**1.Window animate starting**

1. Staring 窗口的启动：

AMS调用setAppStartingWindow来创建starting窗口。



在setAppStartingWindow函数中创建StartingData数据结构发送一个ADD\_STARTING消息到android.display线程中。

addStartingWindow函数的解析：

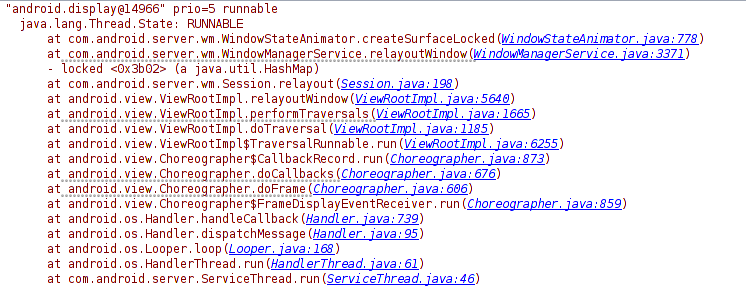
创建title为Starting + packageName的PhoneWindow对象，获取其DecorView root view通过addView函数加入到WindowManager中。



StartingWindow在setview的时候请求一个vsync信号进行relayoutwindow动作：

在relayoutwindow中mDrawState 进入DRAW\_PENDING状态。



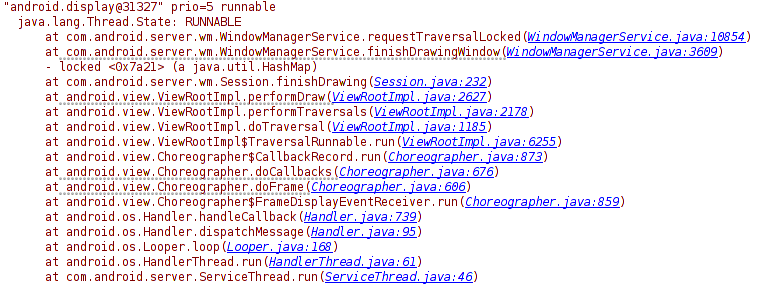


在android.display线程做完relayout window 之后，StartingWindow进行绘制的请求，当绘制完成后调用finishDrawing修改mDrawState 进入COMMIT\_DRAW\_PENDING状态。

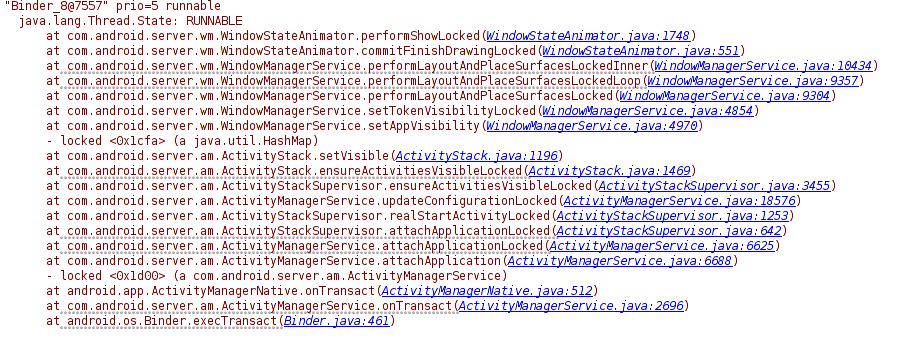




开始StartingWindow的performLayoutAndPlaceSurfacesLockedInner请求。



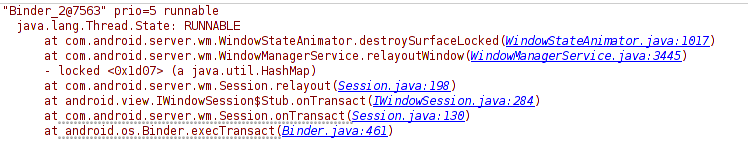
在performLayoutAndPlaceSurfacesLockedInner函数中当w.mHasSurface为ture的时候调用winAnimator.commitFinishDrawingLocked()函数检查该窗口是否处于COMMIT\_DRAW\_PENDING或者READY\_TO\_SHOW的状态，当处于该状态的时候将mDrawState 置为READY\_TO\_SHOW状态。



窗口绘制完成显示的时间点：

在handleAppTransitionReadyLocked函数中为starting窗口加载窗口动画，为窗口动画做好准备。

退出的窗口做relayout动作：



1. Starting 窗口的remove动作：

当真正启动的窗口完成了绘制的动作，进行starting 窗口的remove的动作。