Introduction to

A modern 2D graphics library

Eduardo Lima Mitev
elima@igalia.com

October 14th, 2014
What is Skia?
What is Skia?

A library for drawing 2D Text, Geometries and Images
What is Skia?

Focus on accurate, high quality and high performance rendering
What is Skia?

Cross-platform: Linux, Android, ChromeOS, Mac, Windows, ... and hardware architectures too: x86, x86_64, ARMv7, AArch64, MIPS, ...
What is Skia?

Open source, licensed under the New BSD free software license
Skia in a typical software stack

Figure 1. Skia in the software stack
Originally developed by Skia, Inc, acquired by Google in 2005, then released as open source
History

Sponsored and managed by Google ever since
Similar projects

Cairo, QPaint (Qt), Direct2D, Cocoa Drawing
Why should I care?
Who uses Skia?
Who uses Skia?

Android  Chromium  Chrome  Firefox  Firefox OS
Who uses Skia?

Combined user-base could be above 1.5 billion
How Skia works?
How Skia works?

(Figure 2. High level architecture)
Skia backends

- Raster
- OpenGL (ES)
- PDF
- XPS
- Picture
API overview
API overview

**SkCanvas**: main drawing API (drawRect, drawText, drawLine, drawPath, etc)
SkPaint: encapsulates styling of draw calls (color, path style, blending mode, font, etc)
**SkDevice**: abstracts the backend (SkBitmapDevice, SkGpuDevice, SkPDFDevice, etc)
SkPicture, SkPicturePlayback: records and replay draw operations
Main features
Main features

Linear transforms and perspective (3x3 matrices)
Main features

Shaders, xfermodes, mask filters, path effects
Main features

Antialiasing, transparency, filters
Deferred drawing
(SkDeferredCanvas, SkPicture)
Main features

Subpixel text rendering
Ganesh: the OpenGL(ES) backend
Backend where most of the work is currently focused
Ganesh: the OpenGL(ES) backend

Accelerates 2D Canvas in Chromium/Chrome and Firefox
Ganesh: the OpenGL(ES) backend

Expected to accelerate full web content rendering in the future
Some features:

- Accelerate and combine effects with on-demand shaders
- Batching and merging of draw operations
- Geometry shaders
- Accelerated path rendering if available (NV_path_rendering)
Performance
Great tools available:

- Benchmarks in Skia repository under /bench
- Skia-telemetry
- skiaperf.com
- Chromium/Chrome’s about:tracing can also help
Lack of public benchmarks against similar libraries
The future of Skia
Backend work:

- Enhance PDF for Android and Chrome
- Many GPU backend changes
Platform support:

- C++11 enablement
- GLSL ES 3.0
Roadmap

APIs and New Features:

- Shareable pictures across process boundaries
- sRGB support, in partnership with Chrome
Dev and Test Infrastructure:

- New correctness testing framework
- More frequent recapturing of web archives for buildbot and cluster telemetry testing
Contributing to Skia
Contributing to Skia

Three well defined roles: developer, contributor, committer
Contributing to Skia

Report bugs to the issue tracker
Test: try Skia in your platform/hardware. Feed back!
Contributing to Skia

Contribute code: fix bugs, implement features in the roadmap

beginners, look for issues tagged GoodFirstBug
Contributing to Skia

Benchmark: compare Skia perf against other libraries in your platform/hardware; publish results!
For references and more info:

https://sites.google.com/site/skiadocs/
Thank you!

Q & A