



OpenGL Desktop/ES for the GStreamer pipeline

Matthew Waters (ystreet00)

GStreamer conference 2015

8 October 2015

Who Am I

- Australian
- Work - Centricular
- Graphics - OpenGL

Quick Introduction – OpenGL/ES

- OpenGL born from SGI in 1990's
- Cross-platform 3D API
 - X11, wayland, OS X, iOS, Android, Windows, Embedded Linux

Quick Introduction – GStreamer OpenGL/ES

- Minimum target OpenGL ES 2.0
 - Essentially the beginning of GLSL support
- Versions supported – OpenGL ES 2.0/3.x Desktop
2.x/3.x/4.x
- Platforms supported – Linux (X11 + Wayland), OS X, Windows, iOS, Android, Embedded Linux
- Various elements available – glimagesink, glcolorconvert, glvideomixer, gltransformation, gloverlay, glectures_*, etc

OpenGL Transfers

- GstGLUpload
 - Transfers from GstVideoGLTextureUploadMeta, EGLImage, SystemMemory into GLMemory
- GstGLDownload
 - Simply a caps transformation from GLMemory to SystemMemory
- GstGLColorConvert
 - Converts to/from YUV/RGBA
 - RGBA channel reordering

GStreamer OpenGL/ES - Elements

- Required integration is GstGLDisplay and GstGLContext
- Base classes GstGLBaseFilter, GstGLBaseMixer, GstGLFilter, GstGLMixer
- GstGLSyncMeta for synchronising OpenGL access

What's New?

- OpenGL 3 core profile
- Improved integration with existing OpenGL library/applications
- Sinks for gtk+, qml, CoreAnimation
- 3D video elements/library object – glviewconvert, glstereomix, glstereosplit

What's New?

- New subtitle/logo blending in OpenGL
- New transfer elements – glupload, gldownload + helper bins, glsrcbin, glfilterbin, glmixerbin, glsinkbin
- Improved GL context propagation
- PBO usage for texture transfers

What's Happening?

- Importing dmabuf fd's
- Shaders
- GL thread == streaming thread
 - Reduce the possibility of deadlocks and the requirement for intricate lock patterns
- GL pixel formats

Demo

- Shader
- Modelling

What Next?

- Vulkan?
 - Ease multi-threading?
- Generic OpenGL buffer based GstMemory?

Thanks!

ystreet00 in #gststreamer on freenode