

![GStreamer Logo](https://dreambox.guru/eh2018/gstreamer-logo.svg "GStreamer Logo") #
Programming Workshop Commandlines

fraxinas@schaffenburg.org (<mailto:fraxinas@schaffenburg.org>)

MRMCD 2018 Darmstadt

prerequisites

It helps if you've watched Gstreamer Einmaleins @ Easterhegg 2018

- https://pads.schaffenburg.org/p/GstreamerEinmaleins#/

You'll need

- gcc (build-essentials)
 - git
 - and your favourite code editor
-

installation

Arch / Antergos / Manjaro:

```
pacman -S gstreamer gst-plugins-{base,good,bad,ugly} gst-libav
```

Ubuntu or Debian:

```
apt-get install libgstreamer1.0-0 gstreamer1.0-plugins-base gstreamer1.0-  
plugins-good gstreamer1.0-plugins-bad gstreamer1.0-plugins-ugly gstreamer1.0-  
libav gstreamer1.0-doc gstreamer1.0-tools
```

Fedora:

```
dnf install gstreamer1-devel gstreamer1-plugins-base-tools gstreamer1-devel-  
docs gstreamer1-plugins-base-devel gstreamer1-plugins-base-devel-docs  
gstreamer1-plugins-good gstreamer1-plugins-good-extras gstreamer1-plugins-ugly  
gstreamer1-plugins-ugly-devel-docs gstreamer1-plugins-bad-free gstreamer1-  
plugins-bad-free-devel gstreamer1-plugins-bad-free-extras
```

helpful tools

- devhelp is a very helpful glib API documentation browser
- A Pixelflut Server for testing:

```
sudo pacman -S libevent glew glfw-x11 or  
sudo apt-get install build-essential libevent-dev libglew-dev libglfw3-  
dev  
git clone git@github.com:defnull/pixelflut.git  
cd pixelflut/pixelnuke && make  
./pixelnuke
```

- netcat for testing or mimicking a pixelflut server

Setting up the workdir

```
git clone https://github.com/fraxinas/gst-pixelflut.git
cd gst-pixelflut
```

step_00 Boilerplate

```
git checkout step_00
autoreconf --force --install
./configure
make
```

```
gst-inspect-1.0 pixelflut
export GST_PLUGIN_PATH=$PWD
gst-inspect-1.0 pixelflut
```

step_01 DEBUG Category

```
GST_DEBUG=:DEBUG gst-inspect-1.0 pixelflutsink
git checkout step_01 && make
GST_DEBUG=pixel*:DEBUG gst-inspect-1.0 pixelflutsink
```

step_02 PAD Templates

```
gst-launch-1.0 fakesrc ! pixelflutsink
git checkout step_02 && make
gst-launch-1.0 fakesrc ! pixelflutsink
```

step_03 Properties

```
git checkout step_03 && make
gst-inspect-1.0 pixelflutsink
```

step_04 Finalize, Start, Stop

```
git checkout step_04 && make
GST_DEBUG=pixel*:DEBUG gst-launch-1.0 pixelflutsink
```

step_05 Render function

```
git checkout step_05 && make
GST_DEBUG=pixel*:LOG gst-launch-1.0 audiotestsrc ! pixelflutsink
```

step_06 Pad Templates and set_caps function

```
git checkout step_06 && make
GST_DEBUG=GST_CAPS:LOG gst-launch-1.0 audiotestsrc ! pixelflutsink
GST_DEBUG=pixel*:LOG gst-launch-1.0 videotestsrc ! pixelflutsink
```

step_07 Accessing the pixel data

```
git checkout step_07 && make
GST_DEBUG=pixel*:TRACE gst-launch-1.0 videotestsrc ! pixelflutsink
```

step_08 Socket Operation

you may need explicitly specify host=\$IP property from here on...

```
git checkout step_08 && make
export PXFHOST=127.0.0.1
GST_DEBUG=pixel*:DEBUG gst-launch-1.0 videotestsrc ! pixelflutsink
host=$PXFHOST
```

step_09 More properties and clever sending

```
git checkout step_09 && make
gst-launch-1.0 videotestsrc ! pixelflutsink host=$PXFHOST offset-top=360
offset-left=480
```

step_10 Canvas size and alpha transparency

```
git checkout step_10 && make
gst-launch-1.0 videotestsrc pattern=pinwheel ! video/x-raw, width=800,
height=600 ! pixelflutsink host=$PXFHOST
gst-launch-1.0 videotestsrc ! alpha method=green ! pixelflutsink host=$PXFHOST
```

Test application

```
git checkout testapp
make && cd test
./gstpixelflutsinktest -h $PXFHOST
./gstpixelflutsinktest --uri
https://download.blender.org/peach/bigbuckbunny_movies/big_buck_bunny_480p_h264
.mov -i 20 -x 480 -h $PXFHOST
```

Hyperlinks

GLib

- <https://developer.gnome.org/glib/stable/>
- <https://developer.gnome.org/gobject/stable/>

GStreamer

- <https://gstreamer.freedesktop.org/documentation/>
- <https://gstreamer.freedesktop.org/documentation/plugin-development/basics/>
- <https://gstreamer.freedesktop.org/documentation/application-development/>
- <https://gstreamer.freedesktop.org/data/doc/gstreamer/head/gst-plugins-base/libs/html/>
- <https://developer.gnome.org/gobject/stable/howto-gobject-chainup.html>

GStreamer Workshop @ MRMCD 2018

- <https://pads.schaffenburg.org/p/GStreamerWorkshop#/>

GStreamer Einmaleins @ Easterhegg 2018

- <https://pads.schaffenburg.org/p/GStreamerEinmaleins#/>
-

Q & A

Write me

fraxinas@schaffenburg.org (<mailto:fraxinas@schaffenburg.org>)

fraxinas in #gstreamer on freenode irc
