Copyright (C) 2018 by Kevin C. O'Kane

This program described herein is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA.

August 22, 2018

This document is subject to change without notice and is a work in progress.
Table of Contents

1 Overview.................................................................................................................................7
  1.1 README - Warnings and Possible Pitfalls........................................................................7
  1.1.1 Supported Operating Systems.......................................................................................11
  1.2 Required Hardware and Software......................................................................................11
    1.2.1 Operating System Requirements..................................................................................11
    1.2.2 Hardware Requirements...............................................................................................11
    1.2.3 Window Manager........................................................................................................11
    1.2.4 Other Running Windows...............................................................................................12
    1.2.5 Graphics Support..........................................................................................................12
    1.2.6 Hardware Timing...........................................................................................................13
  1.3 Goals....................................................................................................................................14
  1.4 Basic Design.......................................................................................................................14
  1.5 Performance.........................................................................................................................15
  1.6 Workspace Switching...........................................................................................................15

2 The Automation Console............................................................................................................16
  2.1 Overview.............................................................................................................................16
  2.2 Audio / Video Players........................................................................................................18
    2.2.1 Mixxx Player................................................................................................................18
  2.3 Browser................................................................................................................................18
    2.3.1 JACK and Browsers......................................................................................................18
  2.4 Console Layout....................................................................................................................18
    2.4.1 Audio control areas for Loopback Modes 0 and 1.......................................................18
    2.4.2 Audio control areas for Loopback Mode 2..................................................................19
  2.5 Casting Console Buttons and Controls................................................................................20
    2.5.1 File Select, Randomize, Text Entry, and Play...............................................................20
    2.5.2 YouTube Videos...........................................................................................................22
    2.5.3 Full Directory Select........................................................................................................23
    2.5.4 Playlist Select, Shuffle and Play...................................................................................24
    2.5.5 File Search....................................................................................................................24
    2.5.6 Program Search.............................................................................................................25
    2.5.7 Stream V and Stream I..................................................................................................25
    2.5.8 Panel Button..................................................................................................................25
    2.5.9 Full Screen Button..........................................................................................................25
    2.5.10 Program Buttons...........................................................................................................25
    2.5.11 Seq Program Button......................................................................................................26
    2.5.12 Rand Program Button...................................................................................................26
    2.5.13 Stop Players Button.......................................................................................................26
    2.5.14 Pause Button................................................................................................................26
    2.5.15 Prog <<..........................................................................................................................26
    2.5.16 Next Button..................................................................................................................26
    2.5.17 Skip Button...................................................................................................................26
    2.5.18 Prog < Button................................................................................................................27
    2.5.19 Prog > Button................................................................................................................27
    2.5.20 Show Saver Button.......................................................................................................27
    2.5.21 Show Slides Button........................................................................................................27
    2.5.22 Show Video Button........................................................................................................27
    2.5.23 Cam0 / Cam1 Buttons....................................................................................................27
1 Overview

1.1 README - Warnings and Possible Pitfalls

This is an on-going project. If you find errors or have suggestions, send them to the email address given on the title page. You may want to check back for updates from time to time.

The following is a list of possible problems you may encounter and how to correct them.

1. This is a work in progress and this manual is being updated as changes are made to the code. Some of the figures will be out of date and not show newer features. These will be corrected as soon as possible. Please check back for frequent updates.

2. This was developed and tested using Linux Mint **18.3 with Mate**. It works, subject to theme management, with:

   - Linux Mint 18.3 Sylvia
   - Linux Mint 19 Tara with Mate
   - Linux Mint 19 Tara with Cinnamon
   - Linux Ubuntu 18.04 Bionic Beaver
   - Linux Mint LMDE 3 Cindy (requires manual install of VLC)

3. The current version in the distribution (versions 2.22 and following) have been tuned to Linux Mint 19 (Tara). This has involved some adjustments to spacing of items to conform to differences and corrections in Mint 19 versus Mint 18. Versions numbered 2.21 and earlier were tuned to Mint 18. This system is not presently compatible with distros that doe not use `apt-get` to updates their software.

4. Different desktop themes result in different colors and spacing of GTK widgets. That is, the same set of GTK buttons may be larger or smaller depending on the underlying window manager theme setting. The code was tuned with the desktop set to the Mint-X theme with Mint-X borders and GNOME icons. The window manager was Marco with Compositing. Other settings may also work but these are the settings that have been tested.

5. There were issues involving newer versions of the camera processing libraries which have hopefully been corrected. If you have a problem with the cameras, please run the camera install and build scripts again. Be certain to run these with the `sudo` command as they change ownership of files to the user running the script. If you run them as `root`, the files may be owned by `root`. Use `sudo`.

6. Rather than write new versions of existing software, this system uses a large number of commonly available standard Linux system packages. Not all these packages, 

---

1 ‘Sorta’ works but Ubuntu still hasn’t fixed their window manager - avoid.
however, are normally installed by default. Thus, the files install.script, install-camera.script, build-camera.script, build-pavumeter.script, and build-pavucontrol.script contain `apt-get install` commands that will install those packages needed for this system to work.

You must run the files install.script, and install-camera.script first and then run the others.

See in.script below for the correct order.

Some packages, however, might have been overlooked and their installation may not be in the installation scripts. You may need to manually install these. Please send an email if you find any that are missing so they may be added to the installation scripts. The file installation scripts must be run as root since it modifies the system. Sorry, only Debian related (.deb) installation available at the moment.

7. After the install scripts have run, you will probably need to reset permissions in the Automation directory. Go to Desktop, open a terminal and type:

   ```bash
   sudo chown -R yourUserId Automation
   sudo chgrp -R yourUserId Automation
   ```

8. When installation scripts finish installing packages, you will need to compile the software with the command `./compileAutomation.script`. This script should not be run as root.

9. The files install-camera.script and build-camera.script build a modified version of the standard Linux program guvcview. This program controls and displays the cameras. The standard system version, however, is difficult to control so a modified version is included. The distribution comes with a binary version but you may need to re-build this for your distribution. This is done by install-camera.script, as root, and then a second file, build-camera.script, also as root. The first script file installs a number of video and audio related system libraries not normally installed by default and the second builds a local version of guvcview.

10. Be sure all script files are executable:

    ```bash
    chmod u+x *.script
    ```

11. Installation summary:

    ```bash
    sudo ./install.script
    sudo ./install-camera.script
    ```
sudo ./build-camera.script
sudo ./build-pavucontrol.script
sudo ./build-pavumeter.script
./compileAutomation.script

There is a script file, in.script, that contains these commands. Note: in.script is NOT the same as install.script.

You will see many messages and several warnings. These are normal. Note that the final line should not be done as root. To run the result, type:

./runAutomation.script

12. The distribution file program.lst is empty and, as a result, the program buttons will be empty. You need to populate program.lst with full path files names of video and/or audio files. See section 2.5.32

13. You should run the VLC player at least once and adjust its default settings. For example, VLC will pop-up an annoying request to be allowed to access the Internet. You may eliminate this in settings. You will want to make other settings adjustments to VLC as well but you should keep a list of any changes as they may need to be undone if they conflict with the Automation system. Some VLC settings you should set (in Tools | Preferences | Interface), turn off (uncheck) the following:

A) Save recently played items.

B) Continue playback.

C) Allow Metadata Network Access (optional - you may want to leave this turned on). If you leave this setting checked, you will be asked the first time you run VLC if it should be left on. If the first time you run VLC is in the automation console, this will cause issues with window placement.

14. The Automation code expects that certain files will be in certain directories. These are discussed below. Attempting to play 'hide the files' will not result in a happy outcome.

15. A ‘compositing’ or similar windows manager is required (System | Preferences | Look and Feel | Desktop Settings | Windows). Marco + Compositing works. Others may not. If the correct settings are made, video will not be transmitted by the encoder.

16. It is possible that this system may work on a virtual operating system but probably poorly or not at all not due to the real-time nature of audio and video processing. The results will most likely be dependent upon the speed of your memory and CPU.

However, that being said, I was able to get it to run in Panel mode on Oracle’s Virtual
Box by giving it 3 CPUs (out of 6 on an AMD FX 6100). Note: the FX6100 is a hybrid machine and not truly a 6 CPU system. While casting in virtual mode, I found that I was able to switch to another real mode workspace without loss of video composition in the virtual machine. That is, the virtual instance continued to cast from the virtual screen and not the visible, real screen. If you try this, be aware that smaller video windows generate less CPU overhead than larger windows.

17. If you have an external mixer, you will want to use LOOPBACK 2 mode (see section 2.4.1). If you do not have an external mixer, you will want to use LOOPBACK 1 mode. In LOOPBACK 2 mode, the volume meter optimal value is about 0.900.

18. If you are in LOOPBACK 0 or 1, pavucontrol will appear along with two volume meters. In LOOPBACK 2, only one volume meter appears. The program pavucontrol is in the box at the lower left of the Automation display. Check that the audio levels for players, output, and so forth are at 100%. PulseAudio has a habit of silencing things and not telling you. You may also want to disable System Sounds unless you like random noises in your cast. The Microphone setting is trial and error depending on your microphone's output levels.

19. Modified versions of guvcview, pavucontrol and pavumeter are included both in source and executable form. These are used by the system. The modifications reduce the screen footprint and give the modules names that can be easily accessed and controlled by window manager code.

20. You MUST disable audio cards that will not be in use, for example, the audio on your web cam. If you do not, the system may attach to the wrong input devices. You should also check that your audio devices are not muted and that Built-in Audio Analog Stereo (or similar) is the default input device. Run pavucontrol (PulseAudio Volume Control at least once before using the system to establish configuration defaults. This program can also be used to disable any extra audio devices.

Most issues involve audio. PulseAudio is especially devious and will try, whenever possible, to default to the wrong setting. Please check the sections on audio in this document.

Note also: hot plugging audio devices - inserting or removing a plug while the system is running - generally results in Pulse resetting defaults to values you won't want. Don't hot plug devices but if you do, reboot.

21. Video Decode and Presentation API for Unix (VDPAU) is required if you have a supported graphics processor. If your graphics card is supported, be sure to install its drivers. This enables some graphics processing to be offloaded to the GPU and thus reduce CPU load. The software is found in libvdpau_r600.so (or similar). You need to install this manually with Synaptic.
22. If you want to use the slide show you MUST update the file address of the first slide to display in `automation.config` (SLIDES_DIR). Other slides will be selected randomly from the same directory.

### 1.1.1 Supported Operating Systems

The system works with Debian based distros including:

- Linux Mint 18.3 Sylvia
- Linux Mint 19 Tara with Mate
- Linux Mint 19 Tara with Cinnamon
- Linux Ubuntu 18.04 Bionic Beaver

Some manipulation of the screen may be necessary to make available the full 1920x1080. There are, at present, no Red Hat based distribution installation scripts.

### 1.2 Required Hardware and Software

#### 1.2.1 Operating System Requirements

This system has been developed and tested on Linux Mint 18.3 with Mate. It also appears to work with Linux Mint 18.3 Cinnamon and Linux Mint 18.3 XFCE with appropriate adjustments to the themes and windows (Mint X theme and Metacity + Compositing).

Since the system makes use of window manager facilities, it may not work correctly on other versions of Linux with other window managers. It does not appear to work with Ubuntu 18.04 but may with other Linux distributions with a GTK 3+ based window manager. If not, the display will not be correctly organized and buttons will overlap on another.

#### 1.2.2 Hardware Requirements

Due to the large number of threads and processes required to effectively run this system, at least a quad-core machine is desirable.

Overall hardware load can be minimized, if needed, by not running screen savers, slide shows, or cameras. Using Panel Mode also reduces video compressor requirements.

The main system requires a screen size of 1920 by 1080. It is possible to run with a smaller screen but without the automation system.

#### 1.2.3 Window Manager

This software is configured to run with Linux Mint 18.3 with Mate and similar systems. With appropriate adjustments, it will work with Cinnamon and XFCE. Due to differences in GTK usage, the software may or may not work correctly on other distributions of Linux.

For proper operation, it depends on certain window manager features. If these are not
present, all or part may not operate.

The following is a list of some Mint 18.3 Mate window manager options. They may be set in:

Preferences | Look & Feel | Desktop Settings | Windows

when using the Mint Custom Menu Bar.

The following options do NOT work:

- Metacity
- Marco

A Compositing manager is required for some features.

If you use the above options, although you will see video on your screen, the video will not encode correctly when you use ffmpeg.

The following options DO work:

- Marco + Compositing
- Marco + Compton
- Metacity + Compositing
- Metacity + Compton
- Compiz

Please adjust your settings accordingly.

### 1.2.4 Other Running Windows

The system makes use of wmctrl to control window visibility. This program makes use of window names to control window behavior. You can see a list of currently running windows on your system by running the command:

```
wmctrl -l
```

When you run this system, you need to close all other windows first as the names or portions of names of some of these may conflict with or duplicate that names of windows that wmctrl will control.

### 1.2.5 Graphics Support

Generally speaking, performance will be improved if your Linux system has access to a supported graphics card. Generally speaking, NVIDIA based cards are well supported by Linux but driver installation may be required.

The video rendering software will make use of the graphics card GPU by means of VDPAU (Video Decode and Presentation API for UNIX) package. This option (libvdpau1) should be installed if you have active, compatible, video drivers.
1.2.6 Hardware Timing

The system makes use of several standard Linux programs including image display programs, players, screen savers and camera display software as well as detached processes, shared memory, and multiple threading.

Once initialized, the visual windows associated with these standard programs are repositioned and resized, as needed, into the console area. As each of these programs takes a finite amount of time to load an initialize, the system has built-in delays between the time when the service is requested (for example, a player) and the time when it is repositioned and resized. You will notice these delays at times.

The built-in delays may be too large or too small for your particular system. That is, your system may be much faster than the one on which the code was developed and the delays are unnecessarily long.

On the other hand, the delays may be too short for your system. When this happens, the signals to reposition may arrive before the program has fully initialized and is accepting commands.

At present, with the exception of the cameras, the delays are coded in the Automation.c. It is expected that these will be changeable from automation.config in the future.

As noted, the load and initialization times for some features will depend upon the speed of your hardware and other loads concurrently running on your machine. Among these are:

1. Audio latency. Depending on the speed of your CPU and the number of processors, the audio may or may not be correctly aligned with video. This is due to the different processing for audio and video. There are settings in stream.script and VLC to compensate for this. Also note, the audio channels on some videos available on the Internet are not aligned correctly so compensation may not be advisable.

2. Camera initiation. When starting a camera, a delay is taken while the camera initializes. If the camera does not fully initialize, subsequent control of the camera may not be successful. This may be remedied by increasing (or decreasing in the case of fast cameras) the delay in the file automation.config.

3. Other windows. The system console incorporates windows that are created and maintained by other programs. As with cameras, there is a delay built in when these are started to give them time to fully initialize. These delays may need adjustment depending on your system.

Also, the basic internal clock for the system operates on a one second interval. As a result, some activities may not synchronize until the next clock cycle. The effect of these is minimal but may be visible.

Bottom line, if your hardware is slow, it may be necessary to increase the built-in delays to
libcompensate.

1.3 Goals

Internet casting has been growing in popularity in recent years. While there are many commercial, and some open source, products available in the Windows ecosystem to support these activities, there are relatively few in Linux.

For actual casting, OBS (Open Broadcaster Software) is perhaps the best known. It runs on both Windows and Linux and has many sophisticated features to support casting and recording applications. Unfortunately, OBS requires considerable CPU resources, has complicated settings that do not always produce the best broadcast image for a given bit rate, and has more features than the typical user may need. Also, OBS is mainly a casting/recording tool. It lacks broadcast automation and playback facilities. These must be provided by other packages such as Mixxx, VLC, smPlayer, and so forth.

The attempt here is to develop an open source system that makes use of existing Linux resources, where available, that is compatible with OBS but has a simpler, less resource intensive profile, suitable for lower power machines, with support for:

1. high resolution full motion (24 fps) video
2. low streaming bit rates
3. built-in streaming encoder
4. broadcast and multi-media automation support.

The system makes use of existing software packages where ever possible. In some cases (pavumeter, pavucontrol, and guvcview), slightly modified versions of these standard packages are included in the distribution.

Existing software packages are started as independent processes. Since most of these take a period of time to initialize (for example guvcview can take several seconds to initialize a camera, depending on the camera), you will notice delays at times when first starting a service. These are due to built-in sleep commands that pause until the service has begun. The duration of these can be altered and, in some cases, depending on your hardware, may need to be increased.

1.4 Basic Design

The package consists of:

1. a control panel to schedule and automate the playing of local and Internet accessible media files;
2. an ffmpeg based encoder capable of sending a 24 frame per second (fps) full motion video to an RTMP (Real Time Messaging Protocol) media streaming server at a high
level of compression;
3. PulseAudio loop-back configuration to merge and control audio send to the encoder.
4. Slide show, screen saver, camera, browser, and YouTube video controls.

1.5 Performance
In Linux, the most widely used audio and video casting system is OBS. OBS offers a number of features for both recording and casting but these can come at a price on lower power machines with limited graphics processing support, as is the case in Linux.

In OBS, multiple sources of video, audio, slides, video capture and so forth are organized into scenes. OBS captures the contents of these and allows the user arrange them into a window the contents of which are encoded and sent to the streaming server. Multiple scenes with differing layouts are possible. OBS supports a wide range of options for resolution frame rates, encoding and so forth.

A problem that arises, however, is that, on Linux systems, where video graphic card support in minimal, the real time processing to handle this can be overwhelming to even high end machines. Even simple casts can require more CPU power than many systems provide, especially if high compression rates are needed. Full motion video is often difficult to attain.

The concept used here is simpler. A fixed area of the user’s display monitor is reserved as the casting area or canvas. Anything placed in this area (players, video sources, browsers, terminal windows, etc.) will be encoded and transmitted. Audio is obtained and controlled through the PulseAudio system (a JACK version will be available soon).

While this approach is not as flexible as OBS, it is far simpler and requires, in most cases, less than half the CPU time as OBS to cast the same content at the same frame rate but at a one quarter the bit rate.

The system is suitable for laptops and portable applications where OBS will not run. A very limited standalone version can be made to run on a Raspberry Pi. In this mode, the control panel is not used and the casting area is smaller. Consequently, the CPU load is reduced.

1.6 Workspace Switching
If you switch workspaces while casting, the audio from a player in another workspace will continue to be streamed although the visible casting area will switch to the new workspace. See below in section 2.6 for details on the size of the areas being cast.
2 The Automation Console

2.1 Overview

The system consists of a casting console that controls the audio and video presented to the screen.

You have three streaming options:

1. Panel Mode: 1324x750 resolution. The area streamed corresponds to the corresponding to the beach image shown in Figure 1. Panel Mode is engaged by clicking the Panel button. The text in the Panel button will turn red to indicate that Panel mode is active.

   In Panel Mode, the encoder will transmit only the contents of the area noted above. Video players, screen savers, slide shows, etc. will initially display only if they appear in this area. An example of a VLC video playing is shown in Figure 2. Note: the background picture is optional.

   In Panel Mode, you have the option of enlarging, minimizing or rearranging video players, slide shows, the camera feed and so forth as needed.

2. Full Screen Mode: 1920x1080 corresponding to an entire screen\(^2\). Full Screen Mode is enabled by clicking and highlighting (red) the Full Screen button. Clicking the Full Screen button turns off the Panel button if it was highlighted. Full Scree Mode is similar to Panel Mode except the entire desktop is streamed. This mode may be unsuitable for casts targeted at tablets or phones.

\(^2\) Screens with at least 1920x1080 are required at the moment. To use another resolution requires a modification of the source code.
Figure 1 Automation Console

Figure 2 Automation Console With VLC Video
2.2 Audio / Video Players

The VLC audio/video player is the default although others may be used. In these examples, the default VLC player is shown. VLC comes with a built-in equalizer and a built-in compressor as well as some other features that may make it a good choice. The VLC window is normally opened with a minimal interface. A control-H will turn the interface back on (if the window has focus). Normal VLC keyboard shortcuts will work if the window has focus.

2.2.1 Mixxx Player

The Mixxx player causes problems as it connects (as of this writing) directly with ALSA and this blocks Pulse Audio sources. Mixxx can be used with JACK, however.

2.3 Browser

The Chromium browser is used for YouTube videos. You should terminate any other instances of Chromium before starting the system as other instances may cause interference.

2.3.1 JACK and Browsers

Most browsers at present do not connect directly with JACK Audio.

2.4 Console Layout

Figure 1 shows the basic automation console. The panel mode casting area is the area encompassing the beach scene in the upper right. Figure 2 shows the console with a video playing in the casting area using VLC.

The active areas of the console are as follows:

2.4.1 Audio control areas for Loopback Modes 0 and 1

In these modes, two internal system volume meters (instances of pavumeter) and one instance of Pulse Volume Control (pavucontrol) will be shown as seen in Figure Error: Reference source not found. Pulse Volume Control is the larger box at the lower left while the instances of pavumeter are to its right.

In LOOPBACK modes 0 and one, audio is mixed internally before being presented to the encoder. No external mixer is required.

Pulse Volume Control can be used to monitor and set audio levels of the microphone and all internal audio sources (players).

The top meter, labeled L Mon and R Mon, shows the sound levels being presented to the encoder.

---

3 Choice of VLC or MPV is a compile-time option.
4 The versions of pavumeter and pavucontrol provided in the distribution have been modified slightly from the versions normally included in Linux in order to reduce the size of the associated visible windows.
The lower meter shows the sound levels for the microphone input channel (upper left) L Mic and R Mic (pavumeter).

The numbers appearing in the title area of both meters give an volume average level. A level of 1.00 is optimal.

Both the meters and Pulse Volume Control are separate windows that float above the console.

2.4.2 Audio control areas for Loopback Mode 2

In LOOPBACK mode 2, audio mixing is done by an external mixer. Only one instance of pavumeter is displayed as seen in Figure 2. This is the level being received by the Mic/Line In that is presented to the encoder.

When using LOOPBAK 2 mode (refer to Figure 3):

1. Disable audio cards not in use.

2. Connect the computer line out (socket 2 in the figure) (or speakers or headphones, depending on your hardware) to one of the mixer input channels (such as Line 4/5 in Figure 4).

3. Connect your microphone to the appropriate mixer input channel (usually the XLS connector shown in the upper left of Figure 4).

4. Connect the mixer main output to your computer’s audio Line In (socket 6 in the figure) or microphone in as necessary.

5. Optional: connect your mixer’s phones out to either your ear phones or studio monitor speakers.

To test, start the VLC player playing an audio file. Adjust the external mixer controls to send an appropriate level to your audio card. Generally, you want the lights on the mixer to just touch the 0 dB level and rarely exceed it. Adjust the gain and volume on the microphone channel as well.

Adjust the audio card's input volume (use pavucontrol) for proper levels. Test the microphone similarly. Some fiddling with the computer audio input setting will be required for optimum, distortion free sound.
2.5 Casting Console Buttons and Controls

2.5.1 File Select, Randomize, Text Entry, and Play

File Select

There are two File Select buttons. Each opens a standard Linux GTK File Chooser.
window whose initial directory, the first time you use it, is:

$HOME/Desktop/Broadcast.

From there, you may navigate to other directories.

In the File Chooser, you may select a video or audio file to be played. The name of file selected will appear in the adjacent text box (shown with the shadow designation *music file name*).

Once you have selected a file, subsequent usage of the File Select button will cause the File Chooser to open in the same directory from which the selection was made however you may navigate to other directories if you wish. The File Chooser pop-up is not visible in the gray Normal mode casting area but would be in Panel or Full Screen mode. An example of the pop-up File Chooser for a File Select button is shown in Figure 5. Selection of a file does not initiate play of the file.

![Figure 5 File Chooser Example](image)

**Randomize**

The Randomize buttons will select a random media file from the directory from which the last file was chosen by the corresponding adjacent File Select button and place the file name into the corresponding text box. The Randomize button does not initiate play. The media file will play when the corresponding Play button is clicked. Use of the Randomize button is meaningful only if a directory has been identified by prior use of a corresponding File Select button.

**Music File Text Entry Box**
The box may receive entries from the File Select button or the Randomize button. You may also past entries into this box. File references must be full system file references (path from root). You may also place URLs of videos if the videos are directly playable. They may *not* be YouTube URLs as these are not directly playable by VLC.

**Play**

The Play buttons cause the named file, or URL to play in a window in the gray area.

### 2.5.2 YouTube Videos

**YouTube URL, Text Entry Box and Play Button**

The button labeled *YouTube URL* inserts the main YouTube URL into the YouTube text entry box to its right. If you click the corresponding *Play* button, the top page of YouTube will appear in a small Chromium window in the casting area. You may search and play YouTube videos from here.

The YouTube text entry area is also where URLs pointing to YouTube videos may be pasted (but not typed).

The YouTube video whose URL is placed here will be played in the Chrome window that will appear in the gray casting area when the adjacent *Play* button is clicked (see Figure 6).

**YouTube Button**

The YouTube button in the center turns on or off the display of the Chrome window containing a YouTube video. When the window is not visible, it will be in the tray at the bottom of the screen. See Figure 7 which also shows other video elements in the tray.
2.5.3 Full Directory Select

Full Directory Select Button

Clicking the Full Directory Select button will bring up a directory chooser box with which you can navigate and select a directory.
Play

The Play button causes the music/video contents of the selected directory to play in random order. If the window has focus, the keystroke ‘n’ will advance to the next randomly selected file\(^5\).

A directory containing a DVD commercially formatted movie may be selected.

2.5.4 Playlist Select, Shuffle and Play

Playlist Select

The Playlist Select button functions in a manner similar to File Select. It is used to select a standard system playlist file (.m3u) such as may be produced by VLC or smPlayer. The full playlist file address will appear in the adjacent text box or you may paste the full path information of a playlist. Only playlists may be used, not regular media files. Regular media files will not play from this entry box.

The only acceptable playlist formats are .m3u or .pls.

Shuffle

If you click the Shuffle button, the contents of the playlist file will be played in random order.

Play

If you click the Play button, the contents of the playlist will be played in consecutive order.

2.5.5 File Search

Search Box

The uppermost search box (identified by a magnifying glass icon), searches through the music library for files whose names satisfy the search expression entered into the search box. The search begins when the user hits enter. The search expression may be any regular expression accepted by grep. Up to 20 files found will be displayed. If you click on one, it's path information will be entered into the box labeled Search Results. The music library searched is:

$HOME/Desktop/Broadcast

Play

Clicking the adjacent Play button will cause the search results entry to play.

\(^5\) Note: if VLC has focus, all keyboard shortcuts are available.
2.5.6 Program Search

The second search box searches through entries in the file `program.lst` from which the 12 program buttons are populated. The search expression is any expression accepted by `grep`. A results box will appear and the program buttons will rearrange such that the selected entry will become the upper rightmost button.

2.5.7 Stream V and Stream I

The Stream V and Stream I buttons toggle the stream encoder on or off. When the encoder is on, the stream bit rate will be shown in the area labeled (at this moment) stream inactive. This buttons invoke the bash script file `stream.script` which you must edit to insert your streaming host’s URL and your casting key. The encoder will not function until you do so. As currently configured, Stream V turns on the section of code in `stream.script` presently configured for VaughnLive while Stream I turns on the section presently configured for iVlog. You may change these to target other streaming servers.

2.5.8 Panel Button

The Panel button tells the system that you will be casting the panel area (rather than full screen). If you are already casting (with `ffmpeg`), neither the Panel nor the Full Screen button will not respond if clicked. They are locked during streaming.

2.5.9 Full Screen Button

The Full Screen button tells the encoder that you will be casting the entire screen (1920 x 1080). It will not respond if you are already casting.

2.5.10 Program Buttons

The three columns of large buttons in the center of the control panel are the program buttons. Each button shows the name of a file read from the file `program.lst`. The file names are shown with path information omitted (the `program.lst` file must containing path information - see below).

Each file name that appears has been processed so that: (1) special characters removed, (2) the first character of each word following a blank capitalized, and (3) all other text reduced to lower case.

Optionally, the numbers in the parentheses following the file name are (1) the number of seconds of play time for the file and (2) how many times the file has been played during this session. Whether these timing numbers appear depend on settings presently in the code but eventually in `automation.config`. 
Left clicking on a button causes the named file to play.

Right clicking on a button causes the display of buttons to reorganize such that the button right-clicked moves into the first position (upper leftmost position).

2.5.11 Seq Program Button
The Seq Program button will begin to play the media files whose names appear in the program buttons. The files will be played in the order (sequential) in which they appear. The first file to be played will be the one designated by the upper leftmost button. As each file is played, the buttons shift. The currently playing file is always in the upper left button. A media file may be moved to the first (upper left) button by right clicking on it. The files whose names appear in the buttons are loaded from program.lst which may be organized into a program play list.

2.5.12 Rand Program Button
The Rand Program will randomly select media from the program.lst file. The program buttons will be rearranged such that the currently playing media file is in the upper left button.

2.5.13 Stop Players Button
The Stop Players button halts program play and closes the player.

2.5.14 Pause Button
The Pause button will cause the currently playing sequential or random program to pause at the end of the current file and wait until the button is clicked again. This allows for announcements between files.

2.5.15 Prog <<
The Prog<< button returns the program buttons to the beginning of the program.lst file.

2.5.16 Next Button
The Next button, when in Seq Program or Rand Program mode, causes play of the current media file to end the current file and to advance to the next program file.

2.5.17 Skip Button
The Skip button works only in random program mode. It causes the system to skip the currently next scheduled selection and chose another. It can take up to 1 second before an new selection appears.
2.5.18 Prog < Button
This button shifts the program button page one page to the left.

2.5.19 Prog > Button
This button advances the program button page one page to the right.

2.5.20 Show Saver Button
Saver displays the screen saver in the casting area. The screensavers are assumed to be in /usr/lib/xscreensaver/ and the selection of which saver is made in automation.config.

2.5.21 Show Slides Button
Initiates the slide show. The slide show is derived from the directory and image specified in the automation.config parameter SLIDES_DIR. The value for SLIDES_DIR must point to an image. Subsequent images will be randomly chosen from the same directory once you initiate the slide show (right click on the first slide displayed and select slide show).

The images should be edited to be no wider than 640 pixels.

2.5.22 Show Video Button
Causes currently running VLC player to appear in the casting area.

2.5.23 Cam0 / Cam1 Buttons
Toggles camera visibility.

The program guvcview should be run before using any camera as it sets configuration values in $HOME/.config). The cameras are assumed to be named /dev/video0 and /dev/video1.

This button invokes a modified version of guvcview (provided in the distribution, see bellow).

Setting your cameras to a lower resolution will result in a lower transmitted bit rate. The frame rate should be adjusted for best performance. A rate of 25 FPS seems best in most cases. However, since camera performance varies widely, your results may be different.

Due to variations in hardware speeds, the delay factor for the camera may need to be adjusted. When the system starts a camera, it waits until the camera has activated and is visible on the screen before moving to camera to it’s assign location. If the camera does not position itself correctly, this is a sign that the delay factor needs to be
increased. This can be set in `automation.config`.

You may start additional cameras manually before running the casting system. If there is any difficulty, try the command:

```
guvcview --gui=gtk3
```

On some systems, attaching two cameras to the same USB hub can cause issues. You may want to first check the cameras with `gvcview` to see if they are operating correctly before you try to access them from the casting system. When the cameras start, it is normal for error messages to appear in the terminal window from which the casting system was launched. Most of these can be ignored but, in some cases, they may indicate problems.

### 2.5.24 YouTube Button

Toggles the visibility of a YouTube window.

### 2.5.25 V Lock

The V Lock button locks the player to a size and location on the screen so that, when the player is restarted (as it is in Random and Sequential mode), it will return to the same location. Some features (scenes, for example) automatically set this option.

If you want the player to appear in a place and size of your own choosing, drag and resize the player then click this button. The player will return to the place you designated each time it is restarted or until a new video lock is established.

### 2.5.26 Casting Area Background Radio Buttons

The radio buttons on the left side of the screen casting area control the still card display in the casting area. The options include test patterns, color bars, Linux logo, and various backgrounds.

For those buttons named `img1` through `img9`, the system will look for files named `img1.png`, `img2.png`, ... in the subdirectory `backgrounds`. You may replace these with your own images.

Background image files must be scaled to 1323 by 748 or they will not fit.

### 2.5.27 No Deco / Deco Buttons

The No Deco and Deco buttons turn on and off window decorations in dependent windows. The decorations usually consist of the window title bar and frame. When a window has no frame or title bar, you may not drag or resize it. This is illustrated in Figures 8 and 9.
2.5.28 Scenes

There are (at present) six scene buttons. These automatically arrange contents in the casting area. The layout of each scene is controlled by scripts in `automation.config`.

See Figure 10 gives an example showing a layout consisting of the screen saver, two
cameras and the video player.

The script that created Figure 10 is the following:

```plaintext
SCENE 0 cam0 600 400 526 300
SCENE 0 cam1 1450 400 526 300
SCENE 0 saver 600 1 500 300
SCENE 0 player 1450 1 500 300
SCENE 0 startPlayer 2
```

Each line identifies the scene button (zero in this case), the resource (cam1, cam2, saver and player), the horizontal and vertical coordinates of the upper left corner of each window, and the width and height of the window. Two other resources categories not shown are `utube` and `slides`. They function in the same manner as the cameras and saver. In the case of `utube`, however, the player must already be running before a scene button associated with its display is clicked.

Thus, cam0’s upper left corner is 600 pixels from the left edge of the screen and 400 pixels from the top edge. The camera window is 526 pixels wide and 300 pixels tall.

The final line, `startPlayer`, tells the scene to start, if not already running, the VLC player in random program mode (2 means random, 1 means sequential and 0 means do not start the player).

Each scene button can be similarly configured and there are examples in the distribution `automation.config` file.
2.5.29 Chyron Buttons
The chyron buttons cause text of images to be superimposed on the casting area. Each image must be a .png file with a transparent background. The images are stored in sub-directories of Automation named chyron1, chyron2, chyron3, ... chyron6. The images found in the directories will be played as a slide show.

2.5.30 Clr Chyrons Button
The Clr Chyrons button turns off all visible chyrons.

2.5.31 Clr Scrn Button
The Clr Scrn button clears all content from the casting area. Players, savers, cams, and so forth are parked in the tray at the bottom of the screen. Any background is removed.

2.5.32 Program Buttons
The 12 visible program buttons contain entries taken from the file program.lst which is loaded when the system is started.

A left click on one of these buttons will cause the file referenced to play. A right click will cause the buttons to shift such that the button right-clicked will appear in the first position (upper leftmost).

The Prog< and Prog> buttons will advance (or retreat) to the next twelve entries.

If the Seq Program is clicked, the entries referenced in the program buttons will play one after another (sequentially) beginning with the entry in the upper left button (see note above about right-clicking a button).

The maximum number of program entries is determined by the compile time defined symbol SOUNDER_MAX. The default value is 4096 and is set in Automation.c

2.5.33 Network Bit Rates
Below the Show Video button there is a line that updates every three seconds showing your total system network activity in kilo bits per second. This is obtained from a background script named speedmon.script that writes the files NetTmp1 and NetTmp2 every three seconds. This is total system network activity, not just that due to the casting encoder. The script speedmon.script queries the first active network interface. You may need to alter the script on machines where network activity is handled by an alternative network interface.

2.6 Casting Modes
There are two casting modes corresponding to two resolutions. In each mode, if you
switch to a different workspace, the area of the new workspace that corresponds to the casting area from the previous workspace is the area transmitted. There is no loss of audio due to switching workspaces as can be the case in OBS.

### 2.6.1 Panel Resolution Mode

In panel cast mode, the entire panel will be transmitted when `ffmpeg` is on. The resolution of the cast is 1324 wide by 750 high. In panel mode, the console window will retreat to the upper left of the screen and attempts to move it will fail.

In panel mode, typical bit rates at 24 frames per second are around 800 k bits per second.

### 2.6.2 Full Screen Resolution Mode

In full screen mode, the entire desktop is transmitted when `ffmpeg` is on. The resolution of the cast is 1920 wide by 1080 high. If your screen is too small for these dimensions, do not use this mode or, recompile `Automation.c` with the dimensions of your screen. In this mode, you can move anything anywhere.

In full screen mode, bit rates at 24 fps are typically about 1200 k bits/second.

### 2.7 Screen Saver Mode

It is not uncommon in casts, when there is no meaningful video available, for example, when an audio only file is playing, to show a still card, and animation or a pre-formatted slide show.

The screen saver mode used here consists of a display of animated patterns taken from the `xscreensaver` package which is part of many Linux distributions. There are many others. Selection of which screensaver to use is set in `automation.config`.

The current default screen saver is `morph3d`. The full list of available screen savers is visible in the directory:

```
/usr/lib/xscreensaver
```

and a sample of these can be seen in:

```
System | Preferences | Look and Feel | Screensaver
```

When selecting a screen saver, you should check the amount of video it generates and the amount of CPU time it uses. Some other savers that might be of interest are `flyingtoasters`, `deco`, `fuzzyflakes`, `gears`, and `glnake`.

Generally speaking, screen savers generate a great deal of video and thus may increase bandwidth. They should be used sparingly.
2.8 Slide Show Mode

The slide show is a display of images that part of the cast. The slide show presenter is Phototonic.

You should run and set Phototonic’s preferences before you use it here. Ideally, slide show should be set to random, time should be set to 10 seconds per slide, and not to display file name in the slide.

The slide show directory is set in automation.config and points to the first slide. Other slides will be displayed from the same directory.

Due to the way Phototonics works, an initial image is required.

So, you should ensure that the entry in automation.config points to an initial image. In the distribution, the entry must be changed prior to using the slide show (the entry points to a file on my machine).

3 Casting a Presentation

It is possible to cast a presentation to a remote audience by way of a streaming server using presentation slides, images, audio and video recordings.

3.1 Converting a Slide Show to Images

If you prepare a presentation using LibreOffice Impress or a similar product such as Microsoft’s PowerPoint and you save the presentation as a PDF, a program in the distribution named ImpressConvert-bin will convert the PDF to slides that can be shown by the Automation system. Here are the steps:

1. Create your slide show using LibreOffice Impress or a similar product.
2. Export the presentation as a PDF.
3. Create a directory and copy or move the PDF to the directory.
4. Copy to the directory the file ConvertImpress-bin.
5. In a caja (or similar file explorer) displaying the contents of the directory, set the properties of the directory such that the files will be displayed in the order created with oldest files first and newest files last.

Since the .png files are created in the order in which they appear in the PDF, it is important that the directory be ordered by creation date/time in order that the slides will appear in the directory in the order that they appeared in the original presentation.
Thus, when you cut/paste the file names from the directory into the text editor (see below), they will appear in the text editor in the correct order.

You will probably want to set the directory display to Compact.

6. Open a terminal window in the directory and enter:

   ConvertImpress-bin pdfname

   where pdfname is the name of the PDF file without the file extension.

7. The program will take some time to run depending on the size of the presentation.
   When it is finished, in addition to the PDF file and ConvertImpress-bin, the directory will include a set of .png files whose names will be the titles of the presentation slides.

8. In the caja or similar explorer, select the .png files and paste their names into the text editor and save the result. This file will be your program.lst file. You may edit it to add additional material including references to audio, video or additional images (.png or .jpg).

The distribution has a sub-directory named MumpsTutorial that contains a PDF that you may experiment with.

When you have finished editing your presentation file, copy it to program.lst in the main Automation directory (making a backup as needed).

### 3.2 Presentation Example

Using the PDF in MumpsTutorial, these are the steps to create the program.lst file for a presentation.

First, copy the PDF and ConvertImpress-bin file to the directory in which the slides will be created and set the directories properties such that items will be arranged by modification date (which in this case is creation date). This is shown in Figure 11.

---

6 Note: your slide titles should not contain characters that would be invalid in a file name.
Next, open a terminal window in the directory and start the conversion process. Note that the name provided to `ConvertImpress-bin` is the name of the PDF with no filetype extension. If you elect to include a filetype extension, this program will not work.

Figure 12 shows the terminal window. The program can take quite a long time to run. You will notice files being created in the background `caja` window. Initially, these are sequentially numbered but at the end, their names will be changed to the slide titles from which they were generated. Your presentation should not contain characters in slide titles that would be incompatible with file names (backslash, for example).
When `ConvertImpress-bin` finishes it will generate a numbered list of the file names it created as seen in Figure 13. You will note in the background `caja` window that these files are now present in the directory. There will also be a file with the same name as the PDF but with the `.txt` extension. This file contains the text and titles of the slides and is used to rename the slides. Each slide title except the first is preceded by a control-L character.

Figure 13 `ConvertImpress-bin` Finished

Figure 14 shows part of the list of files generated by `MumpsTutorial.pdf`. 

36
In Figure 15 we have highlighted (selected) the .png files. One way this can be done by a left click on the first (leftmost) file and then, holding down the SHIFT key, right clicking on the last (rightmost). This will highlight or select all the intermediate files as well as the ones clicked. Be careful not to select the PDF, the .txt file or ConvertImpress-bin.

Once you have selected the files, right click on one of them and select Copy (or you can use control-C or the Edit menu).
Next, open the text editor and right click in it and select *Paste* (or you could use control-V or the Edit menu) as shown in Figure 16. This will cause the file names with directory information to be pasted into the editor as shown in Figure 17.

You may save this file. If you want to run is as a presentation, save it as *program.lst* in the *Automation* directory.

You may also edit the file to add additional lines referencing images or media files as shown for media only casts.
Figure 16 Paste .png Files into Text Editor Window
Figure 17 File References in Text Editor

Figure 18 shows the result when displayed in the Automation console. You should resize the player (VLC) as needed and you may start cameras as needed. Remember to use V Lock to cause the Automation console to remember where you have placed the player.

If you engage Seq Program, the console will cycle through the slides in sequential order. You must click the Next button to advance. After about 10 minutes, the slide will advance to the next (if you were in Seq Program mode) or disappear if you started it standalone from one of the program buttons.

You may cast the entire screen or only the upper right rectangle as discussed above. In either case, casting bit rates for static slides are very low.
4 Encoder and stream.script

The builtin encoder is ffmpeg. It is invoked from the bash script file stream.script. It is normally invoked by Automation-bin.

4.1 Standalone Use

The script file stream.script can be invoked directly. If invoked directly, you should provide five command line parameters:

stream.script vert-offset hor-offset widthxheight stream loopback

where:

vert-offset is the vertical offset of the upper left hand corner of the casting area from the top edge of the screen.

hor-offset is the horizontal offset of the upper left hand corner of the casting area from the left hand side of the screen.

widthxheight (example: 640x380) is the width and height of the casting area in pixels. Note there must be a letter x between the two numbers and no blanks are permitted. The numbers should be even.
stream is used to select the correct server and casting key.

loopback determines which audio mode to use (0, 1, or 2).

When you invoke stream.script directly, the area of the screen you select with the parameters will be cast. If you use LOOPBACK 0 mode, you must first run the script loopback.script. This script is not required for LOOPBACK 1 and LOOPBACK 2 modes.

You should select even values for screen height, width and offsets. The encoder may reject odd values.

Standalone mode is appropriate if you have a smaller screen size or limited CPU resources. In standalone mode, the automation system is not operational.

5 Installation

5.1 Desktop Theme

GTK desktop themes influence the manner in which widgets (buttons, labels, lists, etc.) are displayed. The widgets in this project are tightly grouped to save screen space. If the default theme expands the padding or spacing of these, they may overlap or otherwise not display correctly. The theme Mint X is the default at present with Linux Mint 19 with MATE. Other themes should work as well but you should check their effects on not only this software but others as well.

You can set the theme by going to:

System | Preferences | Look and Feel | Appearance

In the pop-up box, select Customize, and under Controls, select Mint X or similar.

Usage of incompatible control themes may cause erratic layouts.

5.2 Installation

The distribution comes with binary executables which are compatible with Linux Mint Mate 19 and related Ubuntu based systems (see section 1.1.1) Unzip the distribution onto your Desktop directory. Do not attempt to install it elsewhere. It will not work.

The installation script files consist of several apt-get commands that, if executed as root, install the needed code if it is not already present. Others compile code.

The following is the order in which to execute the installation scripts:

```
sudo ./install.script
sudo ./install-camera.script
sudo ./build-camera.script
sudo ./build-pavucontrol.script
```
sudo ./build-pavumeter.script
./compileAutomation.script

1. Create a directory on your Desktop named Broadcast
2. Put some video or audio files into this directory (or subdirectories of same).
3. Make your PATH contain the current directory. Add the following, at the end, to .bashrc

   PATH=$PATH:$HOME/bin:./bin:/sbin:

Close & reopen your terminal window for the above to take effect.

See Section 6.4 for information on what directories are required and their contents.

5.2.1 Pulse Audio
Selection of which Pulse audio mode is made by the variable LOOPBACK which can be set in automation.config.

1. Using the Pulse loopback.script file (LOOPBACK 0). In this approach, the microphone input and the player outputs are routed and merged into a dummy internal sink named mywiretap.monitor which becomes the input for ffmpeg. This is discussed in section Error: Reference source not found.

2. Using ffmpeg itself to merge the microphone and player sources into the output stream (LOOPBACK 1). This works but can create issues with regard to synchronizing the audio with a player and/or camera video. The lines in script.stream that do this are:

   -thread_queue_size 1024 -f pulse -i $SINK -thread_queue_size 1024 \n   -f pulse -i $SOURCE -itsoffset $DELAY -thread_queue_size 32768 -filter_complex amerge \n
3. Using an external mixer to merge the mic input and player output into a stream which is fed back in through to line in / mic in on your audio card (LOOPBACK 2). In this mode, disable all audio cards except your main card. Then:

   a) Connect your microphone to the appropriate input on your mixer.
   b) Connect the Line Out or Phones out of your audio card to one of the input channels on your mixer.
   c) Connect the output of the mixer to your line in in on your audio card. Set the Input

---

7 .bashrc is in your home directory.
in pavucontrol (aka PulseAudio Volume Control) to Line In and unamplified.\(^8\)

d) In LOOPBACK 2 mode, the volume meter optimal value is about 0.900.

When you play an audio file on your machine, the output will go to your mixer and then into the microphone input and then be captured by ffmpeg. You may need to adjust your system audio settings (see pavucontrol also known as PulseAudio Volume Control). Some fiddling may be required.

In LOOPBACK 2 only one meter - the mic input meter - will be displayed as this is the only audio stream. In LOOPBACK 2 mode, the volume meter optimal value is about 0.900.

5.2.2 JACK Audio

All audio is merged by means of Jack connections and the final result is send to ffmpeg. This section will be expanded at a later date as it is probably not the one you want to start with. The BASH code, however, is present in stream.script.

5.2.3 VLC

If you are running Linux Mint LMDE, you will need to manually install VLC. It is not included in the base installation.

There are several VLC settings you should pre-set before using the system:

2. Allow Only One Instance.
3. Allow metadata network access: no.

6 Running the Console and Casting

6.1 Environment

You should close other windows you may have running on your system. Possible window name overlaps may result in the console misplacing dependent windows.

For example, if you have an instance of Chrome or Chromium running, these may interfere with the instances that may be started by the console.

6.2 Set Up program.lst

Replace the contents of the file program.lst with your music file names. This is the file that will

---

\(^8\) Microphone In may also be used.
be used to load the buttons with music. The file must consist of full file references, one per line, for the music you want to have appear in the buttons. For example:

/home/you/Desktop/Broadcast/RockMusic1/processed.full/VenturaHighway.mp4

The above should be an absolute file address on your system where you is your user id. You can get a list of absolute addresses for a directory of media files on your system by the following:

1. Open a file explorer window to show the file names in compact format (see the edit preferences option).
2. In the window displaying the files in compact format, type ^a (control-a) followed by ^c (control-c). This means all, and copy.
3. Open the text editor (xed) from Applications | Accessories
4. In the text editor window, type ^v (paste). The edit window will now contain the file names. Remove any references to non-media files.
5. Save this as program.lst in the Automation directory. The files in this list, in the order they appear, will be shown in the playlist buttons of the control console.

If you successfully build program.lst, you will now see the file names listed in the buttons. Click on one and it should begin to play. If there is no program.lst file, the buttons will be empty.

The maximum number of entries in program.lst is determined by the compile time defined symbol SOURDER_MAX which is set to 4096 in the distribution.

Typical entries in my program.lst look like:

/home/okane/Desktop/Broadcast/RockMusic1/processed.full/Bob Lind - Elusive Butterfly.mp4
/home/okane/Desktop/Broadcast/RockMusic1/processed.full/Bob Seger - Against the Wind.mp4
/home/okane/Desktop/Broadcast/RockMusic1/processed.full/Bob Seger - Night Moves.mp4
/home/okane/Desktop/Broadcast/RockMusic1/processed.full/Bon Jovi - Its My Life.mp4

Figure 19 program.lst

6.3 Server URL and Casting Key

To begin a cast, first modify the SERVER and KEY variables in stream.script to point to ivlog or vaughnlive or what ever other service you intend to use that supports the rtmp (Real Time Messaging Protocol). This information should be available at the streaming server host site.
6.4 Directories
The system has been configured to assume that certain files will be located in fixed locations. In future versions, some of these will be relaxed but, for the moment, these locations are required.

1. The system assumes that the environment variable $HOME contains the location of your home directory (this is normally the case in most Linux distributions).

2. The system assumes that $HOME/Desktop/Broadcast is the directory in which your media files are located. The directory may contain sub-directories. Symbolic links are permitted but the search function does not follow them at present.

3. The system assumes that $HOME/Desktop/Automation is the address of the directory containing the systems software.

4. The slide show default directory and initial slide should be set in automation.config. The slide show requires an initial image file by the name of image.jpg to be located in this directory.

6.5 Executing the Console
You initiate the system with the command:\n
./runAutomation.script

This will cause the main console to initiate and start the floating windows.

As the system initiates, several small windows will open, and, ultimately, reposition themselves on the screen. This can take a few seconds as there are delays built into the startup procedure to allow these windows to initialize. On some slower systems, it is possible (but not likely) that the built-in delays may not be sufficient. If this is the case, they should be increased in the main code module (Automation.c).

You will also see messages in the terminal window that initiated the system. Some will be minor error messages and this is normally not a problem. Many will be from other software packages such as Phototonics and guvcview. If it crashes, however, there were problems.

Once the console has finished initialization, you may initiate casting by clicking the Stream on/off toggle button. This will cause the script file stream.script to execute and attempt to make contact with the server.

When the stream is running, do not move the main window as the streaming encoder has locked onto the casting area. If the casting window is moved, the encoder may no longer see its contents.

---
9 Note: the ./ is not required if you included the current directory in PATH as shown above.
When contact is established, the STANDBY icon will change to ON-AIR.
You may begin playing media files at any time.
You may terminate the console (and all floating windows) by clicking the X box in the upper right corner of the main window.
When the encoder is running, you may switch to another desktop workspace. The sound channels will not be affected (as is sometimes the case in OBS). Contents within the casting area on the new workspace will be encoded and cast. Thus, you may have more than one scene available for casting.

6.6 Selecting a Casting Mode
You need to select a mode and resolution for your cast. The default mode is set in automation.config (see section 8.1). The default will be either Panel Mode or Full screen mode. If Panel Mode is the default, the Panel Mode button will be highlighted. See section 2.6 for details.

6.7 Enabling Slides, Saver, and Camera
Because they take time to initialize, if you intend to use slide shows, the camera and/or the screen saver, you should initiate them before you begin your cast. See section 6.4 for details on slide shows. The determination as to which screen saver will be used depends on a setting in automation.config (see section 8.1).
Each option, savers, slides and camera, even if not visible, will add to overall CPU load. The camera and screen saver use the most CPU time while the slide show (Phototonic) uses the least. You may terminate any of these during your cast by right-clicking on the associated button in the system launcher panel and selecting close.

6.8 Starting the Stream
Click the appropriate Stream button to start the stream. Clicking it again halts the stream. Once the stream has begun, the console may not be moved nor may you switch mode (panel or full screen).

6.9 Playing Audio and/or Video Files
There are several ways in which to play a file:

6.9.1 The Rand Program Button
The Rand Program button will randomly select and play files found in the program list buttons (loaded from program.lst). It will continue doing so until you click the Stop Players button.
If you click the Pause button, the player will pause between selections and wait for you to click Pause again.

If you want to skip to the next randomly selected file, click the Next button.

When in random program mode, the selection to be played after the current selection will appear in a line just beneath the background images radio buttons.

Note: in Playlist or Full Directory modes, the Next button does not work. In those cases, you give the player focus and type the letter n.

6.9.2 The Seq Program Button
The Seq Program button will cause the files in the program buttons to play in consecutive order beginning with the file listed in the upper leftmost button. A file moves to the upper leftmost button by right clicking on its button.

If you click the Pause button, the player will pause between selections and wait for you to click Pause again.

If you want to skip to the next file, click the Next button.

Note: in Playlist or Full Directory modes, the Next button does not work. In those cases, you give the player focus and type the letter n.

6.9.3 File Select / Randomize / Play Buttons
There are two rows beginning with File Select buttons. The are used to play individually selected files. The File Select button allows you to navigate to a file to be played. The Play button plays this file. The Randomize button will randomly select another file from the same directory as the file selected by the File Select button (do not use it until you have selected at least one file and, thus, a directory).

6.9.4 Full Directory Select / Play
This button is used to identify a directory and the Play button will cause the files in the directory to be randomly selected and played until the Stop Players button is clicked.

In this mode, the Next and Pause buttons do not work. To advance to the next file, give the player focus and type the letter n. You can give the player focus by clicking on its header bar or by clicking on its button in the launcher panel. Do not click in the play area as this will pause the player.

6.9.5 Playlist Select / Shuffle / Play Buttons
The Playlist Select button is used to locate a m3u playlist file. The Shuffle button will cause the contents of this file to be randomly played. The Play button will cause the contents to be played consecutively. Play will end when you click the Stop Players button. In this mode, the
Next and Pause buttons are not functional. Use the n keystroke as noted above.

6.9.6 Insert Youtube URL /Play Buttons
The Insert Youtube URL button will insert into the adjacent text box the main URL for Youtube. Alternatively, you may paste the URL of a specific Youtube video. Clinking the Play button brings up an app version of Chromium that will display and play the URL selected. If you use the main Youtube URL, you will need to use the Youtube search function to locate the video you want to play. Play will end when the Stop Players button is clicked. Next and Pause are not operative in this mode.

6.9.7 File Search and Play Button
The File search box will search the Broadcast directory (and sub directories) for files whose names match the search criteria. The list of located files will be presented. Selecting one will cause it’s name to appear in the adjacent play box and clicking Play will cause it to play. The search criteria are not case sensitive and may include only a fragment of the name of the file. Full grep compatible regular expressions are also accepted.

6.9.8 Program Entry Search
The program entry search box will search the files named in program.lst and cause the buttons to rearrange such that the file found will be in the upper leftmost box. Clicking the button containg the name will cause it to play.

6.9.9 Program Buttons
If you click on a program button, its file will play and its button will move to the upper leftmost position. You may navigate through the program buttons with the Prog < and Prog > buttons. Note: the first time a page of buttons is displayed, the system queries each file on the page for play time. This can result in a slight delay. Once the time have been loaded, they will not be loaded again for the duration of the current session.

7 GTK Configuration

7.1 Overview
The main console display shown in the examples is produced by the Gnome GTK+ toolkit. The widgets produced by GTK interact with the GTK window theme and settings on your desktop to render a visible result. The package was developed with Linux Mint 19 with Mate using the default theme and font settings. It appears to work well with Mint XFCE and Cinnamon.

However, due to differences in themes, window managers, and default fonts and font sizes
used in other distributions, some tuning may be needed.

There are three ways to tune the visual display:

1. Adjust your window manager theme and font.
2. Modify the file `Automation.glade`
3. Modify the file `automation.css` (requires re-compilation). In particular, change the value of:

\[
\text{font-size: 10px;}
\]

...to higher or lower and restart.

It is anticipated that there will be a configuration script that will automatically adjust these settings for the major Linux distributions in the near future. At present, however, Linux Mint 18.3 is the best option.

### 7.2 Automation.glade

The layout of the buttons and boxes is controlled by the file `Automation.glade` which is built and maintained by the `glade` tool (installed by `install.script`).

Within limits, adjustments to the appearance of the console may be made by altering this file, saving the result, and, in some cases, recompiling the code (usually not needed if only `Automation.glade` is changed). The Glade tool displays `Automation.glade` as shown in Figure 20.

### 7.3 automation.css

The file `automation.css` contains code that overrides some of the default CSS settings used by Glade and GTK. These include setting the size of the buttons (default buttons are usually much larger), color of the button text, and the size of the button font.

On some versions of Linux, the default font size is either too large or too small. You may compensate for this in `Automation.c`

Other distribution differences, such as added window panels (Ubuntu, for example), may require modification of settings in `Automation.c` which control the placement of dependent windows in the console.

If you modify this file, you need to recompile `Automation.c`

Do not mistake this file for `automation.config`. They have different functions.
8 Options

For the most part, options at present are set in Automation.c as variables and defined symbols.

Options are also set in stream.script and loopback.script.

8.1 automation.config

The file automation.config contains run-time configuration options that are user settable without recompilation.

The file format is very specific and sensitive to errors. The format rules are:

1. A comment begins with a pound sign (#) in column 1. The remainder of the line is ignored.

2. A parameter line consists of:
   (a) a parameter name beginning in column one.
   (b) the name must be typed exactly - it is case sensitive - with no embedded blanks
   (c) one (1) blank
   (d) the value (numeric or string)
If an option appears more than once in *automation.config*, the last instance is the one used. The version of *automation.config* provided, for example purposes, has alternative several duplicate settings. Only the last is used.

The following lists the options available at present and their default values.

### 8.1.1 Default View

panelView 0

If zero (0), the console will begin in Full Screen Mode. If one (1), it will begin in Panel Mode. No other values are accepted.

### 8.1.2 Camera Delay

Different cameras take different amounts of time to initialize. A camera must initialize and begin sending a video stream before the casting system can manipulate it. The delay options are the number of seconds that the casting system will wait until a camera initializes. These numbers will be specific to your machine and camera(s)

CAM0_DELAY 2

The number of seconds to wait until the camera initializes. At present, only on camera is supported (CAM0)

CAM1_DELAY 5

### 8.1.3 Screensaver

SS_LIB /usr/lib/xscreensaver/flyingtoasters
SS_NAME flyingtoasters

The location (SS_LIB) and name (SS_NAME) of the screen saver.

### 8.1.4 Number of Program Buttons

Number of program buttons. This number must be larger than the number of lines in the file *program.lst* or not all lines will be loaded.

SOUNDER_MAX 4096

### 8.1.5 Play Time Load

Whether to load the play times of entries in *program.lst*. Loading the play time can be slow as each file musty be interrogated. There is also a compile time option to decide if the play times should be loaded (*PLAY_TIME_LATE*) when the program starts (long delay for large *program.lst* files) or when the buttons are loaded.

PLAY_TIME_LOAD 0
### 8.1.6 Scene Configuration

The scene buttons can be configured with script lines of the form:

```
SCENE 0 cam0 600 400 526 300
```

The number following the `SCENE` keyword is the button being configured (0 through 5) followed by the resource or option, followed by the coodinated or the upper left corner of the window and its width and height. For example,

```
SCENE 0 cam0 HOR VER WIDE HIGH
```

where HOR is the number of pixels from the left edge of the screen of the window’s upper left corner; VER is the number of pixels from the top edge of the screen of the window’s upper left corner; WIDE is the width of the window in pixels; and HIGH is the height of the window in pixels.

The possible resources are: `background`, `chyron`, `cam0`, `cam1`, `saver`, `player`, `slides` and `utube` where player refers to the VLC player. The resource `utube` refers to browser app window displaying a YouTube page. The YouTube page must be already running when the button referencing it is clicked.

The code:

```
SCENE 0 startPlayer 2
```

Tells the button (scene zero in this case) to start the VLC player if not already running and to begin in random program mode. The number 1 means beginning in sequential mode and the number 0 means do not start the player.

The following are some examples. Numbers are in pixels. Horizontal and vertical are the number of pixels from the left edge or top edge of the screen, respectively, of the upper left corner of the object.

```
SCENE 0 cam0 600 400 526 300
```

Start `cam0` at horizontal offset of 600, vertical offset 400, width 526 and height 300.

```
SCENE 0 saver 600 1 526 300
```

Start the screen saver at 600 horizontal, 1 vertical with a width of 526 and a height of 300.

```
SCENE 0 player 1450 1 526 300
```

Place the VLC player at horizontal 1450, vertical 1 with a width of 526 and a height of 300.

```
SCENE 0 chyron1 file:///home/....globe.gif 1100 250 0 0
```

Start the chyron1 at horizontal 1100, vertical 250, width 0 and height 0.
Start the image *global.gif* as *chyron 1* at horizontal 1100 and vertical 200. File reference points to actual file or a directory. If a directory, the *.png* files will be displayed in alphabetic order at 5 second intervals.

**SCENE 0 startPlayer 2**

Start the player (if not already running) in random play mode.

**SCENE 0 background 7**

Turn on background number 7. Backgrounds are numbered consecutively beginning at zero (blank). A list is in the comments in *automation.config*.

**SCENE 0 slides 1000 150 500 300**

Start the slide show at horizontal 1000, vertical 150, width 500 and height 300.

**SCENE 5 utube 100 100 500 400**

Start the YouTube player (chromium) at horizontal 1000, vertical 150, width 500 and height 400.

### 8.1.7 Loopback Modes

1. LOOPBACK 0 means use the PulseAudio loopback option for merging the audio streams.
2. LOOPBACK 1 means use *ffmpeg* to merge the audio streams (default).
3. LOOPBACK 2 means that you have an external mixer and that the audio streams will be mixed in the mixer and the result returned to the system through LINE IN or Microphone In. This is the most efficient option and gives you the most control over the audio stream. In LOOPBACK 2 mode, the volume meter optimal value is about 0.900.

### 8.1.8 SLIDES_DIR

SLIDES_DIR specifies the file address of the first slide to be displayed by the slide show. Additional slides will be taken from the same directory.

### 9 Animations

#### 9.1 Audio Animation

Audio files that do not contain video may be animated to show audio levels with *ffmpeg*. The program *ffmpeg* can be used to process an audio file and produce a video equivalent containing the same audio track but with a video display showing the frequency and sound levels. The following are several examples:
9.1.1 Showwaves

![Figure 21 Showwaves Audio Conversion](image)

```bash
ffmpeg -i input.mp3 -filter_complex \\
"[0:a]showwaves=s=1920x1080:mode=line:rate=25,format=yuv420p[v]"\\
-map "[v]" -map 0:a showwaves.mp4
```
9.1.2 Showfreqs

```bash
ffmpeg -i input.mp3 -filter_complex \
"[0:a]showfreqs=s=1920x1080:mode=line:fscale=log,format=yuv420p[v]\n-map "[v]" -map 0:a showfreqs.mp4
```
9.1.3 Avectorscope

Figure 23 Avectorscope Audio Conversion

```bash
ffmpeg -i input.mp3 -filter_complex \
"[0:a]avectorscope=s=1920x1080,format=yuv420p[v]" -map "[v]" \ 
-map 0:a avectorscope.mp4
```

9.1.4 Showspectrum 1
Figure 24 Showspectrum 1 Audio Conversion

```bash
ffmpeg -i input.mp3 -filter_complex \
"[0:a]showspectrum=s=1920x1080,format=yuv420p[v]" \ 
-map "[v]" -map 0:a showspectrum.mp4
```
9.1.5 Showspectrum 2

Figure 25 Showspectrum 2 Audio Conversion

```bash
ffmpeg -i song.mp3 -filter_complex \
  showspectrum=mode=separate:color=intensity:slide=1:scale=cbrt -y \
  -acodec copy video.mp4
```
9.1.6 A Little of Everything

Figure 26 Multiple Format Audio Conversion

```
ffmpeg -i input.mp3 -filter_complex \
  "[0:a]avectorscope=s=640x518,pad=1280:720[vs]; \ 
  [0:a]showspectrum=mode=separate:color=intensity:\ 
  scale=cbrt:s=640x518[ss]; [0:a]showwaves=s=1280x202:mode=line[sw]; \ 
  [vs][ss]overlay=w[bg]; \ 
  [bg][sw]overlay=0:H- \ 
  h,drawtext=fontfile=/usr/share/fonts/TTF/Vera.ttf:\ 
  fontcolor=white:x=10:y=10:text='"Song Title"' \ 
  by Artist'[out]" -map '[out]' -map 0:a -c:v libx264 \ 
  -preset fast -crf 18 -c:a copy output.mkv
```
10 FAQ

10.1 No Sound
Check that one or more of the volume controls has not been set to silent. If you, for example, set the output of `mpv` or `vlc` to silent, it will remain at silent until you change it - even if you reboot!

10.2 No Sound, Yet Again
Check that sound cards not being used are turned off (see above) and reboot. The file `loopback.script` attempts to determine which card is your main audio card and sometimes gets confused.

10.3 No Sound, Even More
Check your Sound Preferences to see if you have Input set to Built-in Audio Analog Stereo (the usual case for most systems) as shown in Figure 27.

![Sound Preferences](image)

Figure 27 Sound Preferences
10.4 File Names

Rarely, media file names have a blank character appended at the end of the name. If this occurs (it is rare), the file name will not be recognized as accessible by the `access()` Linux function. Any files that do not load into the program should be checked to see if they contain blanks at the end of their names.
<table>
<thead>
<tr>
<th>Channel</th>
<th>Duration</th>
</tr>
</thead>
<tbody>
<tr>
<td>OBS</td>
<td>14p.</td>
</tr>
<tr>
<td>Open Broadcaster Software</td>
<td>14</td>
</tr>
<tr>
<td>Options</td>
<td>51</td>
</tr>
<tr>
<td>Oracle’s Virtual Box</td>
<td>9</td>
</tr>
<tr>
<td>Other windows</td>
<td>13</td>
</tr>
<tr>
<td>Panel Button</td>
<td>25</td>
</tr>
<tr>
<td>Panel Mode</td>
<td>11, 47</td>
</tr>
<tr>
<td>Panel Resolution Mode</td>
<td>32</td>
</tr>
<tr>
<td>panelView</td>
<td>52</td>
</tr>
<tr>
<td>PATH</td>
<td>43</td>
</tr>
<tr>
<td>Pause</td>
<td>48</td>
</tr>
<tr>
<td>Pause Button</td>
<td>26</td>
</tr>
<tr>
<td>pavucontrol</td>
<td>10, 18, 44</td>
</tr>
<tr>
<td>pavucontrol</td>
<td>18</td>
</tr>
<tr>
<td>PDF permissions</td>
<td>33p.</td>
</tr>
<tr>
<td>Phototonic</td>
<td>33</td>
</tr>
<tr>
<td>Phototonics</td>
<td>46</td>
</tr>
<tr>
<td>Play</td>
<td>24</td>
</tr>
<tr>
<td>Play</td>
<td>20</td>
</tr>
<tr>
<td>Play Time Load</td>
<td>52</td>
</tr>
<tr>
<td>PLAY_TIME_LOAD</td>
<td>52</td>
</tr>
<tr>
<td>PLAY_TIME_LATE</td>
<td>52</td>
</tr>
<tr>
<td>player</td>
<td>53</td>
</tr>
<tr>
<td>Playing Audio and/or Video Files</td>
<td>47</td>
</tr>
<tr>
<td>playlist formats</td>
<td>24</td>
</tr>
<tr>
<td>Playlist Select</td>
<td>24</td>
</tr>
<tr>
<td>Playlist Select / Shuffle / Play Buttons</td>
<td>48</td>
</tr>
<tr>
<td>Prog &lt; Button</td>
<td>49</td>
</tr>
<tr>
<td>Prog &lt; Button</td>
<td>27</td>
</tr>
<tr>
<td>Prog &lt;= Button</td>
<td>26</td>
</tr>
<tr>
<td>Prog &gt; Button</td>
<td>49</td>
</tr>
<tr>
<td>Prog &gt; Button</td>
<td>27</td>
</tr>
<tr>
<td>Program Buttons</td>
<td>25, 31, 49, 52</td>
</tr>
<tr>
<td>Program Entry Search</td>
<td>49</td>
</tr>
<tr>
<td>Program Search</td>
<td>25</td>
</tr>
<tr>
<td>program.lst</td>
<td>9, 25p., 34, 44p., 47</td>
</tr>
<tr>
<td>Pulse Audio</td>
<td>43</td>
</tr>
<tr>
<td>Pulse Volume Control</td>
<td>18</td>
</tr>
<tr>
<td>PulseAudio</td>
<td>10, 15</td>
</tr>
<tr>
<td>R Mic</td>
<td>19</td>
</tr>
<tr>
<td>R Mon</td>
<td>18</td>
</tr>
<tr>
<td>radio buttons</td>
<td>28</td>
</tr>
<tr>
<td>Rand Program Button</td>
<td>26, 47</td>
</tr>
<tr>
<td>random program mode</td>
<td>26</td>
</tr>
<tr>
<td>Randomize</td>
<td>20p.</td>
</tr>
<tr>
<td>Raspberry Pi</td>
<td>15</td>
</tr>
<tr>
<td>Real Time Messaging Protocol</td>
<td>14, 45</td>
</tr>
<tr>
<td>Right click</td>
<td>26</td>
</tr>
<tr>
<td>rtmp</td>
<td>45</td>
</tr>
<tr>
<td>RTMP</td>
<td>14</td>
</tr>
</tbody>
</table>
XFCE.......................................................................................................................... 11, 49
xscreensaver.................................................................................................................... 32
YouTube.......................................................................................................................... 53
YouTube Button.............................................................................................................. 22, 28
YouTube URL.................................................................................................................. 22
Youtube URL button......................................................................................................... 49
YouTube Videos.............................................................................................................. 22
PulseAudio Volume Control............................................................................................. 44
/dev/video0..................................................................................................................... 27
/dev/video1..................................................................................................................... 27
/usr/lib/xscreensaver.................................................................................................. 32
/usr/lib/xscreensaver/.................................................................................................. 27
$HOME........................................................................................................................... 46
$HOME/Desktop/Automation......................................................................................... 46
$HOME/Desktop/Broadcast............................................................................................ 21, 24, 46
$SINK............................................................................................................................... 43
$SOURCE........................................................................................................................ 43