XWayland multi DPI

Work-in-progress solution proposal

Roman Gilg
Problem statement

• Common scaling approach on XWayland is problematic:
  – Either ignore xdg-output size and be not multi DPI (GNOME)
  – Or blurry upscaling (KWin, weston, wlroots)
• X11 not designed for multi DPI in the first place.
• Toolkits often support only single scale factor if at all:
  – Qt: QT_SCALE_FACTOR
  – GTK: GDK_SCALE
Solution proposal: max factor rescaling

- Take max. scale factor of all Wayland outputs
- Scale XWayland outputs with quotient: max factor / output factor
- HiDPI aware clients scale with max factor
- Translate input coordinates in XWayland and XWM
- Downscale again in compositor
Solution proposal: example

• 3 displays
• Different scale factors
• Scaled outputs in compositor space
Solution proposal: example cont.

- In XWayland rescaled with 2
- Input scaled with 2 as well
Advantages

- Simple linear transformation with single factor
- High-DPI aware clients can render sharp nicely sized content
- Minimal changes to XWayland
- Backwards compatible
- Games do not hide native resolutions on scaled displays
Disadvantages

• Non high-DPI clients small even on low resolution displays in a mixed setup
• XWM part of Wayland compositor must translate all coordinates between compositor space and XWayland space
• Games show too large resolutions
• Increased resource usage
• Maximum possible Screen size: 16K x 16K (?)
Alternatives

- Multiple screens
- Ignore xdg-output size and use X11 multi DPI toolkit scaling:
  - Qt 5.14 has this improved
  - Complicates relation X11 Screen ↔ compositor space
  - What about GTK?