Proposal: Accelerating the migration to GTK4

Abstract

The GTK4 migration is one of the major refactoring projects on Inkscape's roadmap. Besides migrating to a newer and more actively developed GUI framework version, this will help to significantly improve the performance of the user interface and the canvas (in particular OpenGL).

However, the migration is a complex endeavour and will change hundreds of files in our codebase. For this reason, it is imperative to perform the migration as quickly as possible, because all other work that touches the user interface during the migration time will face conflicts.

For this reason, the development team has decided to accelerate the migration: an external contractor, i.e. a senior developer with GTK experience, will be hired and bring expert knowledge about GTK to the table. Multiple Inkscape developers (e.g. PBS, Rafael Siejakowski, Mike Kowalski) are available to mentor them, and will also be available as consultants for migration as a whole.

To reduce disruptions to the normal development process, a fork of Inkscape will be created at the point where further migration steps would break master. The fork is to be merged back into master once it compiles using GTK4 and SVGs are rendered in the canvas, to keep divergence as small as possible.

Within the scope of this proposal, we would allocate $32,600 for the compensation of the external contractor as well as a stipend for the hiring team, covering 4 months of full type work. There will be a trial period of 8 weeks (320 hours).

This document contains:

- the preliminary schedule
- duties and members of the hiring team
- license of the work to be performed
- time plan
- financials
- job posting

Reasoning

The port to GTK4 will bring the following benefits to the project. It is generally agreed upon that these benefits will vastly outweigh the cost of porting. The developer team agrees that this migration is necessary, urgent, needs to be completed quickly and we don't have the resources internally to accomplish this in a timely manner. The following is a list of the immediate advantages of the migration:

- **Model-view architecture**
  Inkscape's UI contains many long lists: the colour palette, batch export preview, marker selector dropdown, and the font picker to name but a few. In GTK3 these lists must be loaded in full when they are first used, while in GTK4 they can be populated on the fly as elements become visible. In each of the above examples, it is easy to construct set-ups where loading the list requires extensive processing causing long freezes. Porting to the new system will therefore allow several currently unfixable UI performance bottlenecks to be addressed.
  Reference: https://blog.gtk.org/2020/06/07/scalable-lists-in-gtk-4

- **Closing the cross-platform performance gap**
  GTK3 is notorious for running much slower on MacOS and Windows compared to Linux. This is partly due to the quality of GTK's platform backends. In GTK4 much work has been done to improve this situation, but most of it has not been back-ported to GTK3 due to the effort required. Therefore the migration is the only option the project has to close this common source of complaints among users of those platforms.
  References:
  - https://gitlab.com/inkscape/inkscape/-/issues/1614
  - https://gitlab.com/inkscape/inbox/-/issues/6363
  - Essentially, anywhere on the Internet where users can complain, e.g. Reddit.

- **OpenGL**
  Canvas rendering needs to be sped up. One great way to do this would be to write an OpenGL-based renderer. However early experiments have showed that GTK3 is riddled with a number of crippling bugs in its OpenGL codepaths. This has meant that the partial OpenGL acceleration support in 1.3 will not be turned on for users by default, and therefore investing the time in writing a full OpenGL renderer would currently be futile. This situation should be vastly improved in GTK4; although OpenGL was added to GTK3 as an afterthought, it is a core requirement of GTK4, which has forced the developers to iron out these bugs.
  Reference: https://gitlab.com/inkscape/inkscape/-/issues/3429

- **Better relationship with upstream**
  By using a more up-to-date version of their toolkit, upstream GTK developers are more likely to listen to and fix bugs reported by downstream, which often does not happen (quickly, or at

all) when bugs are reported against GTK3.
Example: https://gitlab.gnome.org/GNOME/gtk/-/issues/4705

- **Better UI performance overall**
Merely focusing and defocusing Inkscape's main window leads to an excruciating pause of several seconds as every widget has to update. There are similar jank issues when the Fill and Stroke dialog activates and deactivates, for example – an extremely common operation! These issues again appear to stem from GTK3 inefficiencies in its CSS system. Experimenting with GTK4 demo apps suggests that this issue will also be improved in the new toolkit.
Reference: https://gitlab.com/inkscape/inkscape/-/issues/3225

Here are also some quotes taken from various developer discussions:

- "The bottleneck isn't the canvas drawing (that's slow for everyone on every platform), it's the widget drawing of Gtk3." - Martin Owens (https://gitlab.com/inkscape/inkscape/-/issues/1614#note_806163635)

- "There's a comment by @adam.belis in the wild saying that switching to GTK4 might not solve our performance problems, because we don't even know what they are. Yes it will, having extensively investigated them. Being a GTK4 user will also make the GTK developers more likely to fix any remaining performance bugs we find. So all this is to say that the GTK4 port is even more urgent. " - PBS (https://gitlab.com/inkscape/inkscape/-/issues/483#note_864842280)

Preliminary Schedule

Prepare code base for the branch to GTK4:

- Finish merge requests !2107, !3333, !4959.
- Do all preparation steps as outlined in https://docs.gtk.org/gtk4/migrating-3to4.html and https://wiki.inkscape.org/wiki/GTK%2B_4_Migration.

Branch for GTK4 (created when releasing Inkscape 1.3-beta):

- Successfully compile Inkscape with GTK4, commenting out problematic blocks of code.
- Successfully render SVGs in canvas.
- Fix keyboard input (if necessary).
- Fix mouse input (if necessary).

At this point, consider merging back into the master branch.

- Fix drag and drop (if necessary).
- Fix toolbars (if necessary).
- Fix overall dialog code (if necessary).
- Fix widgets (if necessary).
- Fix individual dialogs (if necessary).

- Fix anything else that comes up pertaining to the migration.

Hiring Team

The hiring team consists of up to 4 Inkscape contributors and one SFC staff member. The hiring team performs the candidate selection, including posing questions to candidates, reviewing the applications and deciding on criteria for selection. The hiring team will be appointed by the PLC as a part of the vote on this proposal.

Furthermore, the hiring team performs the midterm evaluation. They review the performance of the candidate and decide whether to extend the contract to the full 16 weeks (full-time equivalent). An extension requires a unanimous decision of the hiring team.

Members of the hiring team are not eligible to apply themselves.

Members: Jonathan Neuhauser, Tavmjong Bah, Marc Jeanmougin, Thomas Holder

License

The work done by the external contractor will be licensed as GPL2+, and the copyright will be with the SFC.

Tentative schedule

The schedule may be modified by the development team.

March: Vote on the proposal.

Subsequent: Job posting, waiting for applications until April 15

April 15-30: interviews, select contractor

May 1-15: wait on SFC to figure out the contract details

May 15: approx. date of 1.3.x branching

June 1: work starts

Aug 1: target date to merge the fork back in master

Financials

- A total of $32,000 is allocated for the candidate’s compensation, of which $16,000 is allocated for the first 8 weeks.

- $600 as compensation for the Hiring team, split equally among its members (excluding SFC staff).

Job posting

The Inkscape project is looking for a senior C++ developer with experience in GTK3 / GTK3MM.
They will perform some of the essential steps of the GTK4 migration for Inkscape in a four month time period.

What we expect from candidates:

- Significant experience in GTK3 / GTK3MM, ideally some experience in GTK4.
- Ability to work in an asynchronous setting / working across timezones.
- Ideally, experience with FOSS development.

The candidate will perform essential steps in the GTK4 migration (see https://docs.gtk.org/gtk4/migrating-3to4.html). Knowledge of GTK4/GTK4MM is advantageous.
They will work closely together with an internal mentor with intimate knowledge of the Inkscape codebase. Thus, no knowledge of Inkscape is required.

Duration:
The project is scoped to fit a 16-week full time schedule. The first 8 weeks of the project will serve as an evaluation period for both sides. Depending on the progress, the project may be extended to up to 16 weeks (with additional compensation). The schedule may be shifted in order to accommodate part-time candidates. Work is expected to start June 01, 2021. Compensation is $50/hr. Payment terms: Net 30.

Application documents:

- CV
- Portfolio
- Answers to the following questions:
  - Summarize your experience with GTK
  - Motivation to work on this project

Application deadline: April 15, 2023
Interviews will be conducted in the week April, 24 – April, 28.

About Inkscape:
Inkscape is a Free and open source vector graphics editor for GNU/Linux, Windows and macOS. Inkscape uses the standardized SVG file format as its main format, which is supported by many other applications including web browsers.
Inkscape is a member of the Software Freedom Conservancy, a US 501(c)(3) non-profit organization.