

# Delegated Compositing - Utilizing Wayland Protocols for Chromium on ChromeOS

Maksim Sisov - [msisov@igalia.com](mailto:msisov@igalia.com)

Fosdem 2024 - 04/02/2024

# Agenda

- Chromium/Wayland in ChromeOS - Goals and Motivation
- What is LaCrOS?
- Chromium Display Compositor
- Delegated Compositing
- Wayland Protocols
- A Big Picture

# Chromium/Wayland in ChromeOS

# Goals and Motivation

- Improve maintainability
- Let more users receive browser updates (old devices)



# How?

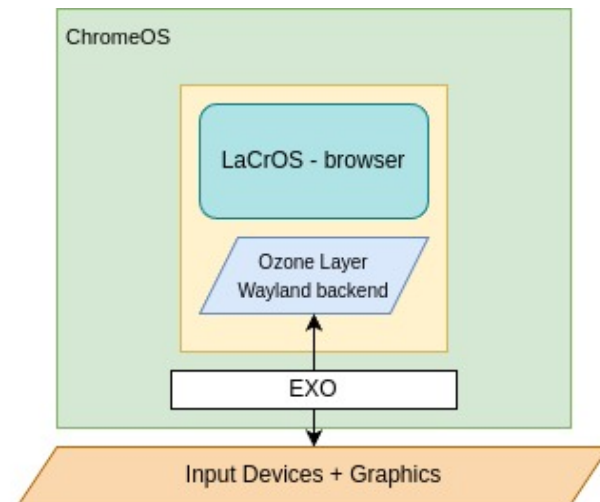
- Decouple the browser from OS (LaCrOS project)
  - ChromeOS (system UI and Ash Window Manager) and Chrome are tied together
- Use Wayland

# Why Wayland?

- ChromeOS has a Wayland server implementation called exosphere
  - Used by ARC (Android apps) and Crostini (Linux apps)
- Chromium is a Wayland client
  - Use Wayland for graphics and event handling
    - Stable Wayland protocols and custom extensions
  - Use crosapi (Mojo IPC) for higher-level features (e.g. file picking)

# What is LaCrOS?

A project to decouple the **Chrome Browser** from the **ChromeOS Window Manager (Ash)** and **System UI**

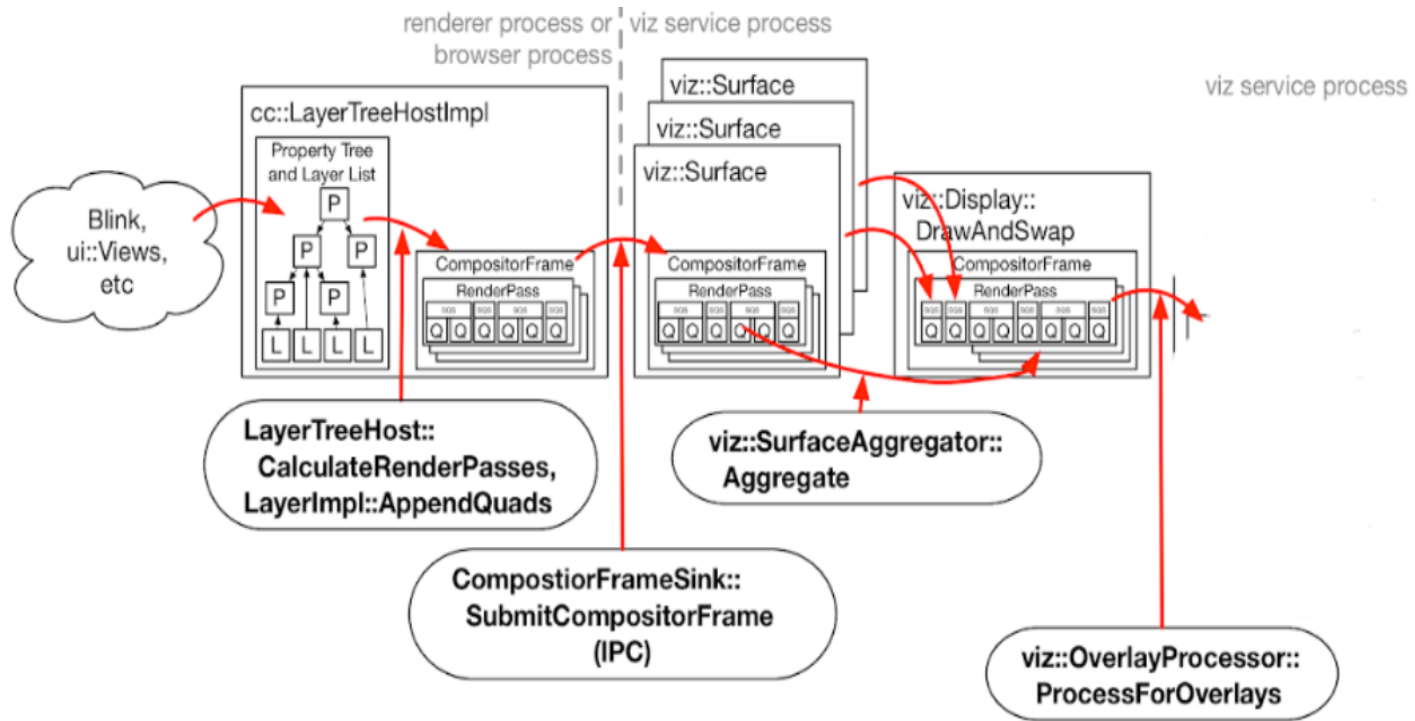


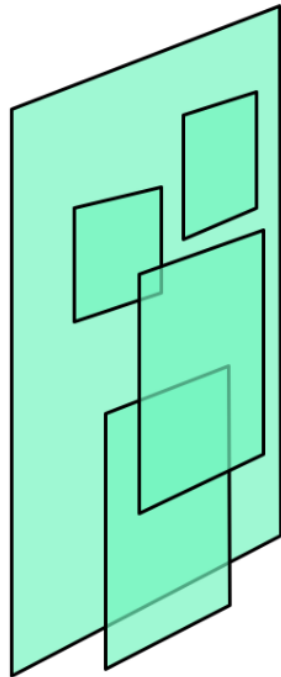
*This split results in performance/resource cost.*

- Why?
- How to mitigate that?

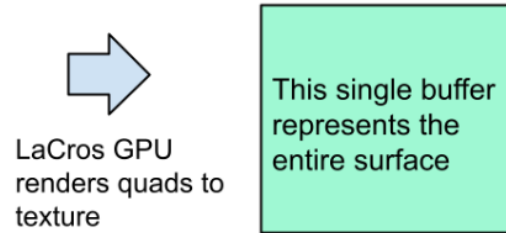


# Chromium Display Compositor

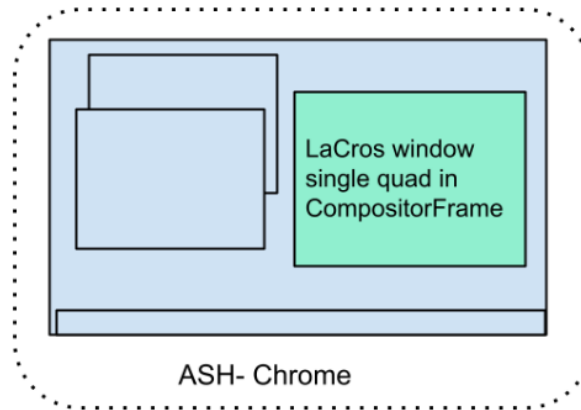




LaCros Aggregates quads that make up this surface.



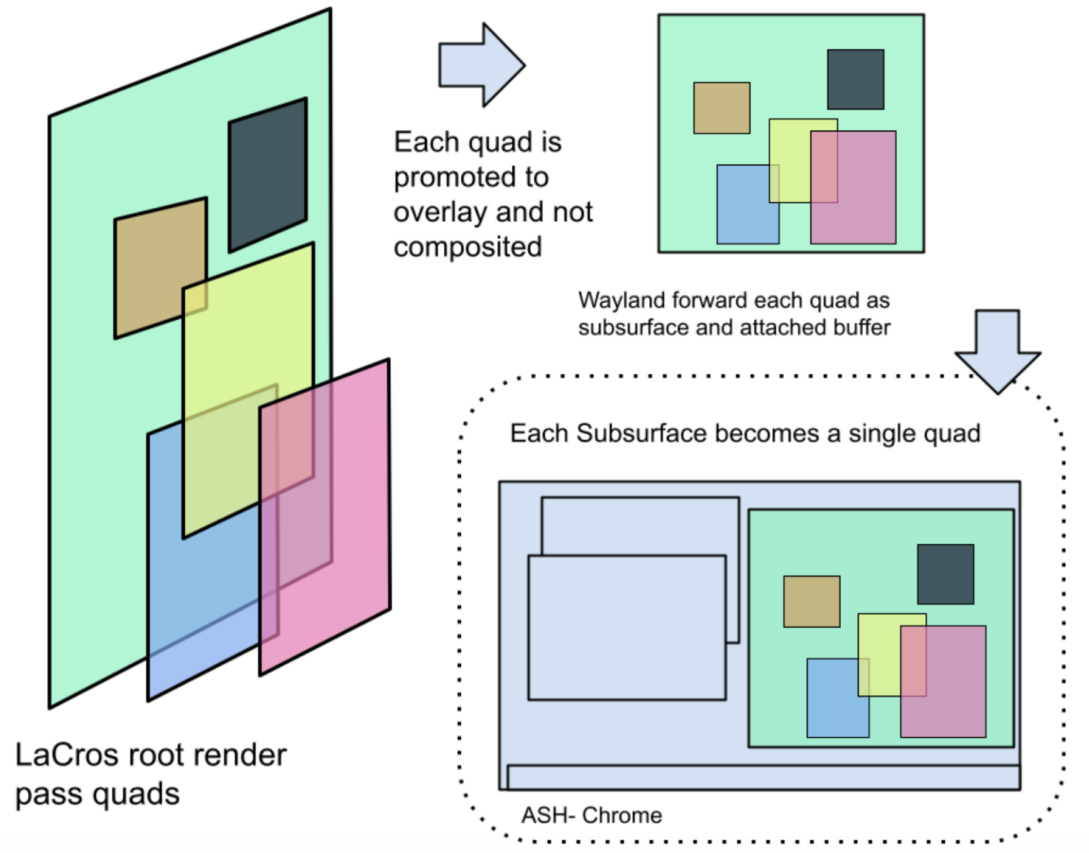
Surface is sent over Wayland  
Along with single buffer



# Double compositing and bigger resources overhead

# How to Fix That?

# Delegated Compositing

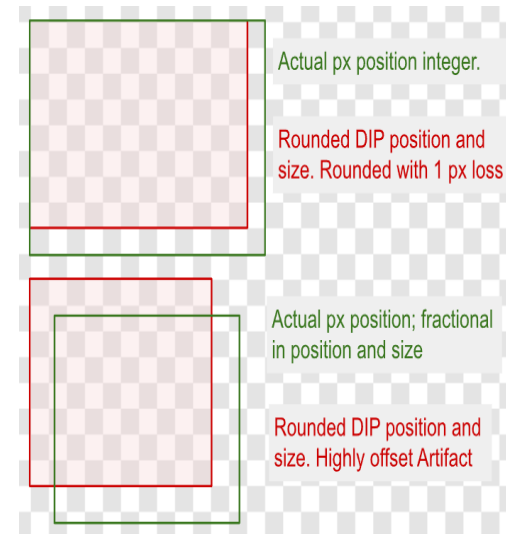


# Wayland Protocols

- ***wl\_subsurface*** - Each quad/overlay is represented by own surface
- ***wl\_buffer*** - Can be attached to multiple subsurfaces
- ***linux\_explicit\_synchronization*** - Fence for each quad's buffer
- Additional protocols:
  - ***surface-augmenter*** - extending wl\_surface (rounded corners and clipping, pixel precision)
  - ***single-pixel-buffer*** - for color buffers

# surface-augmenter

- `wl_subsurface::set_position`
- `wp_viewport::set_destination`



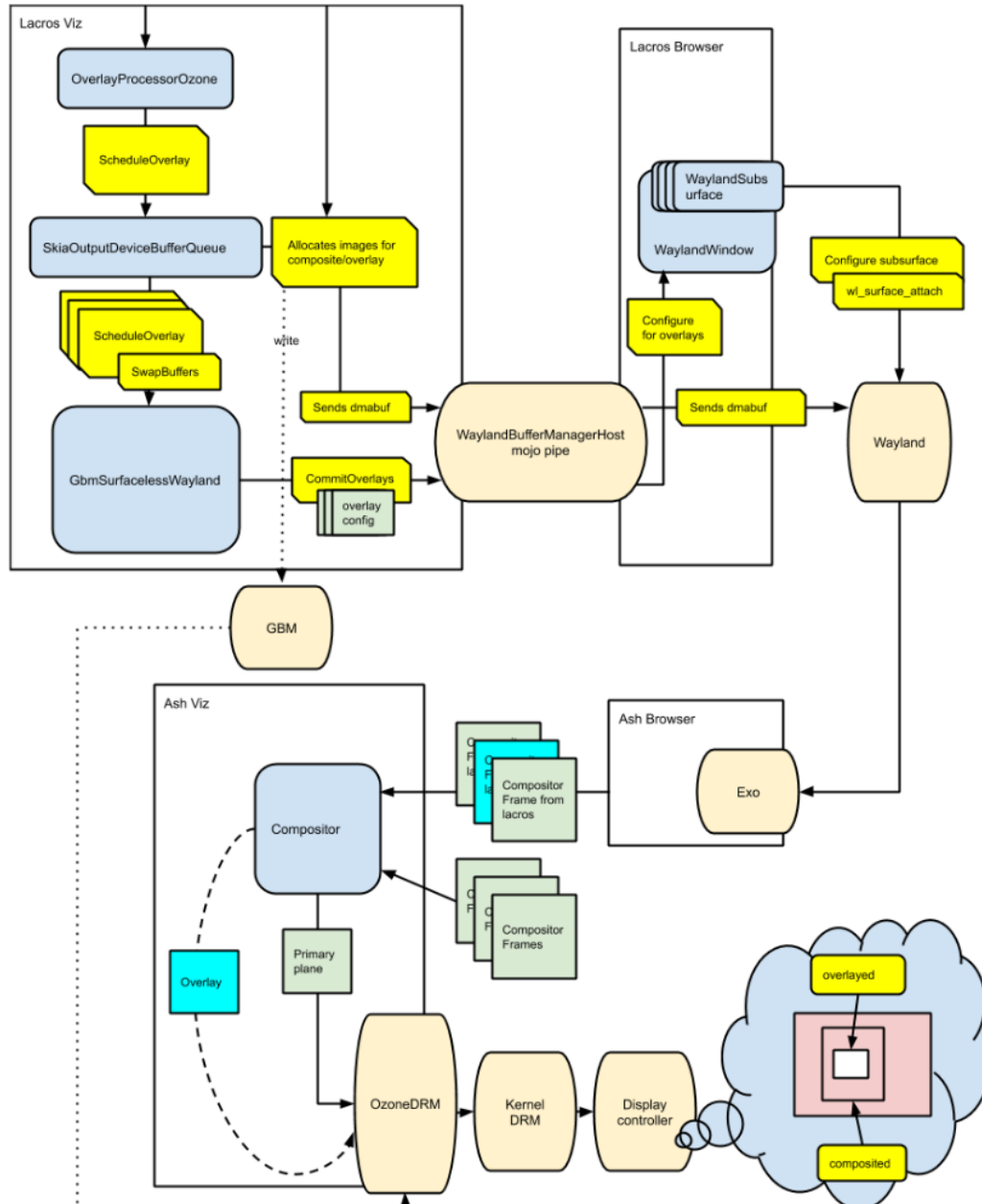


```
augmented_sub_surface::set_position(augmented_surface,  
                                     wl_fixed_t, wl_fixed_t)
```

```
augmented_surface::set_destination_size(augmented_surface  
                                         wl_fixed_t, wl_fixed_t)
```

augmented\_surface::set\_transform

augmented\_surface::set\_trusted\_damage



# Thank you

