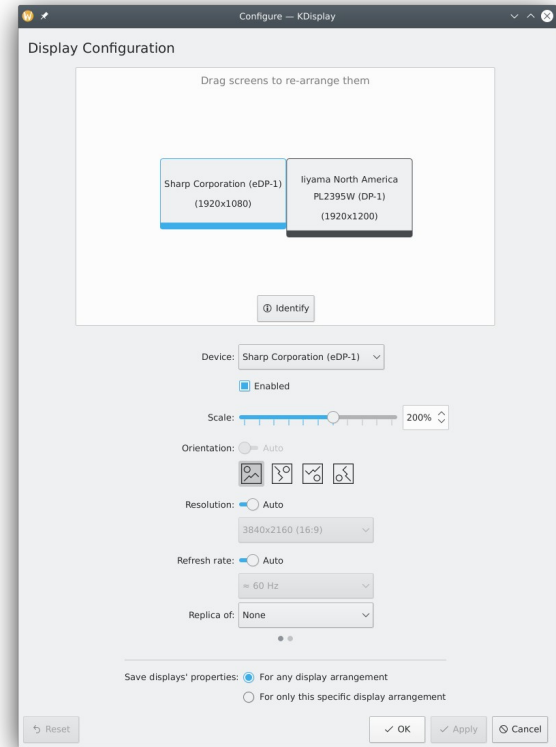


Disman

Universal Display Management





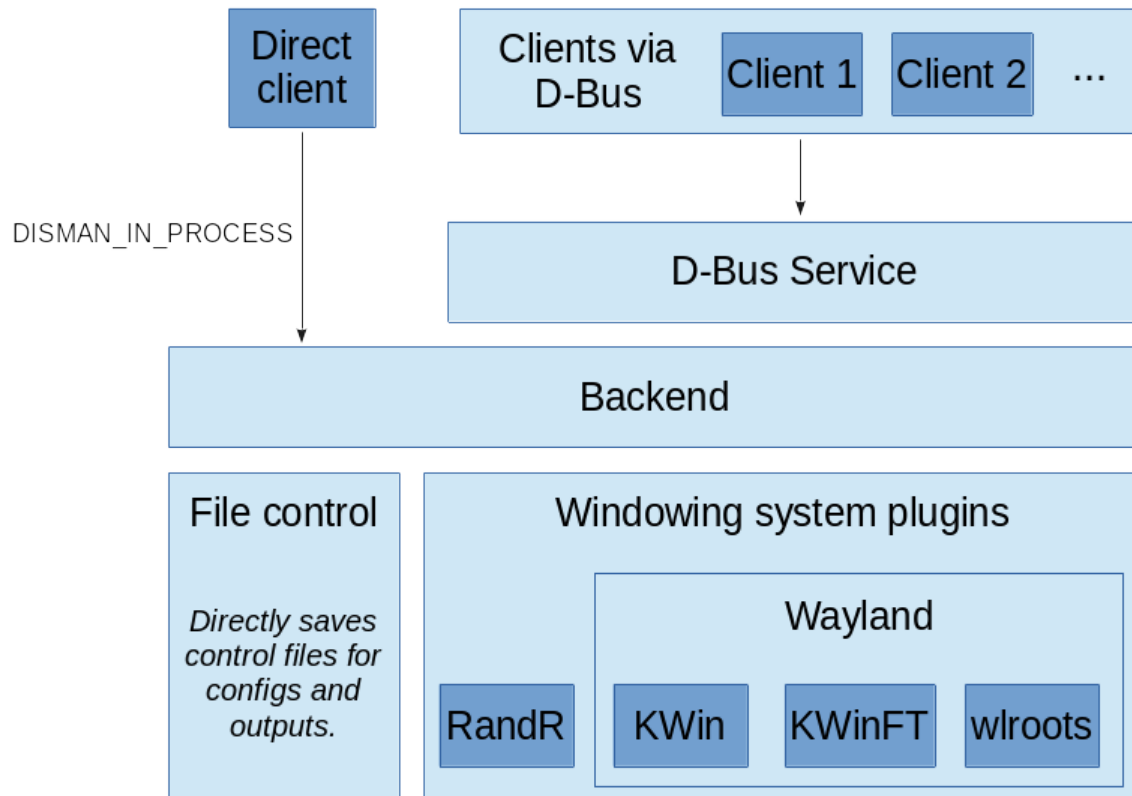
Motivation

- Multi-display setups common
- Configuration can be complex and must be remembered
- Different windowing systems, compositors without common interface
 - RandR on X11
 - Wayland
 - wlr_output_management_unstable_v1
 - kwinft_output_management_unstable_v1
 - KDE's outputmanagement protocol
 - D-Bus interfaces

Disman overview

- Library for frontend creation
 - C++11 with (yet) some Qt
 - CMake, SemVer (not yet stable release)
- Backend plugins loaded at runtime
 - Either in-process or out-of-process via D-Bus service
- D-Bus service
 - With automatic activation, for example via `dismanctl`
- Command line tool *dismanctl*

Disman overview – cont.



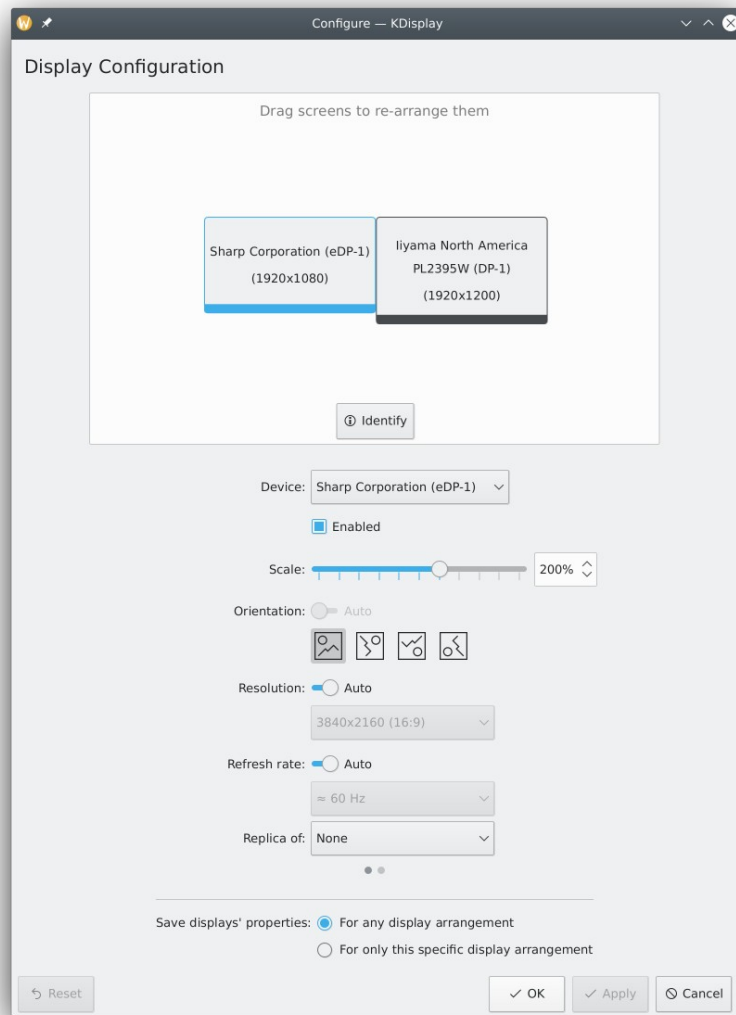


Disman features

- Save and load of display data
 - per combination of displays
 - globally per display
 - override for specific combinations
- Generation of optimal configurations
- Automatic selection of resolution, refresh rate or manual override
- Display replication/mirroring
- Laptop lid detection (via UPower and logind)

KDisplay

- Independent graphical app for changing display settings
- Plasmoid for the KDE workspace (quick change)
- Makes use of Disman





Future plans

- Support for docking of laptops
- Gnome Mutter Wayland backend plugin
- Stabilizing implementation and cleaning up the library API
- Removal of Qt dependencies from Disman, at least its library API
- Stable release 1.0



Project organisation

- Project homes:
 - Disman: <https://gitlab.com/kwinft/disman>
 - KDisplay: <https://gitlab.com/kwinft/kdisplay>
- Merge requests with CI builds and autotested
- Releases aligned with KDE Plasma
- Open development via GitLab issues



Thank you for listening.