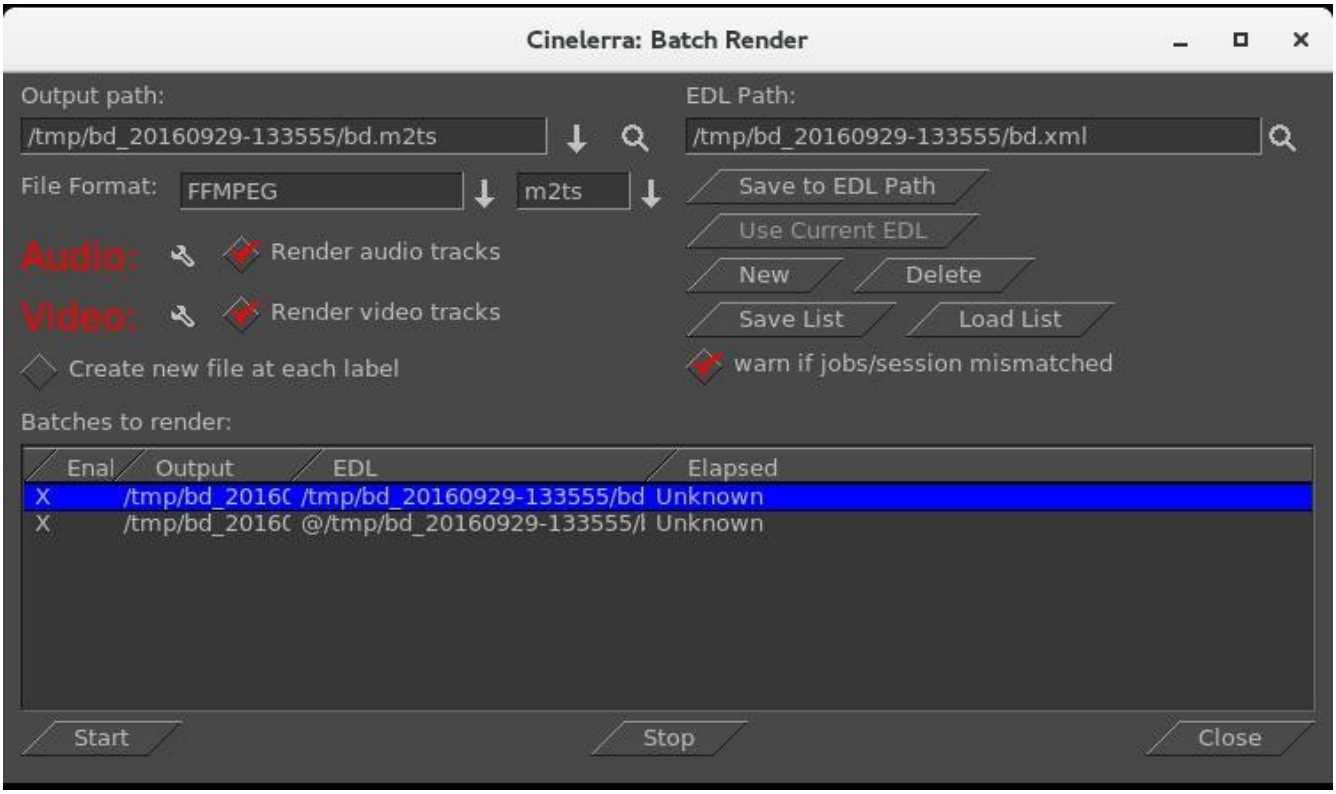
Step 3:
3) Batch render menu appears with "m2ts" format selected for blu-ray or "dvd" format selected for regular/standard DVD when File Format selected is ffmpeg in the previous "Create DVD" menu. It will work just fine without selecting ffmpeg for DVD and may be advantageous not to. Using the audio / video wrench tools (you will have to have the video batch job highlighted to manipulate the video or audio batch job for audio).

> Setup the audio bitrate (192000 recommended). Data rate is 224K default.
> Setup the video bitrate (6000000 - 12000000 recommended). 8Mb/sec is the current default.
> Click OK check button.

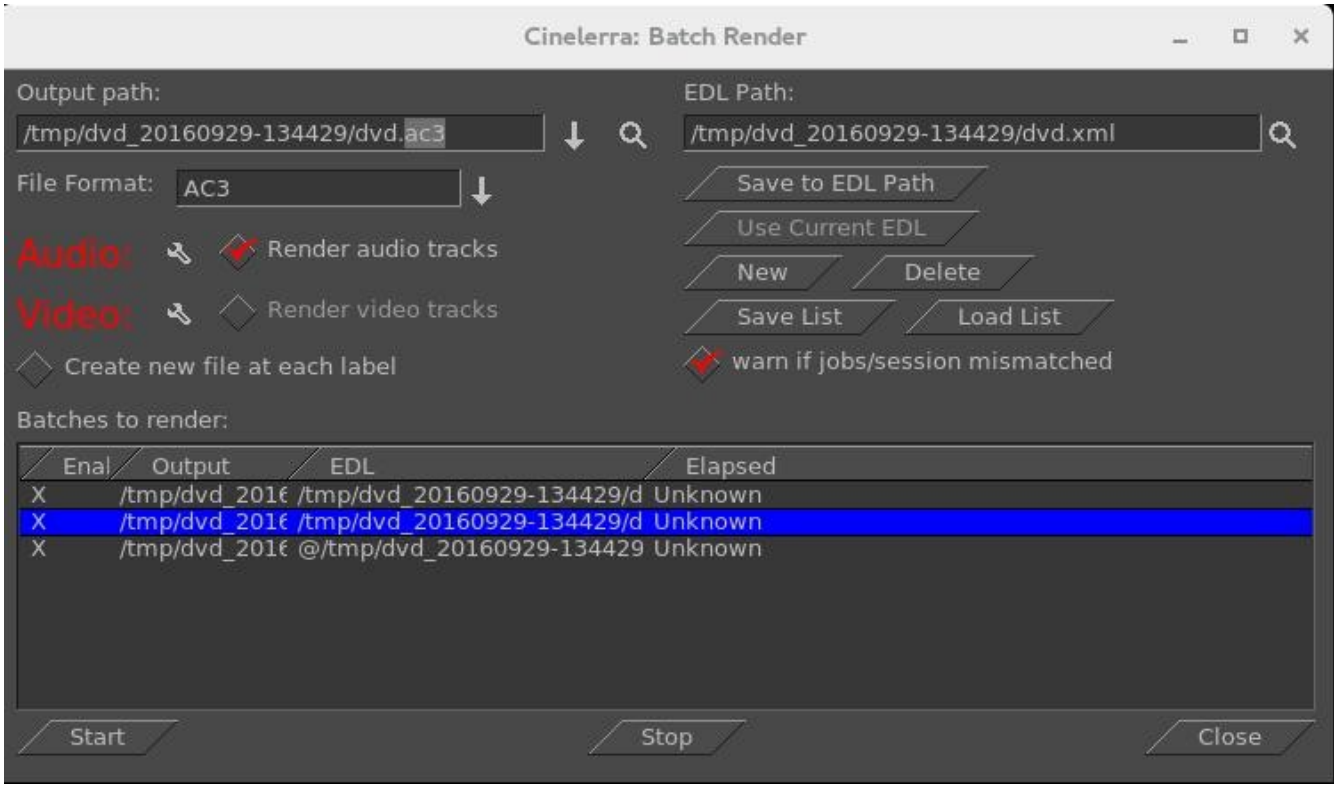
The default bitrate is the largest value possible. The actual 'target' bitrate is calculated based on a formula from the blu-ray/DVD code. It divides the media size (in bits) by the video time (in seconds) to find the bitrate that will "just fit" on the target media. This could create a weird bitrate if the media is large, and the video time is small, so the default/target bitrate is limited to 8Mb/s.

Batch jobs are then built and appended to the job list. Once these batch jobs are built, if you make any changes, you have to start over. You will see listed the batch jobs that are created to perform the rendering/tasks – 2 jobs for blu-ray, one for audio/video processing and one for the scripts. There will be 3 batch jobs created for DVD when not using ffmpeg to include one each for audio and video and then one more for the scripts. When you click on "start", it fires off those jobs and you will see the rendering main window progress bar in the bottom corner reflecting the fact that it is currently processing. Be aware that the Cinelerra program will be totally shutdown AFTER the batch jobs finish.

Screenshot below shows BD blu-ray creation with 2 batch jobs queued up and ready to go.



Screenshot below shows a DVD creation render ready to be started. Note that because it is not ffmpeg, the processing of video is done separately from audio. If you need to modify the video tracks, which you can see is ghosted out, you need to highlight the first batch job listed. Be sure to highlight the first batch job before pressing Start so it runs all of the jobs and be sure the X for job enabled is set.



When you click on “start”, it fires off those jobs and you will see the rendering main window progress bar in the bottom corner reflecting the fact that it is currently processing. Be aware that the Cinelerra program will be **totally shutdown after the batch jobs finish** and you will be at the command line prompt.

This will produce a new directory in your target path which contains a filesystem image file.
For example: /TargetDirectory/bd_20150820-093747
Directory and file names should not be changed at this time because the scripts and programs rely on the given names in order to proceed. You can change them later for your own purposes.

If bluray to test the filesystem you just created, use the command line interface; loopback mount the filesystem image which was generated in the target directory. For example if blu-ray:

```
cd /TargetDirectory/bd_20150820-093747/
mount -o loop,ro ./bd.udfs ./udfs
(check the media using a compatible media rendering tool like ffplay)
umount ./udfs
```

To burn blu-ray media you will need to run from the command line interface. In the examples below, /dev/bd represents your blu-ray writer device (for example: /dev/sr1) and /dev/dvd represents your dvd writer device (for example: /dev/sr0).

Blu-ray Media

For rewritable blu-ray (recommended) (BD-RE):

```
Note - unwritten (virgin) media must be formatted first using:
dvd+rw-format /dev/bd (only done once and does not take very long)
```

To write or rewrite rewritable blu-ray media:

```
cd /TargetDirectory/bd_20150820-093747/
dd if=./bd.udfs of=/dev/bd bs=2048000  (the growisofs command below also works)
```

To write blu-ray write-once media: (BD-R) (no pre-formatting needed):

```
cd /TargetDirectory/bd_20150820-093747/
growisofs -dvd-compat -Z /dev/bd=./bd.udfs
```

DVD Media

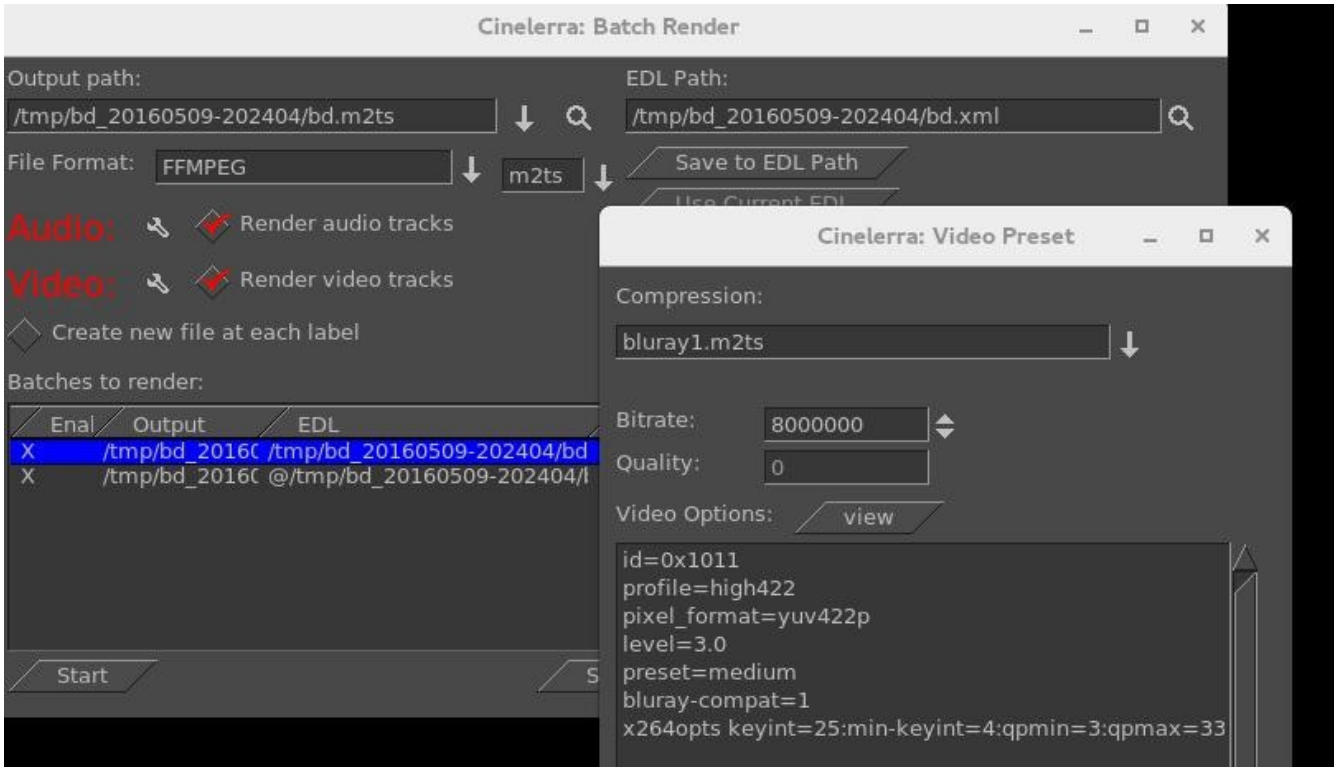
For rewritable DVD (DVD+RW):

```
Note – unwritten (virgin) media must be formatted first using:
dvd+rw-format /dev/dvd (only done once and does not take very long)
```

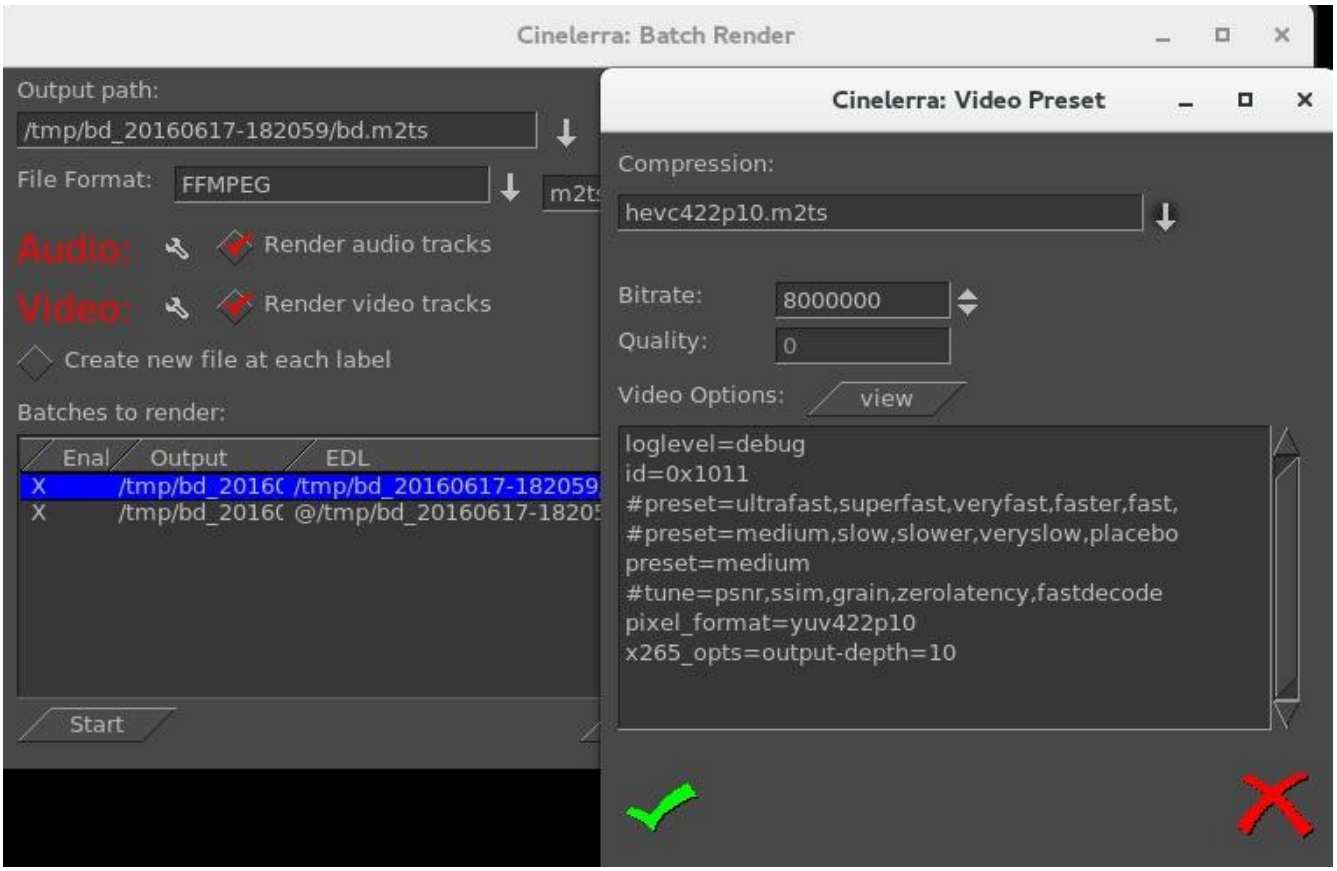
To write a DVD, load blank media and run the following from the command line (requires dvdauthor):

```
cd /TargetDirector/dvd_20160404-175416
growisofs -dvd-compat -Z /dev/dvd -dvd-video ./iso
```

Screenshot showing the availability of 4:2 :2 for a Batch Render seen by clicking on wrench icon.



Screenshot showing the availability of 10-bit high quality 4:2 :2 for a Batch Render seen by clicking on wrench icon. You need specially compiled cinelerra in order to use the 10-bit as opposed to 8 bit.



8.1 Output Terminal Messages from Creating DVDs

Below are examples of what the batch jobs generate and you will see on the terminal screen if you started the cinelerra program in the recommended manner from a terminal window. It is just informational but will let you know if errors. In looking at any of the output, you can safely ignore the errors that read “Unsupported codec with id 100357 for input stream 0” - this comes from “nav-data” (navigation data). The first 2 examples are seen from running the batch jobs; the last 2 are from the single line execution which records the media output to the DVD hardware.

SD Example – Partial Output during Cinelerra run

```
...
FFMPEG::open_decoder: some stream times estimated
Render::render_single: Session finished.
FFMPEG::open_decoder: some stream times estimated
Render::render_single: Session finished.
running /tmp/dvd_20160407-113530/dvd.sh 1 /tmp/dvd_20160407-113530
  INFO: [mplex] mplex version 2.1.0 (2.2.7 $Date: 2012/11/17 01:55:16 $)
  INFO: [mplex] File /tmp/dvd_20160407-113530/dvd.m2v looks like an MPEG Video stream.
  . . .
  INFO: [mplex] MUX STATUS: no under-runs detected.
DVDAuthor::dvdauthor, version 0.7.1.
Build options: gnugetopt graphicsmagick iconv freetype fribidi fontconfig
Send bug reports to <dvdauthor-users@lists.sourceforge.net>

INFO: default video format is NTSC
INFO: dvdauthor creating VTS
STAT: Picking VTS 01

STAT: Processing /tmp/dvd_20160407-113530/dvd.mpg...
STAT: VOBU 32 at 15MB, 1 PGCs
INFO: Video pts = 0.133 .. 22.789
INFO: Audio[0] pts = 0.133 .. 22.789
STAT: VOBU 46 at 21MB, 1 PGCs
CHAPTERS: VTS[1/1] 0.000
INFO: Generating VTS with the following video attributes:
INFO: MPEG version: mpeg2
INFO: TV standard: ntsc
INFO: Aspect ratio: 16:9
INFO: Resolution: 720x480
INFO: Audio ch 0 format: ac3/6ch,  48khz drc, 'en'

STAT: fixed 46 VOBUs
INFO: dvdauthor creating table of contents
INFO: Scanning /tmp/dvd_20160407-113530/iso/VIDEO_TS/VTS_01_0.IFO
To burn dvd, load blank media and run:
growisofs -dvd-compat -Z /dev/dvd -dvd-video /tmp/dvd_20160407-113530/iso
```

BD Example – Partial Output during Cinelerra run

```
...
FFMPEG::open_decoder: some stream times estimated
Render::render_single: Session finished.
+
PATH=/usr/lib64/ccache:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/root/bin:/mnt0/build5/cinelerra
-5.1/bin
+ mkdir -p /tmp/bd_20161224-162059/udfs
++ du -sb /tmp/bd_20161224-162059/bd.m2ts
++ sed -e 's/[  ].*//'
+ sz=19811904
+ blks=13769
+ mkudffs /tmp/bd_20161224-162059/bd.udfs 13769
start=0, blocks=16, type=RESERVED
start=16, blocks=3, type=VRS
start=19, blocks=237, type=USPACE
start=256, blocks=1, type=ANCHOR
start=257, blocks=16, type=PVDS
start=273, blocks=1, type=LVID
start=274, blocks=13238, type=PSPACE
start=13512, blocks=1, type=ANCHOR
start=13513, blocks=239, type=USPACE
start=13752, blocks=16, type=RVDS
start=13768, blocks=1, type=ANCHOR
+ mount -o loop /tmp/bd_20161224-162059/bd.udfs /tmp/bd_20161224-162059/udfs
+ bdwrite /tmp/bd_20161224-162059/udfs /tmp/bd_20161224-162059/bd.m2ts
+ umount /tmp/bd_20161224-162059/udfs
+ echo To burn bluray, load writable media and run:
To burn bluray, load writable media and run:
+ echo for WORM: growisofs -dvd-compat -Z /dev/bd=/tmp/bd_20161224-162059/bd.udfs
for WORM: growisofs -dvd-compat -Z /dev/bd=/tmp/bd_20161224-162059/bd.udfs
+ echo for RW: dd if=/tmp/bd_20161224-162059/bd.udfs of=/dev/bd bs=2048000
for RW: dd if=/tmp/bd_20161224-162059/bd.udfs of=/dev/bd bs=2048000
```

SD Example – Partial Output during writing disc media

```
# growisofs -dvd-compat -Z /dev/sr0 -dvd-video /tmp/dvd_20161224-160756/iso
WARNING: /dev/sr0 already carries isofs!
About to execute 'mkisofs -dvd-video /tmp/dvd_20161224-160756/iso | builtin_dd of=/dev/sr0
obs=32k seek=0'
I: -input-charset not specified, using utf-8 (detected in locale settings)
 75.62% done, estimate finish Sat Dec 24 16:09:51 2016
Total translation table size: 0
Total rockridge attributes bytes: 0
Total directory bytes: 4096
Path table size(bytes): 42
Max brk space used 1a000
```

```
6624 extents written (12 MB)
/dev/sr0: "Current Write Speed" is 4.1x1352KBps.
builtin_dd: 6624*2KB out @ average 0.7x1352KBps
/dev/sr0: flushing cache
#
```

BD Example – Partial Output during writing disc media

```
# growisofs -dvd-compat -Z /dev/sr0=/tmp/bd_20161224-155658/bd.udfs
Executing 'builtin_dd if=/tmp/bd_20161224-155658/bd.udfs of=/dev/sr0 obs=32k seek=0'
/dev/sr0: "Current Write Speed" is 2.0x4390KBps.
    1605632/24524800 ( 6.5%) @0.0x, remaining 1:39 RBU 100.0% UBU   0.0%
    1605632/24524800 ( 6.5%) @0.0x, remaining 2:22 RBU 100.0% UBU 100.0%
    1605632/24524800 ( 6.5%) @0.0x, remaining 3:05 RBU 100.0% UBU 100.0%
    5865472/24524800 (23.9%) @0.3x, remaining 0:54 RBU 100.0% UBU  33.3%
   11829248/24524800 (48.2%) @0.4x, remaining 0:21 RBU  75.8% UBU  37.5%
   17858560/24524800 (72.8%) @0.4x, remaining 0:08 RBU  39.8% UBU  79.2%
   23789568/24524800 (97.0%) @0.4x, remaining 0:00 RBU   4.5% UBU   4.2%
builtin_dd: 11984*2KB out @ average 0.2x4390KBps
/dev/sr0: flushing cache
```

8.2 Helpful Hints for Problems/Debugging DVDs Creation

This section contains helpful hints, how to initially check the results, and some information on determining what might have gone wrong and how to address it.

1) For first time users, taking the defaults seem to work very well when running as root.
2) You may want to use rewritable media to see how it goes before using permanent media.
3) Until you are familiar with the procedure, start with shorter input in order not to waste time.
4) Test the generated output with a compatible media rendering tool before burning DVDs.
5) Check the list of files and file sizes after the batch jobs are complete before burning DVDs.

For blu-ray creation, cd /workpath/bd_date-time directory and look for similar files:
bd.jobs bd.m2ts bd.sh bd.udfs bd.xml
udfs directory which is used as a loopback mount point
Note that the size of bd.udfs should be larger than bd.m2ts because this is the final file which is actually going to be written to the disc media. It contains contents of bd.m2ts and all of the required disc structure.

For DVD creation, cd /workpath/dvd_date-time directory and look for similar files:
dvd.ac3 dvd.jobs dvd.m2v dvd.mpg dvd.sh dvd.xml
iso directory with VIDEO_TS and AUDIO_TS subdirectories of non-zero size

6) The bd.sh and dvd.sh files are script files that you can carefully run manually from some start point to determine where the failure occurred. You must BE CAREFUL and know what you are doing and what directory you are in because dvd.sh contains an “rm” command and will delete files. The script takes a command line parameter of the directory where the file was rendered to and which is usually the directory where dvd.sh or bd.sh was created.

7) There is also a file in the same directory, called bd.jobs. It was the information that was used in creating the batch jobs and may be helpful in determining what parameters were actually used if there are any resulting problems. With enough background knowledge, you can make changes and rerun.

8) For blu-ray check to make sure you do not have any spurious loopback disks mounted that may interfere with the correct generation. Use the df command to check this and then the umount command to unmount these. Also, check to make sure you have used the gsettings command to disable automount.

9) For blu-ray loopback mount the <target>/bd.udfs image, and see if it has the BDMV filesystem written to it, and in particular a subdirectory named STREAM. Look at the results in ./udfs and check for the stream file which should exist in ./udfs/BDMV/STREAM/00000.m2ts and should have the same size as ./bd.m2ts.

```
mount -o loop <target>/bd.udfs <target>/udfs
ls -lR <target>/udfs
du -sc <target>/udfs
umount <target>/udfs
```

Checklist for Troubleshooting BD and DVD Render:

___ Are you logged in as root? This is required in order to loopback mount files for bluray and to write media on /dev/hardware.
___ Did you startup cinelerra from a terminal window so you can see informative messages?
___ Is udftools installed for BD and dvdauthor installed for SD?
___ Do you have loopback not enabled for bluray? At least temporarily, disable automount via:
gsettings set org.gnome.desktop.media-handling automount false
___ Did you have sufficient disk space for working/writing files? In the Create window, the disk space will be displayed in green if sufficient, but in red if less than what fits on the disc media.
___ Did you use /tmp as the work device, then rebooted the computer, which deleted files on /tmp?
___ If the input media is interlaced, did you check the Deinterlace option to eliminate interlacing?
___ Did you change the output name in the Batch Render window after the batch jobs were already created? These filenames have already been written to disk. If you want to change either the Title or the Work_path, you have to start over.
___ Have you selected a Title in the Create window that is a directory that already exists? The program attempts to create that directory and will give you an error message if it exists.
___ Did you replace the /dev/bd or /dev/dvd on the command line with your hardware device name?
___ If a warning was issued in the Create BD/SD window of "* non-standard format" and your bluray reader could not play the disc, did you change to a standard format instead?
___ Did you correctly interpret the frame rate if using interlaced format to be ½ due to interlacing?

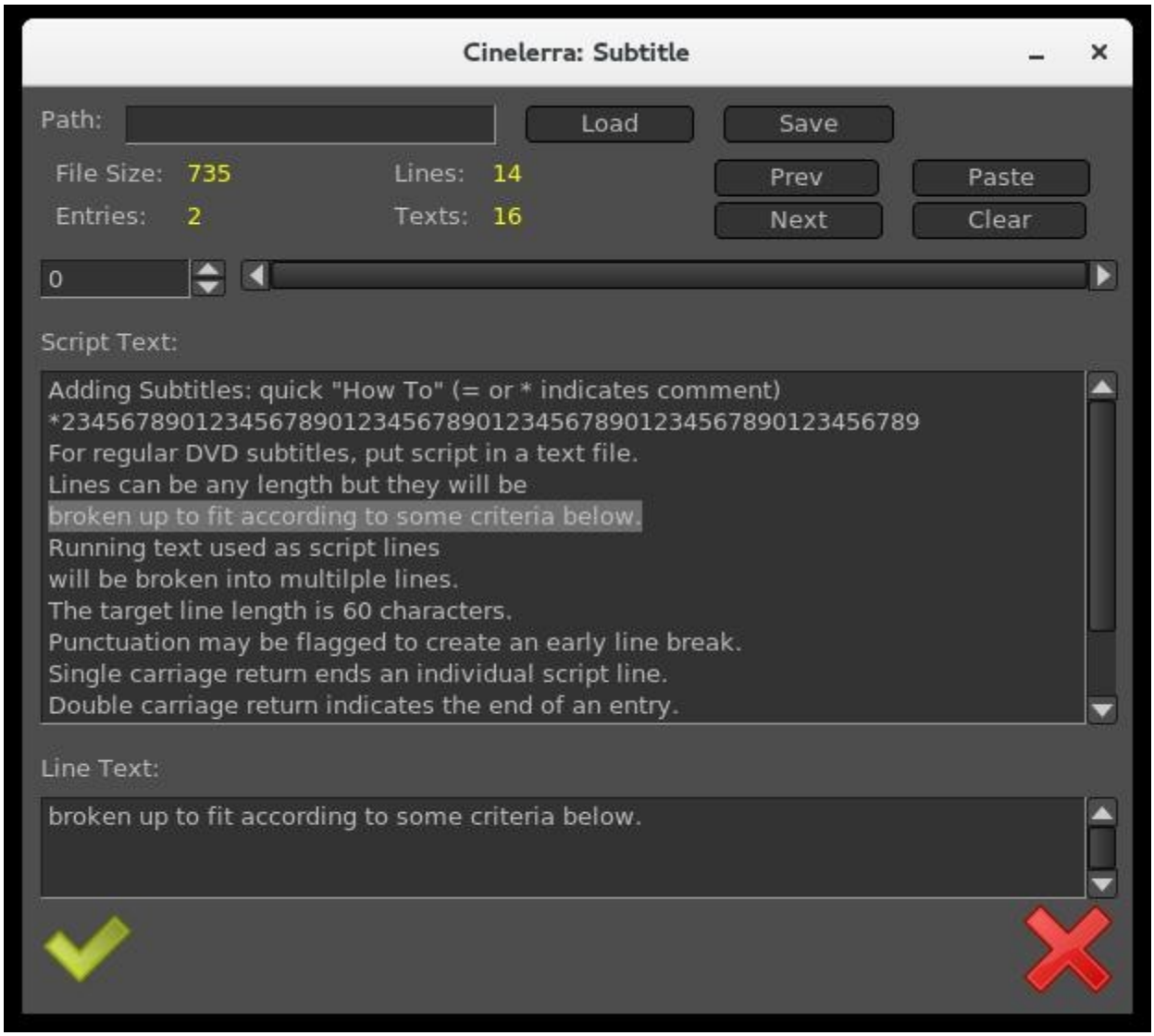
8.3 Additional Features for Blu-ray/DVD Creation

This section describes additional features of creating DVDs which can be very advantageous. It is quite easy to learn “how to” and rather simple, until it becomes tedious to do an entire 2 hour DVD!

8.3.1 Subtitles

DVD (not blu-ray... yet) subtitles are added by using the main window pulldown File→Subtitle which brings up a window allowing you to type the filename of a previously generated text file containing the desired words/lines, the script. After entering the filename, click “Load” to read in your script. By creating a script file ahead of time, it lets you easily add dialog that was already written out and carefully edited for spelling and proper grammar.

Screenshot below shows the Subtitle window you will see.



The format of the script/text input file has specific requirements as listed below:

Lines can be any length but they will be broken up to fit according to some criteria below.
Running text used as script lines will be broken into multiple lines.
The target line length is 60 characters.
Punctuation may be flagged to create an early break.
Single carriage return ends an individual script line.
Double carriage return indicates the end of a entry and helps to keep track of where you are.
The "=" sign in column 1 indicates a comment seen in the script text to assist you in location.
An "*" at the beginning of the line is a comment and not a script line.
Whitespace at either the beginning of a script line or the end will be removed.

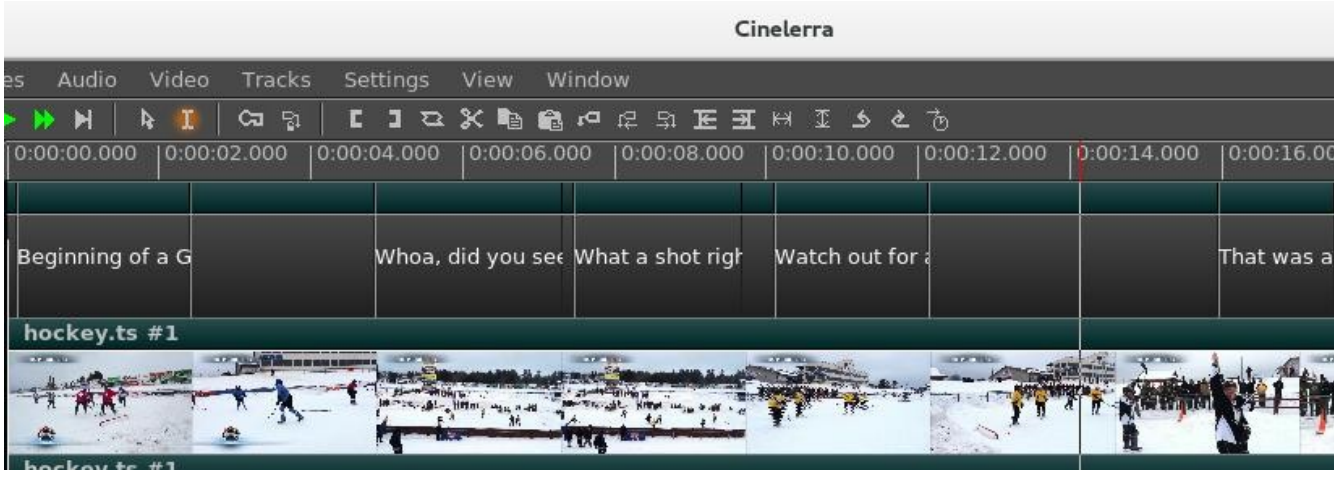
To put the subtitles onto your media, first add a subtitle track via the pulldown Track→Add Subttl . In the Subtitle window, note that there are 2 major textboxes. There is the "Script Text" textbox showing the current entry of text from your input file and there is the "Line Text" textbox showing the currently active text. In your subtitle track, select a timeline region (in/out or drag select with hairline cursor/highlight) to indicate the region where you want the active Line Text to be pasted. Then click the "Paste" button in the Subtitle window to paste the line onto the subtitle track. Silence will be added to the subtitle track in the places in the media where there are gaps.

Editing in the Line Text box can be used to change the active script line. By double clicking the timeline over the subtitle track, you can reselect the active script line. The subtitle text will be reloaded into the Line Text box and can be edited and re-pasted as the new active subtitle text. You can also highlight multiple lines in the Script Text box and paste them (using the usual window paste methodology) into the Line Text box. After pasting to the timeline, the Line Text box will be updated with the next script line. In addition, if you triple click a line in the "Script Text" box, it will automatically become the current line in the "Line Text" box.

When you are finished, before clicking "Save", you must supply a legitimate filename in the "Path" box; your current directory will be used if only a filename but no directory path is supplied. The filename used will automatically have a "-" after it followed by the "track label" and then "udvd' extension added; any extension in the filename will be removed.. If you click OK via the green check box before saving, the subtitle script position is saved with the session. This is convenient for continuing later where you left off.

To reposition the script, use the slider or tumbler buttons:
Slider bar to move through the text entries quickly.
Prev or Next buttons to go to the previous or next script line.

Screenshot below shows what the pasted subtitle script looks like in a portion of the main window.



8.3.2 Dvd Interlaced Chroma

Cinelerra uses 4:4:4 colorspace to edit, so it is necessary to convert interlaced 4:2:0 video to 4:4:4. But you can run into problems, referred to as the “chroma bug”, which you see in DVD media displayed on higher resolution monitors – streaks or spiky horizontal lines are visible in the chroma channel, especially on diagonal edges. The Chroma Bug is specific to MPEG and 4:2:0 encoding. Now you can use the “YUV420P DVD Interlace Mode” when rendering DV directly to mpeg2 through a yuv4mpeg stream and when using video effects on HDV video.

With this option enabled, improved chroma results will be obtained from your DV or HDV source. Editing DV or HDV and rendering it back to the same format does not require any special handling. In order to perform colorspace conversions correctly in Cinelerra and avoid Chroma errors for interlaced 4:2:0 video, check the box as follows:

Settings→Performance→YUV420P DVD Interlace Mode

This option maintains the interlacing in Chroma sample addressing, which ordinarily would be deleted because the upsampling of interlaced chroma fields is normally done using a progressive algorithm. With this mode enabled, the MPEG decoder uses a different algorithm for interlaced frames so that the 4:2:0 format chroma interlacing is preserved.

8.3.3 MPEG utility programs

There are 2 utility programs that come in handy when dealing with DVD media for creating or reading previously written DVDs.

1) mpeg3cc2txt – convert closed captioning data to subtitle data.

This program can be used to scan captured broadcast data streams and convert the closed captioning text data into subtitle track data. The result can be used to add subtitles to DVDs created from the captured media using the text collected from the closed captioning.

usage: ./mpeg3cc2txt [-c cc_service] [-s start:length, ...] [-t track] [-v verbose]
[-w wdw mask] [-x file.xml] [-o file.udvd] file.ts

cc_service	= closed caption service id
start:length	= start:length frames (comma separated list)
track	= video track number
verbose	= verbose level
wdw mask	= bit mask for windows (-1 = all)
file.xml	= filename for edl xml format subtitle data
file.udvd	= filename for udvd format subtitle data
file.ts	= filename for transport stream

To use this program, the input file must be a transport stream (broadcast video) which contains closed captioning services. The service id defaults to one, and the default video track is zero. Either -o or -x must be specified to indicate the output file format desired. If the output file name is a '-' then stdout is selected as the output file.

For example:

```
mpeg3cc2txt -o - /dvb_data/channel5.ts
```

2) mpeg3ifochk – check DVD ifo file for usable features.

For some time, DVD manufacturers have been employing a variety of measures to make reading a DVD difficult on a computer. One technique which is widely deployed is to add a bunch of extra program data, so that correct playback is only likely if you can read the DVD virtual machine data and decode a maze of program data to find the undamaged stream definitions. Only a few streams are created which are machine usable, and dozens are created as decoy streams. The decoy streams fail or introduce errors. This program scans the IFO (info file) playlists and verifies the contents of the stream that does not contain obvious damage. The result is a list of program ids which can be entered into the playback preferences to select a program which qualifies.

8.3.4 HDV-sourced Video Recording on a Blu-ray Disc Without Re-encoding

An MTS file is a video file saved in the high-definition (HD) MPEG Transport Stream video format, commonly called "AVCHD". It contains HD video compatible with Blu-ray disc format and is based on the MPEG-2 transport stream. MTS files are often used by Sony, Panasonic, Canon and other HD camcorders. Legal input for Video – MPEG1VIDEO, MPEG2VIDEO, H264; Audio – MP2, MP3, AC3, DTS, TRUHD.

For creating a blu-ray disc, if you have HDV MPEG-2 media that is in blu-ray format, you can save the original quality of your work, rather than rendering it to another format. Follow the steps below directly instead of going through cinelerra. It has been tested on 10 different MTS files.

```
1) du -sb /yourHDVfile.MTS                  ! Determine the size of your file in bytes.
2) blocks=((size-in-bytes/2048 + 4096))     ! Convert bytes into blocks + a little more.
3) mkudffs /tmp/newfilename.udfs blocks     ! Create a file with that # of blocks + some extra.
4) mount -o loop /tmp/newfilename.udfs /mntX        ! Use a mount point like mntX that is not in use.
5) /<cinelerra_installed_path>/bin/bdwrite /mntX /tmp/yourHDVfile.MTS   ! Substitute cinelerra path.
6) umount /mntX                             ! You must unmount the udfs filesystem
7) growisofs -Z /dev/bd=/tmp/newfilename.udfs   ! Replace /dev/bd with your bluray hardware device.
   OR  dd if=/tmp/newfilename.udfs of=/dev/bd bs=2048000   ! if using rewritable blu-ray; replace bd.
```

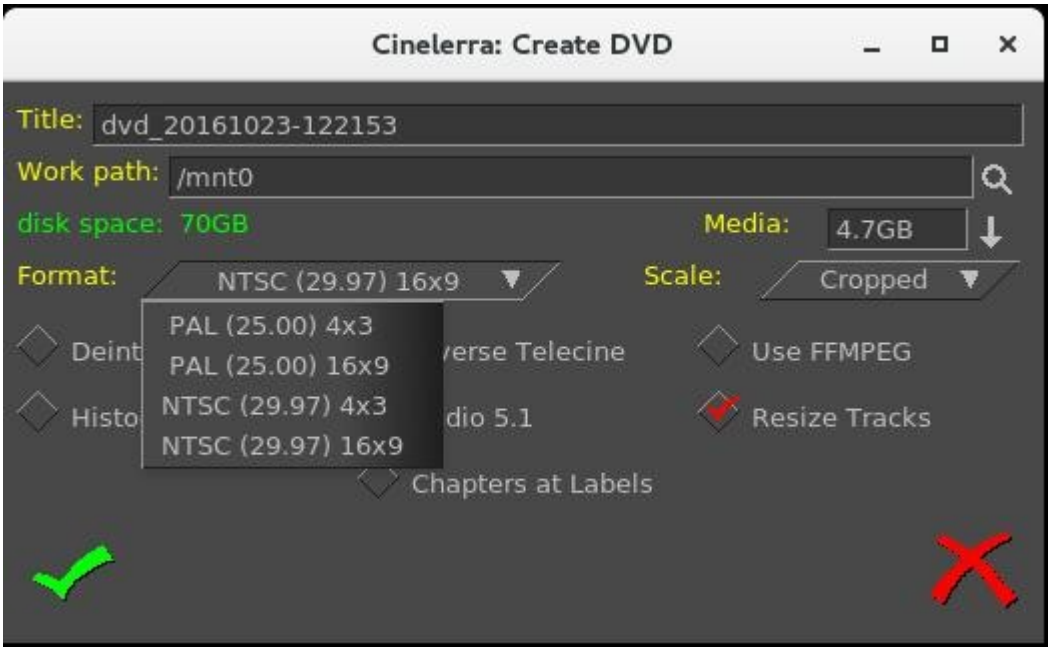
8.3.5 (*FUTURE)* Ultra HD Media Creation

This section is for future expansion of specifics in creating an UltraHD/4K disc/media (abbreviated as UHD for Ultra High Definition). UHD is generally of resolution 3840 pixels wide by 2160 pixels high and is progressive instead of interlaced. It is also referred to as 4K because it is almost 4,000 pixels wide (4 times the number of pixels as on a 1080 display). It has an aspect ratio of 16:9 with 60 frames per second possibility. The same bluray disc media can be used to store UltraHD for single or double layer mode. Only special disc media can accommodate triple layers. Because UHD has a 10-bit color depth, allowing for 1,024 shades of each of the three primary colors red, green and blue, as opposed to the 256 allowed by the current 8-bit standard, it will require Cinelerra 5.1 to be compiled with the 10-bit h.265 library. Currently only Leap has this version available already compiled in a package build.

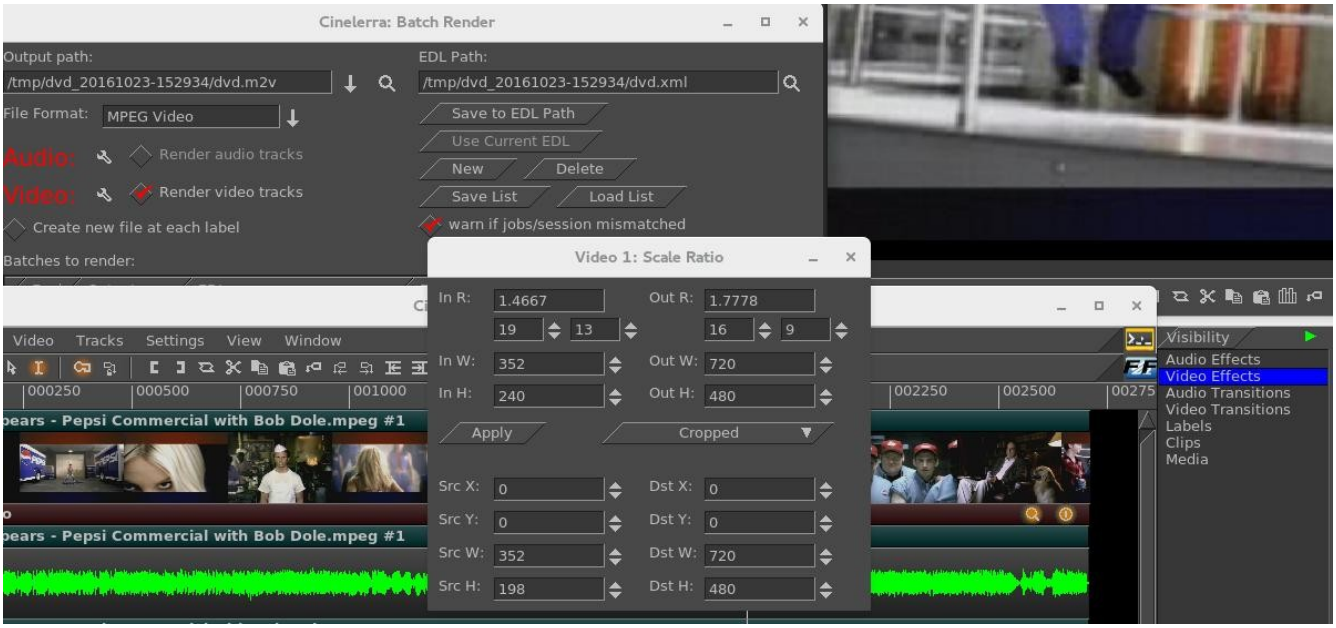
8.4 Complete Example of Video Source with 4:3 Aspect Ratio, Being Transcribed to 16:9 and Creating a DVD to be Displayed on a Digital TV

Illustrated steps to take source input with 4:3 aspect ratio and convert to 16:9, with the bottom of the image being cropped in order to preserve top of video so nobody's head gets cut off are provided here.

0) In order to write to a DVD writer hardware device, you must be “root”!!
1) Start Cinelerra 5.1 to bring up the 4 usual screens with the main track canvas in the lower left corner.
2) Load media via the pulldown File→Load files… by choosing the directory path with the desired file.
3) Bring up the “Create DVD” window using File→DVD Render. This window is shown below.
4) Choose Format: PAL or NTSC with 16x9 aspect ratio for today's digital TV as in below screenshot.
5) Modify the “Work path:” parameter to a disk that has sufficient disk space and you will see the amount of disk space in green letters (/tmp is default, but is often deleted so may be a bad choice).
6) Note that in the following screenshot, Scale of “Cropped” has been chosen.
7) Click the green checkmark on the bottom left side of the window to close it and proceed.

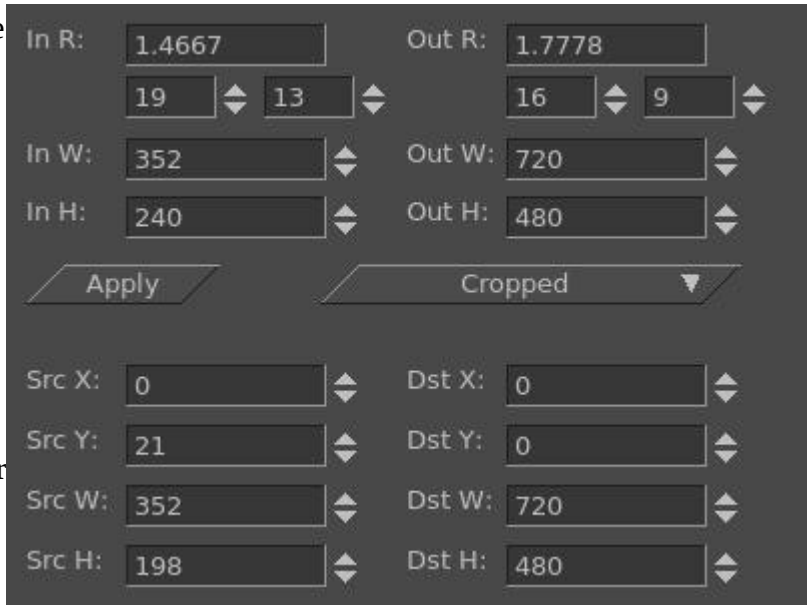


8) Now the “Batch Render” window will appear along with the Scale Ratio brown-colored line below the video in the main track canvas as seen in the screenshot below. Note that in this screenshot the top right most corner of the shot displays the bottom portion of the Compositor window.
9) Next, right mouse click the gold-colored magnifying glass, which is on the right hand side of the brown-colored line. This will bring up the Scale Ratio window that you can see below. Note in the Compositor window, the blue legs are only showing up to the knees.



With the Scale Ratio plugin you can manipulate your video so that it will look the way you want it on a different output Display device. In this case we are going to create a DVD for playing on a Digital TV screen.

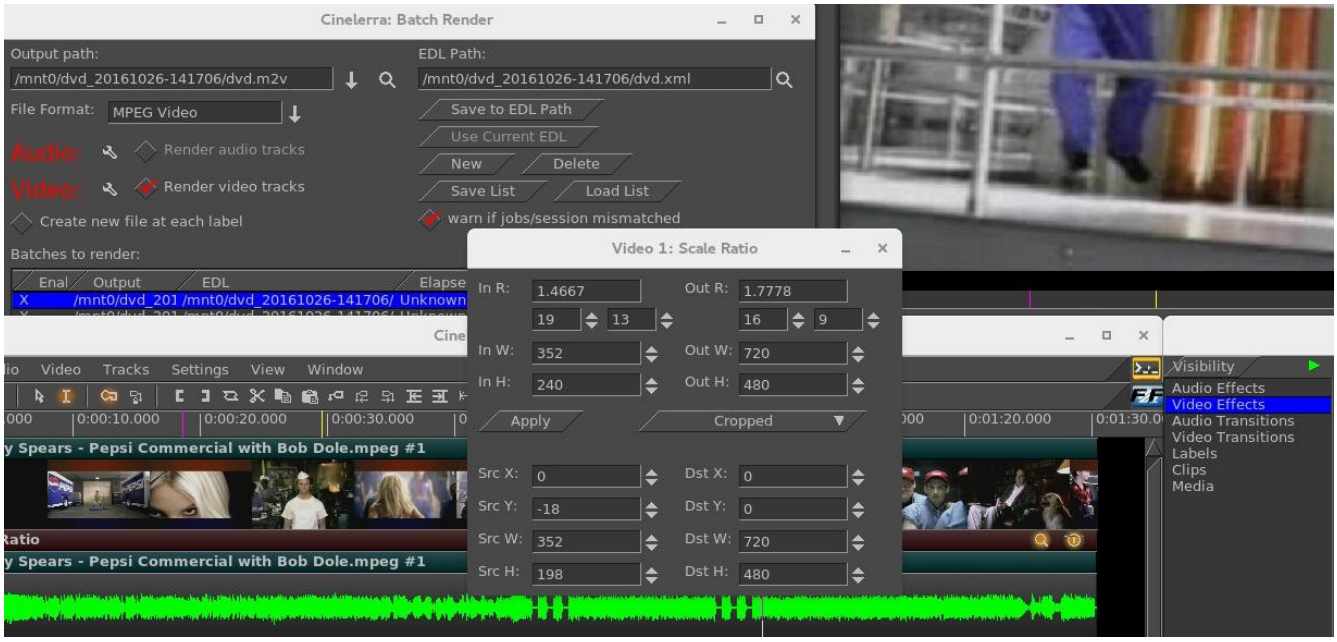
In this screenshot, the left side shows the Input Ratio, Width, and Height of input. The top right half shows desired output values. In this particular case, the input was a YouTube video which was not quite 4:3 aspect ratio.



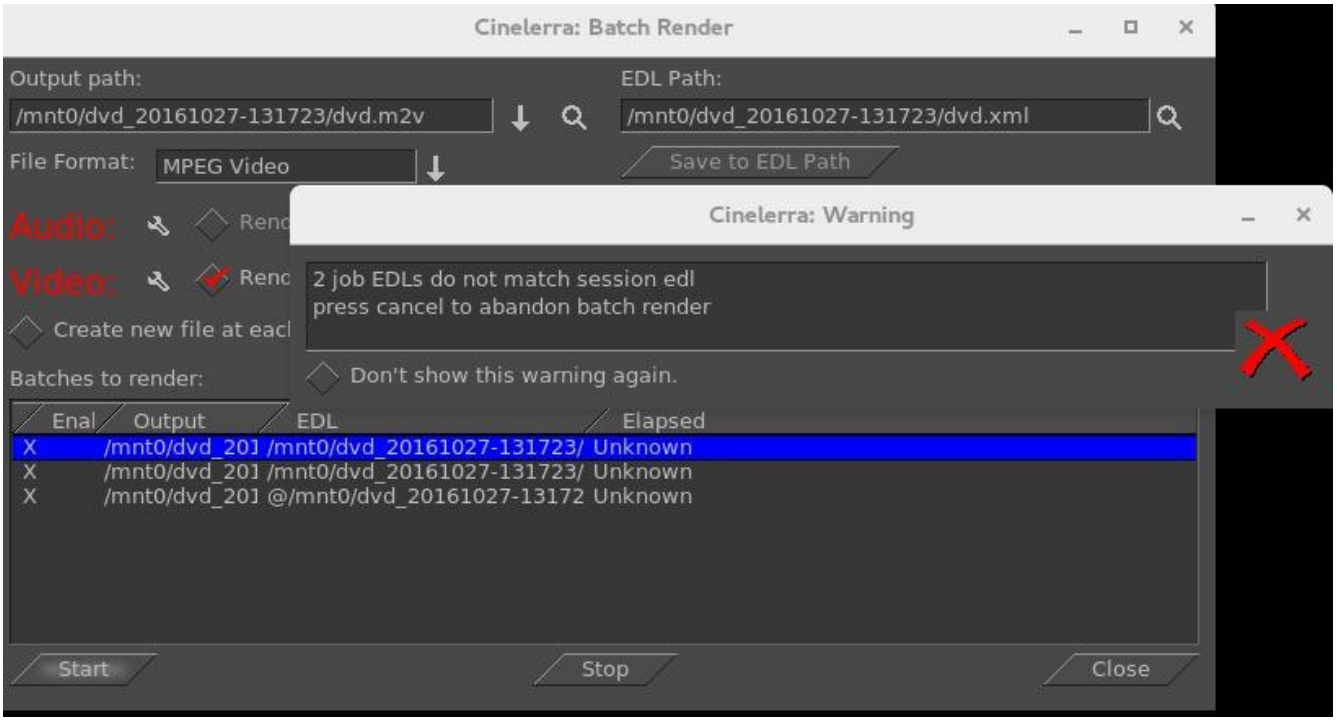
Left and right sides of the bottom portion show the Source and the Destination X, Y, W, H values. As you change the values on the left side, you can see how this will affect the output as you observe the results in the compositor window. For example, as you change the values for SrcY in a “cropped” Scale scenario, you see up/down movement.

Keep in mind that the monitor you are using is NOT the intended output display device – your digital TV is, which most likely will have different looking aspect/pixels, etc.

9) In order to “crop” the bottom of the video in order to preserve all of the image on the top, modify the Src Y value on the bottom of the left hand side in the Scale Ratio plugin. Src Y which was 21 has now been changed to -18. You will see in the Compositor window how the bottom dark colored border is now gone so that none of the top portion which contains a person's head will be chopped off. Compare the next screenshot to the previous and note the blue legs can be now seen to the waist.



10) Click the “Apply” box in the Scale Ratio window.
11) Click the “Save to EDL Path” in the “Batch Render” window for creating a DVD. If you do NOT do this, you will get a Warning box as seen in the screenshot below, to remind you to Save because you have changed the EDL by modifying the scaling parameters in the Scale Ratio window.



12) Next, make sure you have the Timeline set in the Main window at the beginning of where you want to start rendering. Also, make sure the first line in the “Batches to render” section is highlighted as you can see above by the blue highlighting. Click on the “Start” box in the Batch Render window and you will see the video playing in the Compositor window.
13) Cinelerra program will be stopped when done rendering; you will be at the terminal prompt where you will see it has printed out some informational messages (or errors if problems), the last 2 are:

    To burn dvd, load blank media and run:
    growisofs -dvd-compat -Z /dev/dvd -dvd-video /mnt0/dvd_20161027-131723/iso

14) Load a blank or rewritable DVD into your DVD writer device, which will be similar to /dev/dvd as in the “growisofs” line above – something like /dev/sr0 on your computer.
15) Keyin the “growisofs” line, substituting your actual writer device name. Again, you must be root.
16) When back at the terminal prompt, and if there are no errors, keyin “eject /dev/dvd” substituting.
17) Play it on your DVD player connected up to your Digital TV screen.

9. Recording: Web Media / Screen Capture / Television and Playback

The recording application really only does one thing, capture media from some external source, and write it onto a file in a specified way.

Path:	output media file path	
Start time:	weekday / time of day	to begin capture
Duration:	hrs:mins:secs	until end of capture
Source:	channel/device input	selection of input subsource
Mode:	timed	use start time / duration
	untimed	use transport controls

The media file will be written using the format and codec specified in the settings→preferences→ record tab. Only ffmpeg can record both audio and video simultaneously, and some ffmpeg formats require too much cpu to do a realtime compression. A setting which is more likely to be usable (requires less cpu/memory for realtime encoding) is ffmpeg / qt / mp4.qt with bitrates like audio/256000 and video/6000000.

9.1 Record Web Media by real-time Audio Capture/Screen Capture

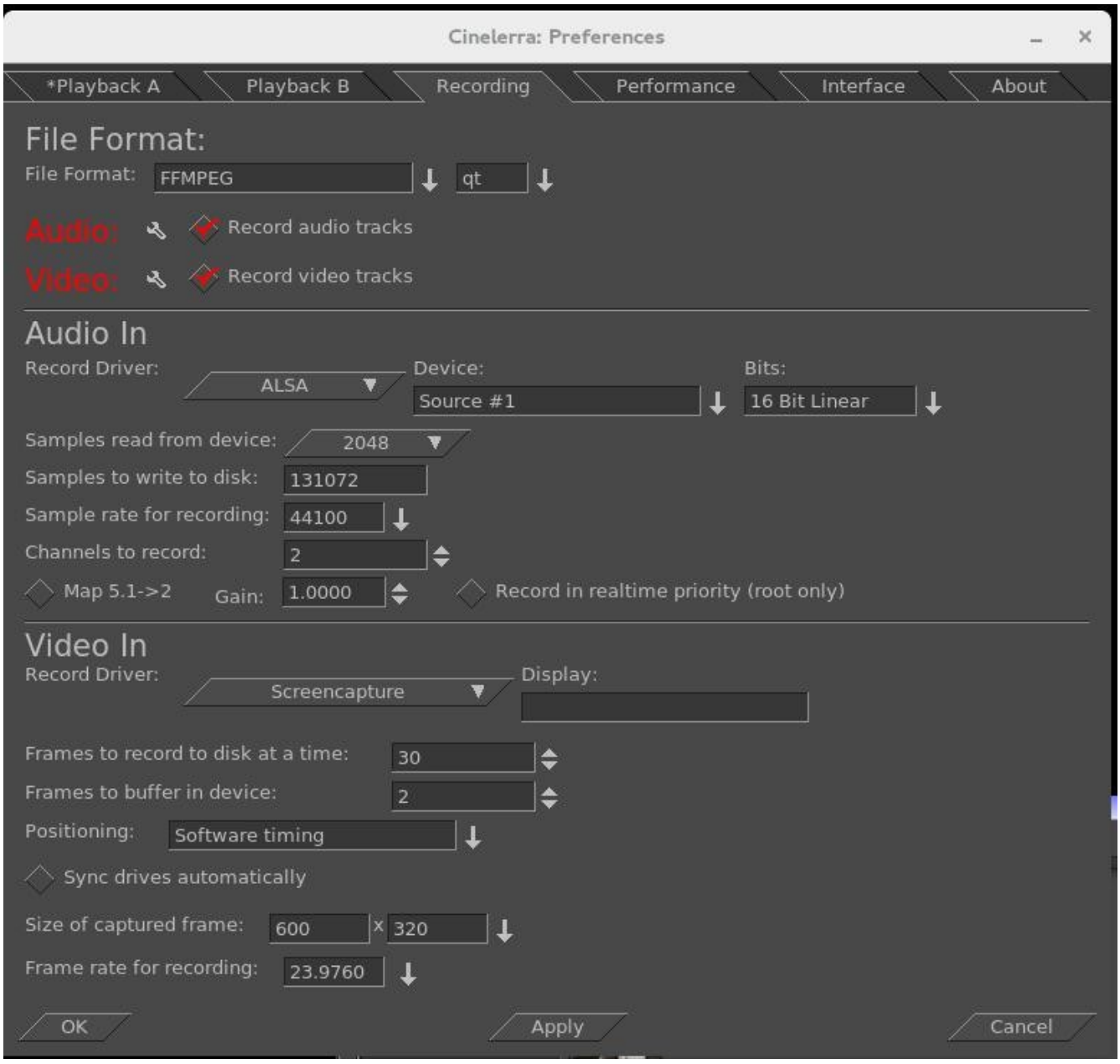
Below describes the necessary steps for recording freely available media from the internet for your own personal use. You have to be on a system using pulseaudio, such as ubuntu, fedora, centos.

1) Start cinelerra and select Settings->Preferences->Recording
   From a terminal (with a wide text window) run: pactl list.
   You will see all of the audio sources and sinks on your system.
   Identify the source associated with the normal output your system uses; example: Source #1.
   Locate the source which monitors your normal audio output. For example: front stereo.

2) Choose a File Format and File Type (for example: FFMPEG & qt).
   Check Record audio tracks.
   Click Audio wrench tool:
   Select mp4, for example.
   Set Bitrate (256000 is a reasonable number).
   Check Record video tracks.
   Click Video wrench tool:
   Select mp4 video, for example.
   Set Bitrate 1000000 - 4000000 (bigger=better, but more cpu)
   Select Audio In->Record Driver: ALSA (or whatever you have instead)
   Set the Bits: 16 Bit Linear (probably, because reasonable and fast)
   Select Device: the source from above pactl list search, for example Source #1)
   Samples read from device: aprox dev buffer size (2k-16k probably)
   Samples to write to disk: 131072 (a good size)
   Sample rate for recording: 44100 (will automatically change)
   Channels to record 2 (probably)
   Uncheck Map 5.1->2
   Gain 1.0

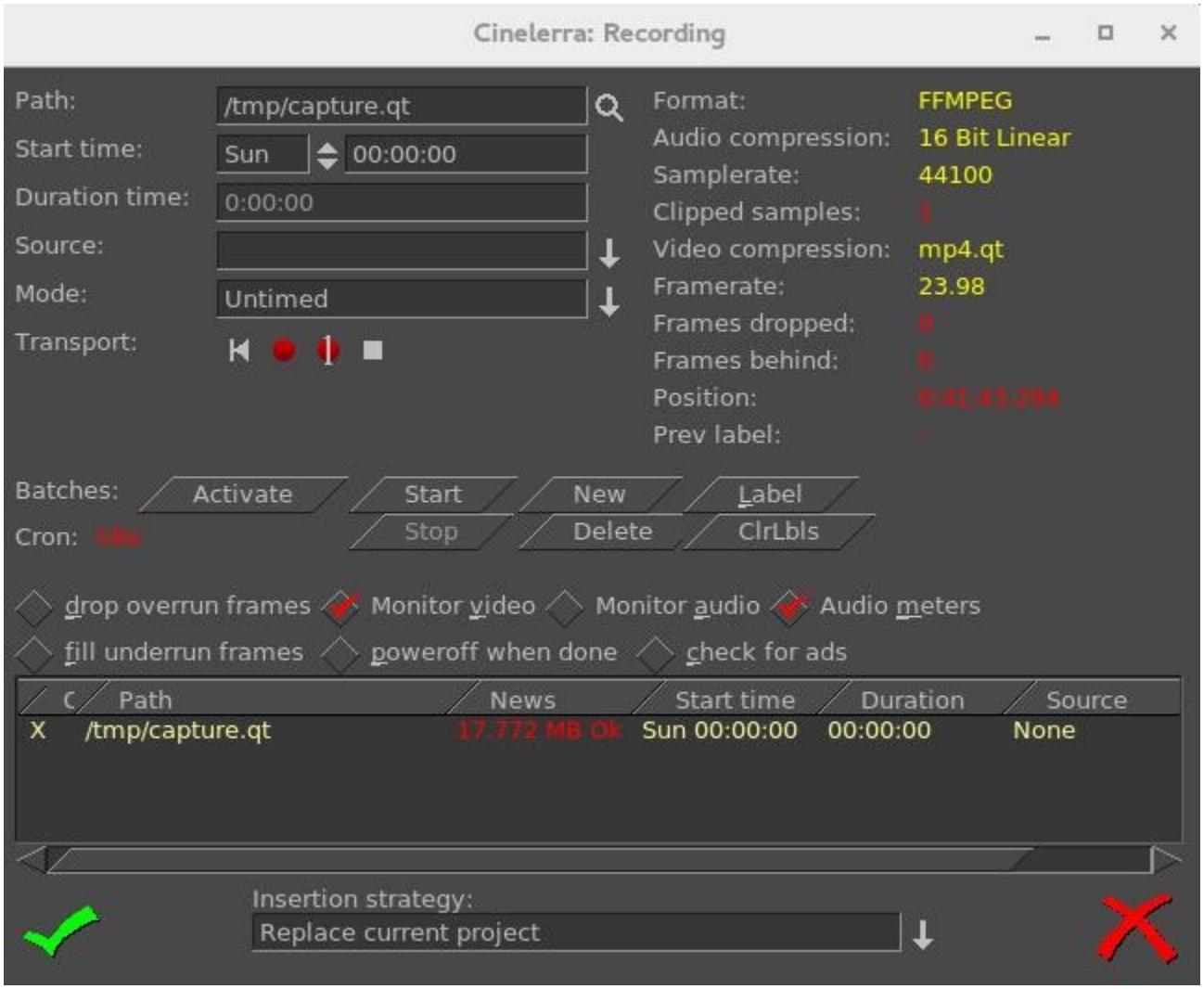
Select Video In->Record Driver: Screencapture
Set the Display: leave blank (probably) or use ":0.0" (default screen/display)
Frames to record to disk at a time: 30 (a good number)
Frames to buffer in device: (2-6) (probably)
Positioning: Software Timing (important)
Uncheck: Sync drives automatically
Size of captured frame: 720x480 (this is your choice, actual capture size; 600x320 youtube)
This defines a “screen capture” rectangular area on the display.
Try to make it the size you need to cover the screen playback area.
Frame rate for recording 23.97 (a good choice, not all choices work)
OK

Screenshot to illustrate some appropriate settings which are described above/below the image.



3) Make sure the Compositor window is up since you will want to see the captured media later.
   From the main window press (lower case r) (move windows and resize if needed)
   Using the mouse pointer, mouse over the “Video In” Recording monitor display.
   Press and hold the left mouse button and drag the display until the desired target screen capture area is properly positioned inside the record monitor “Video In” display. At first this may seem confusing, but keep in mind that what you are actually doing is positioning the portion of your monitor display you want to capture over the “Video In” window.
4) From the “Recording window” select a Path file name for the recording.
5) Select mode untimed.
6) Make sure the “monitor audio” is unchecked else audio becomes looped and is very loud.
7) Press Transport "start recording" (red circle button).
8) Operate the playback of the media source, i.e. start playing the web video.
9) When the media ends, stop on the playback application.
10) Press Transport "stop recording" button (white square button).
11) Before clicking chk-Ok, be sure to set the Insertion Strategy.
12) Press chk-Ok to review the capture, or X to cancel / dismiss application.

Screenshot of what the “Recording” display will look like.



Two things are happening during recording, the data is being written, and it is being rendered. When recording, if video and audio are not in sync, it will usually be video behind audio as opposed to the other way around because video is more difficult to decode. During recording, video frame and audio sample positions are checked and efforts to maintain synchronization are used. Below is an explanation for some of the less obvious settings for recording as shown in the previous screenshot.

Frames dropped – more input than cpu(s) can process. This is the number of frames dropped when the system is too slow to keep up with the video. This value serves as a warning that it is behind and is dropping frames to keep up. You might want to consider stopping to make some parameter adjustments and start over to reduce drops.

Frames behind – active input buffers. This is the number of frames that the video is behind and it will drop frames to get caught up. “Frames behind" depends on the synchronization source, for example, time as the synchronization. This means that frames are not being written as fast as they are being captured and can occur when the system is too slow to keep up. This situation can happen with older systems or cpus where there is more input then can be processed in a timely manner. So that when there are “Frames behind”, frames will be dropped.

The algorithm for determining how many frames to drop is as follows:

If the number of frames behind > 3, only 3 frames will be dropped before it will then do 1 frame;
if the number of frames behind is 2, only drop 1 frame;
if only 1 frame behind, no frames will be dropped.

Drop overrun frames – checkbox when enabled, frames will be dropped. Drops occur when you capture a frame and there is no storage space available to save it, usually because buffer space is exhausted. This situation can occur when there are too many frames coming in too fast and the I/O system is not keeping up in emptying the buffers.

Fill underrun frames – checkbox if enabled, duplicate last frame when no input frame ready for output. As the tooltip states “write extra frames when behind”. This is desirable in order to keep audio and video synched with regards to sample and frame rates. Use underrun padding to fill frames when the input capture rate is lower than the output frame rate. Since demand is constant, you have to output something, so the last frame is duplicated as many times as needed to keep up. You should stop and adjust the frame/sample rates to match the input if not filling underruns, then start over. This situation can occur if the input stops for some reason like lightning interrupts the signal or the internet connection is too slow.

Transport controls - these control buttons mimic the functions of tape recorders from the old days.

Reverse button/left arrow – remnant from the past; does the same as the Start button really.
Red round button – this is the Start button to start recording.
White square – this is the Stop button to stop recording.
Red round button with white line through the center – start recording in single frame mode. The way this works is to just “take a picture now, take a picture now...”. The reason to use this mode is to get a still shot like you would with a camera. Sometimes input is continuous, for example looking at stars with a telescope -- surveillance goes on for hours, but you just want to take a picture now when something of significance interest comes up

Cron – The batch recording watcher, cron, is either Idle or Active. When you start or stop batch recording at specific times, there is a cron thread watching timers to perform the timed action. Idle/Active indicates whether the timers are running.

Position – this is a timebase which tracks frames/samples when obeying frame/sample rate. When a recording starts, it resets to 0. Timing is against audio (when available). Audio time and video time are based on position.

You can select synchronization time source "Positioning" in Settings→Preferences under the "Recording" tab. Possible choices for time base are:

Presentation Timestamps – use time code which is in both the audio and video media input stream. Uses these timestamps to sync the 2 streams.

Device Position – this is the device hardware position of where you are. It is usually only on the audio side.

Sample Position - Sample # or frame # divided by frame rate tells you where you are.

Software Timing – usually used for things like YouTube; it just will "take a picture now"..."take a picture now"… over and over again until you tell it to stop.

Positioning & Timing needs more detailed explanation for complete understanding and application. The overall goal is to maintain media stream timeline synchronization. The reason for providing different "Positioning" options is that different input media may have different timebase standards. Additionally, the input may be damaged. Damaged data can skew the timeline during presentation. The timebase standards make it possible to correctly resynchronize the media presentation to the original time position. For example, the transport layer may have timestamps provided in it. These timestamps record "audio time" and "video time" and are called "presentation timestamps".

The kind of positioning used depends on the input device and the media format. In the transport stream example, the media has already been coded at least once as a stream and contains position information. In other situations, sample position or device position can provide timeline position. If the input device provides data at a variable or asynchronous rate, then software timers can provide timeline position. Sample position is position from the perspective of software, and device position is position from the perspective of hardware. The difference is buffer time. Device position is usually more accurate, but may not be available, or may contain errors.

Examples of media which usually has a particular type of positioning is:

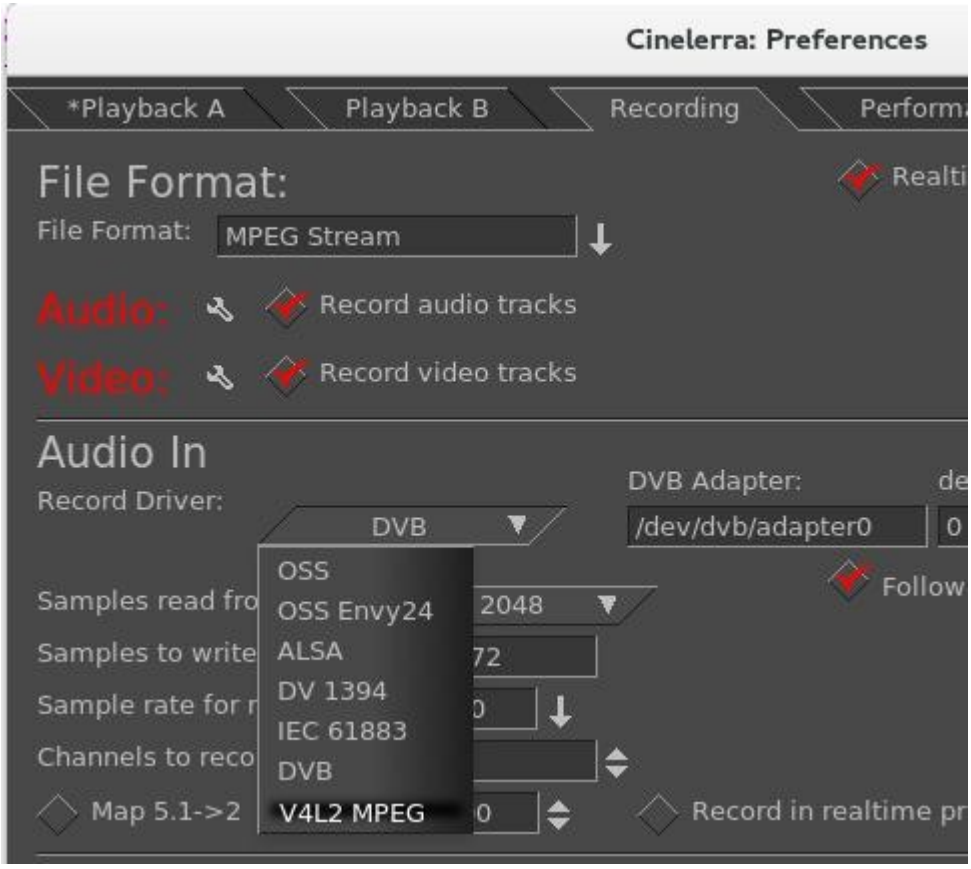
1) mpeg has timestamps therefore Presentation Timestamps is a good recording choice;
2) raw media streams/screen capture have no timestamps, so tell it time via Software;
3) a web cam may supply data at variable rates, so again you would use "Software timing";
4) an example of Device Position is audio (timestamps can come off the device);
5) if Software Positioning working for you, try Sample Position for really bad audio.

Other "Recording" settings are more pertinent to capturing and editing broadcast television with all of its many commercials. Their usage will be explained in more detail in a later section.

Label – create a label. During capture, put a time marker at the designated spot. When the recording gets pulled in later for review, you will see a green arrow marking the spot that was clicked.

ClrLbls – clears all of the previously set labels.
Check for ads - check for commercials. In real time, labels are added when a commercial that matches a previous ad is already in the database (the TDB, Traveling Data Base).

One other noted new feature is a new choice for recording - "V4L2 MPEG". Some digital tuners now stream mpeg data for broadcast data streams. V4l2 has been upgraded to be able to process data from these device streams. Screenshot below shows the available options to include v4l2 mpeg.



9.2 Digital Video Broadcasting (DVB) Capture Configuration

You can do real-time capture of a full transport stream of Broadcast TV. Record television programs using Cinelerra by setting up ahead of time to start recording later at the specified time. Then you can use Cinelerra to watch later and easily fast forward through the commercials or edit the program, deleting the commercials, to watch uninterrupted. Some details may be slightly different than what it is in the United States. There are 3 requirements you must have to take advantage of this capability.

1) You need to have a suitable DVB Adapter such as the Hauppauge WinTV-HVR 950Q usb adapter. Make sure it is the linux version, not the new version.
2) Your DVB adapter on your PC should connect to a broadcast antenna input.
3) For now, you may have to run as the “root” user.

9.2.1 Overview

There is a lot of detailed explanation below to be thorough, but an overview is presented here.

1) Set up Record Preferences.
2) Scan the Channels.
3) Watch TV by selecting the desired channel.
4) Or set up a batch job to record a program and watch later via Shift-S in the main window.
5) Or record a currently running program by using the buttons in the Record window.
6) Load a previously recorded program.

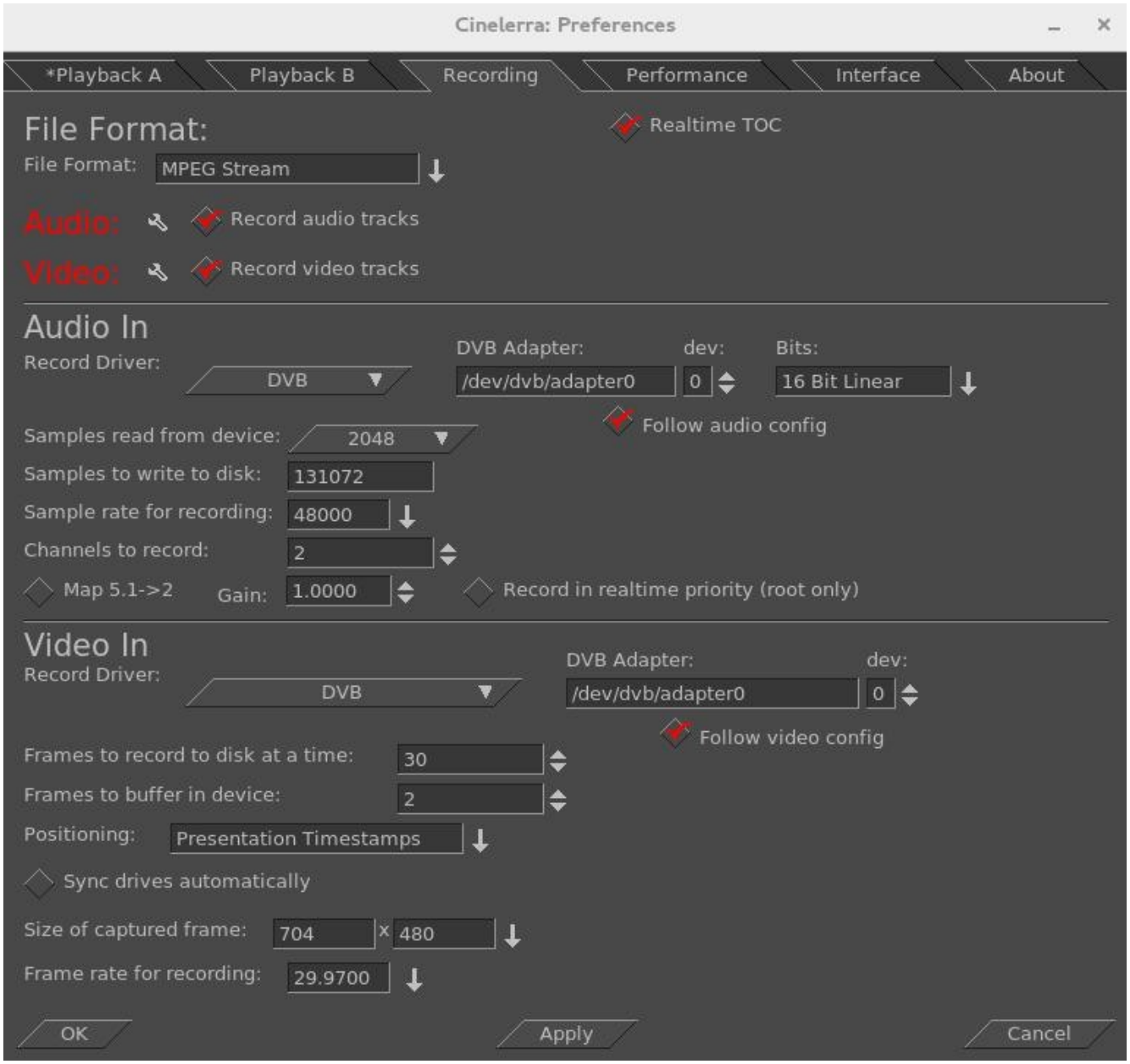
9.2.2 Initial Setup for Digital Video Broadcast Capture

Listed below are the steps for the initial DVB setup. Setup will be preserved until you make changes.

1) Start cinelerra, and click Settings->Preferences->Recording
   Select File Format checkbox: Realtime TOC (Table of Contents)
   It is not necessary to set File Format at this point as it will automatically be set later.
   Check Record Audio Tracks.
   Check Record Video Tracks.
   Select Audio In->Record Driver: DVB
   Set the adapter path = /dev/dvb/adapter0 (probably)
   Set the adapter dev = 0 (probably)
   Set the Bits: 16 Bit Linear (probably)
   Check: Follow audio config
   Samples read from device: 2048 or approximate dev buffer size (2k-16k probably)
   Samples to write to disk 131072 (a good size)
   Sample rate for recording: 48000 (will automatically change, probably to 44100)
   Channels to record: 2
   Uncheck Map 5.1->2
   Gain: 1.0
   Uncheck: Record in real-time priority
   Select Video In->Record Driver: DVB (note, File Format above changes to MPEG Stream)
   Set the adapter path = /dev/dvb/adapter0 (probably)
   Set the adapter dev = 0 (probably)

Check: Follow video config
Frames to record to disk at a time: 30 (a good number)
Frames to buffer in device: 2 (2-6 probably)
Positioning: Presentation Timestamps **(this is important and may not be already set)**
Uncheck: Sync drives automatically
Size of captured frame 720x480 (will automatically change)
Framerate for recording 29.97 (will automatically change)
Click OK

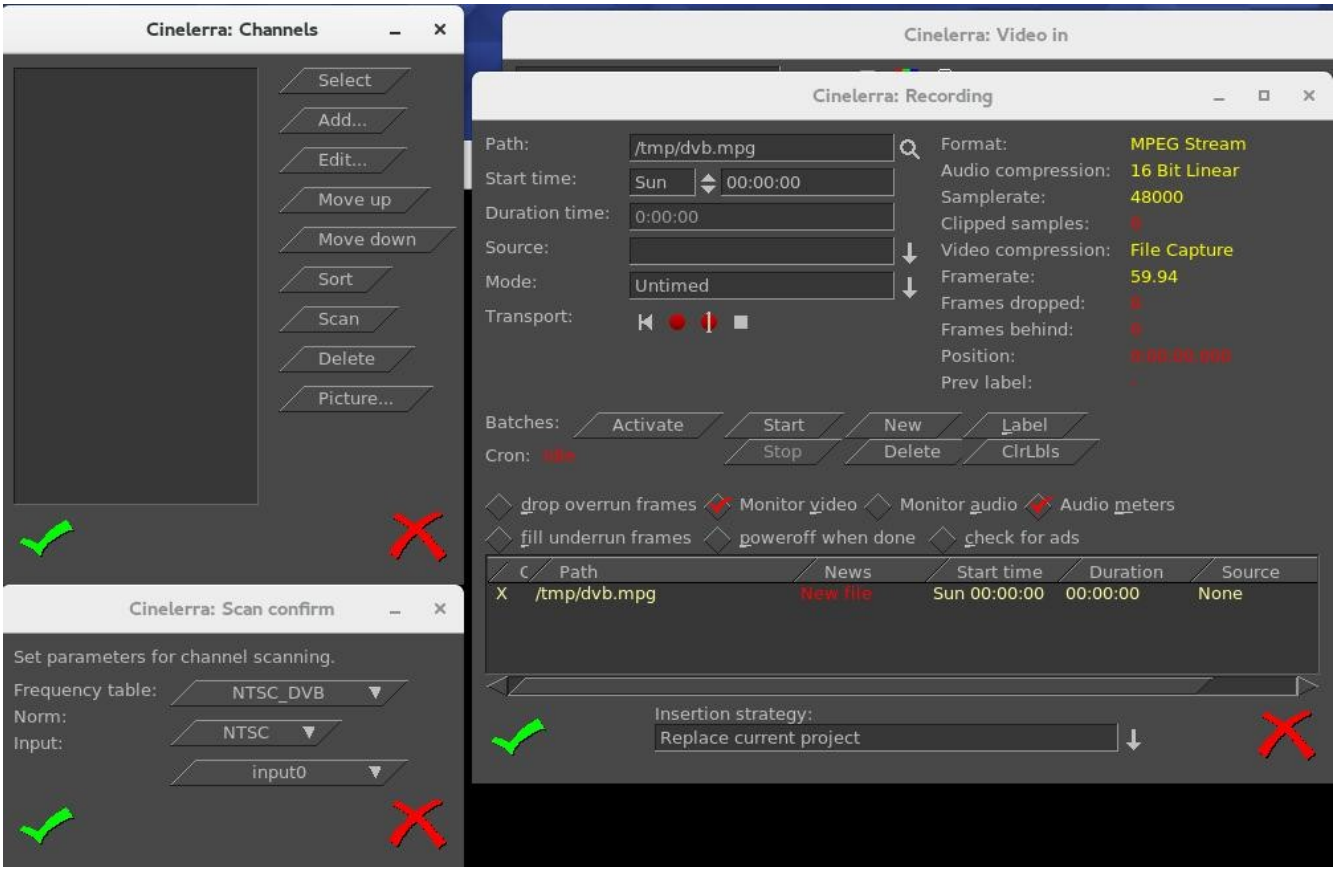
Screenshot below shows the Preferences window as a good example of what you should see.



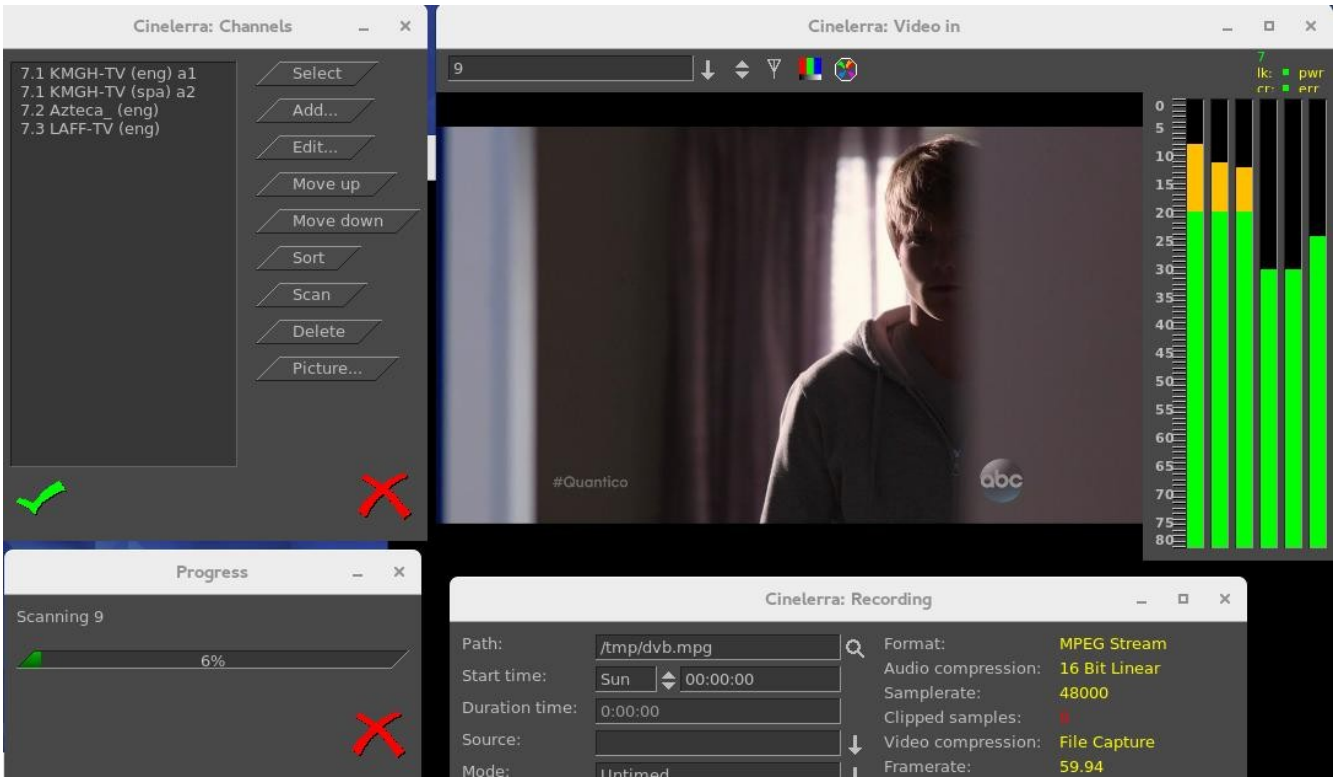
Scan the channels once the devices are setup by performing the following.

2) From the main window pulldown use: File→Record (lower case r) & resize windows.
Two windows appear - the “Recording” and the “Video In”. The “Recording” window should have “Monitor video” checked and “Monitor audio” unchecked. Remember to check the Monitor audio later when you want to listen to a TV progra.
On the “Video In” toolbar, click the antenna button (Edit channels tool tip).
When the Channels window opens, click scan - “Scan confirm” appears.
Make sure Freq table NTSC_DVB, Norm NTSC, and input0 are chosen automatically.
Click OK and then wait for scan to end which may take several minutes.

Screenshot below on the left shows the Channels window and the popup waiting for scan confirmation. On the right side is the usual Recording window with the Video in window hidden behind.



Screenshot below shows scanning in progress. Note the “Video in” with the antenna (3rd symbol).



Any time after the channels have been scanned, you can edit them from the “Recording” window channel selection dialog (shown on left of screenshot above, such as “Sort”). You do not have to scan the channels every time as they are saved in a database referred to as the channel-db.

Click chk-OK when done scanning channels and making any edits you want.

After the channel-db is ready, you can watch any of the selectable channels.

9.2.3 Watching TV using Cinelerra

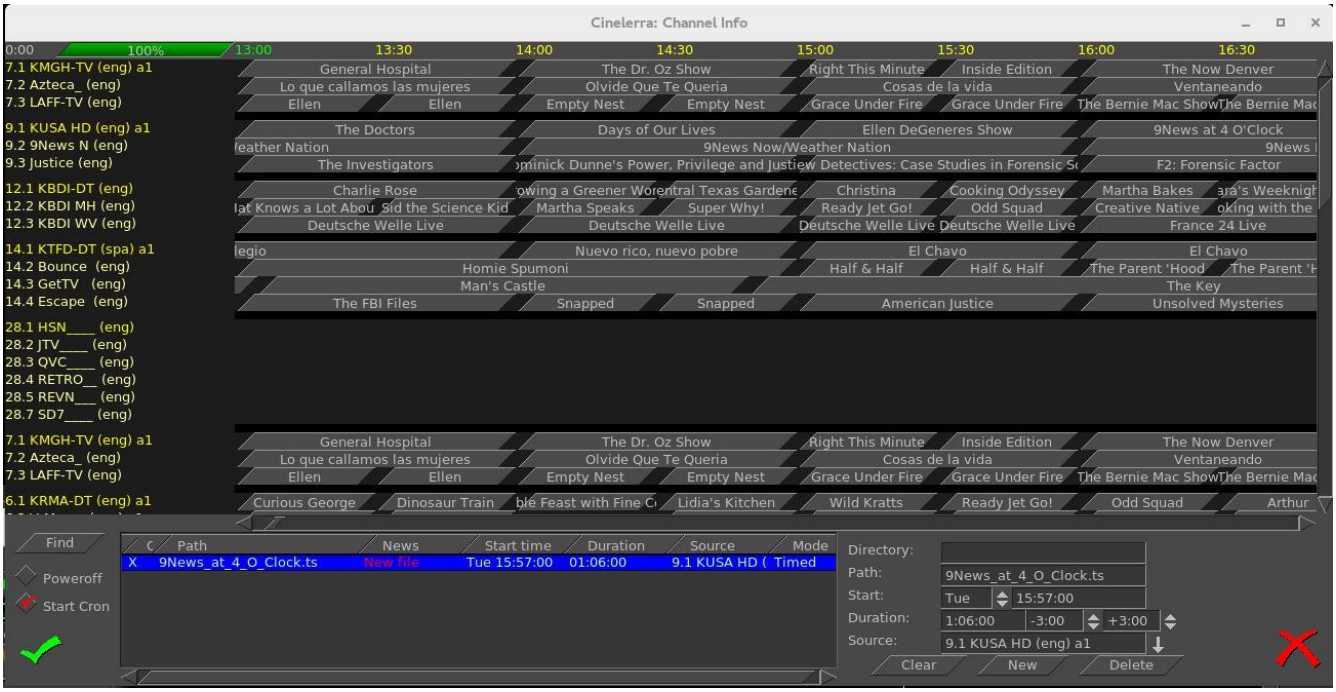
If you have setup for DVB capture as outlined in the section above, just start here to watch TV.

From the main window use File→Record to request the “Recording” application.
In the “Video in” window, press the down arrow on the left to see the channel pulldown. Select the desirable channel, and watch TV. You will have to enable the audio by checking the “Monitor audio” in the Recording window for sound.
Helpful hints: for any computer that you might choose to use that is a little slower, you can improve the available resources for watching TV by unchecking “Audio meters”, checking “drop overrun frames”, and checking “fill underrun frames”. Also, uncheck “Realtime TOC” in the Preferences→Recording window.
Dismiss the “Recording” window with chk-OK, or X to cancel when finished.

9.2.4 Recording TV to Capture at some Future Time via Batch

After the initial setup and channel scan completion, with the Recording application down (so that the /dev/dvb/adapter0 hardware is not in use) you can scan the DVB channel program data by pressing "Shift S" from the main window. It can take a minute or two to scan the channels. The resulting display is a large array of program buttons, with tooltips which are the TV program info text.

Screenshot showing Channel Info window after the "Shift-S" completes scanning TV program data.



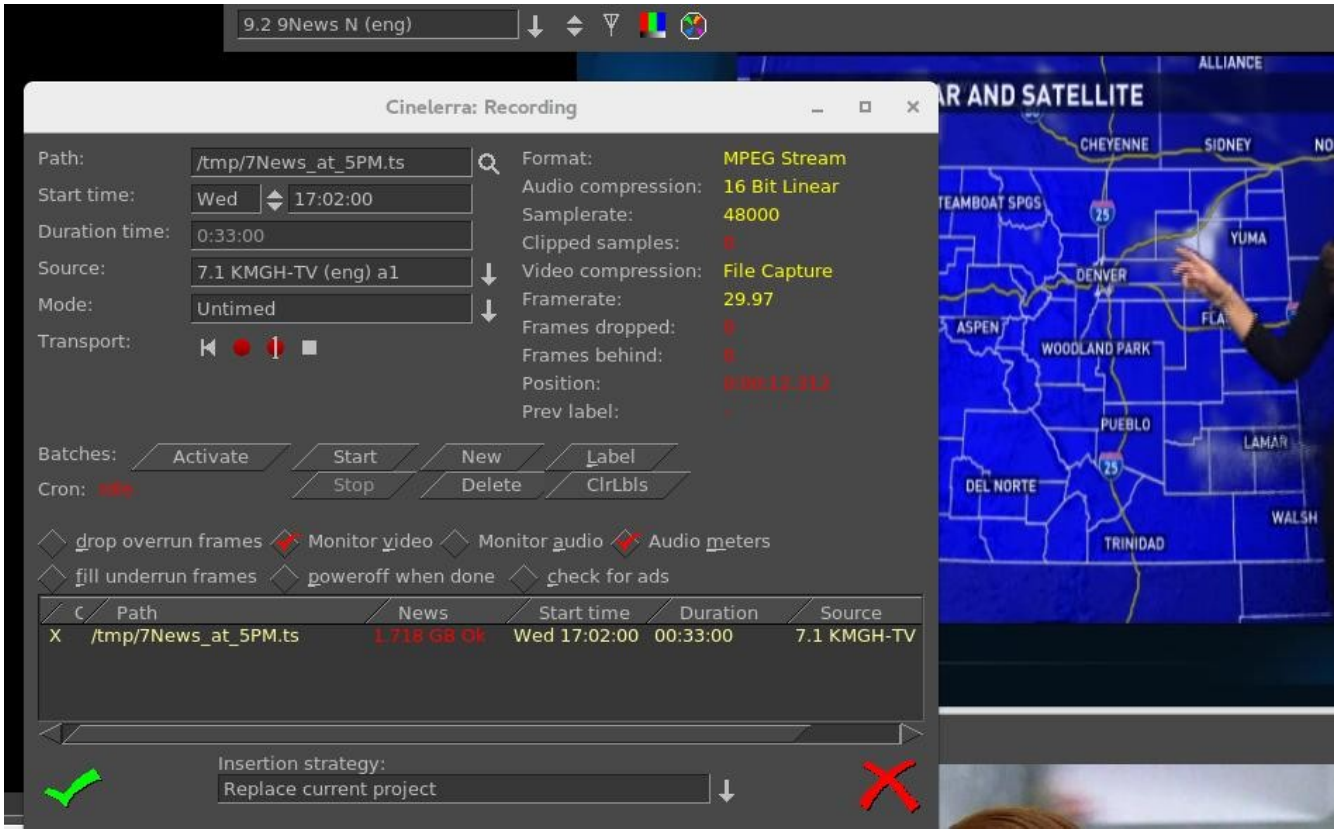
Now if you mouse over a program button, you see the TV program's info. Each of the programs is represented by a button. If you press a button, it preloads the program recording information into the batch record box below. In the bottom right corner box, you should set the "Directory" path of where to write the recording before pressing a button, so that the files will be written to an area of sufficient size to handle the recording (9.5Gb/hr). The two little spinboxes on the Duration line in the right bottom corner are leadin/leadout times to start recording a few minutes early, and end a few minutes late to prevent losing program endpoints. Instead of setting it to the original default of 0, you might want to start 3 minutes (-3.00) before and stop 3 minutes after (+3.00). Also, be sure your computer has the correct time or it will miss. Check to make sure the "Start Cron" option is checked (default). The batch job will start monitoring the channel to be ready for capture once you chk-OK and you will see the "Recording" window and the "Video in" window come up. The computer will power off after completion if the "Poweroff" checkbox is selected (not default). The job line of the active capture in the batch bay will flash continuously once the recording is started.

9.2.5 Recording Immediately to Capture a current TV program

If you have setup for DVB capture as outlined in the first section above, just start here to record a TV program that is currently playing and stop it whenever you want. Media files can be directly captured from the “Recording” window to include Start, Stop, and then loaded via chk-OK.

Bring up the “Recording” window via the main window File→Record (lower case r).
Set the “Path” to the write file Path; be sure there is enough disk space (9.5Gb/hr).
Set “Mode” pulldown to “Untimed” found in the upper left corner area of the window.
Press transport "start recording" button (red round button).
Path will flash and update in batch window and remain flashing while recording.
Press transport "stop recording" button (white square button).
Before clicking chk-OK, be sure to set the Insertion strategy.
Pressing chk-OK attempts to load any newly recorded media data for review..
Or press X-cancel to terminate and dismiss the “Recording” application.

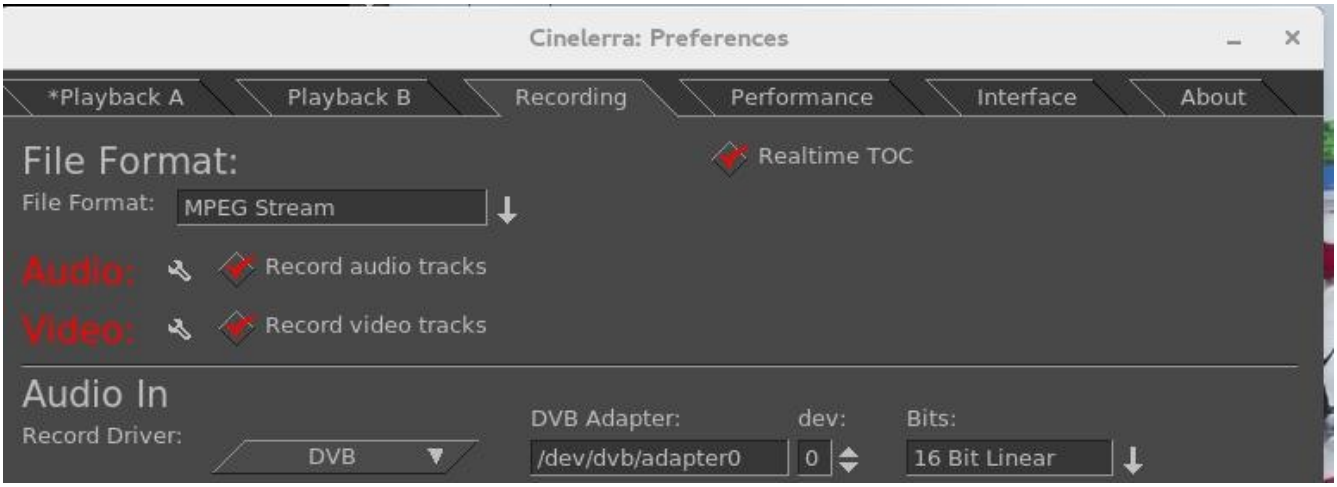
Screenshot showing the recording of a currently running broadcast TV program weather report.



9.2.6 Record “on the fly” Table of Contents (TOC)

If the Settings→Preferences→Recording tab is selected, and the “video Record Driver” is changed to DVB, the file format is switched to MPEG Stream automatically (note that the “audio Record Driver” must also be set to DVB for broadcast recording). This kind of record stream can be captured by simply writing the hardware capture stream directly to the record capture asset file path. But by checking the “Realtime TOC” in the Preferences Recording window, it is possible to scan the stream data “on the fly” on its way to being written while the asset is being captured . The scan is used to generate a TOC – Table of Contents file in the $HOME/.bcast5 directory. This file is used by the mpeg library to view the stream. This avoids the need to scan the file on first open since the TOC builder can run in realtime.

Screenshot showing the red-checked “Realtime TOC” for the “Record Driver” of DVB.



9.2.7 Watching a Previously Recorded TV program

If you have setup for DVB capture as outlined in the section above, just start here to watch a previously recorded TV program. Use the following steps.

In the main window choose File→Load files and select the desired media file to watch.
Broadcast transport streams may consist of multiple programs that all run together on one channel (for example 9-1, 9-2, and 9-3 are all on channel 9). Therefore when you play the recorded program back all of these programs will play simultaneously and that is a jumble. In order to select just the one you want, press Alt-1 in the main window to select Asset Program 1 (or Alt-1...8 to select the desired asset program number) which removes all of the unrelated audio/video tracks except for the selected program. It also resets the session to match the media.
Move timeline to where you would like to start watching.
Be sure the Compositor window is up and you will see the pre-recorded program there. Use either the keyboard, a remote control that has been set up, or Android device for controls. To toggle between fullscreen or original size, press “f” in the Compositor window.

To watch on a large TV screen or monitor which you have connected, you will have to setup for Dual Screen Mode as explained in the next section. In addition, you can either use a Remote Control hardware device or special key setup for manipulating the playback.

9.2.8 Dual Screen Mode

Dual screen mode is useful for displaying cinelerra video/project on a large screen digital Television set or for watching a previously recorded broadcast TV program via Cinelerra. The physical configuration results in one screen (an Xwindow) on one monitor and the other screen on the other monitor or TV screen. Before the mode is usable for Cinelerra, X has to be configured in "dual screen mode" - that is, you must first split the configuration into 2 displays to operate Dual Screen Mode. On some systems, you may be able to get to the dual screen mode by using Xinerama allowing for use of 2 physical displays to act as 1 virtual display.

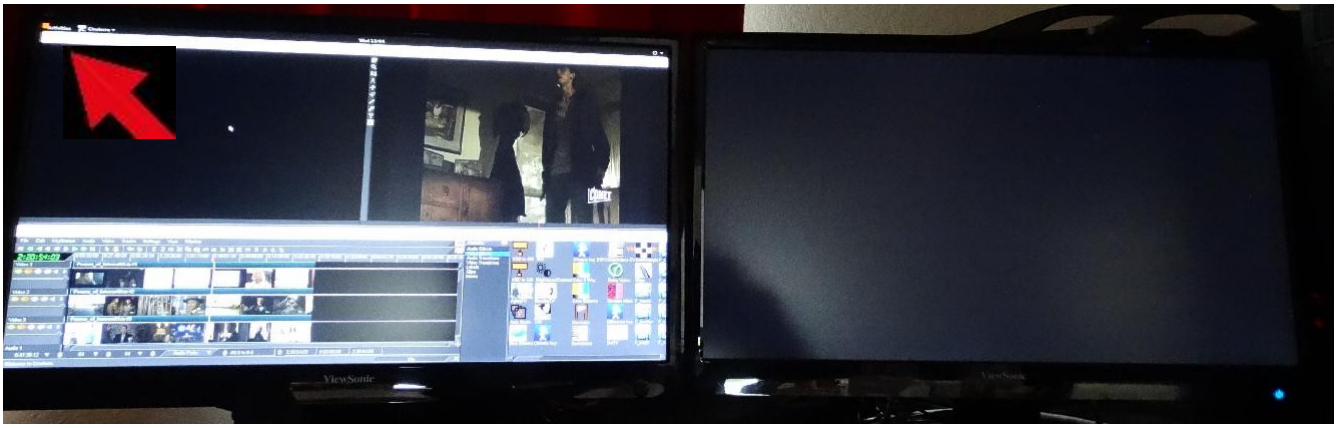
The main menu pulldown, Settings→Preferences→Playback A and B Tabs, must be properly configured in order to take advantage of the Dual Screen Mode capability.

In the main menu Window pulldown there are "Tile left" and "Tile right" options for the 2 monitors. If you have only 1 monitor, you will see no change upon activating these options. Tiling is possible due to the existence of one long horizontal screen display buffer. Tile left takes Cinelerra and moves all of its windows to the 2nd monitor/TV and Tile right puts it back to the 1st monitor. The left and right monitors are the left part or right part of a Playback A/B Xwindow host referenced in the Video Config Playback A/B for separate X screens. Shortcuts for Tile left and Tile rights are the letters a and b on the keyboard once you have set Remote Control mode. Below is a summary of the configuration letters:

a - select playback a and tile windows left (all on monitor or laptop)
b - select playback b and tile windows right (all on 2nd monitor or TV)
c - select playback c and tile windows left, but composer right (TV mode)
This option has the effect of taking the Compositor window and moving it to the 2nd monitor/TV without the surrounding borders interfering with the video.

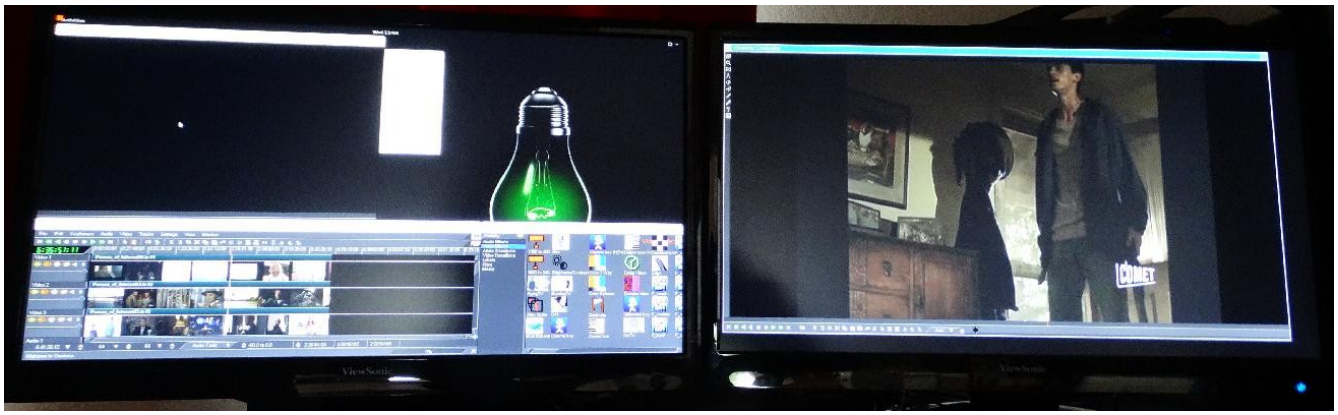
Since Dual Screen Mode is most widely used for either a demonstration or watching TV, you will want to take advantage of Remote Control mode as described in a following section. In the camera shots below, note the square red box that shows up in the upper left hand corner of the Viewer window (arrow pointing to it but it looks more orange than red) which denotes that the Application/Menu key was used to get into Remote Control mode.

Camera shot below shows 2 monitors (could just as well be a big-screen TV for the second monitor) with Tile left where all of the cinelerra windows are on the left most monitor. The big red arrow points to the little red square box (actually may look orange) indicating that Remote Control mode is in effect.

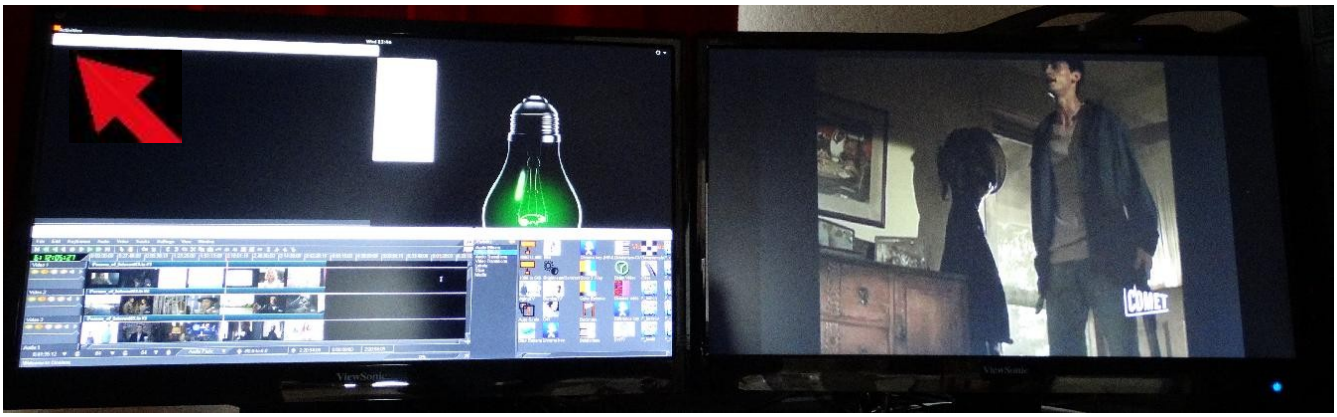


Camera shot below shows 2 monitors with Tile right (b key) where all of the cinelerra windows are on the right monitor. The left monitor shows the Suse distro logo and a couple of xterm type windows.

Camera shot below shows 2 monitors with Compositor window on the 2nd monitor which was obtained through use of the “c” keypress available when in Remote Control mode.



Camera shot below shows 2 monitors with Compositor window on the 2nd monitor in FULLSCREEN mode so that no cinelerra borders are visible to distract from the picture. The big red arrow points to the little red square box (may actually look orange) indicating that Remote Control mode is in effect.



9.2.9 Remote Control for DVB

Cinelerra DVB recording/playback can now easily be done in “couch potato” mode using a remote control. This comes in handy when you want to playback on a big TV screen for multiple person viewing. You can use the Application/Menu key on the keyboard to toggle between standard Cinelerra usage or a Dispatcher methodology through use of a remote control, an Android programmed device such as a tablet, or keyboard. The Application/Menu key on most keyboards is between the Alt and Ctrl keys on the right hand side bottom and depicts a menu on it either with or without a pointer (see image below). An ati-x10 Remote Control device is currently working with Cinelerra and other remote controls may work but have never been tried. The red/orange box on the top left main screen of Cinelerra indicates Dispatcher Method access is activated which allows for keyboard grab which routes all key strokes until toggled off. Below are the currently defined operations.

Remote Control Keys (Application/Menu key toggle for ati-x10 remote)

up arrow -	forward 1 minute
down arrow -	backward 1 minute
left arrow -	back 10 seconds
right arrow -	forward 10 seconds
d -	toggle channel scan menu
f -	toggle full screen
a -	select playback a and tile windows left (all on laptop)
b -	select playback b and tile windows right (all on TV)
c -	select playback c and tile windows left, but composer right (TV mode)
0-9 -	select timeline by 1/10s from beginning to end
a -	toggle audio mute
m -	toggle audio meters
keypad / -	display mpeg ts stream program data
keypad * -	display mpeg ts stream channel schedule date
keypad + -	change channel, next
keypad - -	change channel, prev
keypad . -	start channel change by numeric entry
keypad .. -	clear channel entry
book key -	toggles channel scan (always available)
hand key -	terminates cinelerra (always available)

Plus usual Transport keys

e -	??
f -	toggle full screen
<< -	fast reverse
> -	toggles normal playback
>> -	fast forward
red o -	slow reverse
square -	stop
2 lines -	fast reverse

Ati-x10 Remote

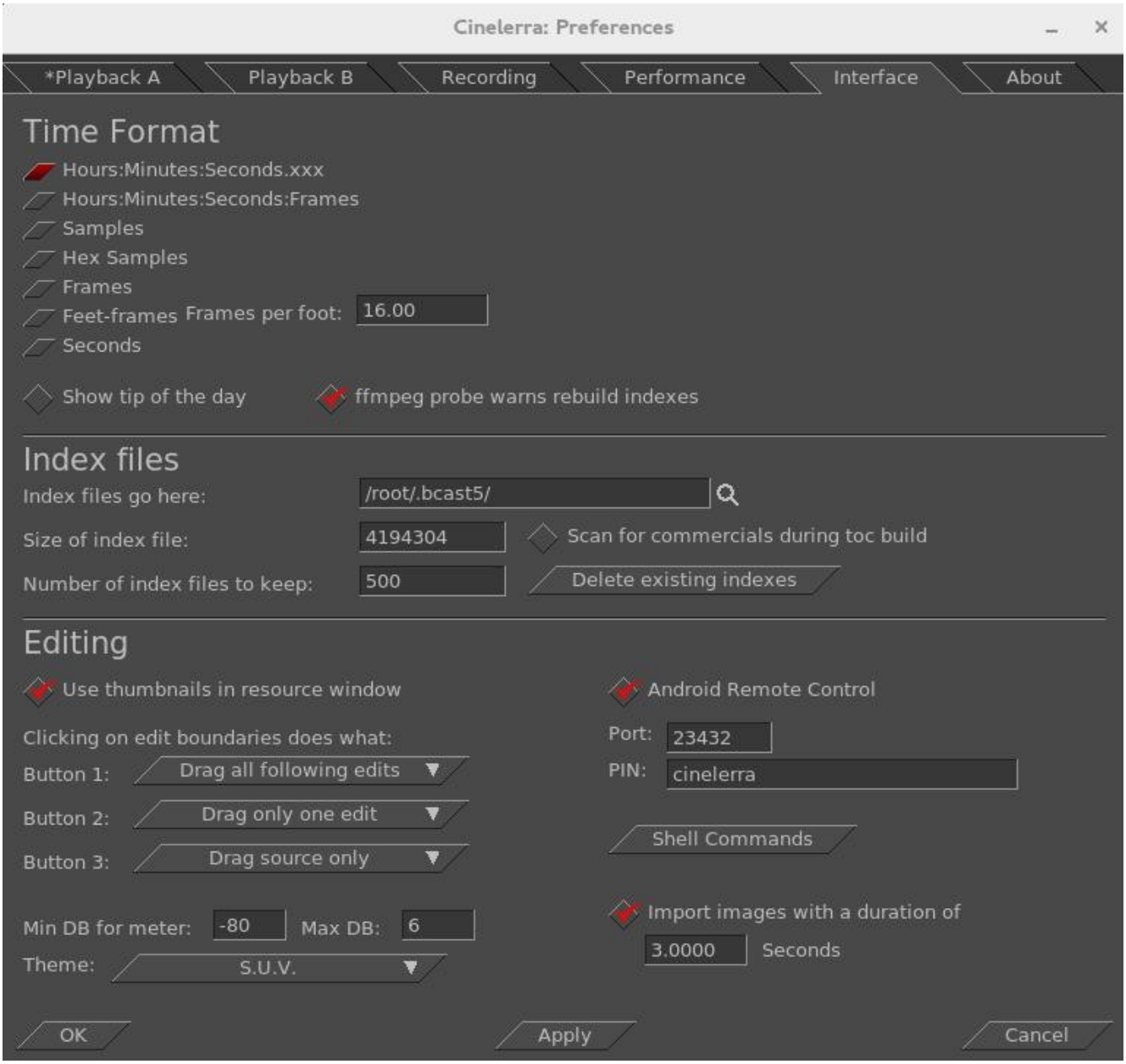
Application/Menu key right bottom of keyboard

The Application/Menu key is used to get into remote control mode, even if there is no remote control hardware device in use, allowing for use of the same defined remote control keys to work on the keyboard. Once the Application/Menu key enables remote mode, the remote control keys don't allow the standard Cinelerra keys to work.

9.2.10 Android Remote Control for DVB

Cinelerra 5.x has android remote interface code. Any device, such as a tablet or a phone, can be used as long as it is running the Android operating system. Programming an Android Remote Control for DVB is a bit complicated at first, but becomes pretty simple after a bit.

1) The Android Remote Control requires you to download and install the android CineRmt app package (apk). To install it simply download it (any way you can) to your phone or tablet and click it to install, if it doesn't run automatically.
2) You should be able to open Settings→Preferences→interface tab. And see Android Remote Control in the editing section. Enable it using the checkbox. The default port 23432 and default PIN cinelerra are OK.
3) After you install the new phone/tablet app in step 1, start it up from your apps menu.



Before you can use it, the Android device must have the ip address of your computer entered into the configuration menu. The PC you are running Cinelerra on and the Android device have to be on the same network. You can tell it is in communication by tapping the Power (menu) button. When the remote is operated with the Power (menu) button, there is a little orange box on the upper left corner of the display to indicate the remote state is active. If you do not see the little orange box when the menu key is tapped, the communication is broken and must be fixed before proceeding. Follow the directions in the paragraph below.

BIG NOTICE: the firewalls in your computer and wifi router can stop this thing dead. Re-configuring a firewall is tricky and varies from distro to distro. If possible, during setup it is suggested that you disable the firewall temporarily. For those with advanced skills, use tcpdump or wireshark to look for udp messages from the IP address of the device.

SECTION NOT COMPLETE – (manualandroid)

Screenshot of an Android Tablet that can be used as a Remote Control for Cinelerra Recording and Playback.

9.2.11 How to Setup a Remote Control Device to work with Cinelerra

SECTION NOT DONE !! This section explains how you can take your own brand of remote control and set it up to easily work the way you want with Cinelerra.

9.3 The commercial DB... What is that? A TDB – Traveling Data Base

While watching broadcast television, it is possible to use Cinelerra to automatically mute commercials being aired during a broadcast capture. Visual muting as well as sound muting is done! You can NOT use ffmpeg for the format when working with the commercial DB. This database that was added is a “novel architecture” shared memory database, a Traveling Data Base (TDB). With TDB you have direct access to the shared database presence (instead of having to go to the server).

Basically it works as follows, but there are a few differences in just how it goes about it:

1) Identify a zone in a timeline as a commercial.
2) Characterize it concisely, in the audio and video timeline.
3) Store the characterization in a database.

Then, when capturing media or building a table of contents for an mpeg transport stream (broadcast capture DVB):

1) Characterize each I-Frame (a few frames per sec).
2) Search the database in real-time to identify the image.
3) If a match is identified, automatically press the mute button.
4) Unmute if the pattern match fails, or the commercial ends.

It mostly works, but it takes more time to setup and maintain the database as the advertisements change frequently or vary by a small amount. It would take a small set of personnel to keep the database current and some finessing of the code.

Currently, to activate the commercial database (db) methods you must do the following as root:

```
cd <cin-path>/bin
mkdir /cinelerra                    (yes, currently only in the / filesystem)
./cin_db /cinelerra/media.db  (to create a new empty commercial capture database)
export BC_USE_COMMERCIALS=1  (in the command shell that is starting cinelerra)
```

This activates the gui tools and database access. As of yet, all of the requisite permission changes necessary to run as non-root have not been worked out (initially just easier to run as root).

When a commercial is "cut", the following occurs:

1) transmits the capture timeline fragment to /tmp
2) scans the snip to characterize it statistically
3) creates a miniature b/w movie of the beginning and end
4) stores all of this into the media.db

There are 2 ways currently to identify a commercial:

1) Load mpeg3 transport media, highlight select the commercial in the timeline, and press "cut commercial" on the toolbar.
2) Use the remote control interface to indicate the boundary in/out cut points in the realtime

capture timeline, and operate a cut command.

The cut command creates a popup to let you know that it is processing the captured data. This can take several seconds depending on the size and media format. The results from the scan are stored in a new record in the “media.db”.

The stats are somewhat disappointing. Image recognition is way past zero, it is not 100 percent. Broadcast streams have tons of minor variances which hamper digitally perfect reproduction from instance to instance of commercial presentation. This puzzle remains to be solved.

To review media stored in the “media.db”, and delete old expired commercial data records, press (Shift-M) from the main window. This will create a media-db window which shows the various captured samples stored there.

Currently, only a customized ati-x10 (kernel mod required) radio remote control works. But this can quite readily be modified for other remote control models.

10. Themes User Interface

Akirad themes have all been added to Cinelerra 5.1 with major thanks to the Cinecutie project for all of their hard work. These modifications provide alternative User Interface environments and allows you to choose your favorite look. Be aware that when you change your theme, Cinelerra will automatically shutdown and restart so that it takes effect. The theme you choose is preserved across sessions.

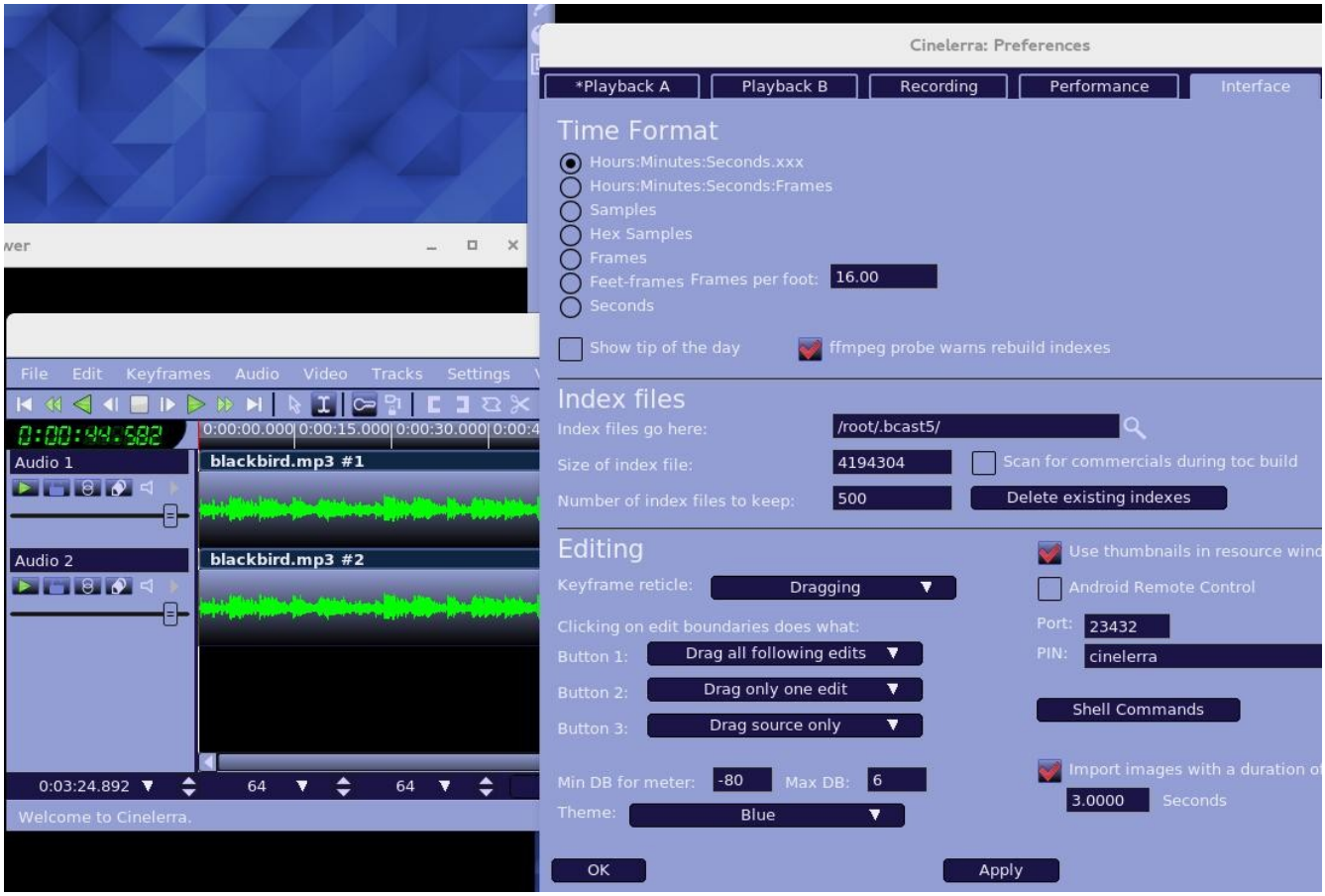
Theme choices available:

- Blond
- Blond-cv
- Blue
- Blue Dot
- Bright
- Hulk
- Pink Lady
- S.U.V. (default)
- Unflat

How To Change your Theme:

To change a "Theme" in the main window pulldown,
select Settings→Preferences then
in Preferences window, click on the "Interface" tab.
In the Editing section in the lower left hand corner,
click on the down arrow next to Theme to see your choices.
Click on your desired choice from the list given.
Check OK, cinelerra will automatically shutdown and restart.

Screenshot showing the "Blue" Theme and the Preferences Window where you can change Theme.



11. Resizable Fonts/Icons Support

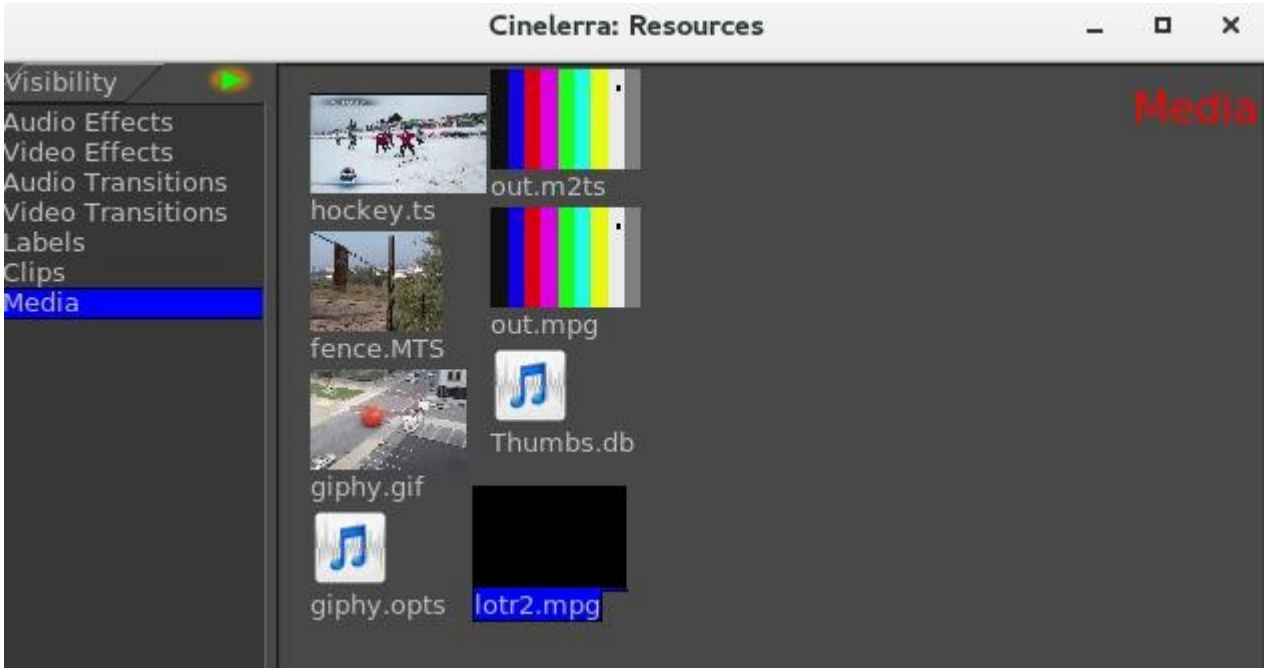
You can increase the size of the characters in the fonts and icons on your Cinelerra system. This will make it easier to read the characters if you have trouble seeing the default small letters, which have been auto-scaled based on the window geometry. The user-friendly font/icon scaling default is 1. To defeat default auto scaling and get any size characters/fonts, override the setting via shell environment variables. For example:

```
export BC_FONT_SCALE=2.0 ! twice as big, may show letters beyond the borders
export BC_ICON_SCALE=1.5 ! 1.5x bigger
```

12. Vicons – aka Video Icons

Vicons are video icons. The Resources window now will play the first 5 seconds of video looped in the area occupied by the video media icons. This is only enabled for the "media" folder in "icon" mode when the mouse pointer is actually inside the Resources window. There is also a "playback" toggle to enable/disable the icon playback activity. This playback toggle is a green arrow to the right of the Visibility label as seen in the screenshot below. It is active when surrounded by a gold background and inactive if no background. When it is enabled/active, if you click on one of the vicons, a view popups that increases the size to 4 times larger. This makes it easier to see if it is the video you are looking for in case you have many similar media files. To conserve memory, the video is stored 8-bits per pixel which results in low image quality. The reason for playing 5 seconds of a video for a vicon is that until the first I-frame, the media frequently does not decode properly. In other words, a lot of media does not begin at the “beginning” point and will not be properly rendered until enough data has been read to assemble a picture.

Screenshot showing the Vicons in the Resources window with the Playback toggle “active”.



13. Toolset Visibility Support

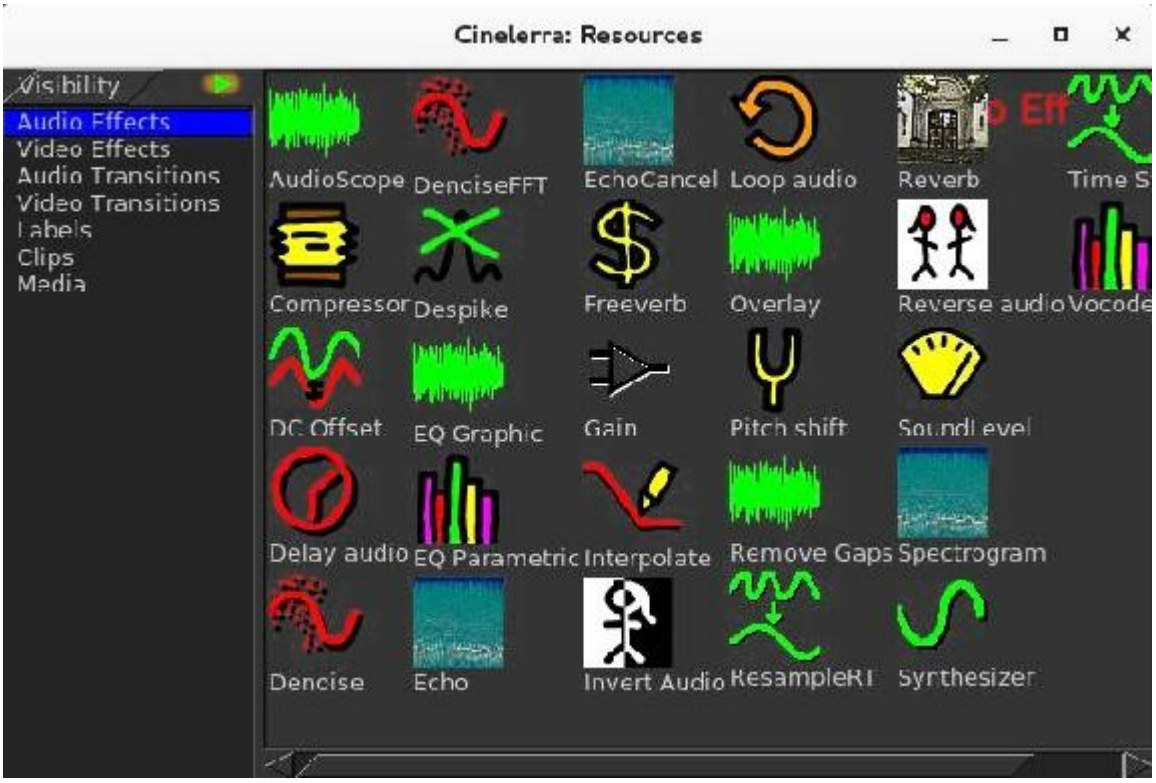
Cinelerra contains many plugins, especially with the recent addition of ffmpeg, and it is somewhat difficult to find the one you are looking for in the Resources window. In 5.1, the plugins have been categorized into the following subsets in the Visibility section of the Resources window to make it easier to locate a particular one:

Audio Effects, Video Effects, Audio Transitions, Video Transitions, Labels, Clips, Media

Assistance in further categorizing these plugins would be welcome. There may be additional categories needed or plugins that are currently miscategorized for a more intelligent grouping. There is a text file called “plugin_defs” located in the path where cinelerra was installed, that currently contains the definitions for plugin groupings. Email any recommended changes. Below is an example:

```
plugin_dirs += colors
colors := brightness bluebanana C41 color3way colorbalance huesaturation …
```

Screenshot showing on the left hand side the Visibility box with Audio Effects highlighted.



Another helpful feature of the “Visibility” toolset in the Resources window gives you the ability to turn off or on any of several sets of plugins. If you left-click the Visibility box, you will see the various categories of plugins, such as ladspa, ffmpeg, colors, transforms, etc. Highlight the set you want to turn on and a check mark appears to show it is active. Highlight again to toggle it off. And handiest of all is the ability to add your own personal directory of plugins which will show up here. See the

attached screenshot which illustrates that all of the plugins are turned off (not visible) except for your own personal set, called “old_plugins”, which just contains the AgingTV plugin. All you have to do to have these plugins become visible is to create a directory, with some name that is meaningful to you, and put your .png files in your cinelerra_path bin/plugins/<your_directory_name>.

Screenshot showing the Visibility categories of plugins with all toggled off except old_plugins.

14. Menu Bar Shell Commands

In order to provide some configuration control, the Menu Bar Shell Commands are available for customization purposes. In the main window on the top line containing the File, Edit, … Window pulldown menus, all the way to the right hand side is the “shell cmds” icon. You will see a small gold-color bordered box with the >_._ inside and if you mouse over it, the tooltip says “shell cmds”. This is a configurable popup which gives you the ability to execute a pre-defined script to do something such as post processing or whatever needs to be done on a repeating basis.

The Shell Cmds popup menu items are configured in:

Settings->Preferences->Interface->Shell Commands

This runs a dialog that lets you Add/Del/Edit the items in the main menu popup. Possible options are:

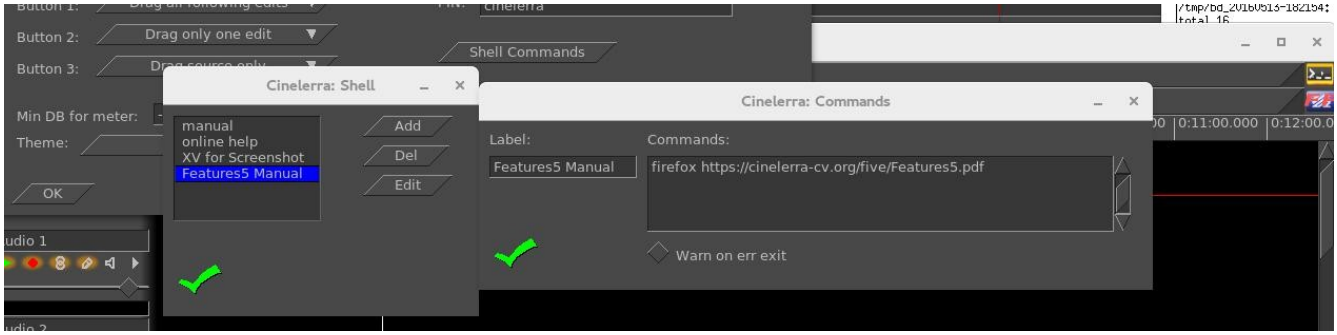
1) Add a new menu item/script:
click Add
in the Label box, change the word "new" to the desired menu label
in the Commands box, type the command lines to be included in the script
(for example: Label => ls; Commands => ls -l
check-OK in the Commands window
check-OK in the Shell window

click on OK or Apply in the Preferences window

2) Del to delete an existing menu item (for example “manual”):
select to highlight an entry in the Cinelerra: Shell listbox
click Del
check-OK in the Shell window
click on OK or Apply in the Preferences window (the entry is destroyed)

3) Edit an existing menu item:
click Edit
select to highlight an entry in the Cinelerra: Shell listbox
click edit (the edit dialog appears)
set the label and/or commands to the new desired values
check-OK in the Shell window
click on OK or Apply in the Preferences window

Next time you click on the “shell cmds” icon, you will see your new labels.



15. Last Play Position Memory

When you play media, the start/end playback positions are saved as if they had been made into temporary labels. They appear on the timeline as purple/yellow hairline markers representing the last start/end labels for the last playback. They can be addressed as if they are label markers using:

Ctrl← tab to the label before the cursor, that is “play start”
Ctrl→ tab to the label after the cursor, that is “play stop”

You can use these markers for re-selection. Additionally, the selection region can be expanded by “pushing” the markers using single frame playback. Use frame reverse (keypad 4) to push the start play marker backward, or use frame forward (keypad 1) to push the end play marker forward.

Another handy feature is to use the combination of Ctrl-shift-arrow (left or right) to select the media from the cursor position (red hairline) to the start or end marker by “tabbing” to the label markers. For example, tab to the beginning of the previous play region using Ctrl-left-arrow to move the cursor to the beginning of last play, then press Ctrl-Shift-right-arrow to tab to the end of the playback region. Now you can clip/play/expand or edit the previous playback selection.

Ctrl SHIFT→ tab cursor to label right of cursor position and expand selection
Ctrl SHIFT← tab cursor to label left of cursor position and expand selection

16. Audio H/W Gain Support

This feature, device level gain, corrects for hardware conditions which some devices may need to be useful. For example, you may need to increase the gain for a weak microphone or a noisy speaker, since it affects rendering when you crank up or down the audio via use of the patchbay. With the audio H/W gain support, you have the ability to fine tune the audio volume by some numerical value for the scale. You are adjusting the scaling of data into the audio driver - H/W scaling is done before it goes into or out of the driver. This is a one time linear multiplication of the sample values, and may offer better control than the logarithmic DB gain controls of the application. These device controls are in Preferences:

Settings → Preferences→ Playback(A/B) → Audio Out → Gain
Settings → Preferences → Record → Audio In → Gain

17. Valgrind Support Level

Valgrind is a memory mis-management detector. It shows you memory leaks, deallocation errors, mismanaged threads, rogue reads/writes, etc. Cinelerra 5.1 memory management is designed to work with Valgrind detection methods. This assists in developing reliable code. Use of Valgrind points out problems so that they can be fixed. For example, when this version of Cinelerra shuts down, it deallocates memory instead of just stopping, thus making memory leak detection possible.

18. Subtitle Track and Subtitle Script Support

A subtitle track is a text only version of track data. It has time position and duration on the timeline, the same as audio or video data. The text for the subtitle is rendered into the track timeline in the edit interval that corresponds to the position where it appears during playback. Its content is text only, and is entered using the interface activated by the menu bar File→Subtitle menu item. The subtitle output data file format is microdvd, and can be used by dvdauthor to create “textsub” text based subtitles. For details on how to utilize this feature, see the section: “Additional Features for Blu-ray/DVD Creation”.

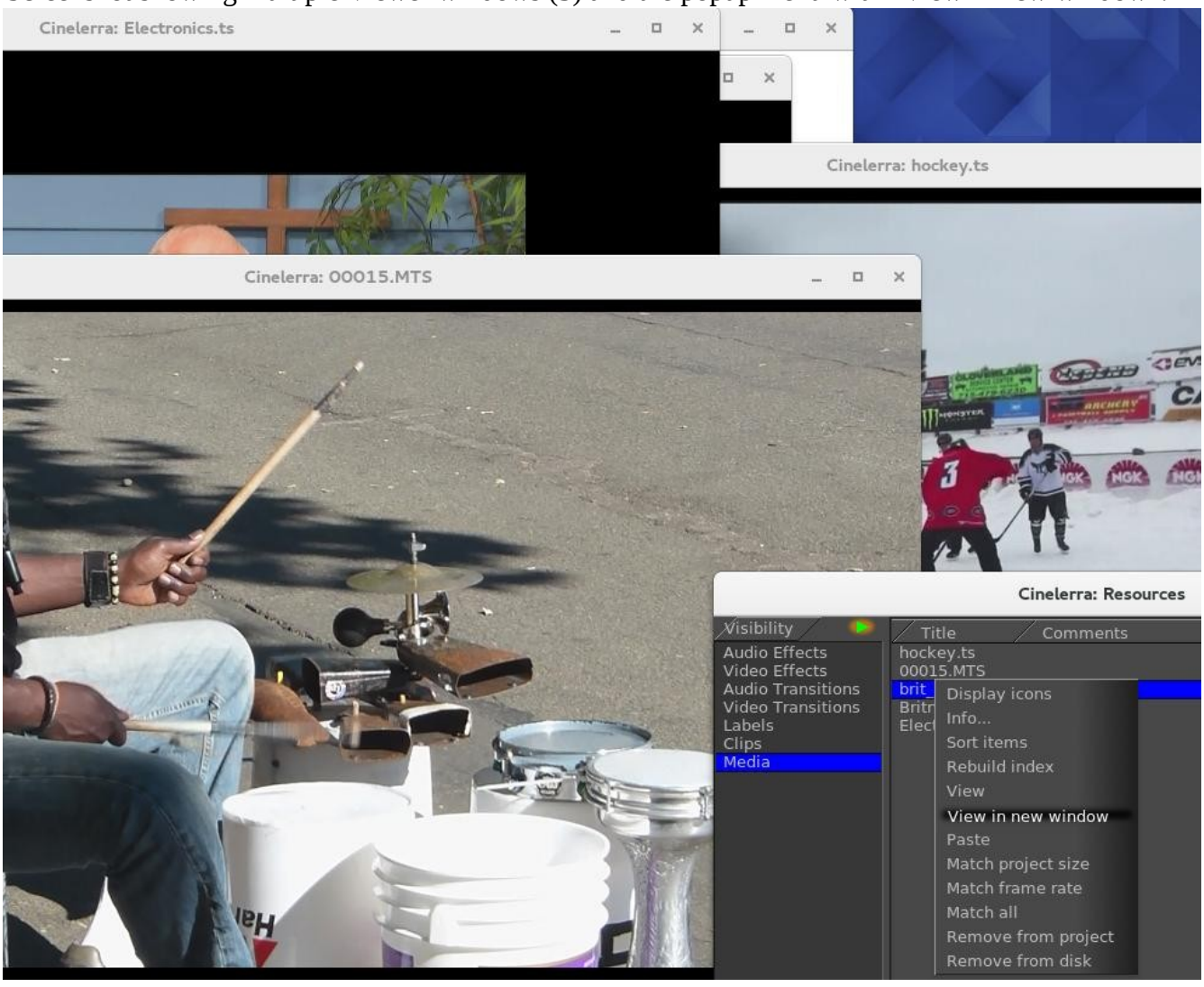
19. CFLAGS has -Wall

When compiling Cinelerra 5.1, a CFLAGS option used is “Wall” where the “W” represents warnings and “all” means all. This causes the compile to check for simple mistakes that can be detected automatically and issue warnings when the code is questionable. It can also detect situations where the compiler will generate incorrect code, like type-punned pointer. By turning on this flag, when new code is vetted for predictable mistakes, the code can be corrected before becoming manifested in the application.

20. Multi-Viewer Window Support

You can create as many Viewer windows as you want in 5.1. These are handy for users who are adept at working with a lot of different clips simultaneously. By bringing up multiple Viewer windows, each clip can be edited in its own area, making it easy to see all of the separate pieces. After you have loaded some media files, to start another Viewer window, right click on one of the pieces of media in the Resources window. This brings up a menu of several options, one of which is “view in new window”. Choose this option and that media will come up in a new Viewer window for you to work.

Screenshot showing multiple Viewer windows (3) and the popup menu with “View in new window”.



21. Overlay Modes – Alpha Blending with PorterDuff and Operators

The purpose of the Overlay Modes is to control the foreground and background stacking and use blending to reshape image object boundaries. It normally makes use of a binary type alpha blending system for all in or all out. To use the available operations in Cinelerra, follow these steps:

1) In the main window, look at the Patchbay on the far left.
2) Click on the small arrow that points to the right, inside the panel to see it expand.
3) A down arrow shows up with a tooltip of “overlay mode”.
4) Use the down arrow to get to the popup menu and choose the desired effect from the 30 possibilities. “Normal” is the default.

Screenshot on the left shows the pulldown in the patchbay and the tool tip “Overlay mode” which provides access to 4 types of overlays. Each will expand further as shown in the screenshots below for “PorterDuffs” and “graphic art”.

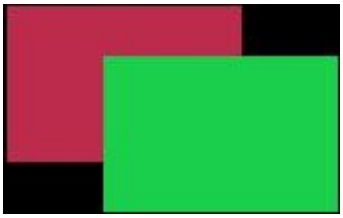
Porter-Duff is the industry standard for alpha blending operations. Only a short explanation follows here, but there is much more information to be found on the internet with complete descriptions and examples. Every pixel has 3 color channels (like RGB), and may have 1 alpha channel value. If there is no alpha defined for a color model, the alpha value is assumed to be 1. Regions of the image are created with the alpha image map. These regions are manipulated using the blending operations described below. Alpha blending is the process of combining a foreground color with a background color which produces a new blended color. The “alpha” channel describes how much opacity is present in a pixel. It may be completely transparent, completely opaque, or any range of translucency.

Conceptually, when the foreground color is completely opaque, the resulting blended color will be the foreground color. If it is transparent, the blended color will be the color of the background. When the value of the alpha channel is 1, the image is all there, if it is 0, there is no image at all, otherwise it is only partially there. In other words, the alpha value goes from 0 to 1, where full transparency is 0 and opaque is represented by 1. Alpha blending models opacity.

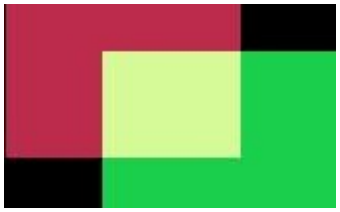
When blending source and destination shapes, the shape boundaries can be changed with the alpha blending effects. There are a total of 10 standard Porter-Duff operators, but there are 30 possible overlay modes used in Cinelerra 5.1. Each is characterized by its value in the four regions: source, destination and both, with the “neither” region always being blank. The source and destination regions can either be blank or filled with the source or destination colors. A specific compositing math formula is used to calculate effect. This is only applicable to RGB; some effort has been made to accommodate YUV, but the effects are not as predictable, and may not be useful.

Below are the results of utilizing the 30 available operations within Cinelerra as listed on a following page. Src is the solid green rectangle and Dst is the solid red rectangle. There are better illustrations of what alpha blending can do, however for consistency sake, these are the results when using standards.

normal

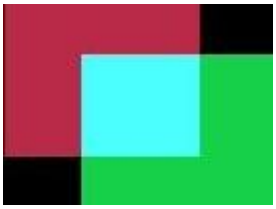


addition



subtract

divide

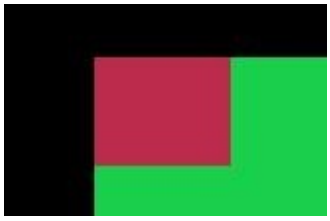


multiply

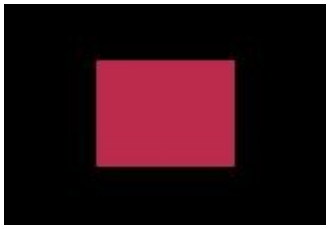
replace

dst

dsatop



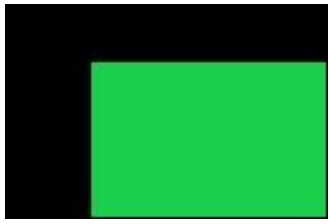
dstin



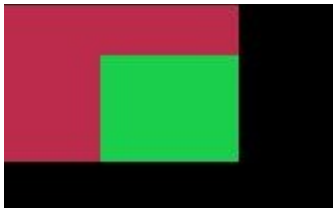
dstout

dstover

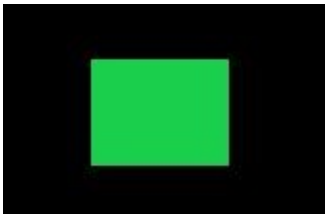
src



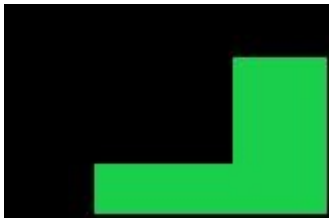
srcatop



srcin



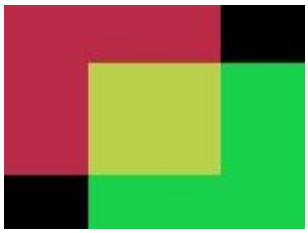
srcout



srcover

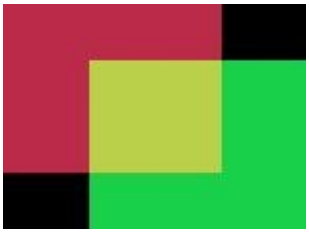
min

max



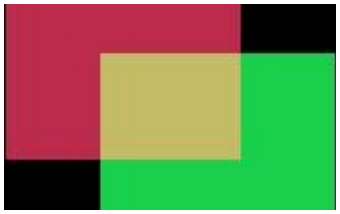
darken

lighten

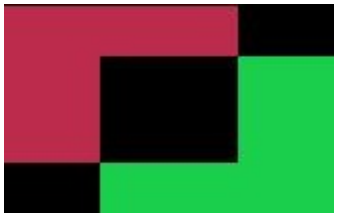


and

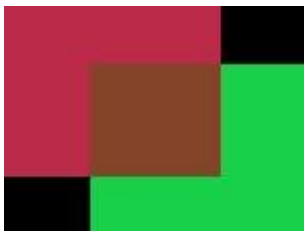
or



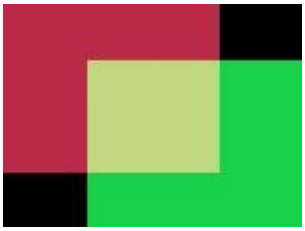
xor



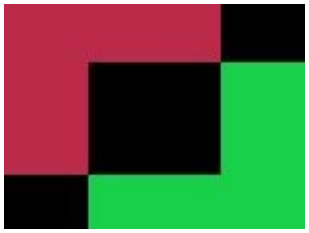
overlay



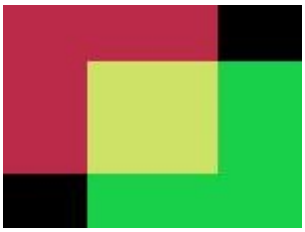
screen



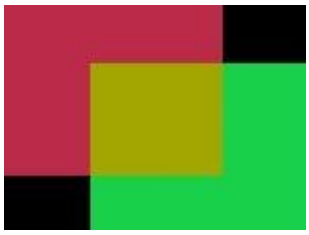
burn



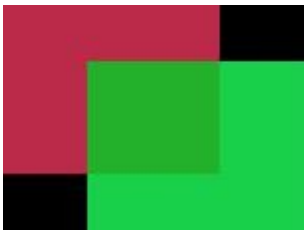
dodge



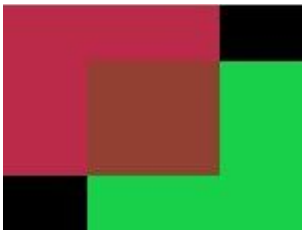
difference



hardlight



softlight



The implementation math forms are listed below, where D=Destination, S=Source, a=alpha, c=chroma (color). Each line describes a pair with the left one for alpha and the right one for chroma.

NORMAL [Sa + Da * (1 - Sa), Sc * Sa + Dc * (1 – Sa)]

Arithmetic

ADDITION [(Sa + Da), (Sc + Dc)]

SUBTRACT [(Sa - Da), (Sc - Dc)]

MULTIPLY [Sa + Da - Sa*Da, Sc*(1 - Da) + Dc*(1 - Sa) + Sc * Dc]

DIVIDE [Sa + Da - Sa*Da, Sc*(1 - Da) + Dc*(1 - Sa) + Sc / Dc]
REPLACE [Sa, Sc] (fade = 1)

PorterDuff

DST [Da, Dc]
DST_ATOP [Sa, Sc * (1 - Da) + Dc * Sa]
DST_IN [Da * Sa, Dc * Sa]
DST_OUT [Da * (1 - Sa), Dc * (1 - Sa)]
DST_OVER [Sa + Da - Sa*Da, Sc * (1 - Da) + Dc]
SRC [Sa, Sc]
SRC_ATOP [Da, Sc * Da + Dc * (1 - Sa)]
SRC_IN [Sa * Da, Sc * Da]
SRC_OUT [Sa * (1 - Da), Sc * (1 - Da)]
SRC_OVER [Sa + Da - Sa*Da, Sc + (1 - Sa) * Dc]

Logical

MIN [min(Sa, Da), MIN(Sc, Dc)]
MAX [max(Sa, Da), MAX(Sc, Dc)]
LIGHTEN [Sa + Da - Sa*Da, Sc*(1 - Da) + Dc*(1 - Sa) + max(Sc*Da, Dc*Sa)]
DARKEN [Sa + Da - Sa*Da, Sc*(1 - Da) + Dc*(1 - Sa) + min(Sc*Da, Dc*Sa)]
AND [Sa * Da, Sc * Dc]
OR [Sa + Da - Sa * Da, Sc + Dc - Sc * Dc]
XOR [Sa + Da - 2 * Sa * Da, Sc * (1 - Da) + Dc * (1 - Sa)]

Graphic Art

OVERLAY [Sa + Da - Sa * Da, Sc*(1 - Da) + Dc*(1 - Sa) + 2*Dc < Da ? 2*Sc*Dc : Sa*Da - 2*(Da-Dc)*(Sa-Sc)]
SCREEN [Sa + Da - Sa * Da, Sc + Dc - (Sc * Dc)] (same as OR)
BURN [Sa + Da - Sa * Da, Sc*(1 - Da) + Dc*(1 - Sa) + Sc <= 0 || Sc*Da + Dc*Sa <= Sa*Da ? 0 : (Sc*Da + Dc*Sa – Sa*Da)*Sa/Sc]
DODGE [Sa + Da - Sa * Da, Sc*(1 - Da) + Dc*(1 - Sa) + Sa <= Sc || Sc*Da + Dc*Sa >= Sa*Da) ? Sa*Da : Dc*Sa / (1 – Sc/Sa)]
DIFFERENCE [Sa + Da - Sa * Da, Sc*(1 - Da) + Dc*(1 - Sa) + abs(Sc * Da - Dc * Sa)]
HARDLIGHT [Sa + Da - Sa * Da, Sc*(1 - Da) + Dc*(1 - Sa) + 2*Sc < Sa ? 2*Sc*Dc : Sa*Da - 2*(Da - Dc)*(Sa – Sc)]
SOFTLIGHT [Sa + Da - Sa * Da, Sc*(1 - Da) + Dc*(1 - Sa) + Da > 0 ? (Dc*Sa + 2*Sc*(Da - Dc))/Da : 0]

The previous math forms are the only truly accurate description of each blending operation, but short descriptions are below where “Source” is the output from the next track and “Destination” is the output from the lower track stacking. Blending starts with new Source and combines it with the previous render stack output, which is referred to as Destination. The new output becomes the next Destination and the next up stack level becomes the new Source. Source is above; Destination is below.

Normal – Normal mode is the default layer mode. The result color is the source color. The layer on top covers the layers below it. If you want to see anything below the top layer when you use this mode, the layer must have some transparent areas. It is “stacked on top”. Math formula used is different than that used by Gimp; there is no SVG equivalent.

<u>Arithmetic Group</u> – standard numerical operations:

Addition – The source is added to the destination and replaces the destination. Addition mode is very simple - the pixel values of the upper and lower layers are added to each other. The resulting image is

normally brighter. The equation can result in color values greater than 255, so some of the light colors may be clipped to the maximum value of 255. Math formula is the same as that used by SVG but different than Gimp.

Subtract – Subtract mode reduces the pixel values of the upper layer by the pixel values of the lower layer. The resulting image is normally darker. You might get a lot of black or near-black in the resulting image. The equation can result in negative color values, so some of the dark colors may be clipped to the minimum value of 0. Math formula used originated from HV cinelerra and is different than that used by Gimp; there is no SVG equivalent.

Multiply – The source color is multiplied by the destination color and replaces the destination. The resulting color is always at least as dark as either the source or destination color. Multiplying any color with black results in black. Multiplying any color with white preserves the original color. Math formula is the same as used by SVG and Gimp.

Divide – Divides source color by destination color. If the source color is white, the result color is the underlying color. The resulting image is often lighter. Math formula used is different than that used by HV/CV and Gimp; there is no SGV equivalent.

Replace – Replace mode will cause any existing destination to be replaced by the source media. Mathematical formula used is the same as used by Gimp; there is no SVG equivalent.

PorterDuff Group – industry standard compositing operators:
DST - The destination is left untouched. Only the destination will be present in the output. Math formula is the same as that used by Porter-Duff and SVG; there is no Gimp equivalent.

DST_ATOP - The part of the destination lying inside of the source is composited over the source. The destination outside the source is dropped. You will see the existing canvas is only kept where the shapes overlap. Math formula is the same as that used by Porter-Duff and SVG; there is no Gimp equivalent.

DST_IN - The part of the destination lying inside of the source is displayed. This effectively allows for cutting out the background using the current layer/feature as a mask. You will see the new shape is drawn only where both the source and the destination overlap. Math formula is the same as that used by Porter-Duff and SVG; there is no Gimp equivalent.

DST_OUT - The part of the destination lying outside of the source is displayed. This is basically a reverse mask, compared to destination-in. You will see the existing content is kept where it doesn't overlap the source. Math formula is the same as that used by Porter-Duff and SVG; there is no Gimp equivalent.

DST_OVER - The destination is composited over the source. Math formula is the same as that used by Porter-Duff and SVG; there is no Gimp equivalent.

SRC - The source is copied. The destination is not used as input. Only the source will be present in the output. Math formula is the same as that used by Porter-Duff and SVG; there is no Gimp equivalent.

SRC_ATOP - The part of the source lying inside of the destination is composited over the destination.

The source outside the destination is dropped. You will see the existing canvas is only kept where the shapes overlap. Math formula is the same as that used by Porter-Duff and SVG; there is no Gimp equivalent.

SRC_IN - The part of the source lying inside of the destination is displayed. This effectively allows the destination to act as a mask for the layer/feature being shown. You will see the new shape is drawn only where both the source and the destination canvas overlap. Math formula is the same as that used by Porter-Duff and SVG; there is no Gimp equivalent.

SRC_OUT - The part of the source lying outside of the destination is displayed. This is basically a reverse mask, compared to source-in. You will see the new shape is drawn where it doesn't overlap the destination. Math formula is the same as that used by Porter-Duff and SVG; there is no Gimp equivalent.

SRC_OVER - The source is composited over the destination. Math formula is the same as that used by Porter-Duff and SVG; there is no Gimp equivalent.

Logical Group – Venn diagram operations of area combinations:
Min – The output color is the component-wise minimum value of the source and destination colors. There is no SVG or Gimp equivalent math formula.

Max – The output color is the component-wise maximum value of the source and destination colors. There is no SVG or Gimp equivalent math formula.

Lighten – Selects the lighter of the destination and source colors. The destination is replaced with the source when the source is lighter, otherwise it is left unchanged. Completely black layers have no effect on the final image and completely white layers result in a white image. Math formula is the same as used by SVG and Gimp's “lighten only”.

Darken – Selects the darker of the destination and source colors. The destination is replaced with the source when the source is darker, otherwise it is left unchanged. Completely white layers have no effect on the final image and completely black layers result in a black image. Math formula is the same as used by SVG and Gimp's “darken only”.

And – This operation intersects source and destination. If either image is not white, it down factors the other image. Usually you will end up with a lot of darker areas. There is no SVG or Gimp equivalent math formula.

Or – This operation is the union of the source and the destination. If either image is not black, it adds to the other image. Usually you will end up with a lot of lighter areas. Math formula is the same as that used by “Screen” operation in SVG; there is no Gimp equivalent.

Xor – The union of the source and destination with the intersection removed. Results in black if they are equal. Shapes are made transparent where both overlap, and drawn normal everywhere else. Math formula used is that used by Porter-Duff; there is no Gimp equivalent and SVG formula differs.

Graphic Art – typical operations from popular "paint" packages:
Overlay – Multiplies or screens the colors, dependent on the destination color. Source color overlaythe

destination while preserving its highlights and shadows. The destination color is not replaced, but is mixed with the source color to reflect the lightness or darkness of the destination. Dark parts on the base layer become darker, and light parts become lighter. Math formula is the same as used by SVG and Gimp.

Screen – The source and destination are complemented and then multiplied and then replace the destination. The resultant color is always at least as light as either of the two constituent colors. Screening any color with white produces white. Screening any color with black does not change the other layer. Using the Screen mode is similar to projecting multiple photographic slides simultaneously onto a single screen. A lighter picture is the result. Math formula is the same as used by SVG and Gimp. This is identical to "Or".

Burn - Darkens the destination color to reflect the source color. Burning with white produces no change. In photography, burning is a technique used in a darkroom to increase the exposure in particular areas of the image. This brings out details in the highlights. Math formula is the same as used by SVG and Gimp.

Dodge - Brightens the destination color to reflect the source color. Dodging with black produces no change. The result color is a lightening of the source color to reflect the underlying layer color by decreasing the contrast. If the source color is pure black, the result color is the underlying color. In photography, dodging is a technique used in a darkroom to decrease the exposure in particular areas of the image. This brings out details in the shadows. Math formula is the same as used by SVG and Gimp.

Difference - Subtracts the darker of the two constituent colors from the lighter. This is the same as "Subtract" except that the absolute value is used. "Subtract" colors which would be black due to clipping become visible instead. Differencing with white inverts the destination color. Differencing with black produces no change. Math formula is the same as used by SVG and Gimp.

Hardlight - Multiplies or screens the colors, dependent on the source color value. If the source color is lighter than 0.5, the destination is lightened as if it were screened. If the source color is darker than 0.5, the destination is darkened, as if it were multiplied. The degree of lightening or darkening is proportional to the difference between the source color and 0.5. If it is equal to 0.5 the destination is unchanged. Using pure black or white produces black or white. The effect is similar to shining a harsh spotlight on the destination. This mode is useful for creating the appearance of shadows on a layer. You might use this mode to combine two photographs and obtain bright colors and sharp edges. Math formula is the same as used by SVG and Gimp.

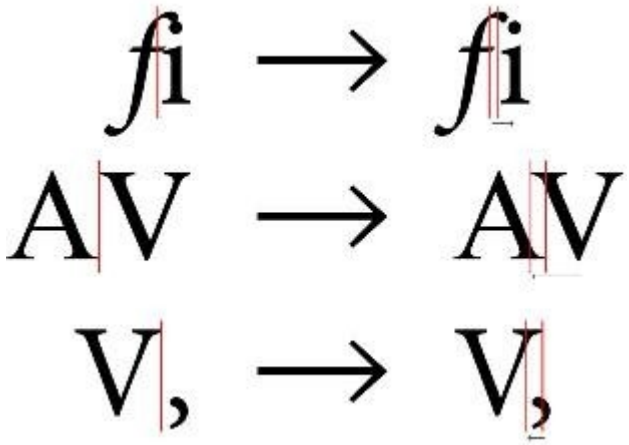
Softlight - Darkens or lightens the colors, dependent on the source color value. If the source color is lighter than 0.5, the destination is lightened. If the source color is darker than 0.5, the destination is darkened, as if it were burned in. The degree of darkening or lightening is proportional to the difference between the source color and 0.5. If it is equal to 0.5, the destination is unchanged. Using pure black or white produces a distinctly darker or lighter area, but does not result in pure black or white. The effect is similar to shining a diffused spotlight on the destination. A layer with pure black or white becomes markedly darker or lighter, but does not become pure black or white. Soft light is not related to "Hard light" in anything but the name, but it does tend to make the edges softer and the colors not so bright. Math formula is the same as used by Gimp; SVG formula differs.

Note: the Graphic Art group operates principally on color, and the others operate principally on alpha.

22. Titler Font Kerning

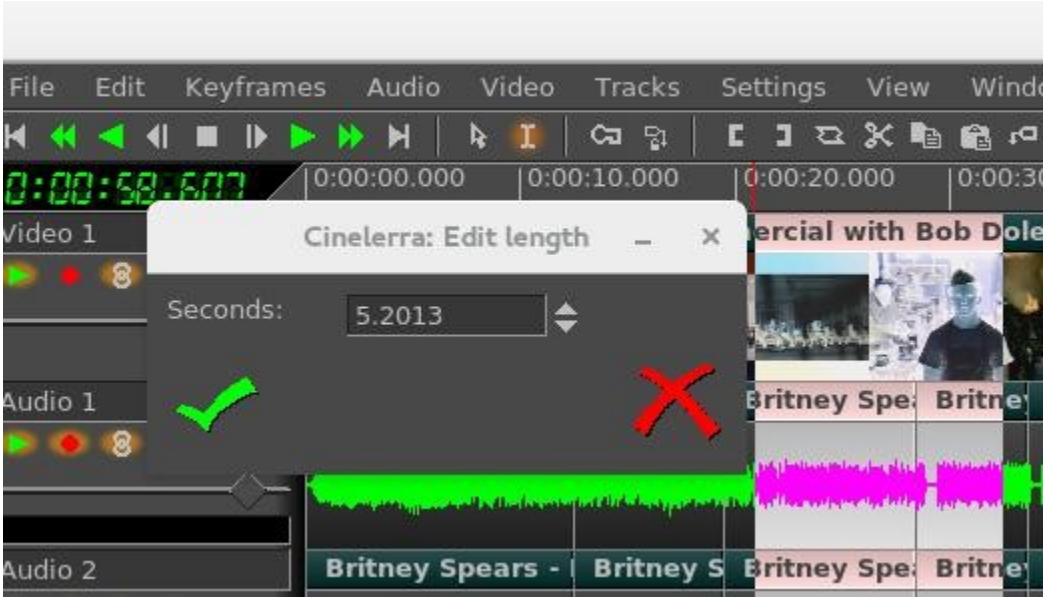
When using the Titler, kerning is applied in order to allow parts of a letter to go outside the standard sized letter box. Kerning is the process of adjusting the space between individual letters. It is not the same as proportional spacing. In kerning, a bounding box is allowed to overlay another bounding box. The philosophy here is to aim at the ability to have boxes overlap in order to make the letters look more visually appealing. Bounding box and escapement are tracked separately. In addition to adjusting individual letter spacing, the program will also expand the render box in order to have any parts of the letter extend outside the standard letter box. Kerning is applied to any and all fonts.

Screenshot shows how letters can be presented without and with kerning. Note this is just an example.



23. Edit Length

To set the length of an edit in the timeline, select the region which contains the edit to be modified. Now select the menu bar Edit→Edit Length… menu item to activate the “edit length” popup. The duration of the edit can be reset by entering the desired edit length in seconds. Pressing OK will change all of the selected edits (in armed tracks) to the specified length.



24. Align Edits, Reverse Edits, and Shuffle Edits

The Align, Reverse, and Shuffle Edits functions are described below in each section with illustrative screenshots on the following pages.

24.1 Align Edits

When loading media, a common problem is that the various audio/video tracks do not always have exactly the same lengths. For example, you might load audio/video recordings from your camera and be dismayed to see that the audio for each segment is a half second longer than the video. If you load a large set of media clips by concatenation, the audio and video will be more skewed as more media is loaded. This feature makes it possible to adjust the edits so the audio and/or video align by adjusting the edits so that the track lengths are consistent. To use this feature, load all of the desired media and select a region which contains all of the edits to be aligned in the timeline. Now select the menu bar Edit→Align Edits menu item to operate the change. The topmost armed track is used as a template reference, and the rest of the tracks are either cut or padded to align the edit boundaries. Besides aligning audio with the video, you can also align video with the audio if the first armed track is audio.

The code performs the following algorithm:

Use the first armed track as the master track (it must contain data).
Collect the “edit project start times” on the selected master track.
Only edits that are 100% inside the selected area will be used.
Set all other tracks to match the “edit times” of the template track, either by putting in silence or cutting the region to align the edits on the “edit times” of the master track.

The start time sequence of media and silence edits along the master track are collected as the target alignment boundaries. All armed tracks after the master track are modified so that if the next edit edge is too soon, it adds silence; if it is too late, edits are shortened or deleted past the point of the next target alignment boundary time. Align Edits works best if there are an equal number of Video and Audio sections. Also, it is better to use cuts instead of adding silence – if there are silence edits together, the algorithm will combine the silence edits into a single edit and results may not be as desired.

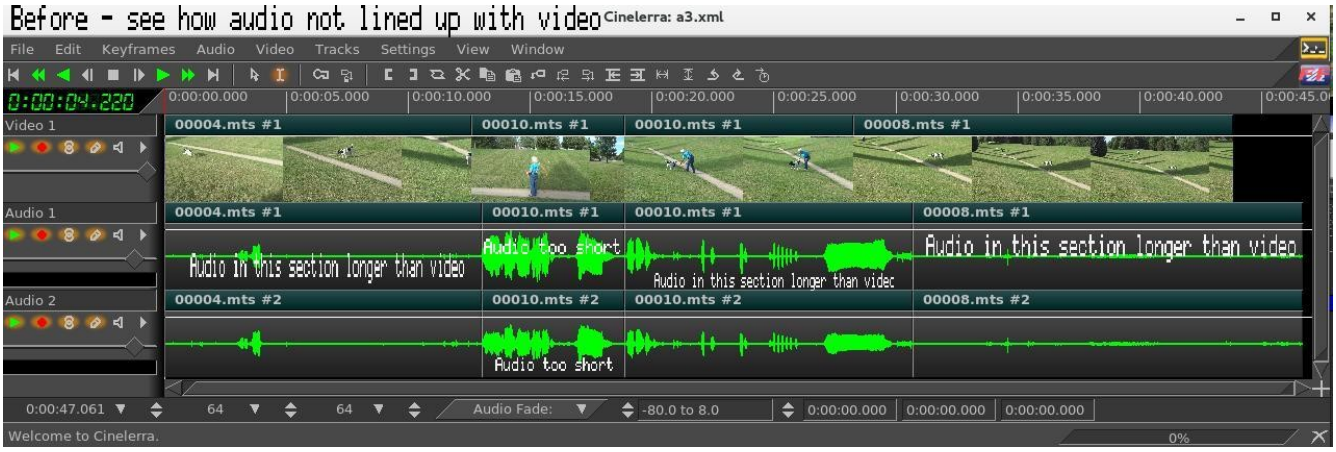
24.2 Reverse Edits

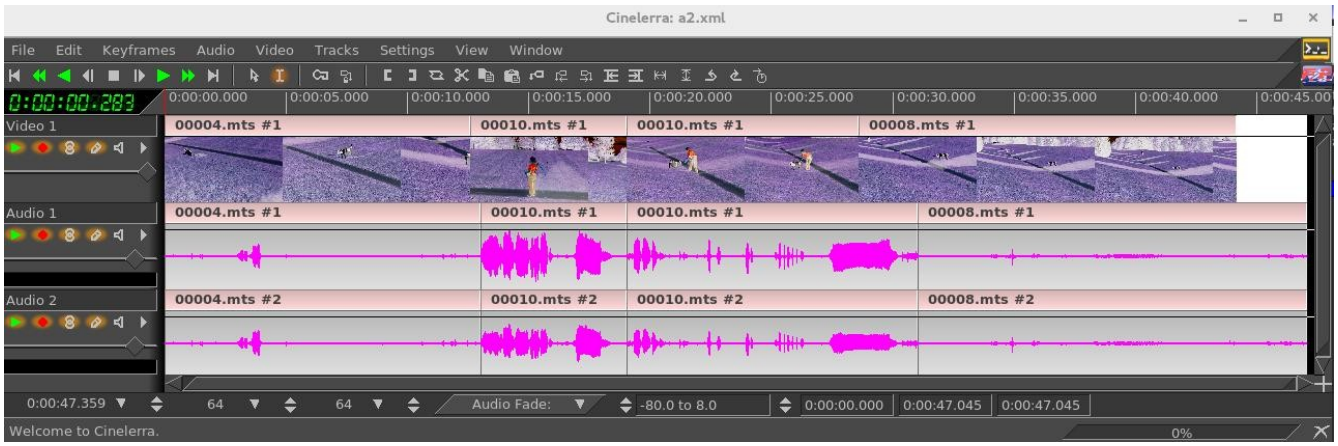
The Reverse Edits can be useful to change the order of 2 edits in the case where you would like to put a “teaser” section that occurred in the middle of a movie at the beginning instead, that is, reversed positions. To operate, highlight completely the edit areas you would like reversed and then use the pulldown Edit→Reverse Edits.

24.3 Shuffle Edits

The file pulldown Edit→Shuffle Edits will randomly exchange the location of the edits. This feature can be used to change the order of the music like you would do from your MP4 player where you have a playlist of your favorite music. Or perhaps you are creating an advertisement background, you can randomly change it, thus the viewer sees a different order of scenes each time shown.

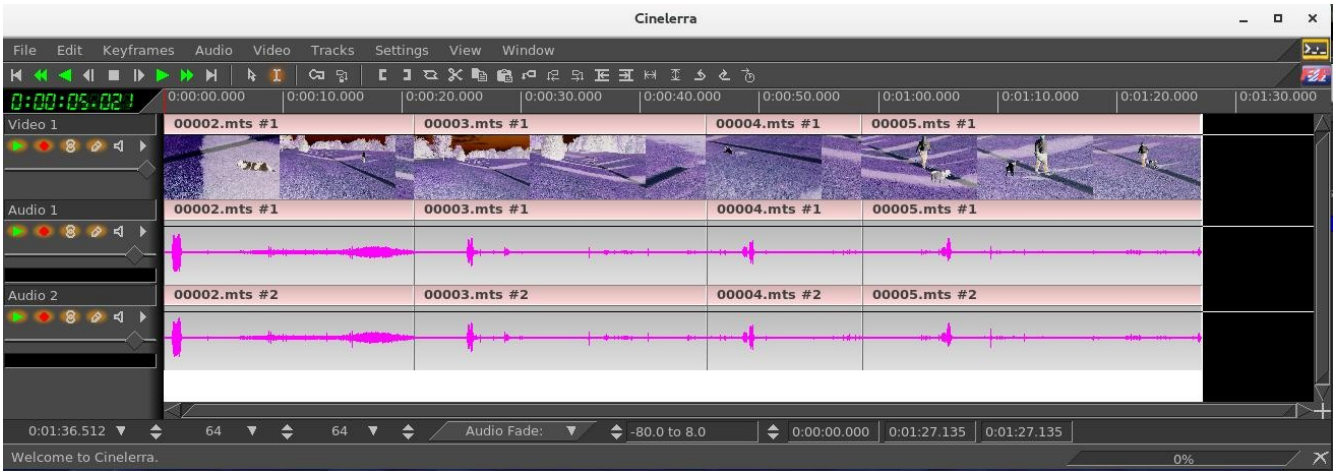
Screenshots below show the Before, the Highlighted Edits to be manipulated, and the After results fo the Align Edits.



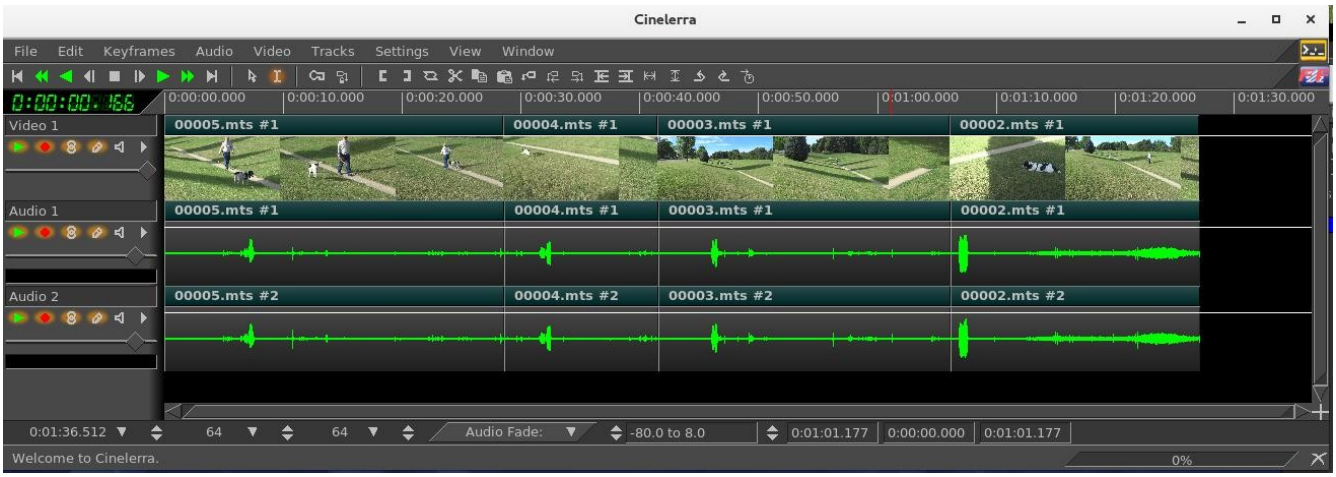


This After screenshot “adds silence” in the second section as noted in red letters.

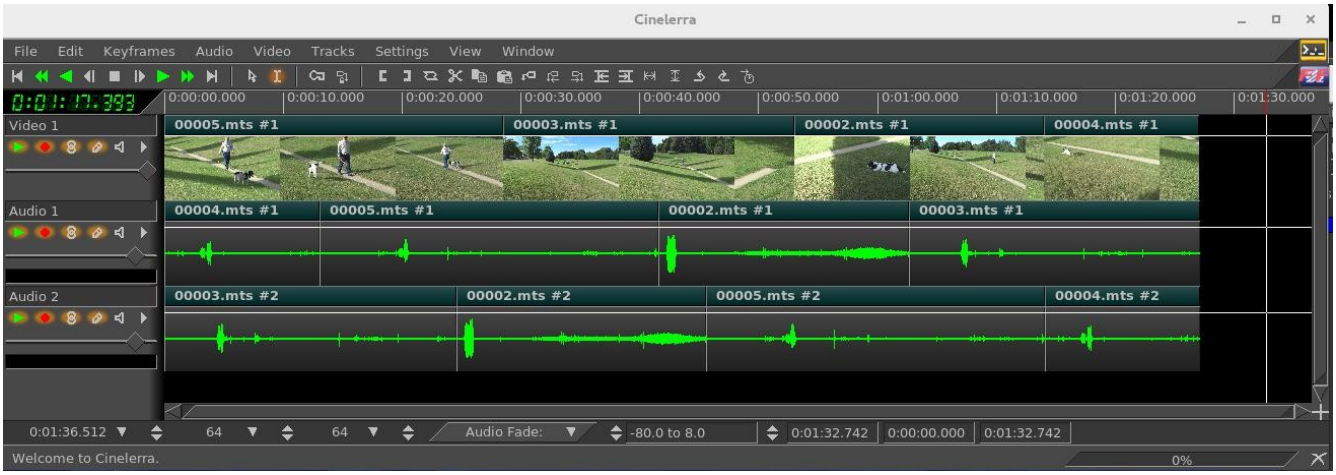
Screenshot immediately below shows the selected/ highlighted area to which Edits will be applied. Note the first edit is 00002, followed by 00003, 00004, and 00005 in that order.



Screenshot illustrating the results of executing “Reverse Edits”. Now you will see the reversed order of 00005, 00004, 00003, and last 00002.

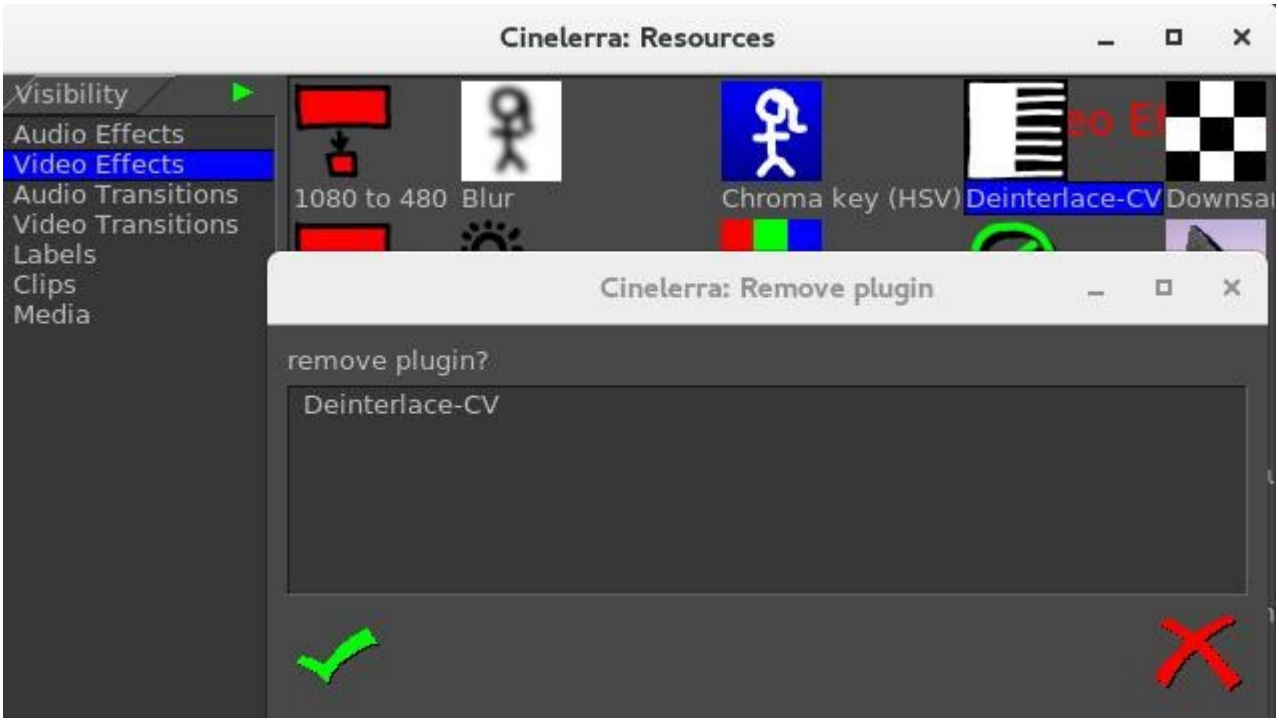


Screenshot illustrating Shuffle Edits of the highlighted area of the first screenshot on the page. Note the permutation of the fragments resulting in 00003 now being first, then 00005, 00002, and 00004 last.



25. Delete Plugins to save Resources Window Space or whatever

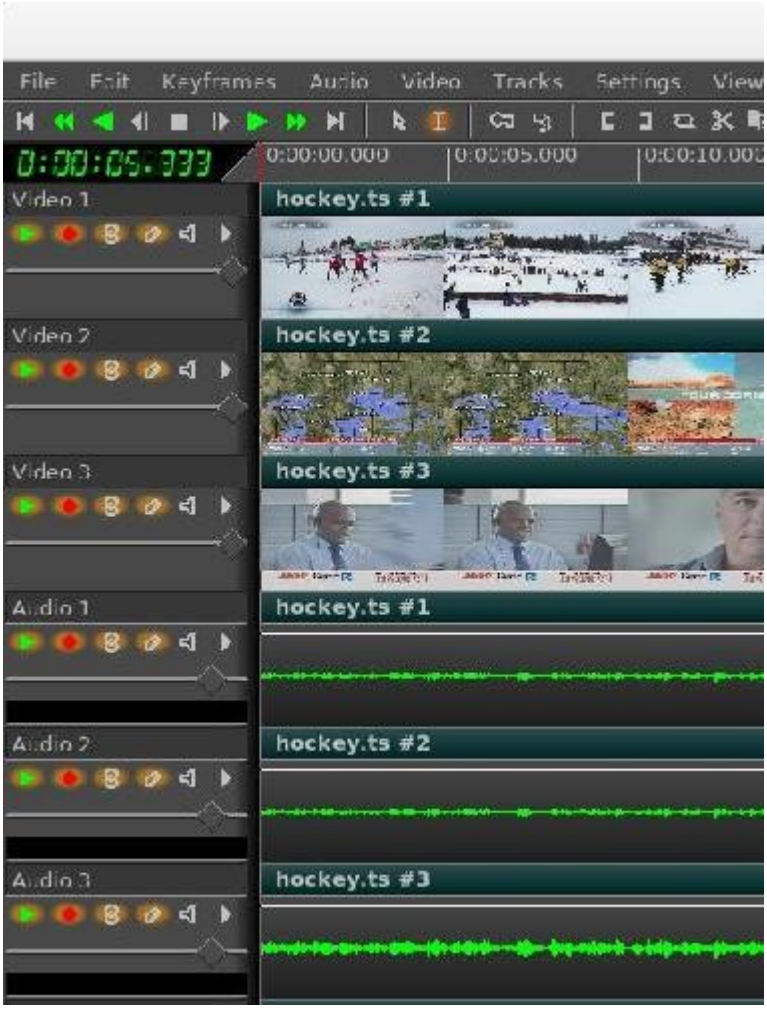
Maybe you just don't ever use certain plugins or would prefer to only find the ones that are useful to you. To save space in the Resources Window so you don't have to scroll to find the plugins you want, a feature to delete others is available. The plugins will be PERMANENTLY deleted, but only until you rebuild or download a new set of Cinelerra binaries. To do this, highlight the plugin you no longer want in the Resources window then keypress Shift-delete. A small window will come up allowing you to change your mind and red-X out or chk-OK to “remove plugin”. This feature may come in handy if you have personnel working on media for you and you only want them to exercise certain functions. Or maybe you can't remember which is the “good” deinterlace plugin out of the available 5 or so and want to delete the extras so as not to be confused. If you have a System install, you will have to be root for this function to be enabled.

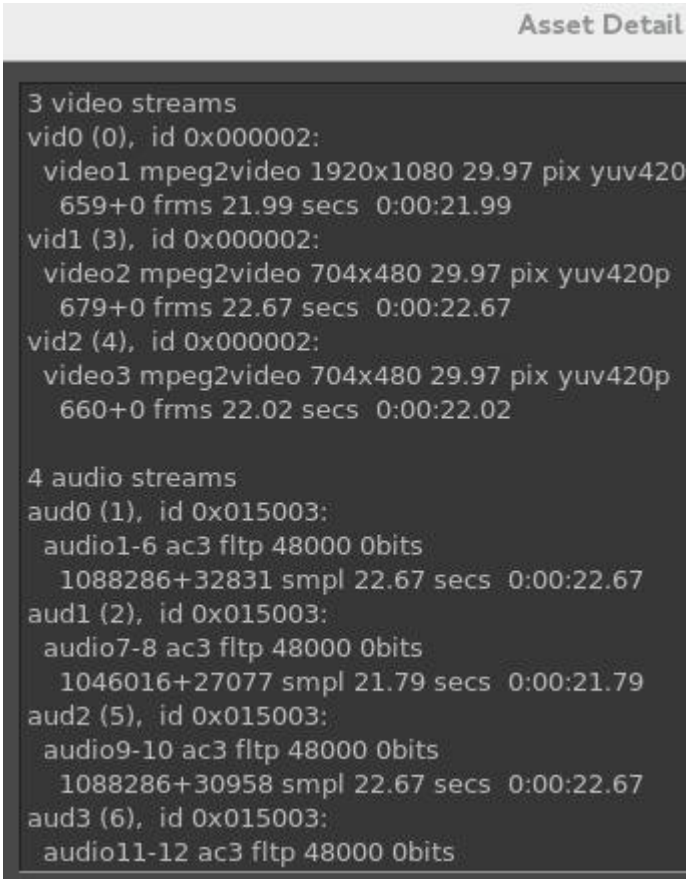


26. Program Selection Support

Some kinds of media have “program” streams, like captured mpeg broadcast stream data. For example, you may be able to “tune” to channel 9, but be able to see 9-1, 9-2, and 9-3 on your TV. If you open a capture of this kind of media, all of the channels are present in the timeline. To select and view just one program, you can use Alt-1 to select program 1, or Alt-2 to select program 2, etc. up to Alt-8. This will remove all of the other unrelated tracks and reset the format. This feature can be used even if there is only one program, by pressing Alt-1, and the effect will be to reset the session format to the parameters from the media probe. Note that there may be several audio “programs” associated to a video stream; for example, there may be dialog in another language or some kind of descriptive dialog. Since the first associated audio is always selected, this may not produce the intended results.

Below are screenshots illustrating multiple program streams. The left screenshot is a partial main Cinelerra window showing a pre-recorded broadcast TV media/audio stream with 3 programs plus several associated audio tracks. The second screenshot of “Asset Detail” provides detailed information on each of the streams obtained through executing the Info Details as explained in the section “Info Asset Details”.

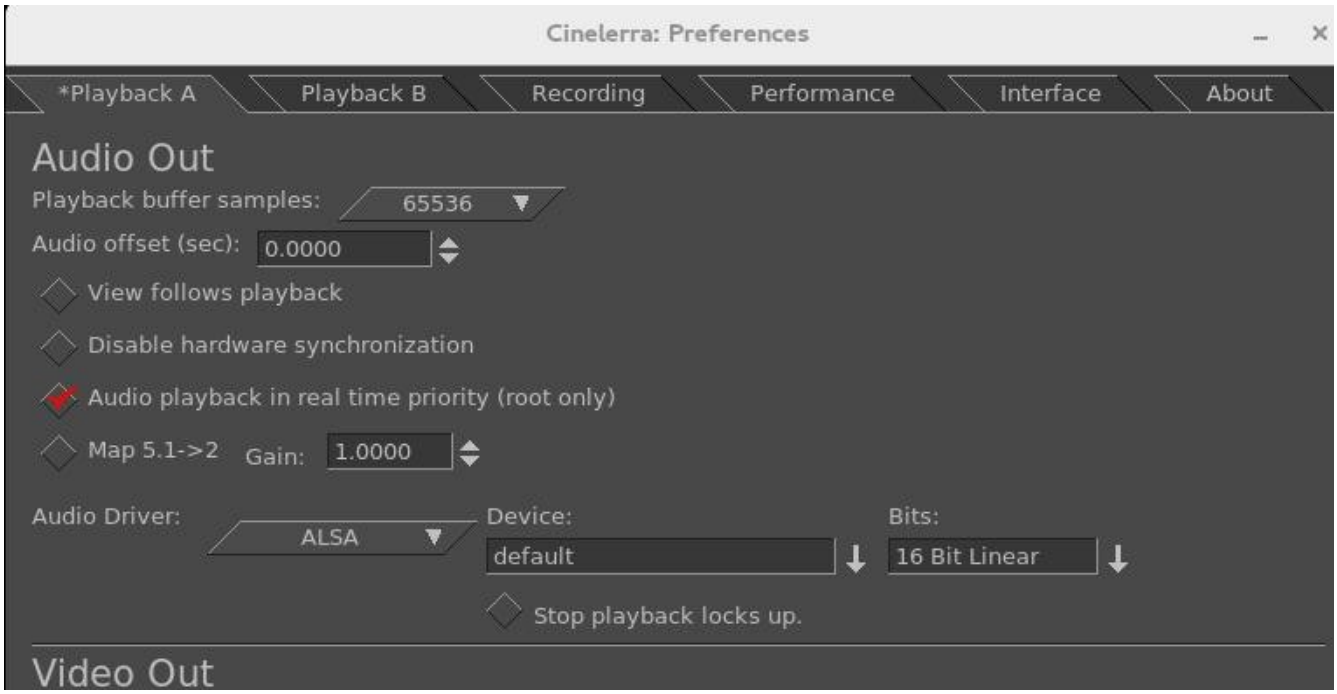




27. Realtime Priority Audio

Only experts might want to use this because it interferes with ordinary time-share scheduling and can lock up the system. When this is enabled, audio gets the first shot and burns audio until audio lets go. To explain, there are 2 kinds of scheduling, “time-sharing” which is the default, and “real time” where the scheduled task must explicitly request scheduling to allow other tasks to execute. Time-share interrupts when you use up your allocated time slice. Realtime priority audio will execute audio decode until it finishes, which may may slow down other types of processing like video decoding. Most decoders use a policy that video may be downsampled to accommodate scheduling, but will never skip audio because it creates a much more obvious defect. This feature helps to make sure audio gets priority over video during decode. To enable this feature, use Settings → Preferences → Playback A/B, and check “Audio playback in real time priority (root only)”. Be sure to check apply in order for this feature to take effect and as stated, you must be root.

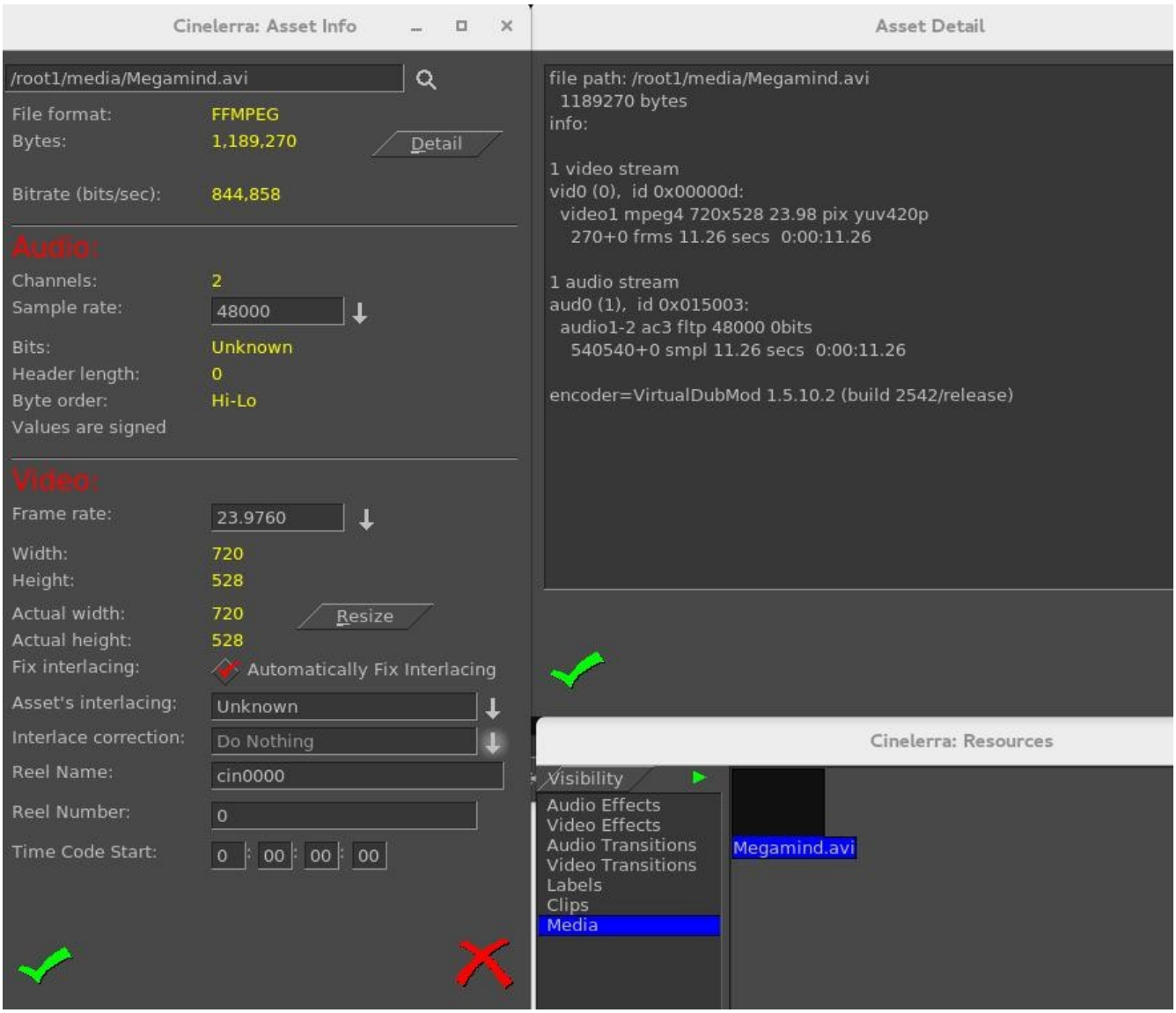
Screenshot showing Playback A will be using real time priority.



28. Info Asset Details

Detailed information is now readily available on any loaded media of type mpeg or ffmpeg. This is extremely helpful in determining what type of media it is, size, resolution, format, and type/number of audio streams. It is especially useful for multiple program streams.

Screenshot below shows the “Detail” box to click on the left side and a simple, typical output in the Asset Detail window on the right side. Also, note the highlighted media in the Resources window.

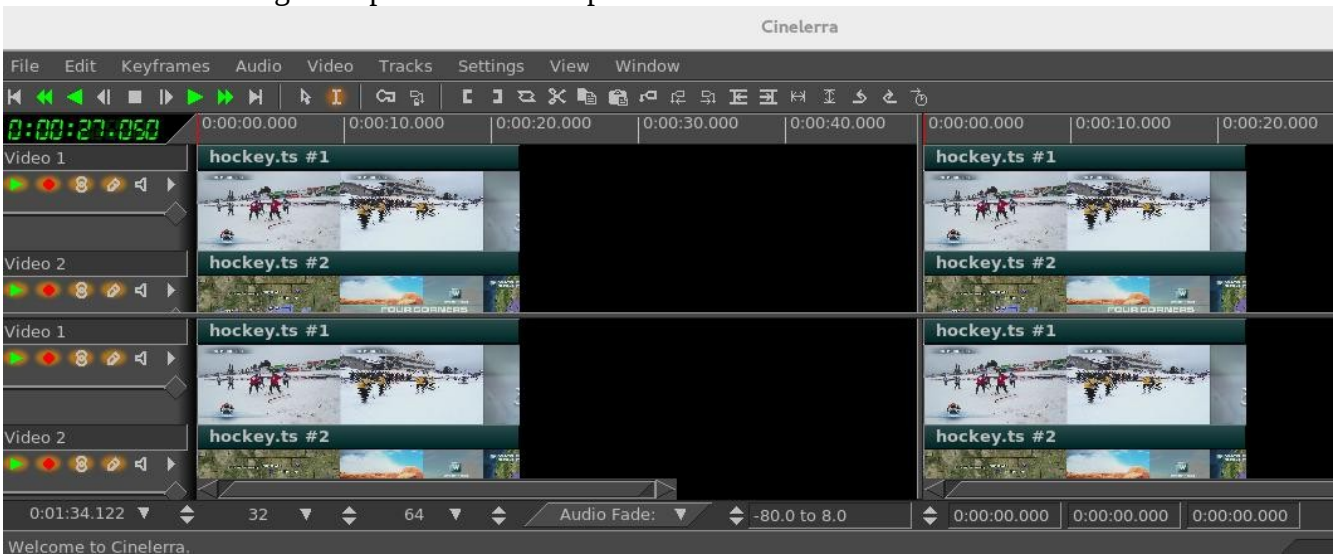


29. Multi-Pane Support

The main Cinelerra edit window holds the Track Canvas which can be divided into 4 panes of track data: 1 or 2 vertical panes and/or 1 or 2 horizontal panes. To split the track, use the Window pulldown, and then click on Split X or Split Y depending on how you wish to split the track. Alternatively, the canvas pane types can be changed using keys <Ctrl-1> for toggle split horizontal or <Ctrl-2> for toggle split vertical. Or the track can be split into panes by using the + widget in the lower right hand corner of the track canvas. Once the track has been divided, you can use the + widget shortcut or the drag bars to change the size of the panes.

Multi-Pane, or split screen, allows you to look at the first part of a movie at the same time as a part that is a long ways away on the timeline which would have been off the screen. By having multiple panes, you can see the 2 parts you want to look at simultaneously and drag/drop easily between the 2. Also, the X pane split is extremely convenient for laptop users and computer monitors with small screens since it can be used with horizontal scrolling with the mouse wheel + Ctrl-key. The Y-pane split makes it easy to see 2 simultaneous drag and drop zones when you have lots of tracks.

Screenshot illustrating the 4 panes that have split the main Track Canvas.



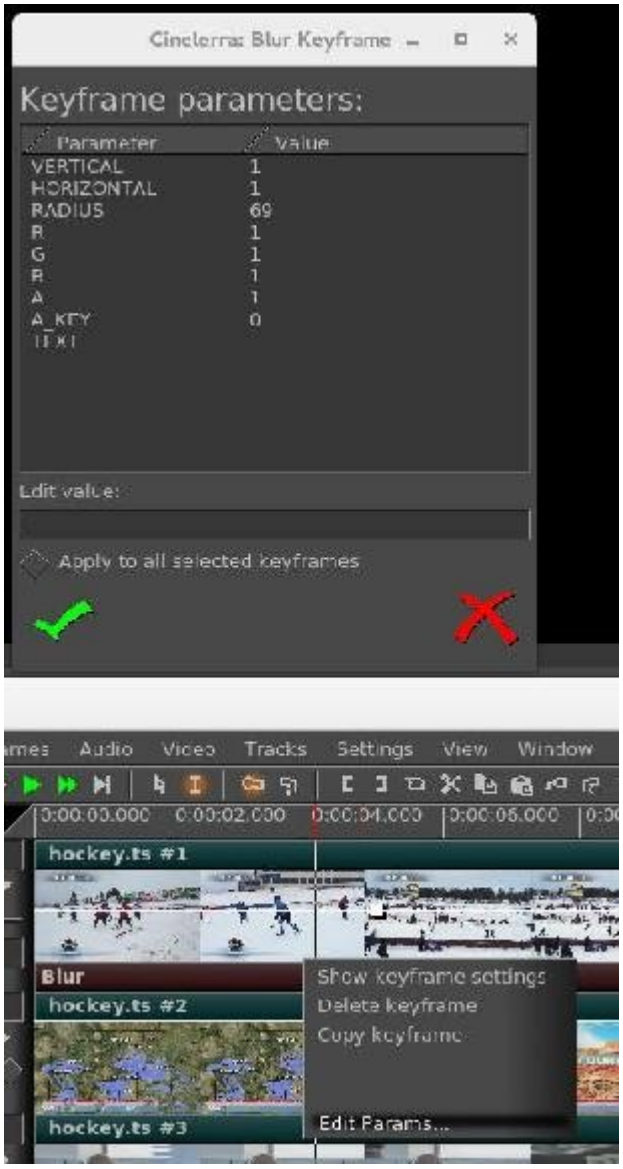
30. Nested Assets

Assets can generally be one of two types: files of media data like codec output or EDL session files. An EDL session could be rendered into a media data file, and then used as a file asset, or equivalently you can open the session as a nested asset. It will be rendered as needed and the rendered output will be used as if it was being read in from a file. The advantage of this capability is that the nested EDL file can be reworked and requires no storage for the intermediate rendering. The disadvantage is that it is re-rendered on every use, which can be time consuming. Nested assets allow for complex stacking of effects, and makes media access much more flexible. This feature can be used recursively, that is, any number of sessions may be stacked and referenced as an asset, as long as all of the rendering resources are available. Nested assets are added to the timeline by using the pulldown File→Load files… on the main menu and selecting the “Insertion strategy” of “Nest asset”. The file will be pasted into the timeline over the current selection or at the insertion point.

31. Keyframe Edit Params

Keyframes are blobs of parameters which affect the media presentation. They are usually associated to a point on the timeline and its parameters may vary using different types of time dependent curves along the timeline. Some kinds of keyframes work with plugins. They are set using the various plugin “show controls” (magnifying glass) icon on the plugin track. It is possible to see all of the keyframe data in a raw format using the Edit Params… popup menu item which you will see when you right click the keyframe icon on the timeline. The keyframe data is stored in “xml” format, and the edit params feature allows you to view and modify the xml data directly. This normally should not be necessary since the plugin's control gui displays the intended parameters, but in a pinch this can let you view and specify just about anything that can be specified in xml. There is no validation checking of the modified data. Use this only if you are confident you know what it does, and be sure to backup the session before you make changes.

Screenshot on the right shows the “Edit Params” highlighted in bottom right corner. Once clicked, the “Keyframe Parameters” window comes up (see top).



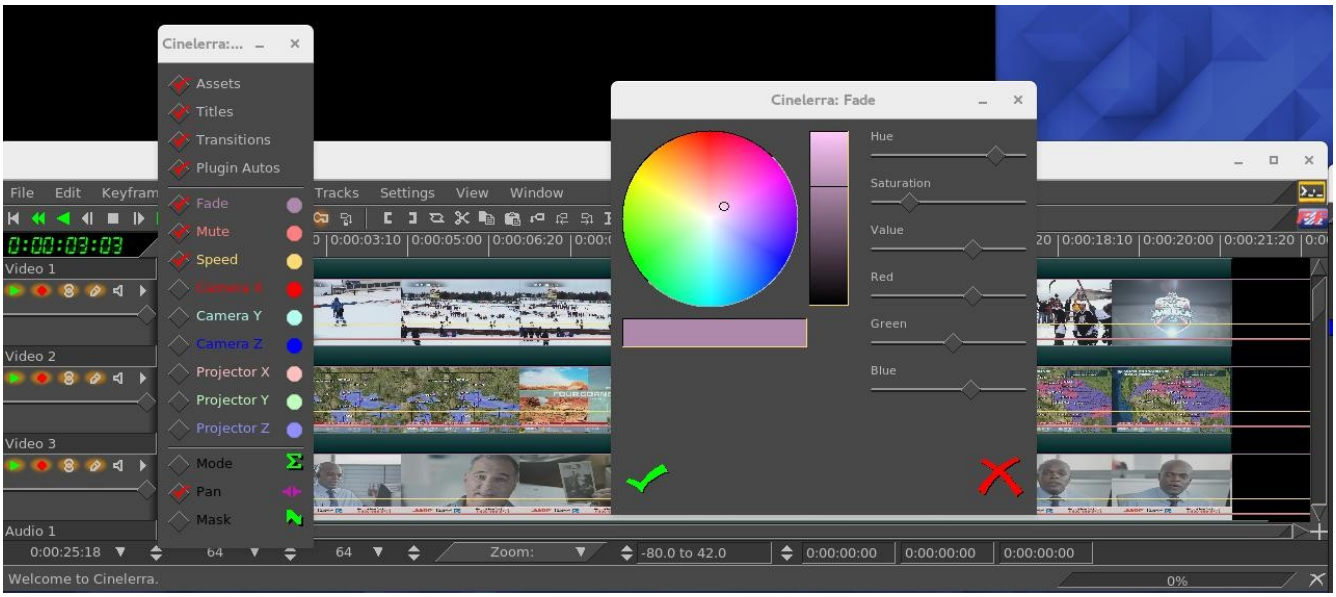
32. Color Coded Keyframe Curves/Different Operational Characteristics

The 5.1 version of Cinelerra has some additional features that help to remove confusion from multiple curves on the track canvas. They are most easily viewed from the pulldown menu of Window→Show overlays but they will operate the same as when used from the View pulldown menu. The most notable differences here are the “Color Coded Keyframe Curves” which have distinct colors associated with each type for less confusion. By clicking button 1 on the “Color Ball” to the right of any keyframe type in the “Show overlays” menu you have the ability to change the colors to whatever works best for your video. The color ball changes made will be retained across sessions.

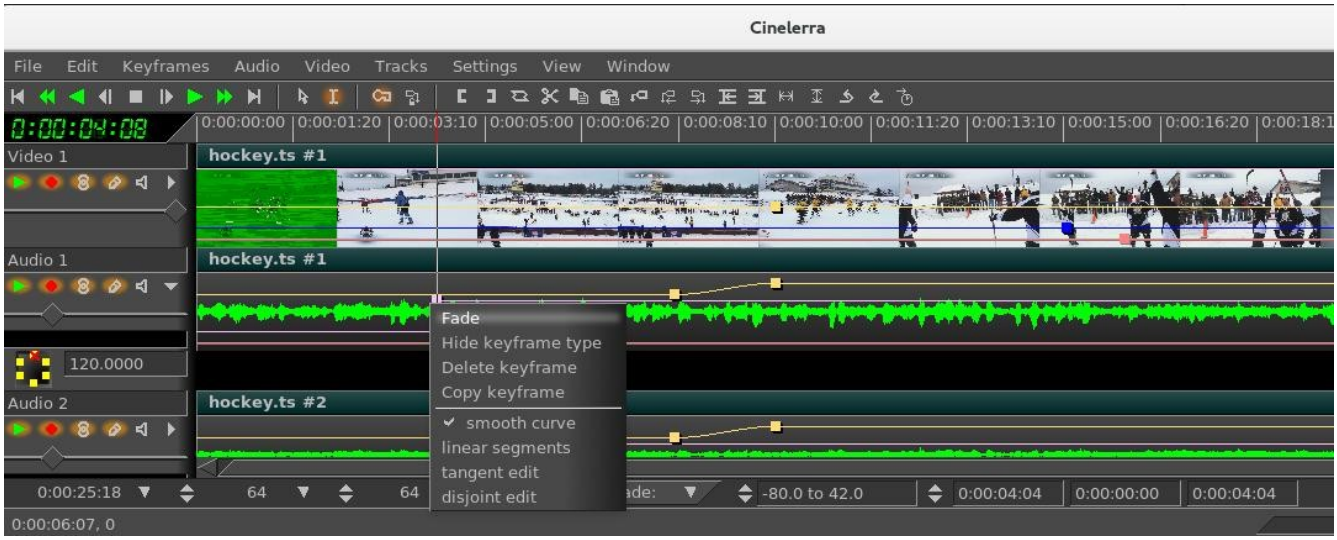
The more important new feature is that you can now click Button 3 on a keyframe box and a menu pops up with the first menu item showing the keyframe type. The top menu item can be activated for immediate access to update the automation keyframe value. Some keyframe types, which have values that can be manipulated in another way than by dragging the color coded line, now show up with a different colored background to make them more visible. Keep in mind that Zoombar ranges/values must be set to appropriate values when working with specific keyframe types, such as Fade or Speed.

There is now a line separating the first 4 items, which are just non-automation type settable values as opposed to “auto” keyframe types. There is also a line separating the last 3 items of Mode, Pan, and Mask which display the symbols associated with each type. Their color is not changeable.

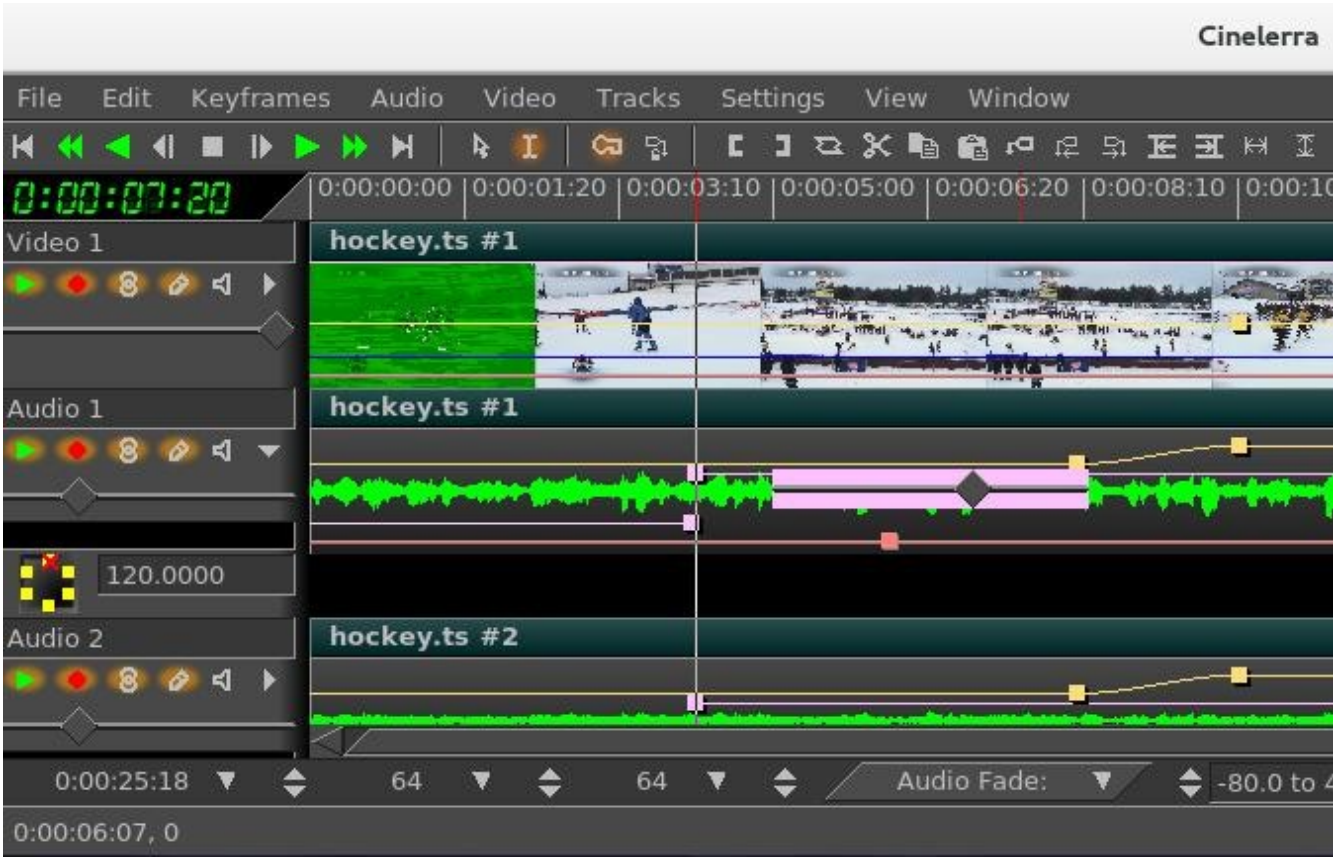
Screenshot below displays the Show overlays popup with all of its options and color coded types such as yellow for Speed and blue for Camera Z. Upon clicking on the associated “color ball” to the right of any keyframe type, for example “Fade” in this screenshot, the color wheel palette window pops up so that you can manipulate the color as desired.



Screenshot below shows several color coded lines for different keyframes, as well as the popup menu associated with the Fade keyframe obtained when the box was clicked. Note the first line of the popup menu allowing you to bring up the Fade slider for manipulation in the next screenshot.

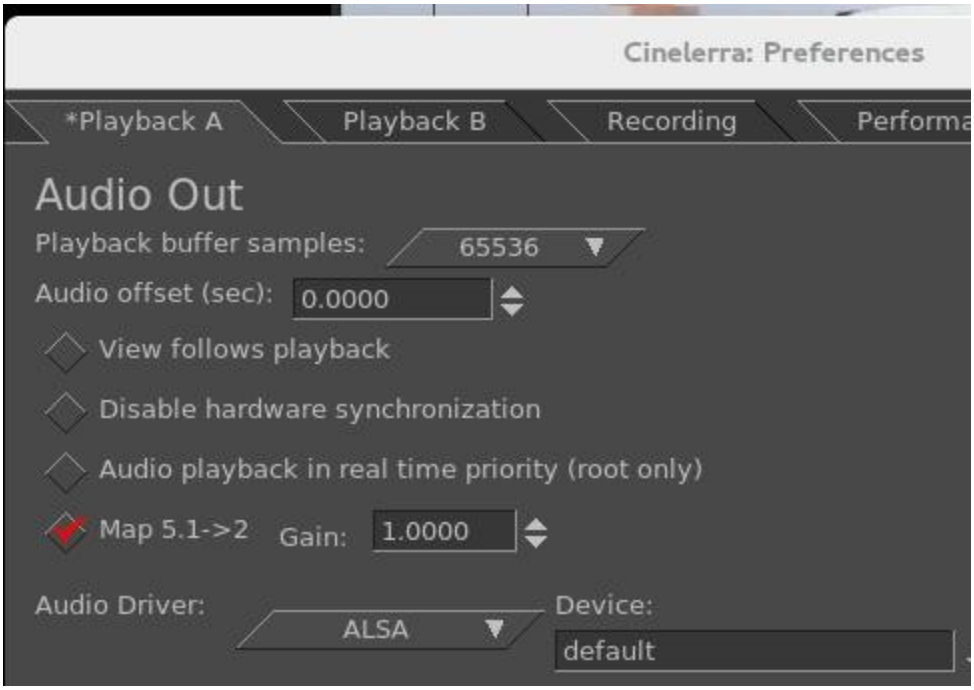


Screenshot below shows the slider bar for the Fade keyframe. It is in the same color as the color coded keyframe type line which is the same color as in the "Show overlays" window.



33. Playback 5.1→2 Driver Downmix

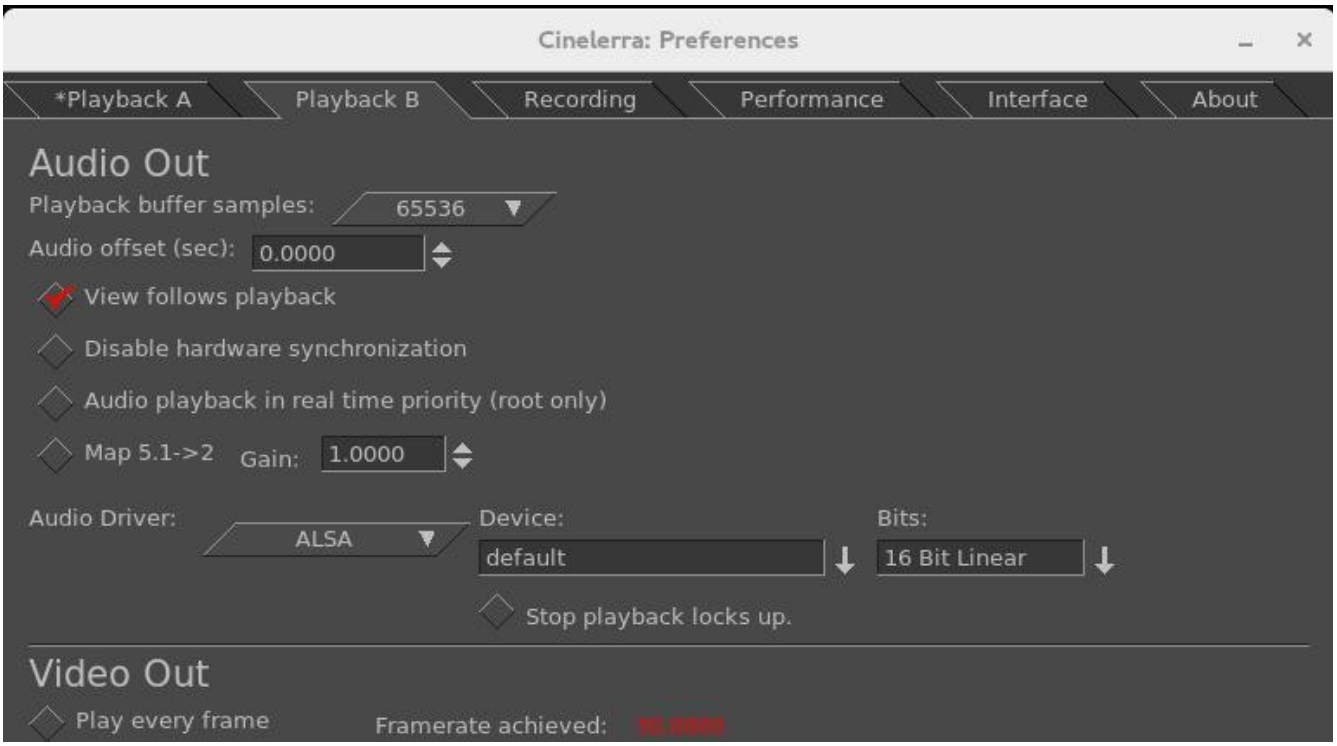
Driver downmix mixes 5.1 down to stereo on the output device side. This is different from the patchbay and menubar functions which reset the pan/mix levels of the input channels. In this way, you can render 5.1 media, and use stereo speakers to listen in the same session setup. This downmix only occurs if the playback is 5.1 (6 channels), the device config is stereo (2 channels), and the mode is set by checking "Map 5.1→2" in the "Audio Out" section of Settings→Preferences→Playback A/B.



34. Multi-screen / Playback Configuration

Cinelerra 5.1 supports 2 separate preferences for the playback configuration. Cinelerra can be operated in a single or dual screen configuration, both by using Xinerama or dual screen configuration of X windows. The Settings→Preferences menu has “Playback A and Playback B” tabs. The target display and audio device configuration can be separate, to support up to 2 display and/or audio device stations. The active configuration displays an asterisk (*) in its selection tab and the selected tab will be made active when OK is pressed. For example: you may have a dual screen monitor system with the left screen showing the cinelerra main window and the right screen showing the composer. Another setup might use a monitor for the left screen and an HDTV as the right screen displaying the composer. When a playback configuration is selected, the audio/video device configuration is switched to the playback selection. The active playback setup can be changed through use of the menu pulldown of Settings→Preferences or via the remote control menu selection (see the section “Remote Control for DVB” for more detail). Yes, you can watch TV on cinelerra instead of cinelerra on TV.

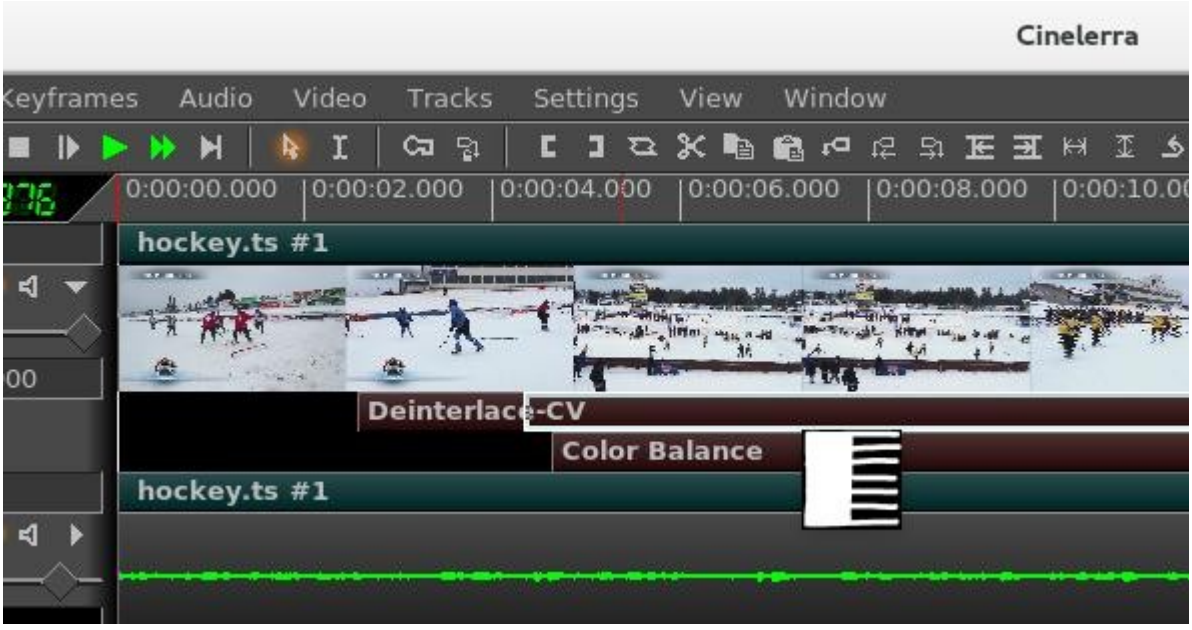
Screenshot showing partial window of *Playback A selected and the second tab for Playback B.



35. “Free” Dragging Plugins

In order to explain how to use the “free drag of plugins”, a short explanation is provided here first. A timeline region can contain a plugin stack. The plugins are normally applied by first selecting a timeline region, and then dragging a plugin from the resources window and dropping it into the selected timeline region. A plugin title bar will appear below the track, which has two buttons. The first title bar button will “show controls” (magnifying glass) and the second is on/off. The plugin position can be changed by using the “drag and drop editing mode” (the arrow icon that is positioned approximately under the video pulldown on the main menu bar) and then selecting a plugin title bar with the cursor for a drag and drop operation. The plugin outline will be drawn over a targetable timeline region, and the plugin can be re-positioned into any plugin track – this is “free drag”!

Screenshot showing 2 plugins, one have already been dragged, and the other Deinterlace in the process of being dragged. Note the gold-colored arrow which enables “allow drag and drop editing mode”.



36. Simple Animation - Filescene using Festival

This functionality was added to cinelerra by the original author to create simple animation. The file type for this animation is “Scene”.

To get started making a simple animated movie copy from the directory: <cin_path>/cinelerra/tests the text2movie and text2movie.xml. You can see what this does via File→Load… text2movie.xml. The file text2movie acts like a normal asset, except changes to it are immediately reflected on the timeline without reloading and the length is infinite. You can just edit the text2movie file to change the script. If the length of the movie increases, drag the right edit handle to extend the edit or use the pulldown Edit->edit length. There is one audio channel created for every character. The frame rate, sample rate, frame size, and camera angles are fixed. To see these values, right click on the asset and look at the “Asset info”.

Currently the functionality that is implemented focuses on dialog between two people. The models are defined in model files saved in Cinelerra's executable directory (for example, /opt/cinelerra/models). The character model and voice is selected separately in the script. The model files have the same name that appears in the script and are usually saved in the directory the script is in, but there is a defined search path, if not. You can create new models for the script without affecting the entire system. These models define the total size of the model along with the images used – the model images are 2D png images because all the animations are baked. Since there is no 3D renderer, no custom movement is supported.

There are currently 2 actions implemented:

1) Character2 can cut off character1 if character1's dialog ends in "...".
2) Inserting "[pause]" anywhere causes the character to pause. This is useful for adjusting the timing of dialog.

This is "simple" animation so you can expect speech synthesis not to be that good. And you will have to adjust punctuation and spelling based on the sound. Since the dialog is rendered on-demand, there is a delay when each character starts to speak but you can split dialog into shorter blocks to reduce the delay.

37. Some Plugins – New or Newly Documented

Echo - is reflection of sound. This plugin could be used to add echoing to video of your canyon hike.

The 3 dials shown here are used to vary these attributes.

Level represents the volume adjustment.
Atten is attenuation which is a general term that refers to any reduction in the echo reflection. Sometimes called loss, attenuation is a natural consequence of signal transmission over long distances.
Offset is the lag in the attenuated echo signal. Offset means adding a DC level to a signal. It offsets the signal up or down in a DC sense without changing the size of the AC part of the signal. When you add an audio clip to the Timeline, the clip plays back from the beginning of the source audio file. The point in the audio file where the clip starts playing is called the offset. By default, a clip's offset is zero, the beginning of the source audio file. You can change the offset so that the clip starts playing from a later point in the source audio file.

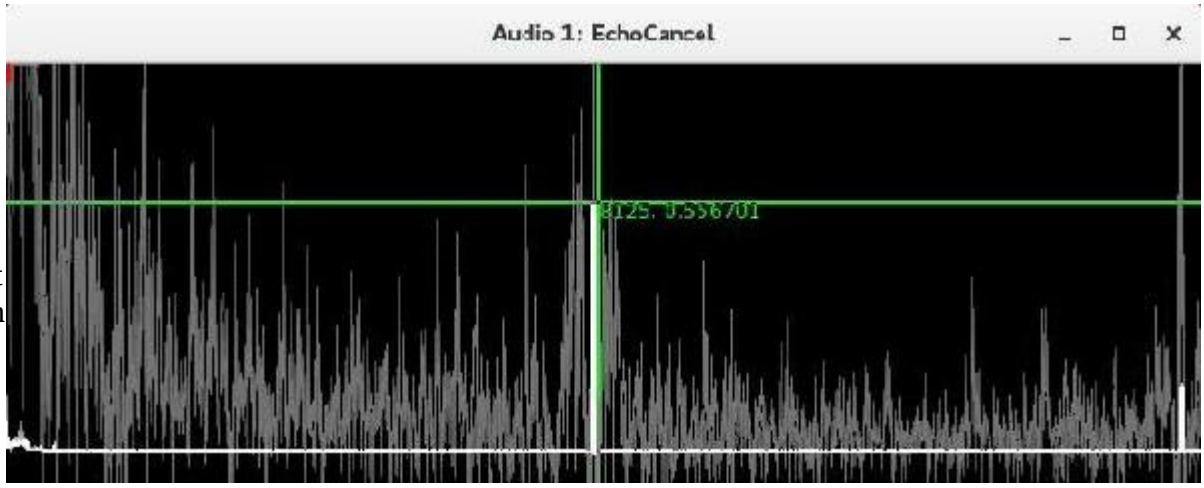
EchoCancel - is the process of removing echos from audio in order to improve the quality. Echo cancel may be needed because an audio recording was done in a room that led to echo generation or there was some kind of unwanted feedback. There are many controls for the EchoCancel plugin which are defined here. However, the first thing you will see when you bring up the plugin, is the top portion that is black which will show a + in the middle when you mouse over it. Once you start playing audio, you will see the cepstrum spectral data inside the window. A cepstrum results from taking the inverse Fourier transform (IFT) of the logarithm of the estimated spectrum of a signal. It is used to identify the period of the echo in the audio. It is recommended to just set the "Mode" to "On" but the below defined parameters can be utilized by professionals.

On/Off Boxes
Normalize: audio normalization adds variable amounts of gain to an audio recording to bring the average or peak amplitude to a target level (the normal amount), on an ongoing buffer by buffer basis. This is to make the cepstrum graphical data appear between 0 and 1. Checkmark appears if On.

Dial Parameters
Level: scale factor used to draw the cepstrum output when "normalize" is not in effect.
History: number of previous cepstrum outputs redrawn as fading graphical data.
X Zoom: X axis scale factor to magnify low frequency cepstrum graphical output.
Damp: echo envelope decay factor used to smooth the cepstrum/correlation data.
Peaks: number of maximal envelope values used in the echo gain calculation.
Cutoff Hz: low frequency cutoff value to prevent beat frequency (heterodyne) echo canceling.

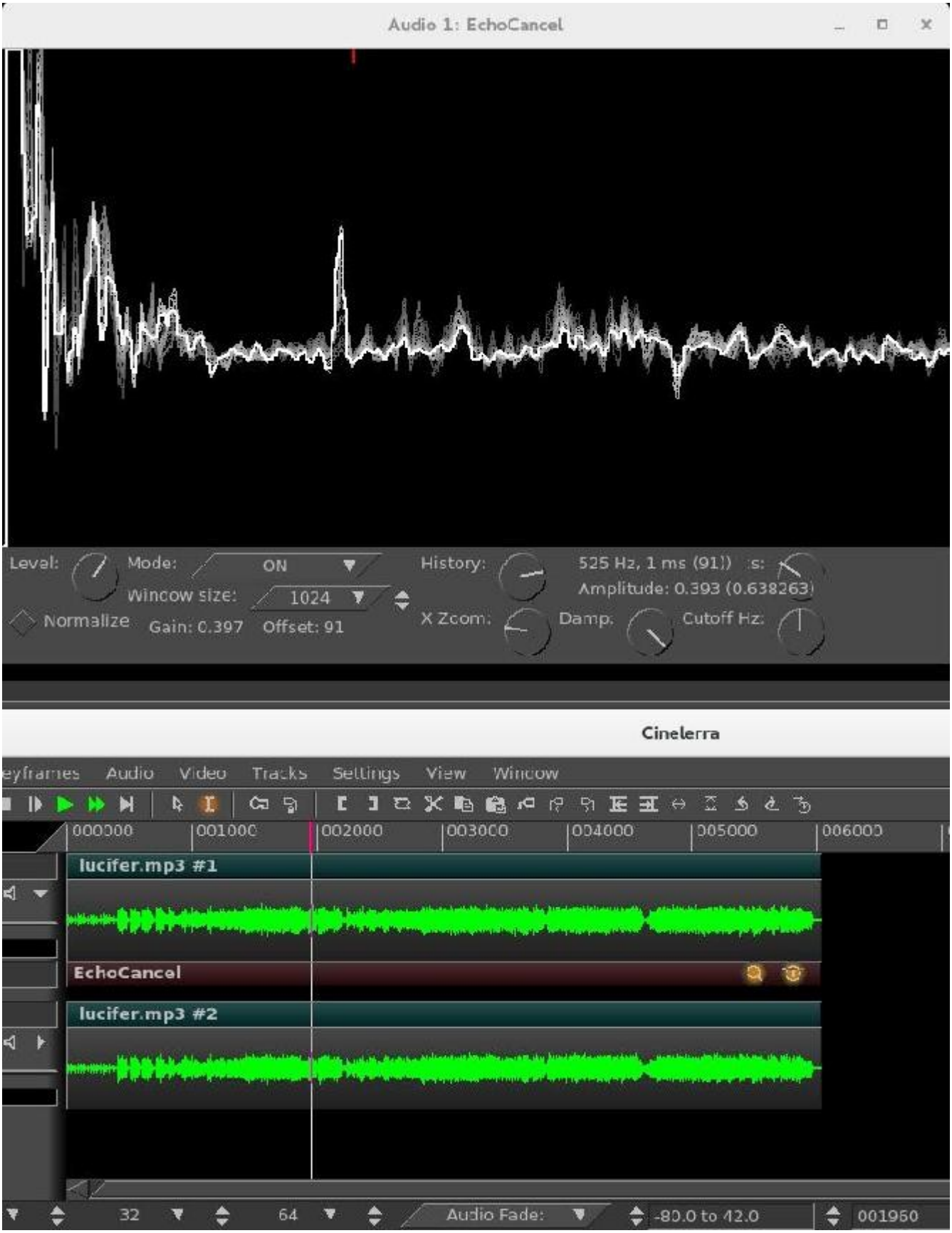
Down Arrow Boxes
Mode: MAN, Off, or On. When Off is selected, the plugin is not active. When MAN is used, the only one peak is used for the echo gain envelope. It is set by pressing mouse button 1 in the ceptstrum graphical output. The Gain and Offset are updated as the pointer drag operation resets the indicated gain and offset values. When On is selected, the echo gain envelope is automatically calculated by cepstrum and auto-correlation of the input audio in the last "window" size audio samples.
Window size: parameter can be set to Default, 1024, 2048, … doubled values... up to 262144.

Values:
Amplitude: the cepstrum value at the drag point during manual envelope selection.
Gain: echo gain setting determined by manual selection.
Offset: echo period setting determined by manual selection. The *Hz* (frequency), *ms* (millisecond duration), and sample *offset* (audio samples) as determined by manual selection.

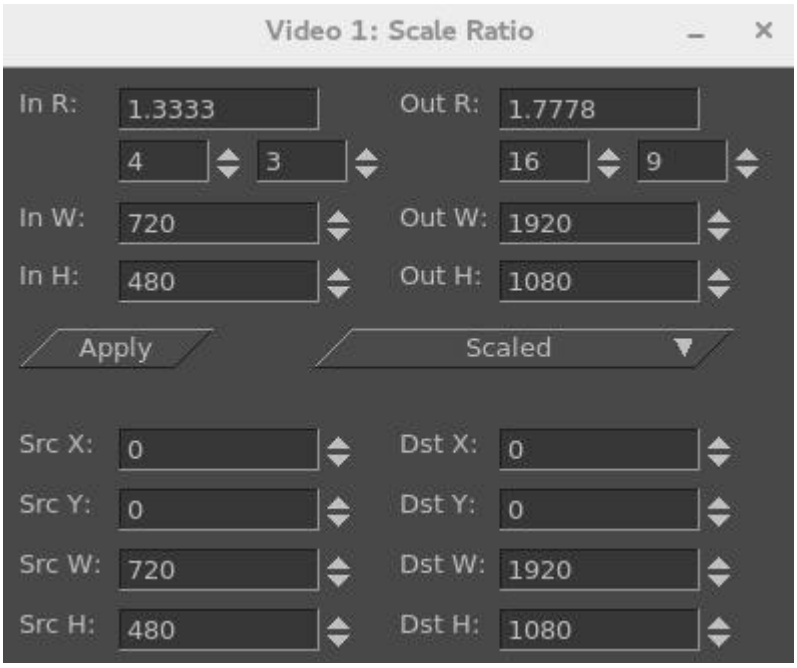


EchoCancel screenshot showing the crosshairs (+) with the offset /gain values in the middle.

Screenshot with Mode set to “On” for the usual situation for most users. This is highly recommended.



Scale Ratio - with the Scale Ratio plugin you can manipulate your video to maintain the pixel aspect ratio (proportional geometry), possibly for a different output Display device.



On the first 2 lines, there is “In R” and “Out R” representing the current input and output aspect ratios. Use the arrows to change to your desired values. Next you have the “In W/H” and the “Out W/H” for Width and Height.

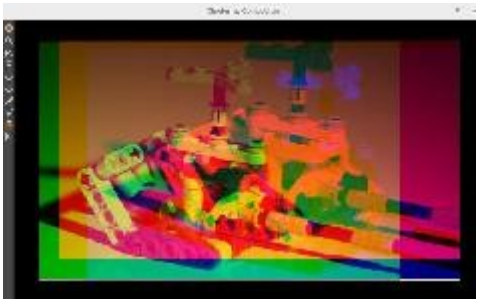
Left and right sides of the bottom portion show the Source and the Destination X, Y, W, H values. As you change the values on the left side, you can see how this will affect the output as you observe the results in the compositor window. For example, as you change the values for SrcY in a “cropped” Scale scenario, you see up/down movement.

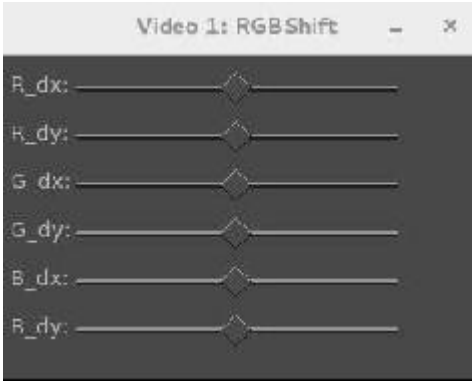
In the middle of the plugin on the right-hand side, you can set the Scale type of None, Scaled, Cropped, Filled, Horiz edge, and Vert edge. The top part (aspect ratio data) is used to compute the bottom part when the “Apply” button is pressed. The bottom part allows you to reposition the image input or output to customize the results. When the in/out aspect ratios are different, the output must be cropped or filled to fit the output and maintain pixel square appearance.

RGBShift - most cameras take the light coming into the lens, and convert that into 3 sets of numbers, one for Red (R), one for Green (G), and one for Blue (B). Some of the older cameras were composed of sensors and originally the RGB sensors were on 3 separate planes and had to be aligned. If they were misaligned in the video, you can use RGBShift to get them realigned. To move a specific color up/down, modify the dy value using the slider bar in the RGBShift window. To move a color left/right, modify the corresponding dx value. Note that the current values of the RGBShift are maintained in the .bcast defaults file and will be retained across sessions.

←- Normal color aligned.

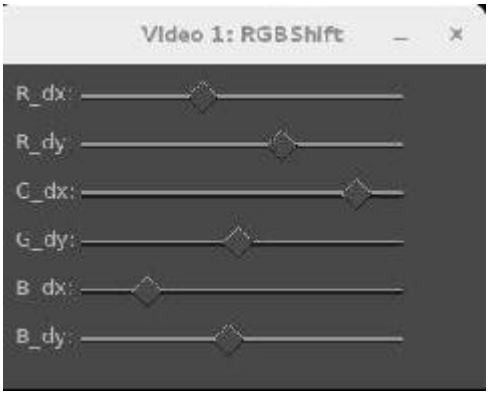
Misaligned color →





←- RGB shift in a good video should be 0's.

You move the R, G, B sliders to align the colors correctly. →



YUVShift – for YUV input video from older cameras using sensors, it is possible to have misalignment of the 3 sets of numbers: Y, which represents the luminance or brightness component, and for U and V representing the chrominance (color) components (a Blue channel and a Red channel that don't contain any brightness). If they were misaligned in the video, you can use YUVShift to realign. To move a specific component up/down, modify the dy value using the slider bar in the RGBShift window. To move a component left/right, modify the corresponding dx value.

Screenshot below shows the blue U component aligned too far to the left. And the red Vcomponent is misaligned too far to the right. Note the U_dx current slider bar set to 0 as shown by the yellow box value in the YUVShift plugin window. All components are currently at zero.

A corrected video image is shown in the screenshot below. Now the red and blue colors are correctly aligned. Note how U_dx is now at +20 and V_dx is now negative to realign the image.abc

38. Cinx and a “Bit” of Confusion

Cinx is the exact same program as Cin. The X (x) represents the roman numeral 10 for 10/12 bit as opposed to 8-bit standard. The third-party libraries used for x264/x265 must be specially compiled with –bit-depth=10 in order to produce 10/12 bit rendered output. This build will not be able to output 8-bit depth which means you have to retain the Cin version also. Whatever build ffmpeg is linked to will determine what bit depth it can output. This is why there has to be separate builds, which is very inconvenient, and hopefully the people maintaining these libraries will work out a better methodology in the near future so that a choice can be made in a single executable.

Keep in mind that the regular 8-bit version works on 8-bit bytes – the standard word size for computers, but the 10/12-bit version has to use 2 words to contain all 10/12 bits so you can expect rendering to be as much as twice as slow.

39. Lossless Rendering

Lossless is a “marketing term” – it simply means that in the compression of a file, all of the original data, every single bit, can be recovered when the file is uncompressed. Nobody describes gzip or gunzip as lossless as nobody expects data to be thrown away. The opposite marketing term is called “lossy compression” where some data is permanently deleted so that when uncompressed, all of the original data can not be exactly recovered. It is generally used for video and sound, where a certain amount of information loss will not be detected by most users or the playback hardware does not reproduce it anyway – it is a trade-off between file size and image/sound quality. The files created will be more than 10 times larger than usual. Most players will not be able to decode lossless as the bitrate will overwhelm the device.

For x264 lossless compression to work, the only color model allowed here is yuv420p. Any other specification will be converted to yuv420p and the data will be modified. Also, keep in mind that the YUV color model has to be converted to RGB, which also modifies the data.

To use x264 lossless rendering - choose File format of “ffmpeg”, “m2ts” in the Render window. Click on the Video wrench, which brings up the Video Preset window and scroll down in the Compression filebox and choose “lossless.m2ts”. Preset=medium is the default, but can be varied from ultrafast (least amount of compression, but biggest file size) to veryslow (most amount of compression, but still HUGE) in the parameter box where you see “qp=0”. This option is also available for bluray creation.

40. Other note-worthy minor Enhancements

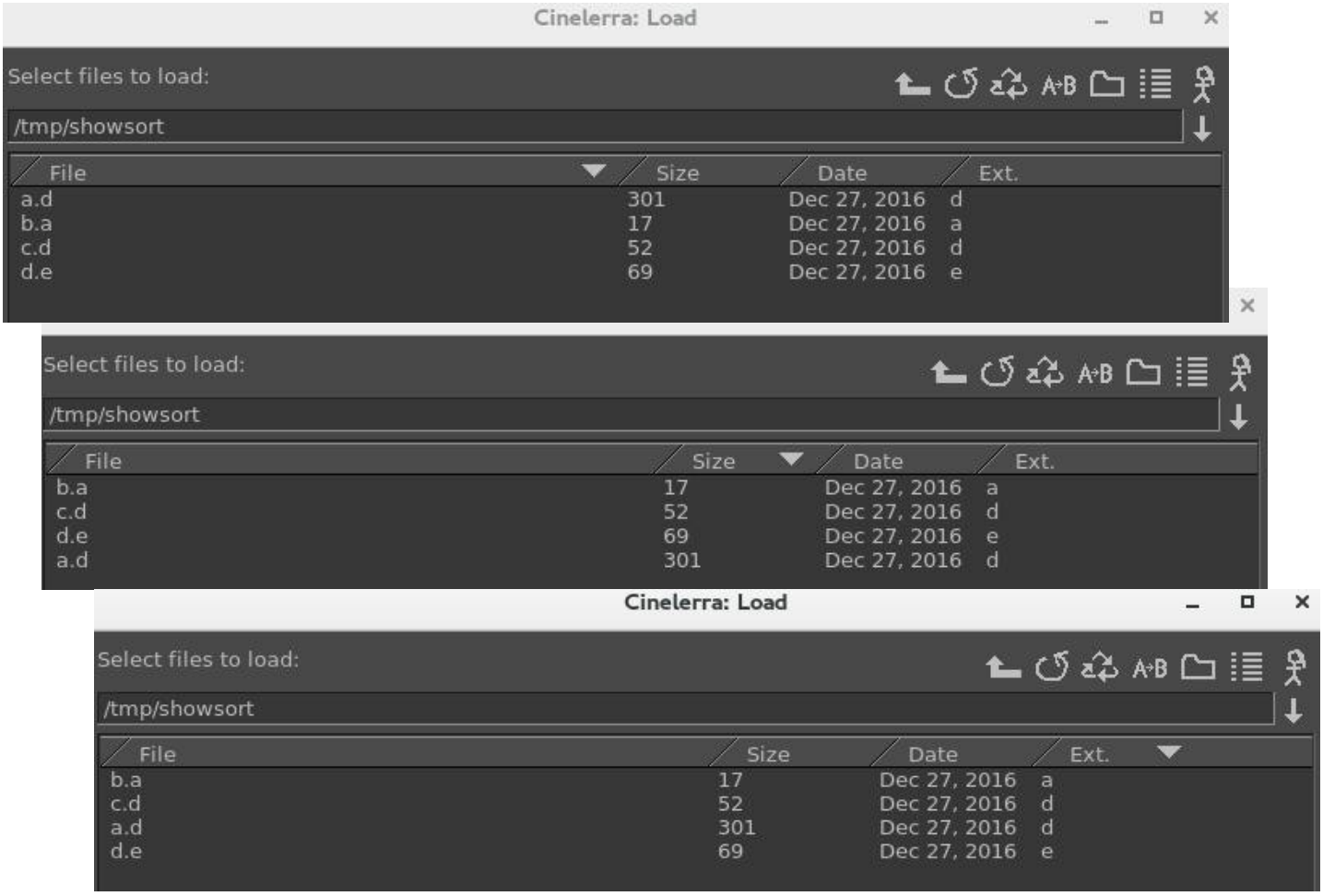
40.1 Replace-type Insertion Strategies

For ffmpeg and mpeg files, when the Insertion strategy methodology in the File→Load files pulldown is chosen to be either “Replace current project” or “Replace current project and concatenate tracks”, the basic session format parameters are reinitialized to match new media. This selects the default asset and determines its width, height, and video length, frame rate, calculates the colormodel, and assumes square pixels to make an intelligent guess about aspect ratio for video. For audio, the sample rate, audio length, and channel count (mono, stereo, or 5.1) are reinitialized. In addition the “Track Size”

will be computed and is reinitialized to match the new loaded media. When using “replace” type insertion strategy, the new asset list is the only media in use so that this update saves the user from immediately needing to change the session format to match the only possibility.

40.2 Sort within Sort in File Load Window

When you use the File pulldown to load files, you can do a sort within a sort when you click on the labeled header box. This is useful, for example, when you want to find the smallest file for a specific extension. In the screenshots below, the first illustrates the default “File” sorted alphabetically; the second shows the “Size” is now sorted; the third shows how after sorting on Size, you sort on Ext. The size sort is maintained within the extension sort so that c.d comes before a.d in the File header box because the size is smaller.



40.3 Playback Speed Automation Support

The speed automation causes the playback sampling rate to increase or decrease to a period controlled by the speed automation curve. This can make playback speed-up or slow-down according to the scaled sampling rate, as “time is multiplied by speed” (speed X unit_rate).

40.4 Keyframe Reticle

In the “Editing” section of Settings→Preferences→Interface there is “Keyframe reticle” with options of Newer, Dragging, or Always. This is used to help in checking edit alignment across tracks. A reticle is a sighting line used to line up visual items, like cross hairs in a eyepiece. The appearance and function of sighting lines can be changed when dragging auto keyframes. To see the effect, create some fader autos and drag a few to see the reticles drawn. "Always" renders a line over all plugins, and "dragging" only over the drag icon. "Never" draws nothing.

40.5 Updatable Icon Image Support

During the build of Cinelerra standard icons are created from .png files. When you start the program, these files are loaded into memory but if the program finds a local .png when it is started, that will override the one in memory. The user's local file must be put into the appropriate directory. For example, to replace the Theme Hulk “blue banana” with a “red apple” instead, create your .png file as desired, and put the file in <target_directory>/bin/plugins/theme_hulk/bluebanana.png.

For most User installs, the <plugin_name>.png file will be located at:
<cinlib_path>/plugins/picon or <cinlib_path>plugins/theme_<theme-name>
For some System installs, the files will be located at:
/usr/lib/cin/plugins/picon or /usr/lib/cin/plugins/theme_<theme-name> (ubuntu-16 distro)
/usr/lib64/cin/plugins/picon or /usr/lib64/cin/plugins/theme_<theme-name> (Leap distro)

40.5.1 Details on where to put Plugin Icons

In order to make the icons available to all themes, which would thus be the default when no theme-specific icon is available, put the png file in the <cinlib>/plugins/picon directory.

The Cinelerra program looks for a plugin icon in 3 places:

1.) First, it tries to find a png file in <cinlib>/plugins/theme_<name> where name is one of the themes.
2.) If that does not exist, it looks in the <cinlib>/plugins/picon directory.
3.) If there is no corresponding .png file for a plugin, the program uses a built-in default:
- ordinary plugins use 3 vertical color bars as a default;
- ffmpeg plugins use the words ffmpeg on blue-colored background as a default icon;
- ladspa plugins use a green-colored audio wave for a default.

Keep in mind these points for newly created plugin icons:

- All included icon images become part of open source, in the public domain, and not proprietary.
- The preferred format is 52 x 52, 8-bit/color RGB or RGBA, non-interlaced.
- Since plugin icons are used by different themes when none for a specific theme exists, it is recommended that a transparent background be used. Otherwise some color background that looks good for one theme may not for another.
- In order to test a new icon, you have to have write permission in the <cinlib_path>/plugins directory so you may have to become the root user to copy the .png file to the correct location.
- If there is currently no theme-specific .png files present, it may be necessary to first create the theme

directory in <cinlib_path>plugins as <theme_name> in order to put the .png files in that subdirectory.
- Make sure that the ownership and file permissions match the existing directory and files, or the data may not always be accessible.
- All ffmpeg icons must begin with ff_<plugin_name>.png (Resources window title will still be F_…)
- Once you have placed the .png file in the correct spot, you will have to restart Cinelerra to test it.
- To submit your .png file for inclusion into a cinelerra branch/version for all to enjoy, it is best to upload it to any datafilehost and notify the community via email with any informative documentation.

40.5.2 Example of new Plugin Icon Testing

For a simple test just copy an existing <plugin_name>.png file into the <cinlib>picon directory with new ff_loop.png filename. This icon will show up in cinelerra and execute the F_loop function.

A theme-based test, if you want a specialized name in the "hulk" theme for F_aeval, create a picon named ff_aeval.png in <cinlib>/plugins/theme_hulk/

```
cd <cinlib>/plugins                          ! go to the correct directory
mkdir theme_hulk                             ! create subdirectory if does not exist
ls -d theme*                                 ! list the theme directories
theme_hulk/ themes/ theme_suv/               ! check for existence (and permissions)
cp picon/gamma.png theme_hulk/ff_aeval.png   ! Copy any example .png file
```

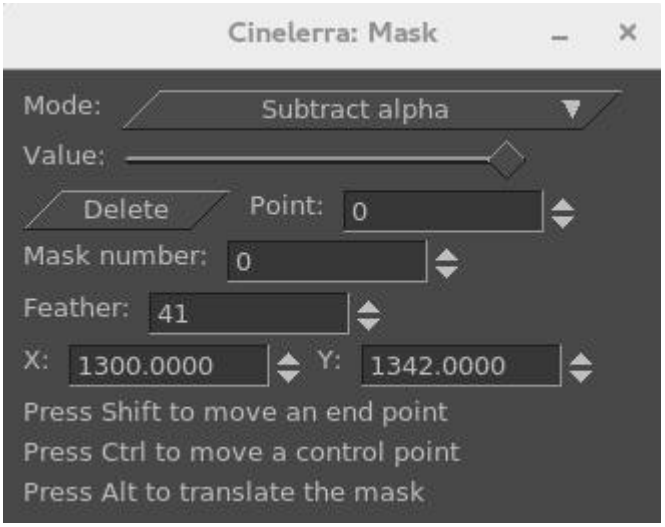
Then restart cinelerra in the hulk theme by changing Settings→Preferences→Interface under the Editing area where the Theme: box exists.

40.6 Masking in Composer Window Key Difference

Usage for Composer masking was changed to create a more consistent user interface and make the meaning obvious. It is easier to remember Control key for control point and Shift for endpoint.

It now works the same way that the fader control point handles are edited in the main window, like using the control key down to qualify the drag selection.

This partial screenshot shows the legend in the Mask info window of the composer outlining usage. ---->



40.7 Plugin – SVG via Inkscape

This plugin allows the user to manipulate an SVG (scalable vector graphics) image with Inkscape without having to leave the program. The associated cinelerra window provides the ability to change the DPI, the Out x/y coordinates, and the Out w/h values.

DPI is "Dots per inch" and it represents the resolution of the SVG image. Since the image is scaled with interpolation mode linear, the edges will look blurry when the input resolution is lower than the output resolution. You can either set the desired DPI value in the window or use the tumbler on the integer text box, then use the “update dpi” button to have the change take effect. Changing DPI causes the entire image to be re-exported via inkscape. DPI changes cause adjustments in the resolution, speed of re-import, and storage needed for the image data.

Out_x/Out_y parameters allow for changing the location of the SVG via the x or y coordinates. Scaling is controlled by the Out_w/Out_h parameters for width and height as they are normal parameters to "overlay".

Screenshot below shows the menu options plugin window and the SVG image in the Inkscape window.

40.8 Hide Keyframe Type

The “Hide keyframe type” in the Popup Menu of a keyframe associated box works in the same manner as disabling the keyframe type in the pulldown View menu. Its usefulness is for convenience. For example, in View you enabled Fade, Mute, and Speed; you do a some work on one of them, and want to work on another one. You can just get the one you are finished with out of sight without having to "think" about which one it is and leaving the area to go to the pulldown menu. Also, right clicking anywhere on the associated keyframe line, brings up the Hide Keyframe Type option.

40.9 Button Operation Preference

To accommodate 2 methods of favorite popups, there is a flag “Popups activate on button up” (the default) in Settings→Preferences→Interface. When unchecked, “Popups activate on button down”.

Screenshot which shows a red checkmark under Flags for “Popups activate on button up”.

40.10 Project SMP cpus Count

On large cpu systems, it can come in handy to downgrade the number of cpus used for some plugins; otherwise it uses all of the processors and splits up the program into too many pieces which may add considerable overhead in high cpu count systems. To restrict the number of processors utilized, change the count number in Settings→Preferences→Performance . That number will be used for the plugin per load balance operation cpu limit, which uses smp-cpus to stripe your data. It does not affect the number of cpus used in any other cinelerra operation besides the plugins.

APPENDIXES

Appendix A: Crash Handling Tips

This section is a handy guide for describing various kinds of software computer system failures. Only some of these various lockups or crashes can be dealt with. Hopefully, it will help to have some hints to know what kind of failure it is, or to save your work or to avoid future problems. For most of this, your name must be root, although you can certainly try to see if it works for you when not root.

System lockups:
When the system locks up, it is usually a system problem. Normally an application program cannot hose the system. It is a major goal of system design to prevent an application (app) from failing a system interface. This does not mean an app can't mess things up.

Cinelerra crash:
This is covered in Section 6 – Crash Dumps for Analysis. Just a reminder that you must be root and by providing a crash dump and as much other information as possible, you will be helping the developer to analyze the problem and fix it so that it can be avoided in the future.

X Server crash:
Keyboard does not respond, screen is frozen, caps lock may operate LED light. Sometimes using ctrl-alt-F1 ... ctrl-alt-F7 (etc.) will allow you to regain control of a VT console. You can use this to login and check logs: eg. /var/log/Xorg.0.log, dmesg, journalctl … etc. If you have another computer, make sure a terminal server is configured (for example: rsh, ssh, or telnet), then remote login via this other computer and check the logs. Most important is to immediately note the current software state, and the very last thing that preceded the crash, i.e. last button click, last keystroke, … or whatever.

Kernel crash:
The machine goes completely dead. The keyboard caps lock LED will probably be flashing. Most likely the only way to see anything after the kernel crashes is to use a serial port console log and usually kdb, the kernel debugger, and special cabling. This requires a lot of setup, and is normally reserved for experts. Login from another computer will not work. Pinging the ip address will not respond since the network stack is part of the kernel. There are some virtual machine setups that will let you debug a “guest” kernel, but this also requires a lot of setup, and affects which kernel is currently under test. The kdb route is preferable.

Keyboard grabs, Server grabs, and Deadlocks:
A grab is an X-server state where all events are forced to just one window event stream. This forces the user to respond to the dialog. Things seems to be working, but no keypresses do anything useful. The system clock and other programs will still be working. The network will work for remote logins. Grabs can be canceled if the /etc/X11/xorg.conf X config contains special setup as shown below:

```
Section "ServerFlags"
  Option     "HandleSpecialKeys" "Always"
  Option     "AllowDeactivateGrabs" "True"
  Option     "AllowClosedownGrabs" "True"
EndSection
Section "InputDevice"
```

```
    Identifier  "Keyboard"
    Driver      "evdev"
     ...
    Option "XkbOptions" "terminate:ctrl_alt_bksp"
    Option "XkbOptions" "grab:break_actions"
EndSection
```

or to $HOME/.xinitrc, add:

```
#  xkb terminate/grab actions disabled in xorg.conf, use:
setxkbmap -option "grab:break_actions"
setxkbmap -option "terminate:ctrl_alt_bksp"
ctrl-alt-bksp = terminate the X-server, may restart automatically
```

If your system is running in runlevel 5 (you might be at runlevel 3 for development), it also allows the user to cancel a keyboard grab using:

```
ctrl-alt-keypad-* = ungrab
ctrl-alt-keypad-/, = ungrab, kill grab owner process
```

Modal forms (always on top, and usually ptr/kbd grab) dialog boxes can hose a system by putting a form over another form holding a grab. This means the form that needs input may never get any, and the effect is a deadlock. Usually you will have to restart X (ctrl-alt-bksp).

Window Manager issues:
The "desktop" window manager can intercept and modify all kinds of user input. Mostly, this is a good thing, but can be a nuisance. If user keypresses can be programmed to trigger actions, then they may be useful to send KILL or INTR to an app that seems to be holding X's attention. For example:

```
killall -INTR cinelerra, killall -9 cinelerra,
killall X, but you must run as root to be able do this
```

The ALT and META keys may be intercepted by the window manager, and this can cause unexpected interface operation.

Appendix B - Developer's Section

B.1 How Builds Really Work and Even More Options

This section describes how builds really work if you want to know more about making changes or understanding the process probably only for a developer or system administrator.

Builds occur in 4 basic steps:

1) unpack/patch source code
2) configure build

3) make build targets
4) installation

So, an example of what happens in 4 steps for a single-user build would be as follows:

1) unpack/patch source code
   git clone –depth 1 “git:/…/target” cinelerra5
   ./autogen.sh
2) configure build
   ./configure –with-single-user
3) make build targets
   make 2>&1 | tee log
4) installation
   make install

A lot of things can be tweaked to change the results. Mostly these changes are parameters to the configure step, which can change important build related items, like the application name, or where and what the target system directories should be. This makes it possible to have several versions at the same time on the same computer if needed. To see what it is that the makefiles use to build cinelerra, look at the resulting toplevel global_config file which is created by the ./configure step.

Building cinelerra requires many thirdparty libraries, and it is recommended that you use the static build version included in the git repo. Some of them are patched, and fix significant bugs. It is important to note that because system installation historically has been with as many shared objects as possible, the defaults are that any system library detected during configuration setup will be used, when the package is built “--without-single-user”, which is the default build. To build with static thirdparty libraries for system install to the system /usr area, use:

.configure --enable-static-build –prefix=/usr

Sometimes, additional package parameters and variables are needed during thirdparty builds. These optional values occur before and after the “configure” and “make” commands during a build. A presentation of the format of the package qualified variable names and how they appear in the build procedure are:

pkg.cfg_vars	prepended to configure
pkg.cfg_params	appended to configure
pkg.mak_vars	prepended to make
pkg.mak_params	appended to make
pkg.cflags	added as CFLAGS+=$(cflags) to pkg.vars
pkg.cppflags	added as CPPFLAGS+=$(cppflags) to pkg.vars

These steps are done for EACH of the packages in the thirdparty build:

<pkg.cfg_vars> ./configure <pkg.cfg_params>
<pkg.mak_vars> make <pkg.mak_params>

The thirdparty Makefile has a set of default vars and params used to build each of the needed thirdparty packages, but you can specify new or overriding values for these Makefile substitutions. These

thirdparty build config changes are specified in the toplevel file: cin_config. By using this file, you can save the configuration changes made for the current build to use the next time you do a new build. For example, to add an include file path to the giflib build, add this line to cin_config:

```
giflib.cflags := -I/usr/local/include/giflib5
```

To have a param/var change apply to all thirdparty builds, use:

```
CFG_VARS, CFG_PARAMS, MAK_VARS, MAK_PARAMS
```

CFLAGS, CXXFLAGS and LDFLAGS are forwarded to the thirdparty build environment via:

```
CFLAGS=-ggdb ./configure --with-single-user
```

However, there is no guarantee that the thirdparty build will honor the environmental flags.

Finally, there are build controls, which enable/disable and set build features.

A few of the more useful ./configure --parameters are:

--with-jobs=n	where n=number of build jobs; defaults to 1.5*cpus+2.
--enable-static-build	build all 3rd party libs; defaults to yes if single-user, else no.
--with-single-user	build installs to <build_path>/bin; no system installation.

The ./configure command builds “global_config”. The global_config is read by the thirdparty/Makefile to create the recipes and definitions used by the thirdparty build.

There are a lot of different options. Thirdparty library build control is available in the configure step of the build. Thirdparty libraries are built on a demand basis. So if you use:

```
--enable-libname=auto  # the static-build enable, or the lack of a system library causes a build
--enable-libname=yes   # this forces the thirdpary build
--enable-libname=no    # this forces no thirdparty build
```

Ffmpeg is a “strongly connected component” in the build linkage and widely influences the cinelerra library demands. It is possible to make small additions to the ffmpeg configuration step using the environment varaiable FFPMEG_EXTRA_CFG. For example, to eliminate the use of libvdpau (an nvidia support library) in the ffmpeg configuration step after you have determined that it is causing an error, use:

```
make clean
autogen.sh
export FFMPEG_EXTRA_CFG=" --disable-vdpau"
./configure …
```

B.2 Thirdparty Parallel Build

The Makefile in the thirdparty build directory employs a set of macros used to create a build rule set of

thirdparty library build dependencies. The standard build sequence of [source, config, build] is used to prepare thirdparty products as static libraries. Build package dependency can be specified in the Makefile std-build macro call. These Makefile macro calls define the rules used for each thirdparty build. The expanded rule definitions may be viewed by using:

```
make -C thirdparty rules
```

Individual package libraries can be rebuilt, via:

```
make -C thirdparty <pkg>-clean;  make -C thirdparty <pkg>
```

The rule targets create the set of thirdparty packages which are built from local source archive copies of thirdparty source code and patches, if needed. The build rule set of dependencies allows for compiling multiple thirdparty programs simultaneously using maximum computer resources. This parallel build speeds up the process considerably. For example, these are full static build timings on the production build machine (full build includes building all thirdparty programs as well as all of cinelerra):

```
 1 cpu   =  61 mins
12 cpus  =   7.5 mins
24 cpus  =   2 mins
```

B.3 Translation Assistance for Program Messages

There are several “po” files for various languages to make cinelerra more usable for non-English countries. A program, xlat.C, assists in providing several variations of text files that can be used in order to allow anyone to help make meaningful translations.

To build a new cin.po file, do the following after moving to cinelerra's directory:

```
./po/xlat.sh > /tmp/cin.po
```

To use the xlat.C program, first compile it with “c++ xlat.C”. You can see the usage help here:

```
# ./a.out
    list csv    ./a.out csv < data.csv > data.po
    list po     ./a.out po < data.po > data.csv
    list po     ./a.out dups < data.po
    list po     ./a.out nodups < data.po
    get strings ./a.out key  < xgettext.po
    gen xlation ./a.out xlat   xgettext.po xlat.csv
    gen xlation ./a.out xlat - text,xlat ... < xgettext.po
```

This program has 6 commands where the desired command is the first parameter to a.out.:

1) “csv” = comma separated value file; to recreate a po from a csv file.
2) “po” = convert po to csv data; for example, what you need to convert ru.po to ru.csv. You can even open the resulting *.csv in ooffice and update the key->value replacements (fields separated by "," only and check quoted fields as text during import). These results can

be "Saved As" a csv file, and then used by xlat.C to reformulate a “po”.

3) “dups” = list only key/value items where either the key=value or value=””.
4) “nodups” = list only key/value items where key!=value and value!=””.
5) “key” = list cin.po key set.
6) “xlat” = overlay translation. This is the most important use and is described next.

The xlat command line parameters specify a new cin.po template, usually created with xlat.sh, and a list of key/value files which are used to build a mapping for the desired translation. The mapping files are added to the mapping in the order they appear on the command line, and any existing key is replaced with the newest definition; so typically the newest key/value data is last in the command parameters. Once the mapping is built, the first parameter, the new po template, is scanned and the keys it contains are used to find the latest mapping in the key/value files. The new value replaces the existing value in the template. For example, to overlay a new map onto an existing po:

```
c++ xlat.C
./a.out po < xx.po > /tmp/xx.csv
./a.out po < new.po > /tnp/new.csv
./a.out xlat xx.po /tmp/xx.csv /tmp/new.csv > /tmp/new.po
```

The first run preserves the existing mapping of xx.po, the second creates new mappings from new.po, and the third merges the original and new mappings to create a po with new included/overriding xx.po.

Of course, you can also use the freely-available “poedit” program to provide translations!