

QEMU GL in Yocto

Zhai, Edwin

OpenGL(Open Graphics Library)

Multi-platform API for writing 3D application

250+ different function calls to draw complex 3D scenes from simple primitives

implementation

- Mesa : a open source implementation of OpenGL

- Proprietary driver : provided by graphics vendor

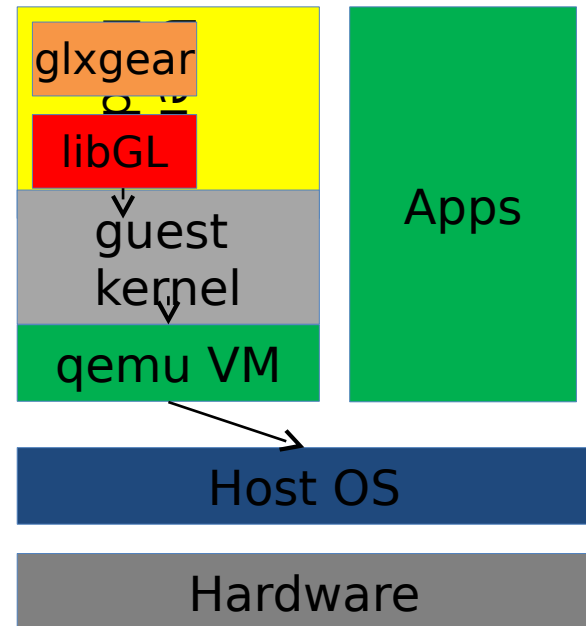
Mesa

```
$ ldd /usr/bin/glxinfo  
    libGL.so.1 => /usr/lib/mesa/libGL.so.1
```

```
glxgears
```

qemu in Yocto

Extra cost because of emulation
Poor performance for GL application
as pure software



qemu GL in Yocto

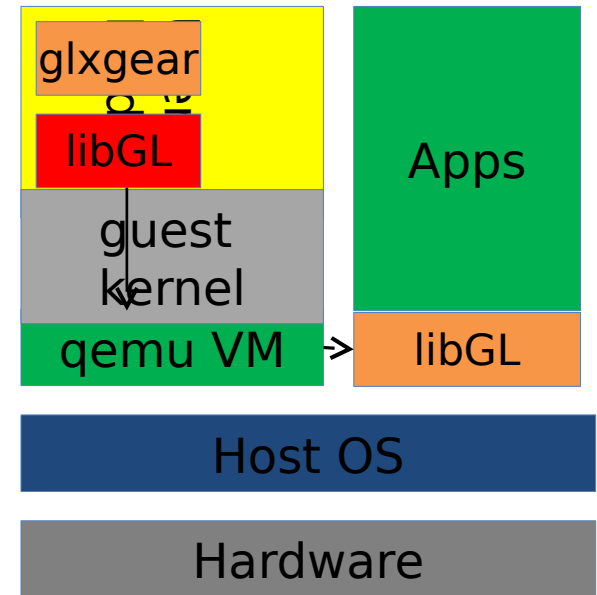
OpenGL API proxy

Guest libGL trap each GL call into qemu

u

Qemu forward this call to host libGL as a proxy

Qemu open a new host window and attach it to origin qemu window



Trap into qemu

Hacked libGL to take place the one in Mesa
mesa-dri_7.11.bb, qemugl_git.bb

How to

int 0x99 in call_opengl_qemu @ qemugl_git.bb

gen_interrupt() @ qemu_0.15.1.bb(native)

=> helper_opengl()

Parameter preparation

Register to pass parameters

Copy guest buffer into qemu directly

argcpy_target_to_host @ decode_call_int

GL capable qemu

Linked with host libGL

Parse each GL call from guest

Forward it to host libGL

*.patch in qemu_0.15.1.bb

qemu-git-qemugl-host.patch

opengl-sdl-fix.patch

opengl-args-copy-fix.patch

Test

mesa-demo as simple test
glxgears

Add more test in future
Sync with other implementation
Meego

Upstream GL status

Common bugs

Host environment issue

Ubuntu 10.04 ~ 11.10 works well

Bug 1885,1886,649,698,1206

Target related

GL proxy is only for qemux86/x86-64 based on mesa-dri

qemuarm/ppc/mips use mesa-xlib

Bug 2066

Guest/host gap

32/64 combination: bug 2221,688

GL extension gap: bug 1927,

Parameter preparation

Gcc optimization or code bugs

Bug 1442

Qemu graphics device model

Bug 111

Common bugs(cont.)

On real hardware

Bug 1840

Yocto install or link issue

Bug 228, 2309

Debug BKM

Debug switch in source code

DEBUG_GL=1 DEBUG_ARRAY_PTR=1 -- put more debug info in lib gl-qemu

WRITE_GL -- ask qemu to record all the GL calls in /tmp/debug_gl.bin, then can replay it with opengl_player.

USE_TCP_COMMUNICATION

GL_REMOVE_EXTENSIONS

gdb + print