

Linebender Office Hours 2024-01-18

 **Calendar:** [iCal](#) [Add to Google Calendar](#) [Browse \(PST\) \(UTC\)](#)

 **Zulip Topic:** <https://xi.zulipchat.com/#narrow/stream/359642-office-hours/topic/2024-01-18>

Other notes: **Previous:** [2024-01-11](#) **Next:** [2024-01-25](#)
All: [Drive folder](#)

Organization: <https://github.com/linebender> **Project Board:** [linebender#1](#)

About

The office hours are a weekly opportunity for the [linebender](#) community to discuss the week's happenings and reach consensus. The meetings are hosted on Google Meet; the link is available in the calendar events (linked above). The office hours are open to anyone interested; just join the meeting at the specified time. Feel free to invite others, although please don't share on social media. The meetings are strictly limited to exactly one hour.

Agenda:

- Project Updates. Please keep things short this week
 - [Xilem](#)
 - [Vello](#)
 - [Glazier](#)
 - [Kurbo](#)
 - [#329](#), [#330](#) (approved) - cleanups
 - [AccessKit](#)
 - Linebender
 - [rfc#2](#) (for discussion) - approval process
- Introductions and welcome
- Tasks from prior week
- winit and Glazier
 - We're seriously considering going the winit route.
 - How much of a priority is AccessKit/accessibility for Winit?
 - How likely are we to get things like application and window menus into Winit?
 - How needed are native window menus - no platform actually still uses them? (Excluding KDE)
 - Application level menus are needed on MacOS (with window specific variants?). I don't use macOS so don't know
 - Does Glazier need to still exist as a layer to make a more user-friendly API over winit's IME?
 - Is it realistic that winit can exist as that underneath layer, or will we end up replacing more and more parts with raw platform interaction?

- Can winit-next expose extensible and potentially replaceable abstractions?
- Branding the Zulip - [#linebender > Zulip Branding](#)
 - TODO: Insert proposed choices here
 - Zulip name:
 - Organization Description:
- Making releases to crates.io
 - [Vello](#). General consensus that we should release this as soon as practicable
 - Already seeing real users
 - Tracking issue: [vello#302](#)
 - Complications because it currently has several (linebender internal) git dependencies further down the stack. See below
 - Documentation could use some work. E.g. readme is more geared towards “try this out” rather than “use this as a library”
This doesn’t need to block a release, but should be considered
 - [Peniko](#) (types for 2d graphics)
 - Required for Vello
 - Should be simple to release. Likely needs some tidy-up of e.g. readme ([peniko#19](#)).
 - Zulip thread: [#linebender > Getting peniko Ready For Release](#)
 - [Parley](#) (“WIP rich text layout library”)
 - Currently owned by Chad
 - Suggestion to move to linebender control
 - Do we have other maintainers? What is the future of this crate?
 - Relationship to cosmic-text: could we position parley as something that cosmic-text could build higher-level features (such as an editor abstraction) on top of (replacing its own layout code).
 - Depended on by Vello
 - [Fello](#) (“Internal font support code for vello”)
 - Owned by chad
 - No longer needed as of (TODO: Link PRs in [fontations](#) and vello?)
 - [Glazier](#)
 - Current Wayland support is in a bad state (sorry!). Can release with this disabled by default
 - No other real blockers to releasing
 - A bit up in the air, given other conversations
 - [Xilem](#)
 - Very early, very few widgets
 - Maybe moving to Masonry
 - Blocked on Vello
- Release policy/governance
 - Zulip thread: [#linebender > Our Ideal crate release process](#)

- For similar reasons as GitHub administration - releases shouldn't be a surprise, so there's no harm in the list of people who have permissions being small (~3?)
- crates.io doesn't currently give us much visibility into/control over this process

Tasks:

☐☐