

Skia

webkit task force

April 13, 2012

1 硬件加速

Skia 软件绘图和使用 opengl 的硬件绘图是使用 **SkCanvas** 类作为绘图接口，区别在于软件绘图是将图形绘制到 **SkBitmap** 上，而硬件绘图则是直接将图形绘制到屏幕上。Skia 的软件绘图和硬件加速的初始化方式

1. 软件绘图: `SkCanvas canvas(new SkDevice(bitmap));`
2. 硬件绘图: `SkCanvas* canvas = new SkCanvas(GrContext, GrRenderTarget);`
`GrContext` 是和 `gpu` 交互的公共接口，管理着 `gpu` 上下文，纹理及字体缓存。
`GrRenderTarget` 是绘制的目标，其中保存了需要绘制的大小，帧缓冲区等内容

Skia 的 `canvas` 绘制是根据 `SkCanvas` 中的保存的 `device` 对象来确定是软件绘制还是硬件绘制。

2 SkCanvas

SkCanvas 是 Skia 的一个核心类，他封装了所有对设备进行的画图操作。这个类自身包含了一个设备的引用，以及一个矩阵和裁剪栈。所有的画图操作，都是在经过栈内存放的矩阵变幻之后才进行的。当然，最终显示给用户的图像，还必须经过裁剪堆栈的运算。**SkCanvas** 记录着整个设备的绘画状态，而设备上面绘制的对象的状态又是由 **SkPaint** 类来记录的，**SkPaint** 类作为参数，传递给不同 **SkCanvas** 类的成员函数。**SkPaint** 类里记录着如颜色，字体，文字大小，文字粗细，渐变等信息。

1. 构造函数，给定一个 `Bitmap` 或者 `Device`，在给定的这个对象上进行画图
`explicit SkCanvas(SkDevice* device);`
`explicit SkCanvas(const SkBitmap& bitmap);`
2. 移位，缩放，旋转，变形函数
`virtual bool translate(SkScalar dx, SkScalar dy);`
`virtual bool scale(SkScalar sx, SkScalar sy);`
`virtual bool rotate(SkScalar degrees);`
`virtual bool skew(SkScalar sx, SkScalar sy);`
3. 画布内画图函数
`void drawARGB(U8CPU a, U8CPU r, U8CPU g, U8CPU b, SkXfermode::Mode mode = SkXfermode::kSrcOver_Mode);`
给定透明度以及 RGB 色，填充整个可绘制区域。
`void drawColor(SkColor color, SkXfermode::Mode mode = SkXfermode::kSrcOver_Mode);`
给定颜色 `color`，填充整个绘制区域。
`virtual void drawPaint(const SkPaint& paint);`
用指定的画笔填充整个区域。
`virtual void drawPoints(PointMode mode, size_t count, const SkPoint pts[], const SkPaint& paint);`
`void drawPoint(SkScalar x, SkScalar y, const SkPaint& paint);`
`void drawPoint(SkScalar x, SkScalar y, SkColor color);`
根据各种不同参数绘制不同的点。
`void drawLine(SkScalar xo, SkScalar yo, SkScalar x1, SkScalar y1, const SkPaint& paint);`
画线，起点(`xo, yo`)，终点(`x1, y1`)，使用 `paint` 作为画笔。

virtual void drawRect(const SkRect& rect, const SkPaint& paint);

画矩形，矩形大小由 **rect** 指定，画笔由 **paint** 指定。

void drawRectCoords(SkScalar left, SkScalar top, SkScalar right, SkScalar bottom, const SkPaint& paint);

给定 4 个边界画矩阵。

void drawOval(const SkRect& oval, const SkPaint&);

画椭圆，椭圆大小由 **oval** 矩形指定。

void drawCircle(SkScalar cx, SkScalar cy, SkScalar radius, const SkPaint& paint);

给定圆心坐标和半径画圆。

void drawArc(const SkRect& oval, SkScalar startAngle, SkScalar sweepAngle, bool useCenter, const SkPaint& paint);

画弧线，用法类似于画椭圆。

void drawRoundRect(const SkRect& rect, SkScalar rx, SkScalar ry, const SkPaint& paint);

画圆角矩形，x, y 方向的弧度用 **rx, ry** 指定。

virtual void drawPath(const SkPath& path, const SkPaint& paint);

路径绘制，根据 **path** 指定的路径绘制路径。

virtual void drawBitmap(const SkBitmap& bitmap, SkScalar left, SkScalar top, const SkPaint* paint = NULL);

绘制指定的位图。

virtual void drawText(const void* text, size_t byteLength, SkScalar x, SkScalar y, const SkPaint& paint);

以(x,y)为起始点写文字，文字存储在 **text** 指针内，长度有 **byteLength** 指定。

Skia 对于字体的绘制是根据不同平台实现的，对于不同平台使用不同的字体引擎，并从字体引擎获得字体图片进行渲染。