

Anant Narayanan

anant@kix.in

<http://www.kix.in/>

Cairo Integration for GGI

A Summer of Code Proposal
Prepared for The GGI Project
March 22nd, 2007

Abstract

The GGI Project unifies different graphics APIs like X11, DirectX, fbdev and many more, hence making the job of developers needing cross-platform deployment of their graphics applications much easier. GGI already does a great job of providing a uniform graphics API, but lacks support for advanced 2D vector graphics.

The Cairo library is already a mature and established 2D vector graphics library. Integrating the Cairo library with libggi would bring lots of benefits to GGI users. The details of how this will be done and why it will be useful are discussed in this proposal.

Why?

As mentioned in the abstract, Cairo [1] is already a mature and established 2D vector graphics library, and is used in a variety of software, most notably the Gtk+ [2] GUI library. Tapping into Cairo's power would bring a lot of benefits, which include but are not limited to:

- Providing GGI users with high-quality 2D vector graphics capabilities.
- Enable the porting of the latest versions of popular libraries like Gtk+ to use GGI as a backend.
- Enable the porting of applications like recent versions of Mozilla to use libGGI.

How?

The best way, to integrate Cairo with GGI, in my opinion, is to create a new backend in Cairo for libGGI. Although the option of "grafting" the libCairo code into libGGI was proposed on the project SoC page, I think that it is not the optimal way of solving the problem. Mainly because it makes maintaining the relevant code a lot more difficult and also because it doesn't add any significant advantages over the backend approach. I will therefore discuss as to how I will go about implementing this new backend for Cairo.

- Cairo already has several backends [3] for a variety of output formats such as glitz, quartz, pdf, xcb and win32. Creating new backends for Cairo is a well documented and straightforward process, especially since there are already several backends in place.
- Creating a backend for Cairo essentially involves mapping the Cairo API [4] into low-level drawing functions of the target backend (in this case libGGI [5]). Sometimes this is not optimal, in which case, some intermediate form such as SVG may be used.

When?

The project will be spaced out roughly as follows:

- April 09 - May 01 : Get friendly with the mentors and the GGI Community ;-)
- May 01 - May 28 : Familiarize self with the libGGI API, Cairo API and the various Cairo backend implementations.
- May 28 - Jun 30 : Begin! Implement the Surface (core) set of Cairo methods
- Jun 30 - Jul 15 : Implement the Font set of Cairo methods
- Jul 15 - Aug 05 : Implement the Drawing set of Cairo methods
- Aug 05 - Aug 15 : Implement remaining utility functions (not all may be required)
- Aug 15 - Aug 20 : Document and finalize project
- Party!

Why Me?

I am an undergraduate student at the Malaviya National Institute of Technology, Jaipur, India; pursuing my Bachelor of Technology in Computer Engineering. I have been involved in the FOSS community for almost 3 years now. I am an active developer and documentation contributor in several open source projects [6], the most prominent of them being Gentoo Linux, GNU Parted and PHP-GTK. I became a Gentoo Developer as a result of my participation in last year's Summer of Code with them. I have worked on porting Cairo to PHP-GTK, and have also used Cairo in my day-to-day application development; which means that I have an intimate understanding of the Cairo API and its behavior. I am not very familiar with the GGI project in general, although I have exhibited the ability to learn and grasp concepts quickly in all my previous and current projects; this will be no exception!

I had applied to six organizations last year, of which four selected me - the highest number of selections for a student in that year. Although I could officially perform only one of those projects, I am continuing work on the other three outside of the SoC. FOSS is something that is already very dear to me, and I can assure you that I will take full responsibility for the maintenance of the modules that result from this project even after the Summer of Code concludes. I take every Summer of Code as an opportunity to "infiltrate" and become part of another new community. Since I am already well-versed in the community dynamics of open source projects, I will have absolutely no trouble in mingling with the GGI community and working with the infrastructure (Mailing Lists; Version Control Systems - I've extensively worked with CVS, SVN and Git; IRC etc.) already in place. In other words, I can get started almost immediately, giving me an effective coding time of almost 4 months, as opposed to the allotted 3. I sincerely hope that my work during the Summer of Code with GGI will eventually lead me to become a full-time GGI developer! You can find out more about me and what I do at my personal home page [7], and you also might want to look at my formal resume [8]. Please don't hesitate to get back to me if any part of this proposal is not clear to you. Thanks for considering this proposal and for your time!

REFERENCES

- [1] <http://www.cairographics.org/>
- [2] <http://www.gtk.org/>
- [3] <http://www.cairographics.org/backends/>
- [4] <http://www.cairographics.org/manual/>
- [5] <http://www.ggi-project.org/documentation/libggi/current/index.html>
- [6] <http://code.kix.in/>
- [7] <http://www.kix.in/>
- [8] <http://www.kix.in/personal/resume.pdf>

The latest version of this proposal will be available at: <http://www.kix.in/soc/07/cairo-ggi.pdf>