

Mesa Governance Workshop recap

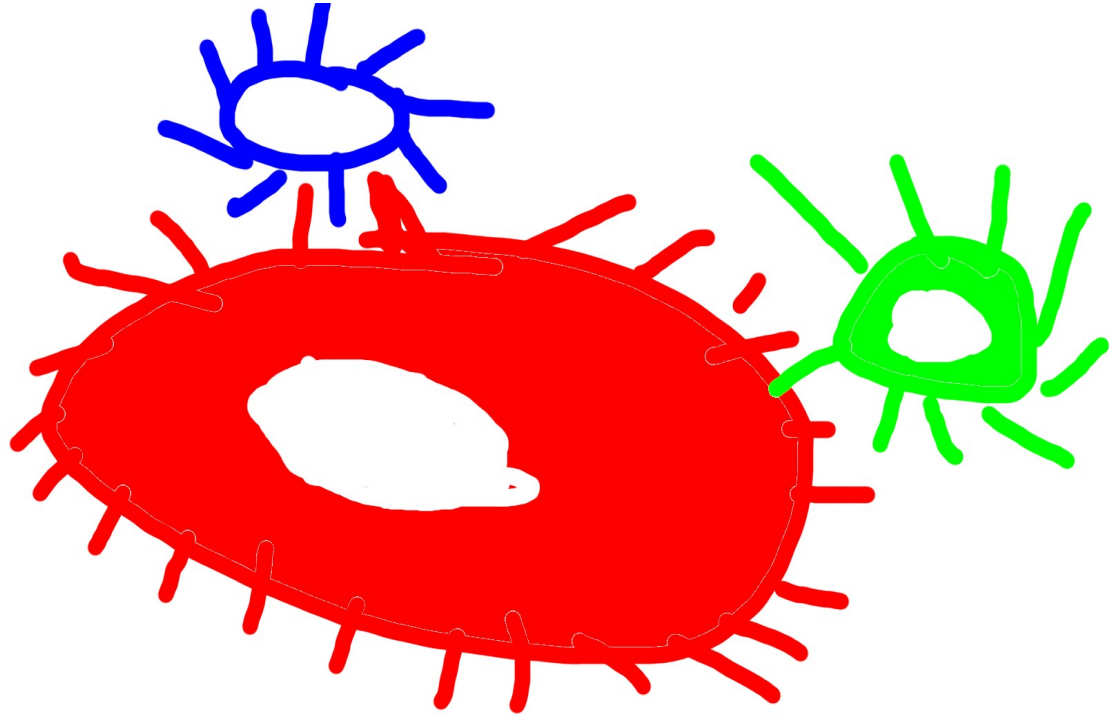


Image credit:
Matthieu Moatti

Why have a “council”?

- If the current process usually works, why even bother?
 - The current process leads to burn-out
 - Takes a lot of bandwidth to oppose proposal, bad things might slip through
- Make sure we have someone passionate enough to drive change when needed
 - Might lack someone willing to oppose proposals
 - Burnout might be an issue
 - Some people might not have the mental bandwidth
- The council speaking as "one voice" avoids some issues

What should a council actually be?

- A conflict-resolution process, not really a decision-making process.
 - Hopefully, rarely needed
 - Ideally not more than a few times per year
 - High barrier to invoke

Conflicts of interests

- Need to be wary of conflicts of interests
 - require abstaining if vote in conflict of interest
 - Sometimes useful to say "this is not my decision" to your employer
 - Trust belongs to the person, not the position
- The council can figure out the details
 - With the right intention but too much steering, the council might not feel empowered enough?
 - Document requirements?
- Is x.org board membership a conflict?

Council Process

- First, we try to form consensus the normal way
 - If that fails for long enough, or enough people thinks it needs to be escalated, we invoke the council
- what should be typical things to "escalate"
 - removing / adding drivers / frontends?
 - setting projects policies?
 - legacy software stack support?
 - big endian support?

Council Process

- Council need to be able to discuss in private
- Review discussions from tickets
- Gather input from stake-holders when needed
- Publish votes or not?
 - Council speaking a one voice has some benefits, including "plausible deniability" in going against employer's company line...
 - Perhaps council should reach consensus instead of voting?
 - What if the council can't agree? Some issues have no middle ground...
- Publish some resolution as free form text?
 - Resolution can be no resolution, or defer until more information is available, etc...

Council input

- Mostly GitLab tickets / MRs
 - non-contributors provide input through tickets / MRs?
 - Lock or turn confidential if needed?
- Distribution input
 - make sure maintainers are properly informed on controversial issues
 - tell them to subscribe to GitLab labels
 - Should we have list of supported distros?
 - no... but relationships with important ones?
 - make sure we don't land controversial changes right before release?
 - It should be a two-way thing!

How is council formed?

- We need a process for choosing the council
 - Nominate based on project activity?
 - Trust is not necessarily tied to commit count
 - Model like jury duty? Form ad-hoc council per issue?
 - Using GitLab tickets for voting?
 - Means votes are public, could be problematic in some cases
 - Elect council at each XDC?
 - We have a voting process for X.Org bod, reuse this
 - Requires voters to be X.Org foundation members

How is council formed?

- Should council even be elected? Can we just assign people?
 - Or elect once and then... deal with changes when needed?
 - max one seat per company?
 - similar to commit access threshold somehow?
- Who should council represent?
 - Make sure every driver is represented?
 - ...or every council member represents **all** project?

Conclusion

- We're all working together, assuming each others have good intentions
 - There's generally trust between contributors
 - We probably don't need to be heavy handed here
- Next step is to create a concrete proposal upstream, based on the input
- Also: we're deleting NVK. It's decided.
 - It had a good run

Thank you!