

# Chromium on Wayland - Status update

Automotive Grade Linux  
F2F meeting (04/2017)

Antonio Gomes & Silvia Cho

**RENESAS**



**igalia**

# Agenda

- Who is Igalia?
- Motivation
- Background
- Discussion

# Who is Igalia?

- Worker-owned, employee-run Open Source consultancy company, based in Spain.
  - ~55 employees around the world.



- Areas
  - Chromium/Blink, WebKit and Servo;
  - Compilers, JavaScript engines (V8, JSC);
  - Multimedia, Graphics (Mesa), Networking, Accessibility.

# Motivation

- Being able to run Chromium natively in Wayland-based systems will leverage its adoption in a variety of systems / environments.
  - Support from **Renesas/AGL**, **GENIVI** (automotive industry consortium for IVI), Bose, Bosch, Volvo, Jolla, Raspberry Pi, Tizen.
  - **Fedora 25** is shipping Wayland by default.
  - Major GUI Toolkits have built-in support, including Qt 5, Gtk+, Clutter, EFL.



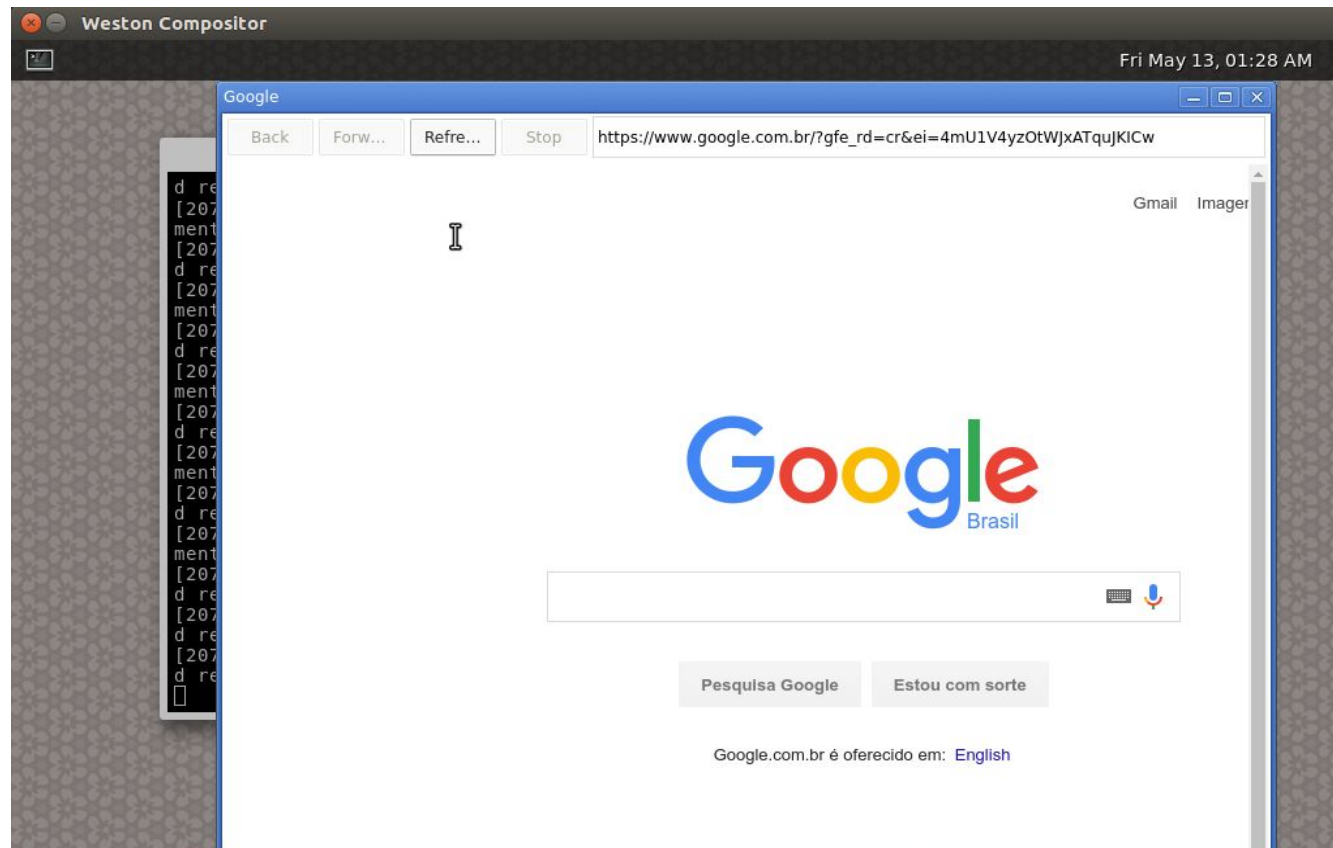
Background

# Background

- [Ozone](#) project
  - Set of C++ classes for abstracting different window systems on Linux.
  - It provides abstraction for the construction of accelerated surfaces underlying the *UI Service (Mus)*, as well as input devices assignment and event handling.
  - `//ui/ozone/`, `//ui/events/ozone/` and `//ui/base/cursor/ozone/`
- Backends:
  - DRM/GBM
  - x11
  - **wayland**
  - cast
  - headless

# Background

- May/16 – started experimenting with Chromium’s Ozone/Wayland.
  - Ported part of the code from [01.org](http://01.org) to Chromium ToT.



Internal “investment”

- `content_shell ozone/wayland`

# Background

- Igalia got in touch with Google/Chromium developers to understand the plans for `//ui/ozone/platforms/wayland`
  - figured about the *exosphere* project and ChromeOS plans for *mash*
    - `//components/exo/`
    - `//mash`
  - figured that the original “desktop integration” approach taken by 01.org did not comply with the way future Linux desktop Chrome was planned.

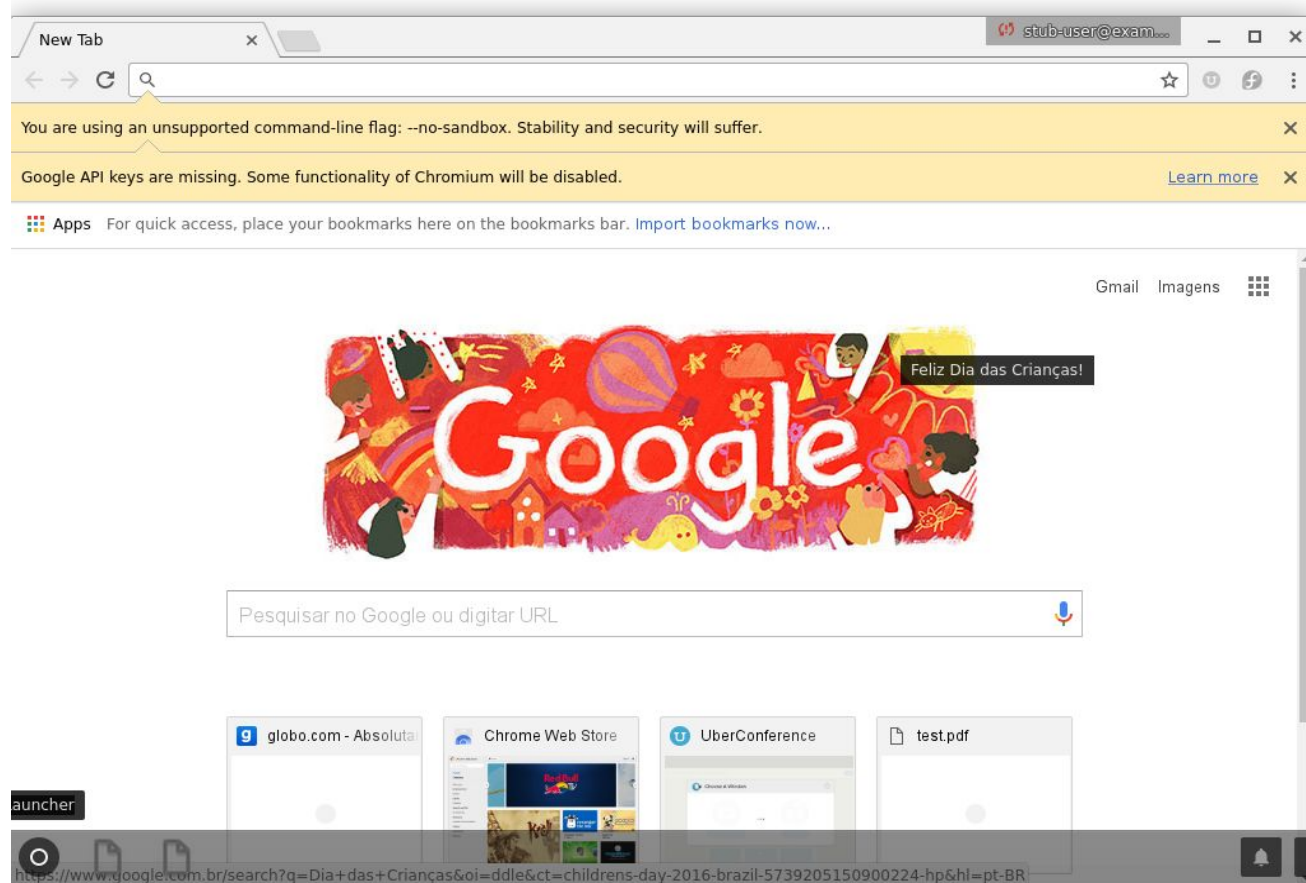


# Background

- Back in Sept/16 ...
  - Ozone/Wayland (by Intel / 01.org)
    - Off trunk.
    - In “maintenance mode” - m49 (december/2015).
    - Good community adoption.
  - Ozone/Wayland (Chromium ToT)
    - Partially upstreamed.
      - still behind in terms of functionality if compared against Intel’s implementation.
    - ChromeOS / mus+ash oriented.
    - Outdated documentation.
    - Limited buildbot coverage.

# Background

- Sept-Oct/16
  - Bringing up of Ozone/Wayland.
  - Start experimenting with “Ozone != ChromeOS”.
  - Design discussions with Robert Kroeger.
  - Try Mojo IPC.
  - [Buildbots](#)
  - [Documentation](#)



# Background



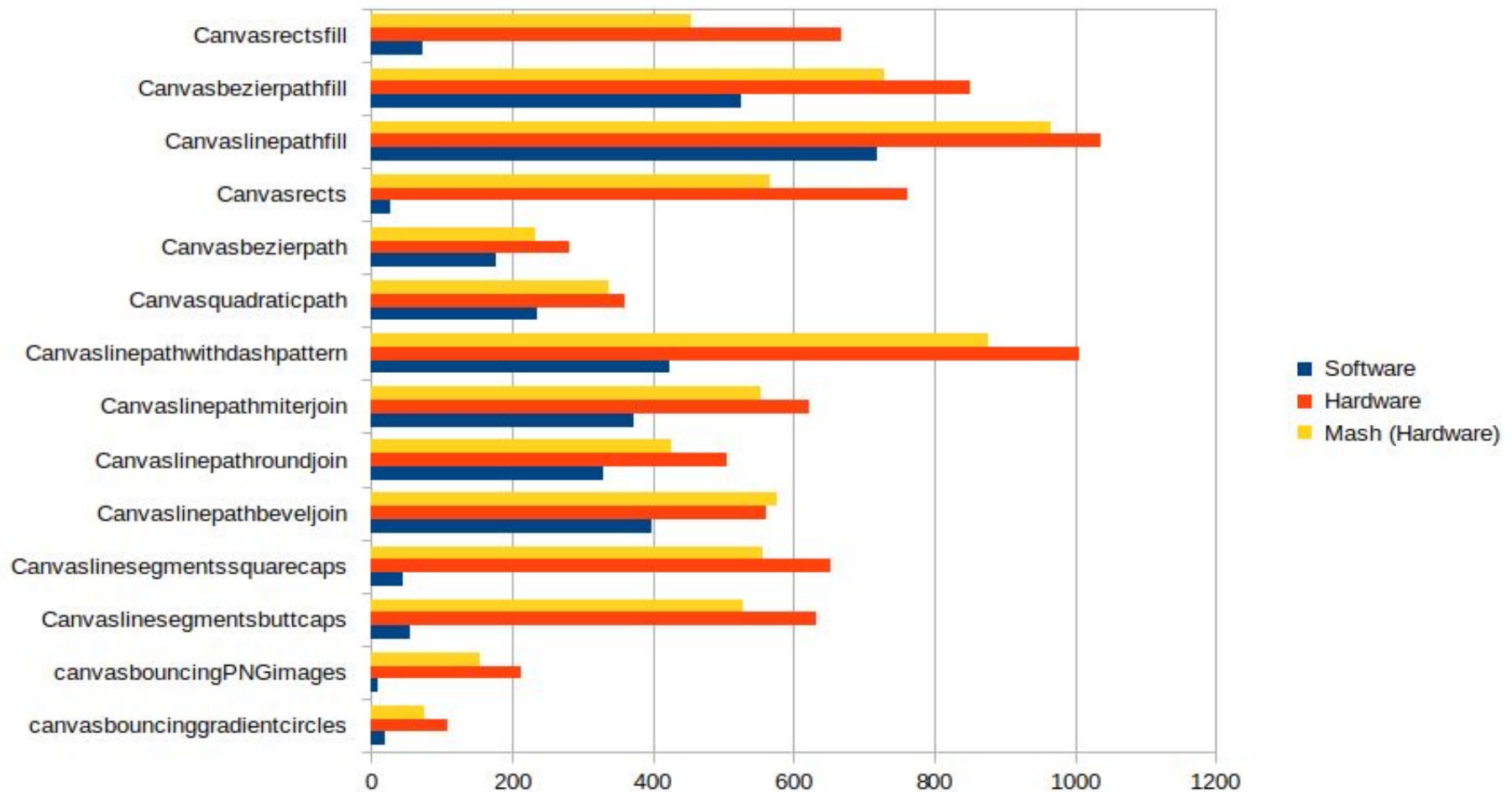
- Nov-Dec/16
  - CES demo: Linux/AGL/Wayland on R-Car M3



# Background

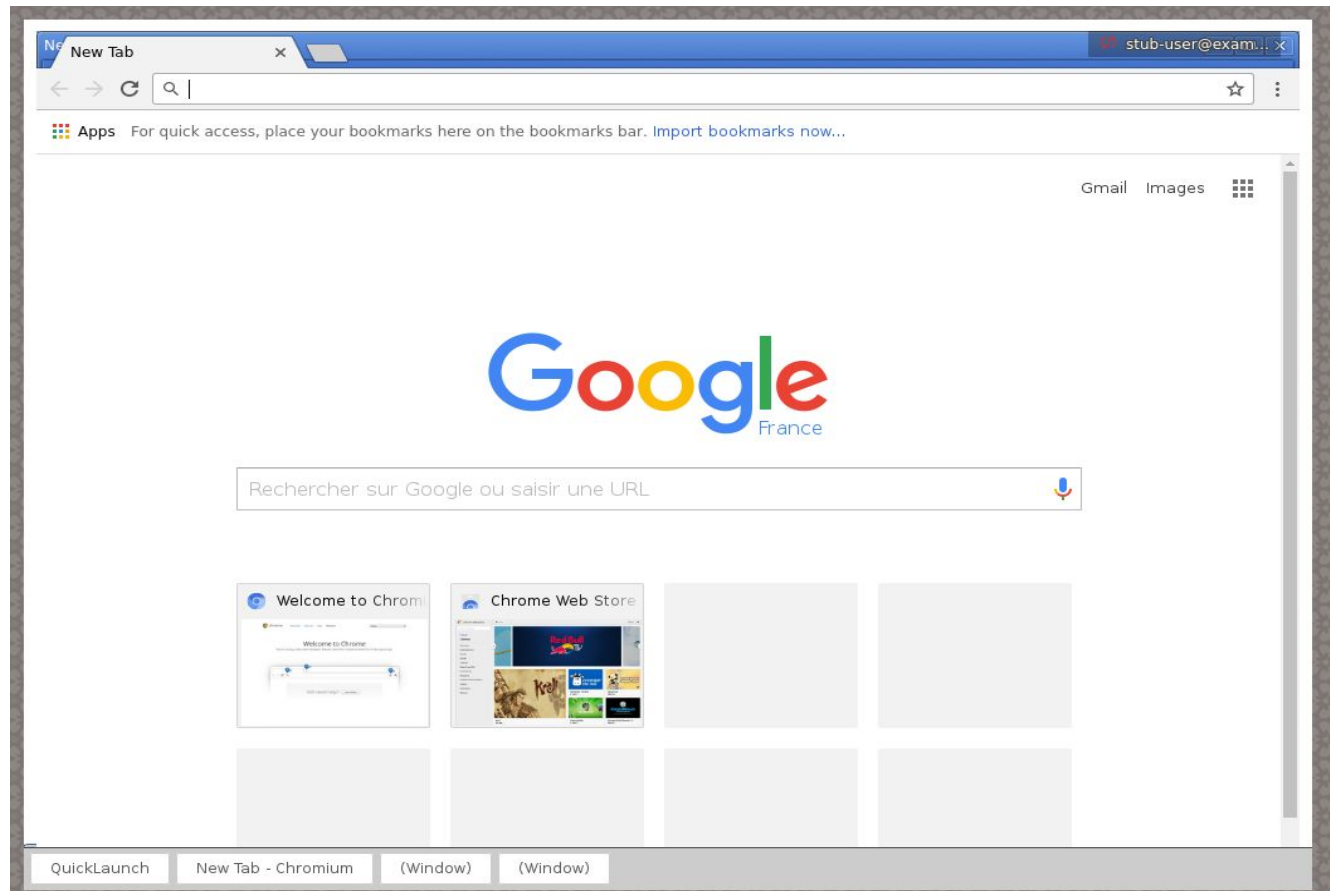


- Nov-Dec/16
  - Performance on BrowserBench GPU tests



# Background

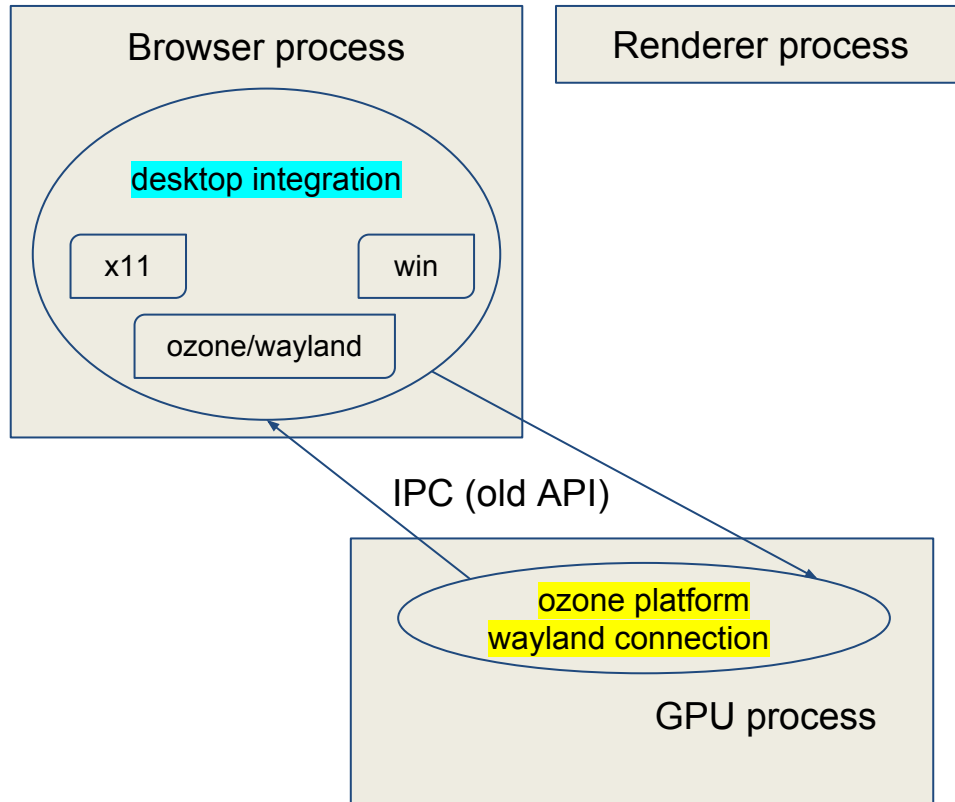
- Since Jan/17
  - //mash/simple\_wm
  - [Analysis of window classes](#)



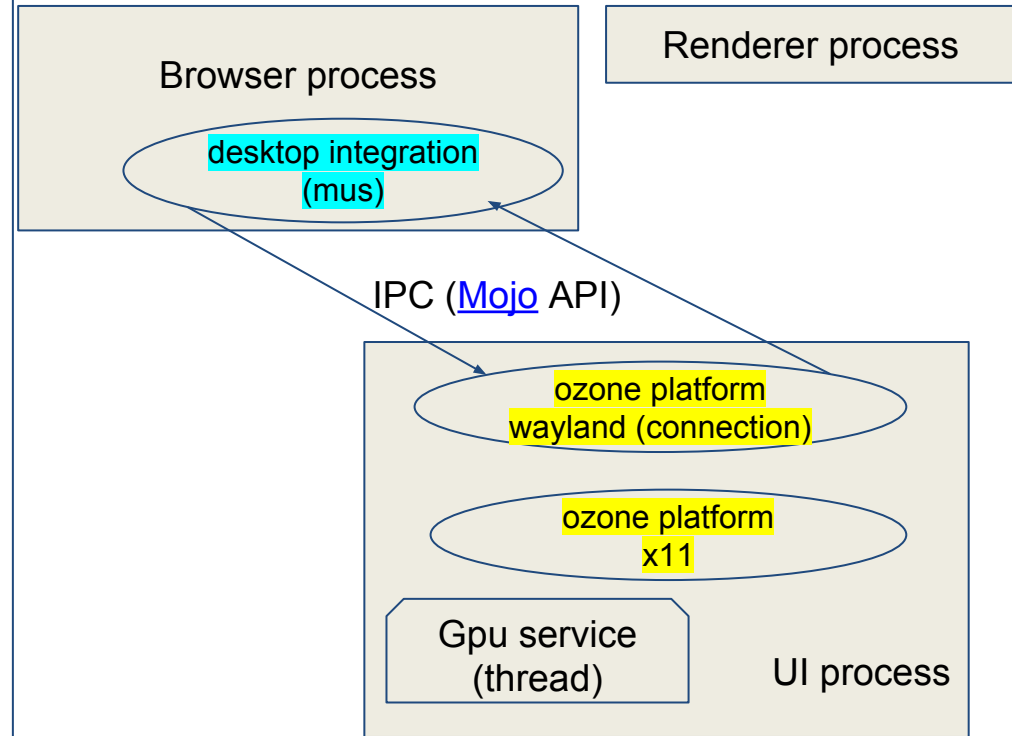
**Chrome / Mus**

# Desktop integration

Linux desktop integration (01.org)



Mus Linux desktop integration



# Discussion: Internal vs External

- Internal-window mode
  - All the aura windows in the system end up sharing a single display.
  - All the ash and Chrome aura windows are embedded within a single top-level acceleratedWidget.
- External-window mode
  - Modify Chrome and Mus so that Mus creates native acceleratedWidget's for each top-level mus window
    - chrome/mus
  - new flag --mus, sibling to --mash

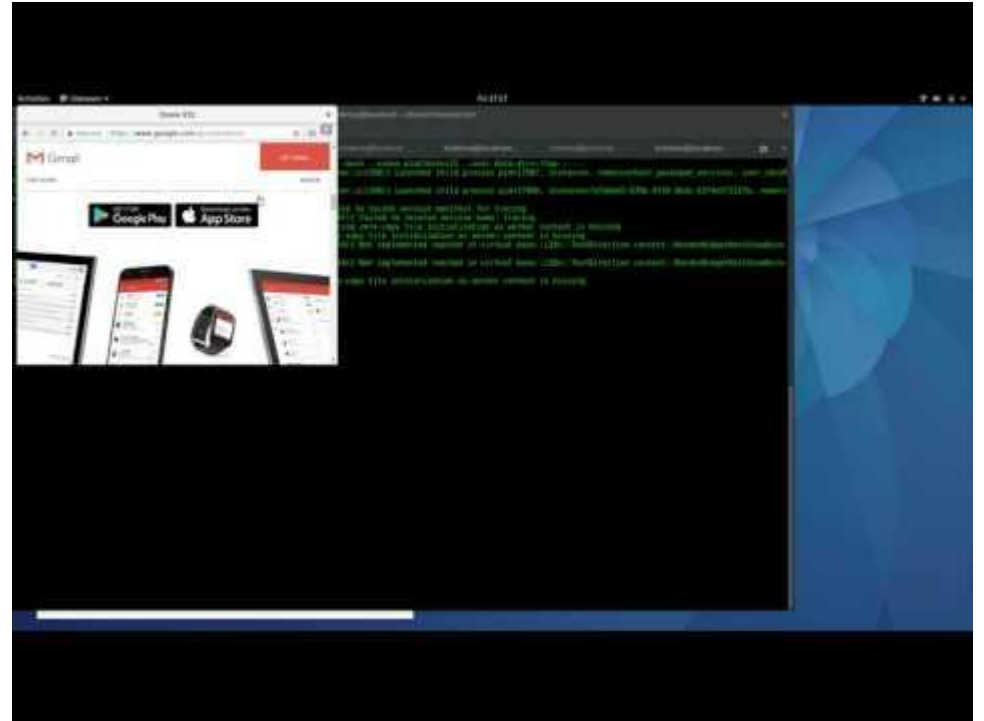


# Discussion: Internal vs External

- External-window mode, original plan proposed by Robert Kroeger (out of date):
  - Create a new “desktop-stub” replacement for Ash?
  - Desktop integration.
    - In essence, a subset of functionality currently provided by Ash is delegated to the native window system.
  - Considering using `//src/mash/simple_wm` as starting point?
  - After talking to sky@ et al, rjkroege@ agreed that this is not the best way to approach to tackle the issue. Alternatively, sky@ proposed to work this out directly on LinuxOS/Ozone builds.

# Mus' External Window Mode

- What is the status today?
  - **Functional, but WIP.**



- Where is this in the code?
  - **Being worked out off trunk, on [GitHub](#).**
- Existing tests
  - **mus\_demo has been extended to launch multiple windows.**

# Mus' External Window Mode - Status

- Today (Chromium ToT):
  - Ozone implies ChromeOS.
    - mus+ash == ChromeOS
- Today (GitHub):
  - Ozone runs on both ChromeOS and LinuxOS
  - Chrome can be ran in
    - mash (ChromeOS + internal window mode).
    - mus (LinuxOS + external window mode).

# Mus' External Window Mode - Plan

- Add external window mode support to Mus.
  - **Ongoing: Igalia**
  - services/ui/demo/ (help from @kylechar):
    - Stub out ScreenManagerOzoneExternal (no delegate) and evolve it.
    - Split up MusDemoInternal / MusDemoExternal.
      - Compile time switch.
    - Use the WindowTreeHostFactory code path on MusDemoExternal.
      - WTHF handles the creation of `ws::Display*`
- Extend Mus to support 'external window mode'.
- Rework internal window mode assumptions in the code.
  - 1:1 relation of `ws::Display` and `display::Display`.

# Discussion: Mus/LinuxOS plan

- Changed Chrome to launch in Mus external window mode.
  - Chrome today launches the same way it ought to, for Chrome/Mus.
  - `$ chrome --mus`
- Continue with desktop integration work (feature completion).

# Discussion: UI / GPU split

- chrome --mash (and --mus) still runs the UI and GPU components in the same process but separate threads.
  - Future: musws and musgpu in separate processes
    - <https://crbug.com/643746>
  - owner: rjkroege@
- Mojo-fication of Ozone/Wayland
  - Use approach similar to Ozone DRM/GBM (ChromeOS)?
  - GBM surface
    - rjkroege: to be discussed later.

# Questions?

[tonikitoo@igalia.com](mailto:tonikitoo@igalia.com) - Antonio Gomes

[mscho@igalia.com](mailto:mscho@igalia.com) - Mi Sun Silvia Cho