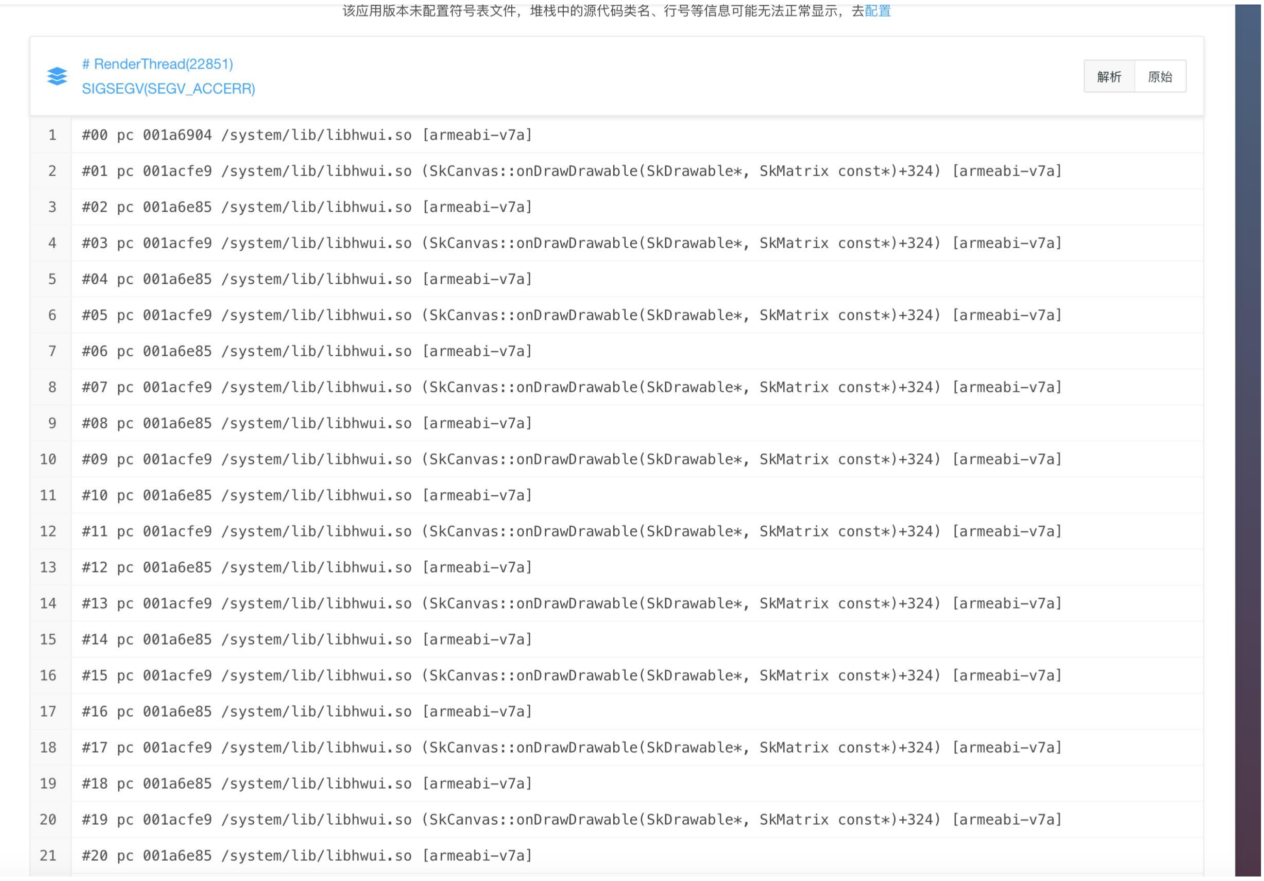
# 引入新技术或者开源库的思考

前言：

> 引入新技术是好事，也是一个组织寻求专业性进步的必经之路。

## 1.IM项目中使用开源库不规范导致的问题

这里分享一下,IM 项目中因使用不规范导致的Native bug，相当致命：



该应用版本未配置符号表文件，堆栈中的源代码类名、行号等信息可能无法正常显示，去配置

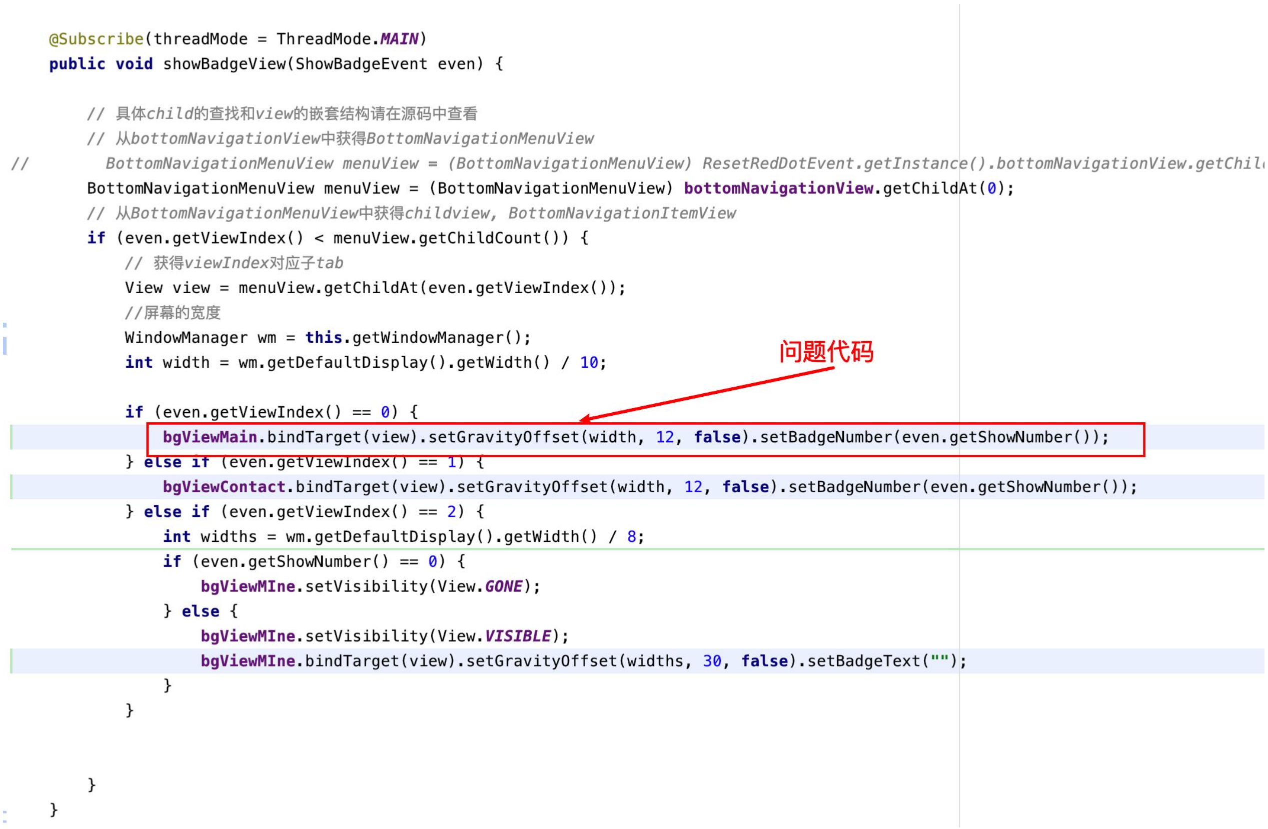
# RenderThread(22851)
SIGSEGV(SEGV_ACCERR)

```
#00 pc 001a6904 /system/lib/libhwui.so [armeabi-v7a]
#01 pc 001acfe9 /system/lib/libhwui.so (SkCanvas::onDrawDrawable(SkDrawable*, SkMatrix const*)+324) [armeabi-v7a]
#02 pc 001a6e85 /system/lib/libhwui.so [armeabi-v7a]
#03 pc 001acfe9 /system/lib/libhwui.so (SkCanvas::onDrawDrawable(SkDrawable*, SkMatrix const*)+324) [armeabi-v7a]
#04 pc 001a6e85 /system/lib/libhwui.so [armeabi-v7a]
#05 pc 001acfe9 /system/lib/libhwui.so (SkCanvas::onDrawDrawable(SkDrawable*, SkMatrix const*)+324) [armeabi-v7a]
#06 pc 001a6e85 /system/lib/libhwui.so [armeabi-v7a]
#07 pc 001acfe9 /system/lib/libhwui.so (SkCanvas::onDrawDrawable(SkDrawable*, SkMatrix const*)+324) [armeabi-v7a]
#08 pc 001a6e85 /system/lib/libhwui.so [armeabi-v7a]
#09 pc 001acfe9 /system/lib/libhwui.so (SkCanvas::onDrawDrawable(SkDrawable*, SkMatrix const*)+324) [armeabi-v7a]
#10 pc 001a6e85 /system/lib/libhwui.so [armeabi-v7a]
#11 pc 001acfe9 /system/lib/libhwui.so (SkCanvas::onDrawDrawable(SkDrawable*, SkMatrix const*)+324) [armeabi-v7a]
#12 pc 001a6e85 /system/lib/libhwui.so [armeabi-v7a]
#13 pc 001acfe9 /system/lib/libhwui.so (SkCanvas::onDrawDrawable(SkDrawable*, SkMatrix const*)+324) [armeabi-v7a]
#14 pc 001a6e85 /system/lib/libhwui.so [armeabi-v7a]
#15 pc 001acfe9 /system/lib/libhwui.so (SkCanvas::onDrawDrawable(SkDrawable*, SkMatrix const*)+324) [armeabi-v7a]
#16 pc 001a6e85 /system/lib/libhwui.so [armeabi-v7a]
#17 pc 001acfe9 /system/lib/libhwui.so (SkCanvas::onDrawDrawable(SkDrawable*, SkMatrix const*)+324) [armeabi-v7a]
#18 pc 001a6e85 /system/lib/libhwui.so [armeabi-v7a]
#19 pc 001acfe9 /system/lib/libhwui.so (SkCanvas::onDrawDrawable(SkDrawable*, SkMatrix const*)+324) [armeabi-v7a]
#20 pc 001a6e85 /system/lib/libhwui.so [armeabi-v7a]
```

经过长时间的排查，找呀找，查呀查，猜呀猜。

皇天不负苦心人，最终找到了罪魁祸首。

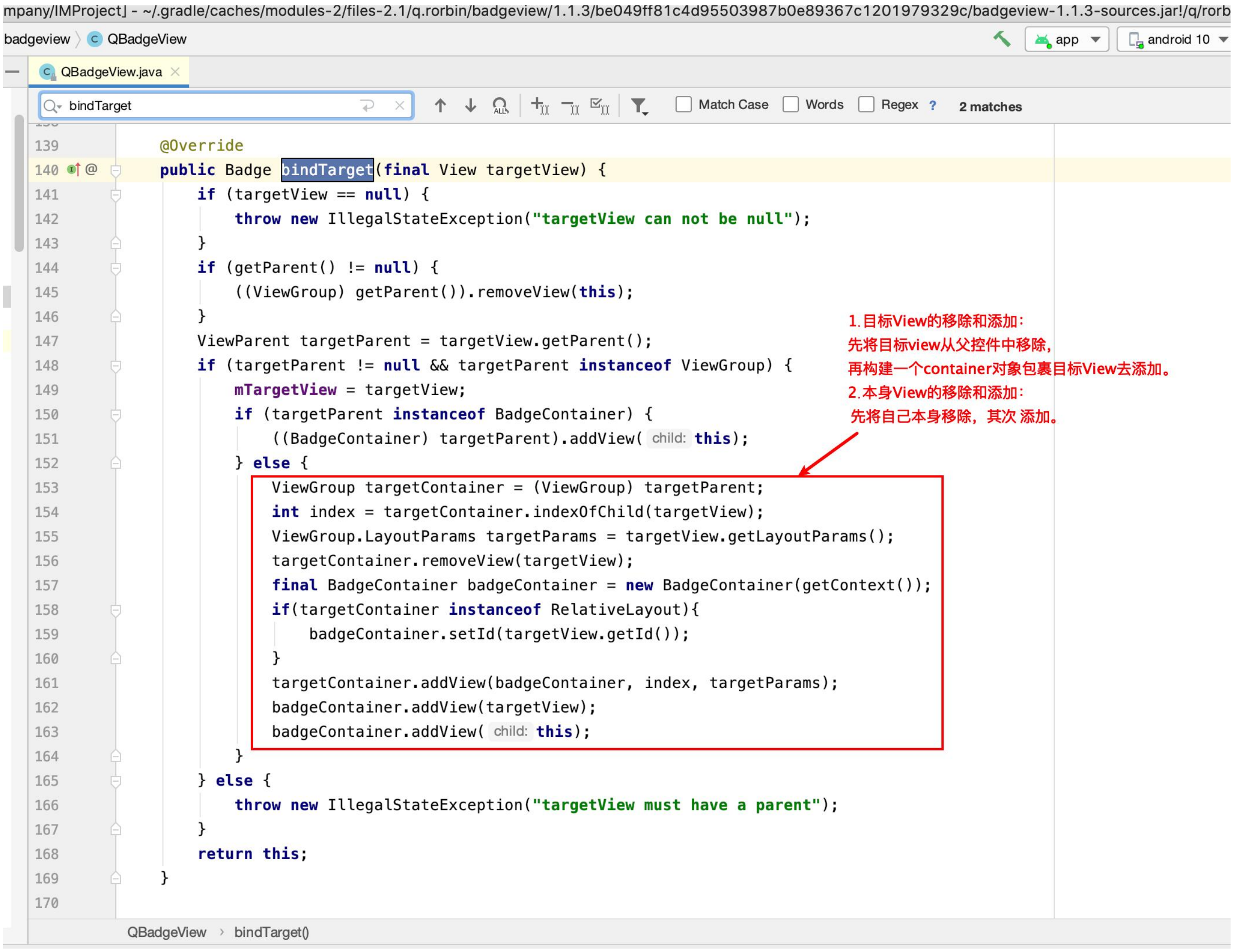
有问题的代码，锁定在这里：



```
    @Subscribe(threadMode = ThreadMode.MAIN)
    public void showBadgeView(ShowBadgeEvent even) {

        // 具体child的查找和view的嵌套结构请在源码中查看
        // 从bottomNavigationView中获得BottomNavigationMenuView
//          BottomNavigationMenuView menuView = (BottomNavigationMenuView) ResetRedDotEvent.getInstance().bottomNavigationView.getChil
        BottomNavigationMenuView menuView = (BottomNavigationMenuView) bottomNavigationView.getChildAt(0);
        // 从BottomNavigationMenuView中获得childview, BottomNavigationItemView
        if (even.getViewIndex() < menuView.getChildCount()) {
            // 获得viewIndex对应子tab
            View view = menuView.getChildAt(even.getViewIndex());
            //屏幕的宽度
            WindowManager wm = this.getWindowManager();
            int width = wm.getDefaultDisplay().getWidth() / 10;

            if (even.getViewIndex() == 0) {
                bgViewMain.bindTarget(view).setGravityOffset(width, 12, false).setBadgeNumber(even.getShowNumber());
            } else if (even.getViewIndex() == 1) {
                bgViewContact.bindTarget(view).setGravityOffset(width, 12, false).setBadgeNumber(even.getShowNumber());
            } else if (even.getViewIndex() == 2) {
                int widths = wm.getDefaultDisplay().getWidth() / 8;
                if (even.getShowNumber() == 0) {
                    bgViewMIne.setVisibility(View.GONE);
                } else {
                    bgViewMIne.setVisibility(View.VISIBLE);
                    bgViewMIne.bindTarget(view).setGravityOffset(widths, 30, false).setBadgeText("");
                }
            }


        }
    }
```

问题代码

再来`BgView#bindTarget()`,看下相应的源码:


mpany/IMProject] - ~/.gradle/caches/modules-2/files-2.1/q.rorbin/badgeview/1.1.3/be049ff81c4d95503987b0e89367c1201979329c/badgeview-1.1.3-sources.jar!/q/rorb
badgeview
QBadgeView
app
android 10
QBadgeView.java
bindTarget
Match Case
Words
Regex
2 matches
QBadgeView
bindTarget()

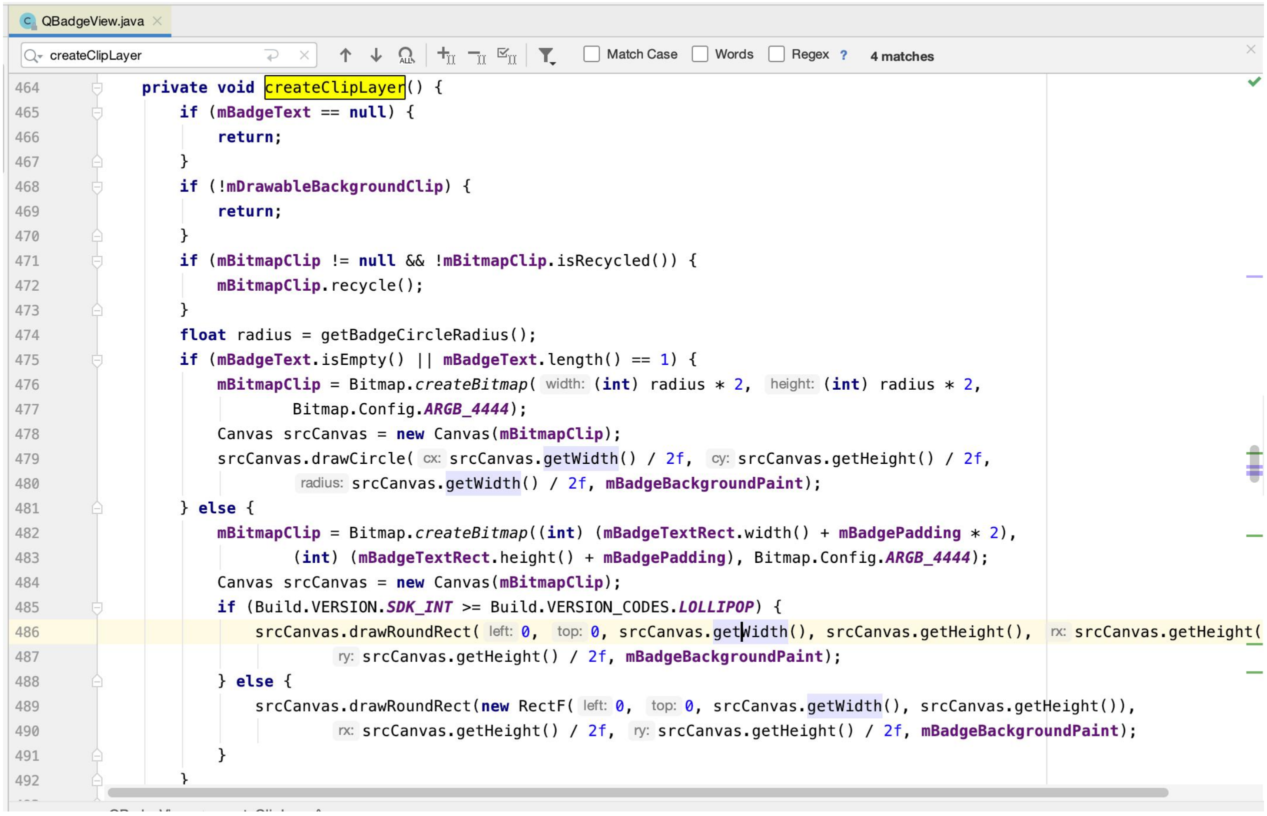
```
    @Override
    public Badge bindTarget(final View targetView) {
        if (targetView == null) {
            throw new IllegalStateException("targetView can not be null");
        }
        if (getParent() != null) {
            ((ViewGroup) getParent()).removeView(this);
        }
        ViewParent targetParent = targetView.getParent();
        if (targetParent != null && targetParent instanceof ViewGroup) {
            mTargetView = targetView;
            if (targetParent instanceof BadgeContainer) {
                ((BadgeContainer) targetParent).addView( child: this);
            } else {
                ViewGroup targetContainer = (ViewGroup) targetParent;
                int index = targetContainer.indexOfChild(targetView);
                ViewGroup.LayoutParams targetParams = targetView.getLayoutParams();
                targetContainer.removeView(targetView);
                final BadgeContainer badgeContainer = new BadgeContainer(getContext());
                if(targetContainer instanceof RelativeLayout){
                    badgeContainer.setId(targetView.getId());
                }
                targetContainer.addView(badgeContainer, index, targetParams);
                badgeContainer.addView(targetView);
                badgeContainer.addView( child: this);
            }
        } else {
            throw new IllegalStateException("targetView must have a parent");
        }
        return this;
    }
```

1.目标View的移除和添加:
先将目标view从父控件中移除,
再构建一个container对象包裹目标View去添加。
2.本身View的移除和添加:
先将自己本身移除,其次 添加。

处理频繁的消息。每次消息的刷新，都频繁经历`remove , add`的操作。

再看下渲染的源码：

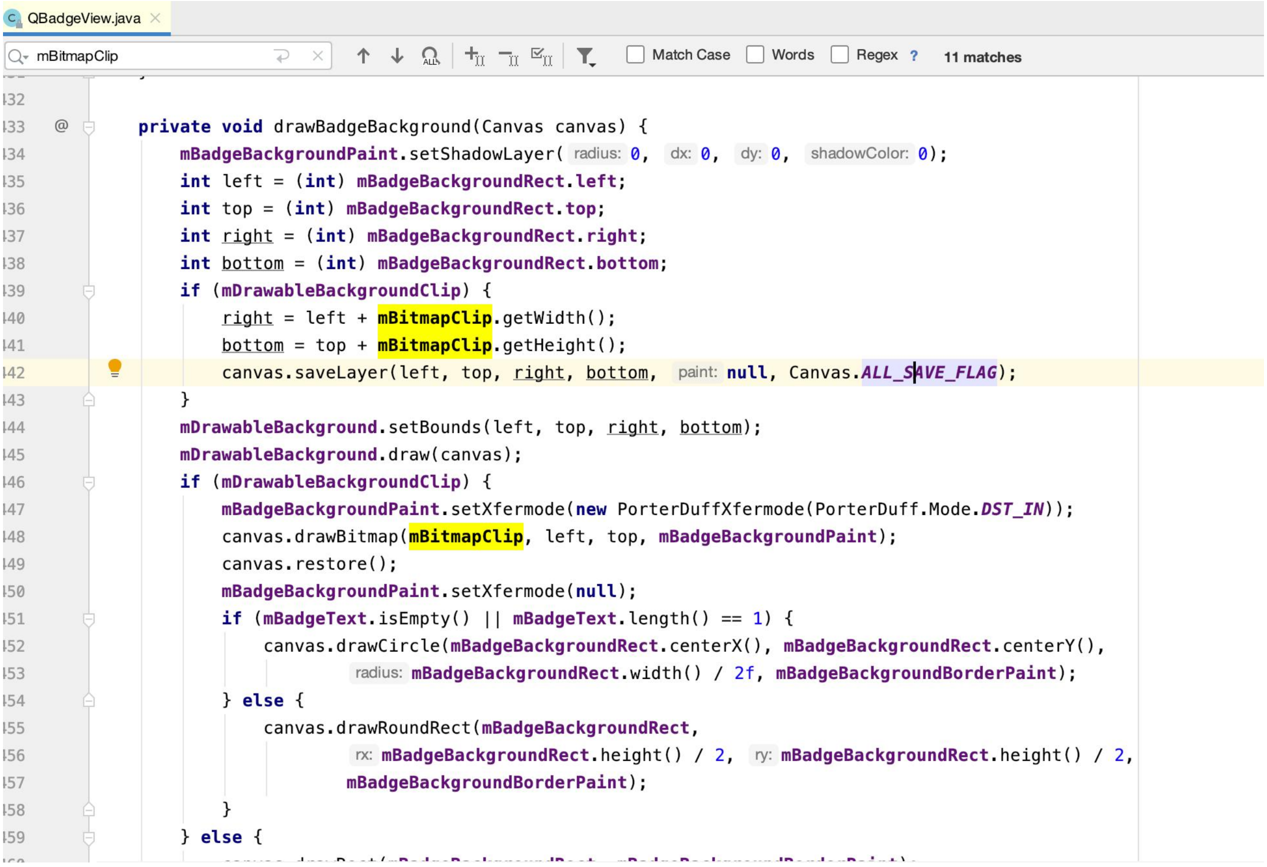
先创建bitmap:


QBadgeView.java

createClipLayer □ Match Case □ Words □ Regex ? 4 matches

```
    private void createClipLayer() {
        if (mBadgeText == null) {
            return;
        }
        if (!mDrawableBackgroundClip) {
            return;
        }
        if (mBitmapClip != null && !mBitmapClip.isRecycled()) {
            mBitmapClip.recycle();
        }
        float radius = getBadgeCircleRadius();
        if (mBadgeText.isEmpty() || mBadgeText.length() == 1) {
            mBitmapClip = Bitmap.createBitmap( width: (int) radius * 2,  height: (int) radius * 2,
                    Bitmap.Config.ARGB_4444);
            Canvas srcCanvas = new Canvas(mBitmapClip);
            srcCanvas.drawCircle( cx: srcCanvas.getWidth() / 2f,  cy: srcCanvas.getHeight() / 2f,
                     radius: srcCanvas.getWidth() / 2f, mBadgeBackgroundPaint);
        } else {
            mBitmapClip = Bitmap.createBitmap((int) (mBadgeTextRect.width() + mBadgePadding * 2),
                    (int) (mBadgeTextRect.height() + mBadgePadding), Bitmap.Config.ARGB_4444);
            Canvas srcCanvas = new Canvas(mBitmapClip);
            if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.LOLLIPOP) {
                srcCanvas.drawRoundRect( left: 0,  top: 0, srcCanvas.getWidth(), srcCanvas.getHeight(),  rx: srcCanvas.getHeight(
                         ry: srcCanvas.getHeight() / 2f, mBadgeBackgroundPaint);
            } else {
                srcCanvas.drawRoundRect(new RectF( left: 0,  top: 0, srcCanvas.getWidth(), srcCanvas.getHeight()),
                         rx: srcCanvas.getHeight() / 2f,  ry: srcCanvas.getHeight() / 2f, mBadgeBackgroundPaint);
            }
        }
```

接着通过Bitmap渲染bg背景：



```java
private void drawBadgeBackground(Canvas canvas) {
    mBadgeBackgroundPaint.setShadowLayer( radius: 0,  dx: 0,  dy: 0,  shadowColor: 0);
    int left = (int) mBadgeBackgroundRect.left;
    int top = (int) mBadgeBackgroundRect.top;
    int right = (int) mBadgeBackgroundRect.right;
    int bottom = (int) mBadgeBackgroundRect.bottom;
    if (mDrawableBackgroundClip) {
        right = left + mBitmapClip.getWidth();
        bottom = top + mBitmapClip.getHeight();
        canvas.saveLayer(left, top, right, bottom,  paint: null, Canvas.ALL_SAVE_FLAG);
    }
    mDrawableBackground.setBounds(left, top, right, bottom);
    mDrawableBackground.draw(canvas);
    if (mDrawableBackgroundClip) {
        mBadgeBackgroundPaint.setXfermode(new PorterDuffXfermode(PorterDuff.Mode.DST_IN));
        canvas.drawBitmap(mBitmapClip, left, top, mBadgeBackgroundPaint);
        canvas.restore();
        mBadgeBackgroundPaint.setXfermode(null);
        if (mBadgeText.isEmpty() || mBadgeText.length() == 1) {
            canvas.drawCircle(mBadgeBackgroundRect.centerX(), mBadgeBackgroundRect.centerY(),
                     radius: mBadgeBackgroundRect.width() / 2f, mBadgeBackgroundBorderPaint);
        } else {
            canvas.drawRoundRect(mBadgeBackgroundRect,
                     rx: mBadgeBackgroundRect.height() / 2,  ry: mBadgeBackgroundRect.height() / 2,
                    mBadgeBackgroundBorderPaint);
        }
    } else {
```

导致渲染线程奔溃的推测的原因：canvas画布绘制drawable对象无效指针。

## 2.团队的新成本

- 如何做到监控
- 如何做到相应的文档，进行团队培训
- 如何合理安排，学习精力

## 3.技术的风险

- 新技术与旧技术的优劣
    - 新技术的优势在哪里，是否显著?
    - 在全部解决方案中，当前技术的优势体现在哪里?
- 新技术的潜藏风险(带来哪些新问题，能否解决掉?)
    - 会带来哪些新问题，是否能够解决掉?
    - 是否存在性能问题、安全问题?
- 新技术的发展前景
    - 是否有持续投入的人或者社区
    - 问题的解决速度
    - 源码质量
    - 文档质量
    - 开源协议