

KWin now and tomorrow

The KDE Plasma window manager's current state and future evolution

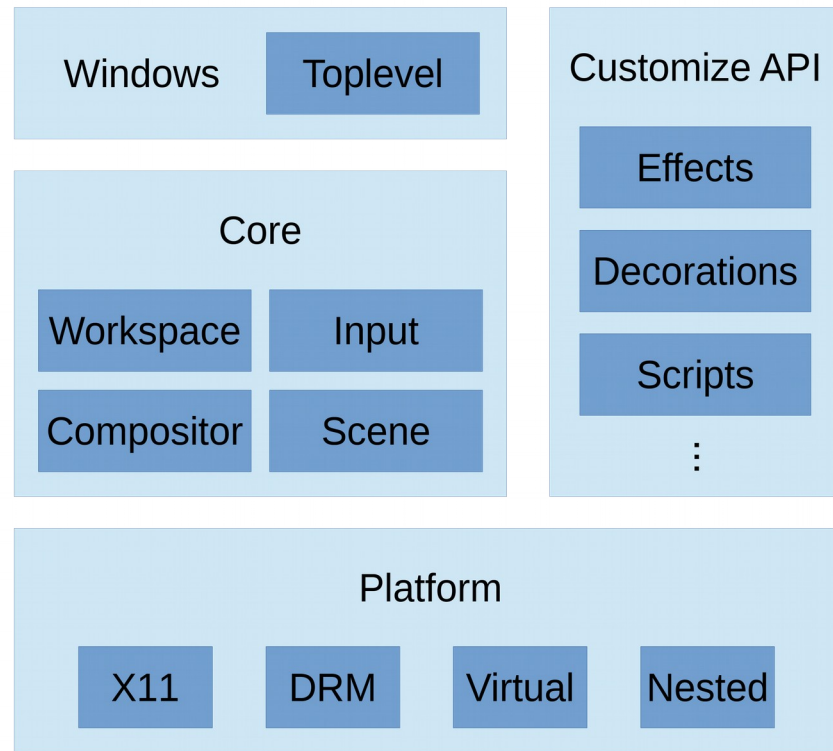
What is KWin?

- First commit on 20th Aug. 1999
- KDE window manager since KDE 2.0
- X11 composite window manager
- Wayland compositor
- First commit 11th June 2011
- Written in C++ and Qt
- Part of KDE Plasma releases, currently 5.17 beta

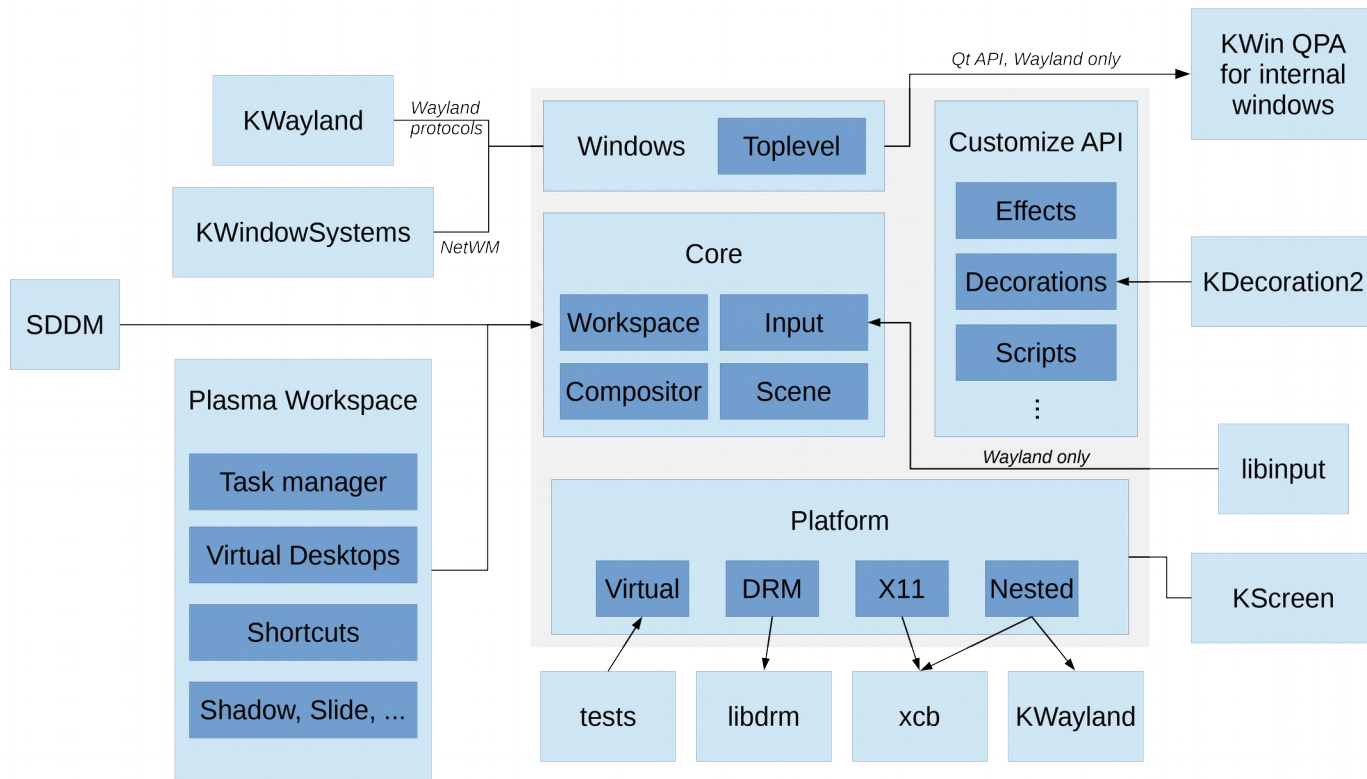


Design overview

- Singletons compose Core
- Qt Plugin systems:
 - Platform: DRM, X11, ...
 - Scene: Gl, QPainter, ...
- Effects + scripting APIs
- Input filters (Wayland)

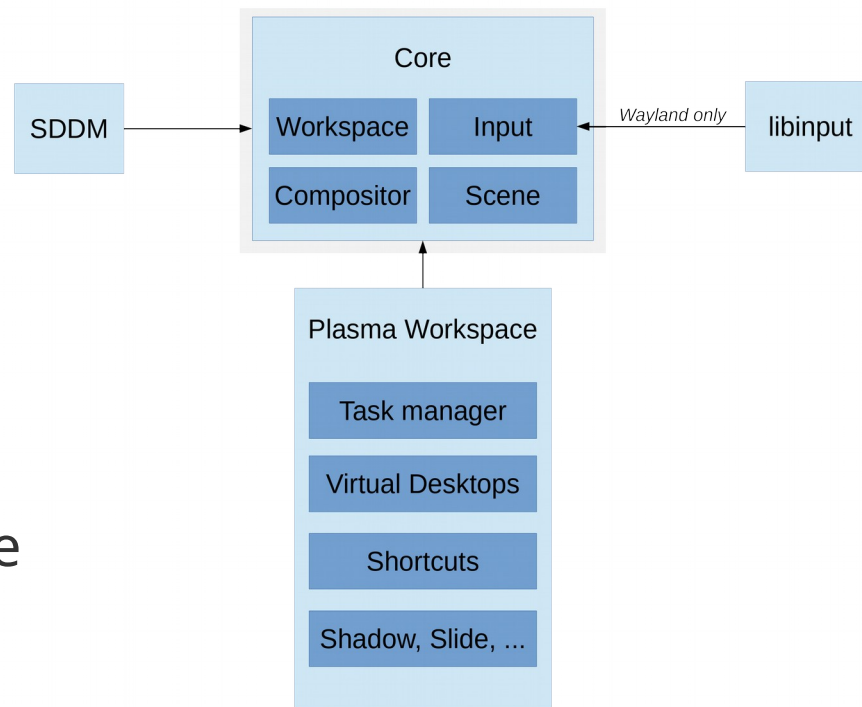


Broader ecosystem



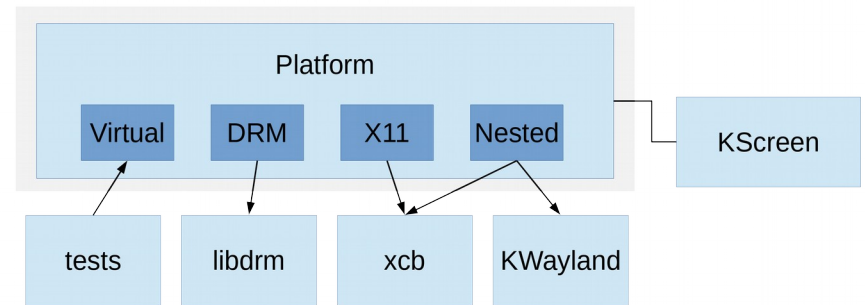
Broader ecosystem and Core

- SDDM display manager
- Startup of Plasma workspace
 - In the past scripts
 - Now binary
- Wayland input through separate thread



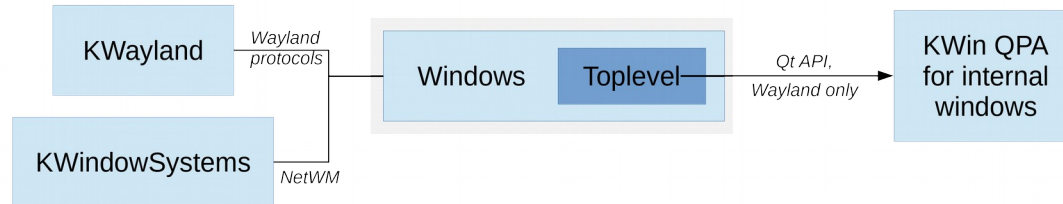
Platform

- Several Platform plugins
- Outputs are on Platform and Core level
- KScreen for output management



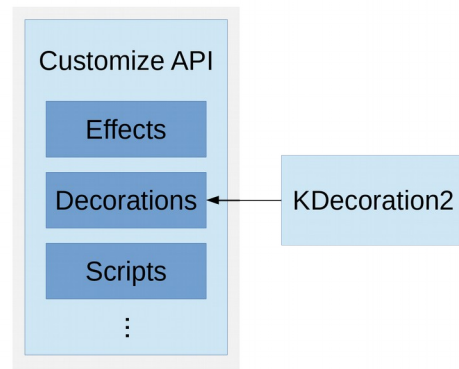
Windows

- KWindowSystems on X11, KWayland on Wayland for windows management
- For internal windows: own KWin QPA (Qt Platform Abstraction) plugin
- Internally represented through inheriting classes

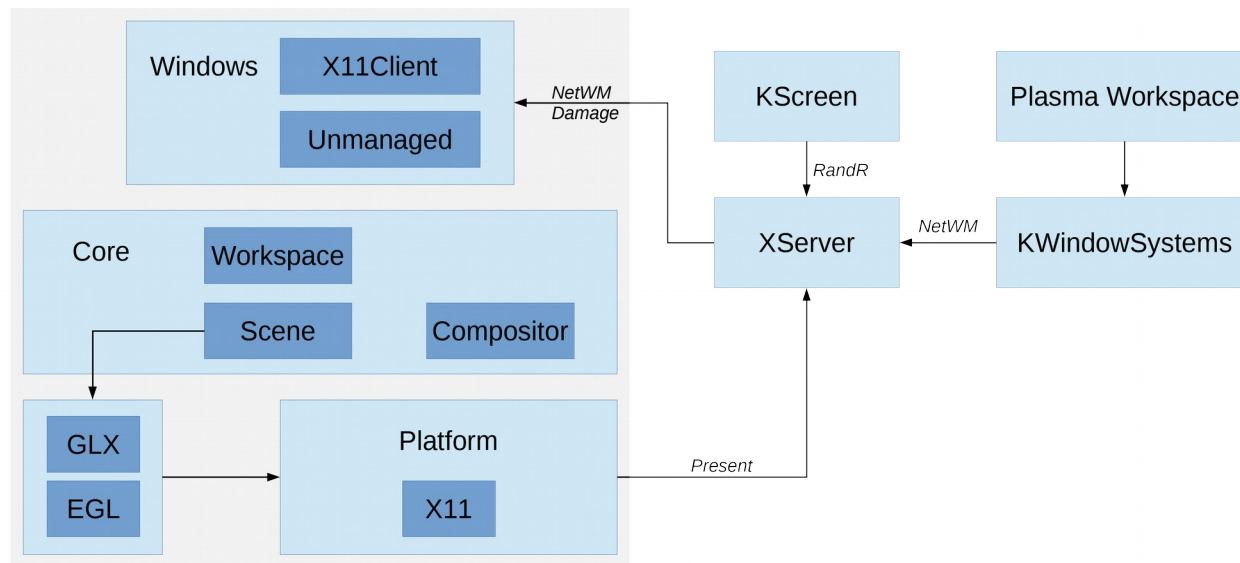


Customize API

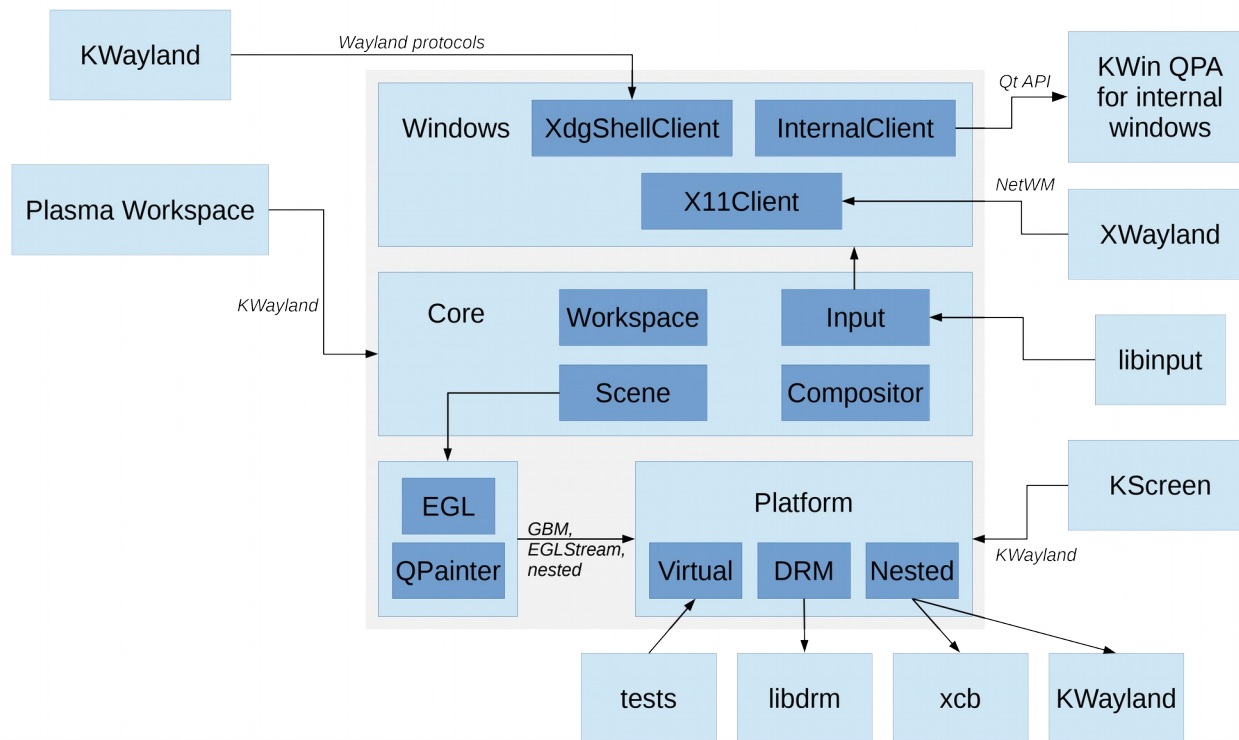
- Effects chaining
- Window decos with KDecoration2
 - Binary (for example Breeze)
 - Text files (SVG or QML) with Aurorae
- Powerful scripting API (for example tiling window manager)



Design as an X11 window manager



Design as a Wayland compositor





Recent and ongoing work

- Internal clients only through Kwin's own QPA
- Redesign of output management (KWin + Kscreen)
- Simplify X11 Compositor and GLX backend
- Night Color on X11

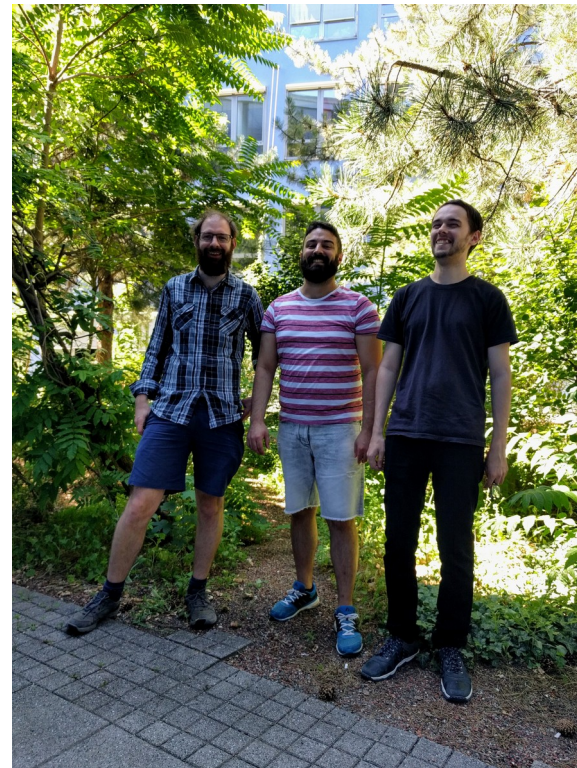


Future work

- Wayland as a KDE community goal
 - Outputs: Robustness, rotation, Xwayland multi DPI
 - Multi GPU
 - Rendering optimization (direct scanout)
 - Input: touch, virtual keyboard, graphics tablet
- X11: composite unredirect, blocking, auto-list
- Further split of X11 and Wayland

The KWin team: core

- David Edmundson
- Roman Gilg
- Vlad Zahorodnii



Vlad Zahorodnii

- Focus areas
 - Customize API
 - QPA
 - Clients geometry
- *IRC*: zzag
- Affiliation: Blue Systems



Roman Gilg

- Focus areas
 - Platform, outputs (KScreen)
 - Compositing rework
 - Task management, outreach
- *IRC: romangg, web: subdiff.org*
- Affiliation: Self-employed, Valve



David Edmundson

- Focus areas
 - Qt usage
 - Plasma integration
 - Everything else
- *IRC: d_ed, web: blog.davidedmundson.co.uk*
- Affiliation: Blue Systems





The KWin team: more contributors

- Plasma team and KDE's Visual Design Group
- Friendly community members
- Supportive companies: Blue Systems, Nvidia, Valve



Team management and cohesion

- Leaving of Martin Flöser, no single project owner anymore
- Challenges:
 - Code review stack
 - Handle disagreements (stability vs. evolution)
 - Find common ground
 - Establish decision processes



Internal and community relations

- Personal meetup: KWin sprint in July
- Plan larger changes in public early and extensively
 - Best: create task with preliminary plan and update on the go
 - At least: create task
 - Do not use IRC or mailing list for keeping track of things but web platform



Divide responsibilities

- Large project
- Best: multiple people are domain experts for everything
- Reality:
 - Work division is a proven concept
 - People need time to learn
- But: respect bus factor, share information (see public relations)
- Goal for core team must still be to have insight into all parts



Delegate work

- Open source community based project knows no hierarchy
- How to delegate work?
- Solution:
 - Leave blank space for others to contribute
 - Accept their solutions
- Example: Frontend finishing of KScreen KCM rewrite



Larger Open Source community

- Qt: Early feedback on new versions, work on QtWayland
- Clients: Promote Wayland protocol standards
- X.Org and freedesktop.org
 - Better communication with upstream and sister projects
 - More work on upstream
 - Help upstream partners to contribute to KWin