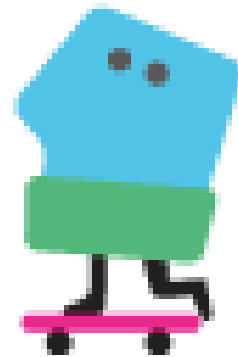


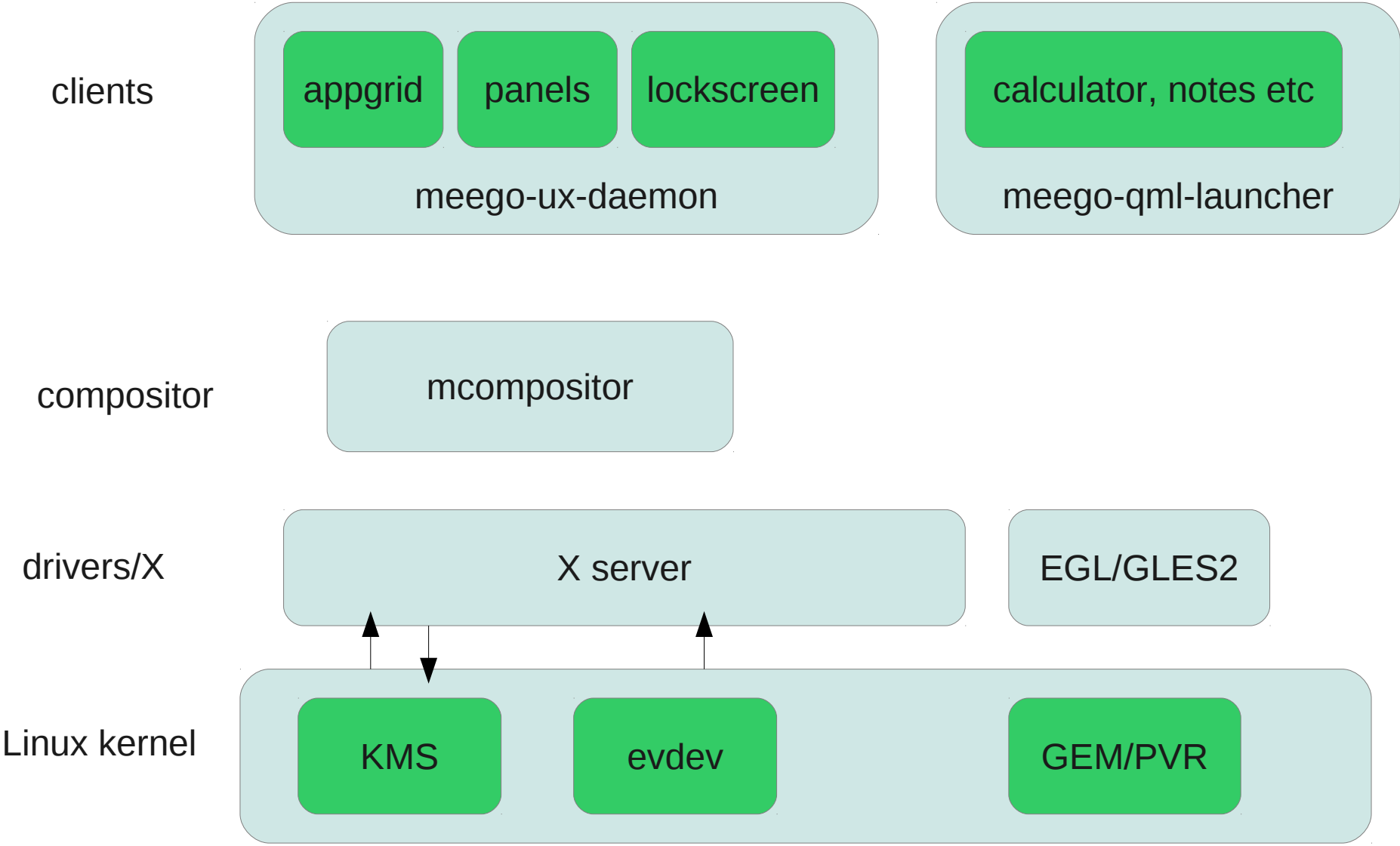
MeeGo Tablet UX on Wayland



What is Wayland?

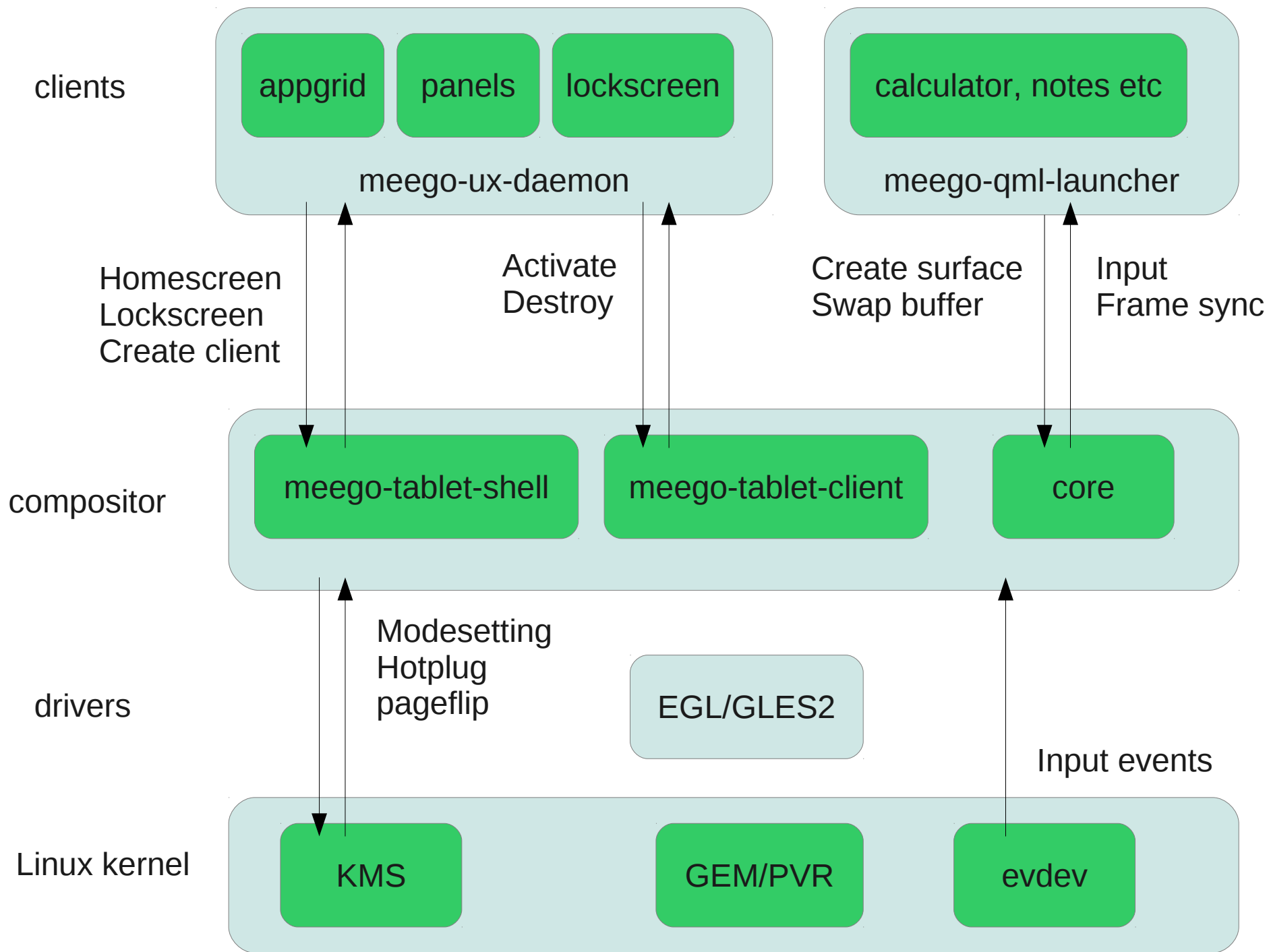
- New graphics system architecture
- A protocol between compositor and clients
- A sample implementation

Current X11 Architecture



New Architecture

- wayland-compositor replaces X and mcompositor
- meego-ux-daemon talks to compositor using a MeeGo specific Wayland interface
- Wayland compositor controls KMS, evdev, backlight, hotplug, idle timeout (screensaver)
- All QML unchanged except for switching to QtQuick 2.0



wayland-compositor

- It's the canonical sample compositor
- Written in C, directly to EGL/GLES2
- No high-level dependencies
- Simple 2D scenegraph, efficient repaint loop
- No overdraw, pageflips to client buffers
- MeeGo implemented in a loadable policy module
- Simple fullscreen animations
- Higher level UI in meego-ux-daemon

QML Scenegraph

- The initial bringup work uses QML Scenegraph
- QML SceneGraph implements QML, like QDeclarativeView, but straight to GLES
- Avoids the awkward QPainter abstraction that gets in the way of efficient repainting
- Scheduled for Qt 5
- We need to backport the current, mostly complete implementation or stay with QDV.

Timing

- Qt 4.8 to come out in “next couple of months”, provides “lighthouse” backend abstraction
- Lighthouse is what makes Qt on Wayland possible
- MeeGo release 1.3 in October, already branched
- We can switch now!
- Or wait for Qt 4.8

Driver Support

- Client side EGL platform (wayland-egl.h)
 - Buffer allocation and sharing
 - Implementation talks to server side interface provided by server side EGL platform
 - Trivial wsegl implementation
- Server side EGL platform
 - Buffer allocation (gbm)
 - KMS integration (pageflip to gbm buffer)
 - Work in progress...

Compatibility

- Wayland applications that are not MeeGo tablet apps
 - Native Qt, SDL, clutter, gtk+ applications
- X applications
 - Need to finish the rootless X server on Wayland project

Remaining Work

- Session management
- MeeGo graphics system type-of-workaround for PVR
- Polish and integration
- Enable other types of clients, SDL for games, X for legacy applications

Decisions

- Switch to Wayland for MeeGo 1.3?
 - When?
- Backport QML Scenegraph
 - Stick with QDeclarativeView?
 - Backport QtQuick 2.0?
- Policy for non-MeeGo QML apps?
 - Not supported?
 - Open up for MeeGo 1.4?