Linux Graphics Meets the ARM Ecosystem

Jesse Barker <jesse.barker@linaro.org>

April 12, 2011

Find us here on the Linaro wiki
Overview

- The Linux Desktop
- The ARM Linux Desktop
- The Subset Approach
- Examples
- Questions
- What's Next
The Desktop
Window system

- Display management
- Resource management
- Session management
- Event handling
- Application programming interface
Protocol Decode

Device Independent X (DIX)

DRI

EXA

DDX

libdrm

Kernel space

User space

evdev

Input H/W

CPU

GPU

Memory

DRM

KMS

GEM/TTM

DC
Toolkits/Frameworks

• Create abstraction layer from the underlying window system.
• Provide uniform look-and-feel across platforms.
• Applications don't have to care which system they are running on.
• New backend to the framework adds a new supported platform for a whole bundle of applications.
Bells and Whistles

- OpenGL
- Video
- Audio
- Compositing window managers
- Animation
The ARM Desktop
What's the difference?

- Most differences are “physical”
  - Screen size and resolution
  - Unified memory pool
  - Power vs. raw performance
- Some API (not necessarily, though)
  - Window system interfaces
  - Rendering interfaces
The Subset Approach

- OpenGL ES 2.0 is explicitly defined as a subset of OpenGL 2.1.
- Both have diverged since the original definition.
- Minimize specialized code (e.g., window system interfaces).
The “big-ticket” items

- Immediate mode
- Fixed-function vertex processing
- Fixed-function fragment processing
- EGL vs. GLX
Examples

- glmark2
- cairo-gles
- compiz
glmark2

- Based upon opensource glmark by Ben Smith.
- Uses 3D Studio Max for model content.
- Uses SDL for window system abstraction.
- Only real differences between OpenGL and OpenGL ES is in initialization of SDL objects.
cairo

- Cairo is a 2D vector graphics library.
- Used by GTK, Webkit, Mozilla, and others.
- Leverages the desktop OpenGL backend.
- Highlights differences between OpenGL and OpenGL ES.
Compiz

- Popular compositing window manager
- Uses an extensible architecture to amend functionality.
- Core handles context and resource management as well as event handling.
- Plugins handle bells and whistles.
Questions?
What's next?

- Android
- Wayland
- ???
Thank You