The pathway to Chromium on Wayland

WebEngines Hackfest
(Oct/2018)

Antonio Gomes, tonikitoo@
Maksim Sisov, msisov@
Jeongeun Kim, jkim@
Agenda

- History
- Desktop Integration
- Current Status
  - UI/GPU split
- Plans
- Demonstration
History

- Started by Intel off the trunk
  - Entered in ‘maintenance mode’.
  - The version was based on m49.
- Igalia started Wayland-support with Mus
  - Extended Mus and Ozone to support ‘External Window’ mode.
- Moved to AURA desktop integration without Mus
  - Comply with the plan from upstream.
  - Simpler design.
History - Design Comparison

Intel

Browser process

desktop integration

ozone/wayland

IPC (old API)

ozone platform
wayland connection

GPU process

w/ mus

Browser process

Desktop integration

aura/mus

IPC (Mojo API)

ozone/wayland
Wayland connection

UI Service

AURA Desktop Integration

Browser process

desktop integration

ozone/wayland
Wayland Connection

IPC (Mojo API)

GPU process
Aura Desktop Integration

- Moved the design without Mus.
  - Dropped the changes for Mus integration.
    - Dropped Mojo IPC for communicating with Mus.
  - Reused Wayland port implementation.
    - ui/ozone/platform/wayland
  - Figured out the changes we have to keep.
  - Re-implemented some changes.
- Continued to support Wayland.
Current Status (Upstream)

- Upstreamed patches allowed to run Ozone/Wayland from ToT (some functionality still has been missing). You can
  - Launch Chromium with `--ozone-platform=wayland`.
  - Do regular multi tabbed/windowed browsing
  - Open popup menus and tooltips.
  - **Have a separate GPU process.**
- Upstreamed ~89% of existing Ozone/Wayland patches.
  - Tracking with crbug.com/578890
- Upstreamed the GPU/Browser process split patches:
  - Including refactoring of the Ozone/Drm to share GBM related code with Ozone/Wayland
  - Upstreamed other related patches.
Current Status (Upstream)

- Window states
  - Maximize / Minimize / Fullscreen
- Session restore (including windows’ bounds and placement)
- Touch events support
- Fixed content_shell support
- Added menu/submenu windows support
- Maksim Sisov and Antonio Gomes are owners of Ozone/Wayland now.
Current Status (Downstream)

- **Window Interactions**
  - Move/Resize (under review for upstream)

- **Implemented additional features**
  - Drag and Drop: WIP
  - Copy and Paste: WIP

- **File Dialog**
  - Reused SelectFileDialogsExtension
Goal:

- Run a separate sandboxed GPU process,
- Be able to use GpuMemoryBuffer framework,
- Improve overall performance.
Design

- Uses Generic Buffer Management (GBM) with drm render nodes on the GPU process,
- Imports prime fds and sends them to the browser process,
- `zwp_linux_dmabuf` protocol is used to create `wl_buffers` based on the send dmabuf prime fds.
- `wl_surface_buffer_damage` is used to identify changed parts of the buffer.
- The communication between processes is done with Mojo.
How buffers are created
How buffers are swapped

[Diagram showing the process of buffer swapping in Wayland, with nodes and edges indicating the flow and interactions between different components.]
GPU/Browser process split

- Outcome:
  - ~10 FPS (~30 vs ~40 FPS) performance gain in the WebGL Aquarium demo on the Renesa R-car M3 board running with yocto image based on the meta-renesas’ sumo branch compared to --in-process-gpu mode.
  - Ability to use GpuMemoryBuffers framework and CPU_READ_WRITE* buffer usages,
  - Chromium security model is followed.
Plans

- Enable native gpu memory buffers support:
  - Better performance with css and skia draws. (under review)
- Improve the GPU/Browser process split:
  - Add missing support for presentation feedback. (DONE)
  - Add Vulkan support.
- Support Tab dragging.
- Upstream all the Ozone/Wayland patches + Ozone/X11 patches.
- Ensure there are no differences for end-users compared to stock Chromium.
Plans (cont.)

- Add PlatformScreen. (crbug.com/872339)
- Add ClipboardOzone support. (crbug.com/875168)
- Add multi-screen support.
- Support non-english keyboard layouts.
- Enable tests in a ToT buildbot.
  - unit_tests, browser_tests
- Integration with AGL
  - analyzing and fixing performance problems of Chromium in AGL.
  - Support Chromium as WebEngine for Web Runtime.
Demonstration

- Chromium ToT
- Chromium/Wayland Project from github
  - Linux Desktop
  - Renesas M3 board
Questions
	onikitoo@igalia.com - Antonio Gomes

msisov@igalia.com - Maksim Sisov

jkim@igalia.com - Jeongeun Kim