



Wltest

automating wayland client & driver testing

Austin Shafer

What is wltest

Open source Wayland test suite:

<https://gitlab.freedesktop.org/ashafer/wltest>

- Tests include resize, minimize, maximize, fullscreen
- Coverage includes:
 - EGL, GLX, Vulkan
 - Wayland and Xwayland
 - Supports running predefined tests on custom applications
- Does not require custom compositor builds
- Detects if test crashes or hangs

Why do we need a new test suite?

Motivation

- Wlcs - <https://github.com/canonical/wlcs>
 - Requires compositor integration
 - Really more of a test framework for compositors, not clients
- xwayland-run - <https://gitlab.freedesktop.org/ofourdan/xwayland-run>
 - Headless only
- No true xdotool cross-compositor equivalent on Wayland for us to automate testing

How wltest is different

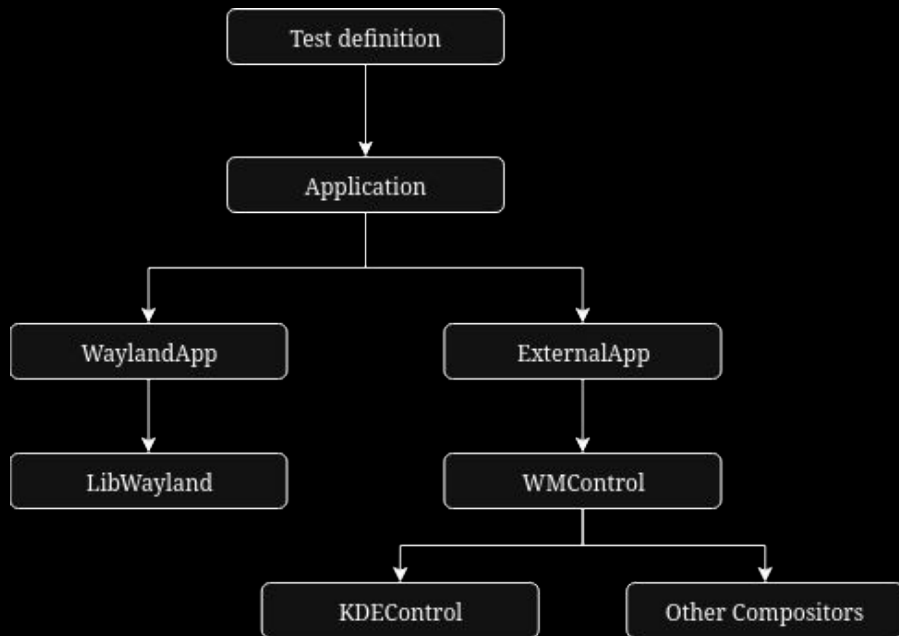
Requirements

- Not headless
 - We want to exercise the presentation pipeline
- No custom compositor builds
 - Validate tests pass in a normal user's setup
- Can run tests with arbitrary programs
 - Useful for testing resize of less common apps
- Coverage of Vulkan, EGL, Wayland, Xwayland

Design

How we avoid requiring custom compositors

- Tests are defined and manipulate an Application
 - Perform a resize to 64x64 without knowing program details
- Application can either be:
 - A window created by wltest itself
 - An external application (vkcube, glxgears, etc)
- Having a common library for compositor controls would be useful



Future Work

And potential uses of wltest

- Support more compositors
 - External application support is currently limited to KDE
- More test cases
- Output validation
 - Currently wltest only verifies that the program changes size and does not crash
 - Trickier than it seems, how can you reliably verify the on-screen contents are correct?
- Please reach out, we want to collaborate

