Running with X11 on embedded platforms

The X11 implementation of Monocle

OpenJFX for embedded ARM platforms is built around the idea that it takes over the whole screen and accesses the screen's framebuffer directly using EGL. This is a lightweight way to render graphics for those devices and OS configurations that support framebuffer access through EGL. However, some configurations only support OpenGL rendering into an X11 window. For these configurations it is still possible to use OpenJFX, using the experimental X11 back-end.

In order to use OpenJFX on an ARM platform with X11, there are a few requirements:

- The DISPLAY environment variable needs to be set to the device's X11 display. The value for `DISPLAY` should typically be "`:0`".
- X11 needs to be running on the device
- EGL libraries needs to be present on the device, with support for using an X11 window as the `NativeWindowType`.

OpenJFX for ARM will attempt to use X11 for its display if it does not recognize the platform it is running on and if the `DISPLAY` environment variable is set. If OpenJFX fails to work with X11, it falls back to software rendering on the framebuffer device. The X11 back-end can be explicitly selected on the Java command-line with the parameter:

```
-Dmonocle.platform=X11
```

When running with the X11 back-end, OpenJFX creates a single X11 window that contains the whole of the OpenJFX window stack. Even if the JavaFX application creates multiple Stages, there are all shown in the same X11 window. The window by default takes up the whole screen, but the window can be restricted in size using the system property `x11.geometry`:

```
-Dx11.geometry=<x offset>,<y offset>[+<width>x<height>]
```

Input

Input is taken directly from Linux device drivers, just as when rendering directly to a framebuffer. Input from X11 events can also be selected, using the flag:

```
-Dx11.input=true
```

In this case only single-point touch support is available, and then only if the touch devices used is recognized by X11. Mouse input is also available, but keyboard input is not implemented for this configuration.

ARM MALI devices

When running Ubuntu on the ODROID U3 device using EGL with X11, there is a bug in the graphics drivers that causes `eglCreateWindowSurface` to fail when called with an X11 Display pointer with the sign bit set. There is a workaround for this in OpenJFX. The workaround is activated with following the command-line flag:

```
-Dmonocle.maliSignedStruct=true
```

OMAP3 devices with X11

The X11 implementation of Monocle does not show a hardware cursor. If a hardware cursor is needed on an OMAP device that does not provide EGL access to the framebuffer (for example, the PandaBoard ES) then the OMAPX11 port can be used:

```
-Dmonocle.platform=OMAPX11
```