



Google
Developers



For Butter or Worse

Smoothing Out Performance in Android UIs

Chet Haase
Romain Guy
Android UI Toolkit Engineers





jank, noun

1. Choppy performance

*“Swiping the home screen feels **janky**”*

2. Discontinuous, surprising experience

*“What's with the **jank** launching that app?”*



butter, tasty noun

1. Smooth performance

*“Home screen swiping is very **buttery**”*

2. Fattening spread and cooking ingredient

*“America’s obesity problem is directly proportional to the deliciousness of **butter**”*



jank, noun

1. Choppy performance

*“Swiping the home screen feels **janky**”*

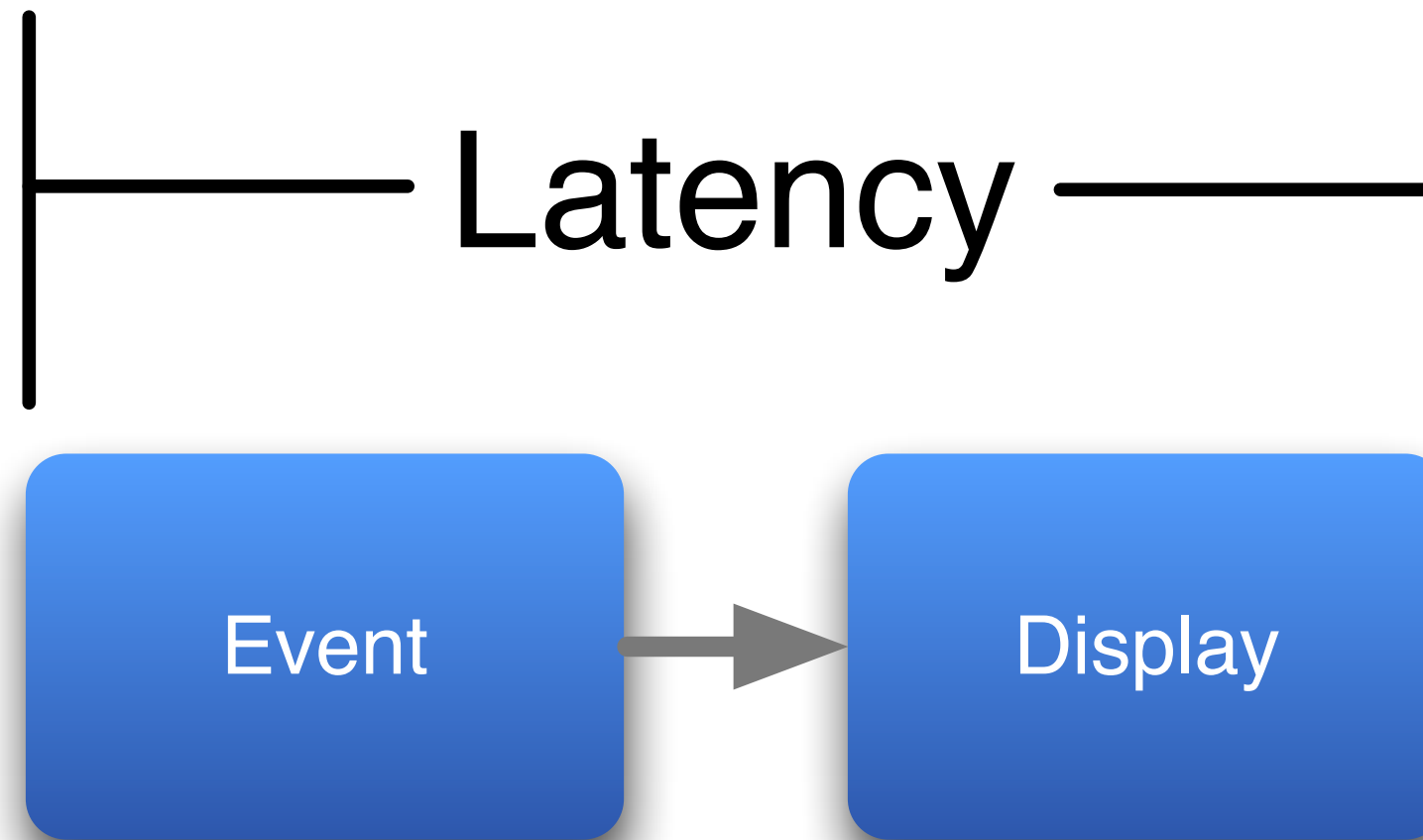
2. Discontinuous, surprising experience

*“What's with the **jank** launching that app?”*



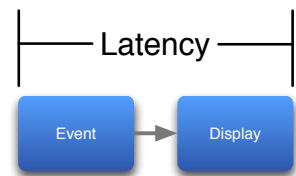
Recipe

- Butter for *eating* is made from cream
- Butter for *Android* is made from
 - low latency
 - fast, consistent framerate



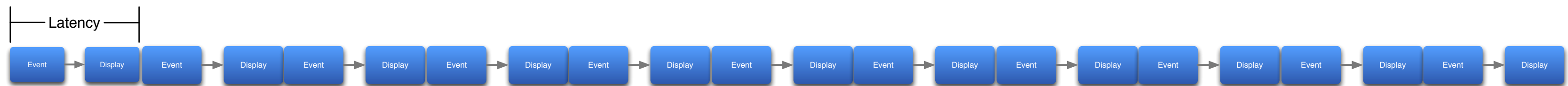
Recipe

- Butter for *eating* is made from cream
- Butter for *Android* is made from
 - low latency
 - fast, consistent framerate

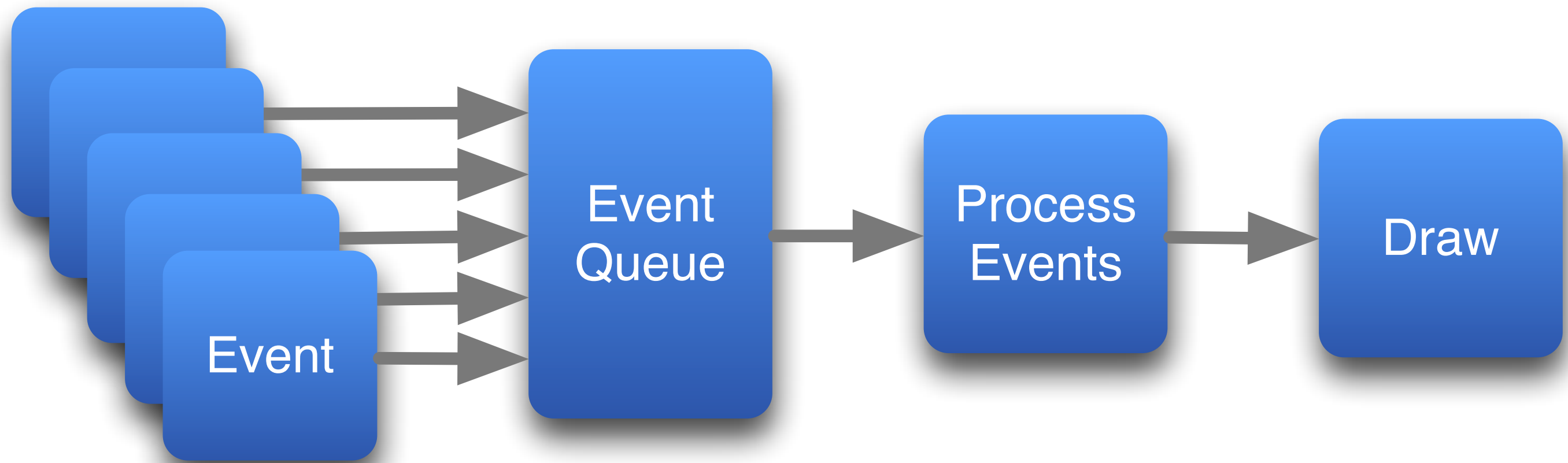


Recipe

- Butter for *eating* is made from cream
- Butter for *Android* is made from
 - low latency
 - fast, consistent framerate



Input Latency



Input Lag (What a drag)



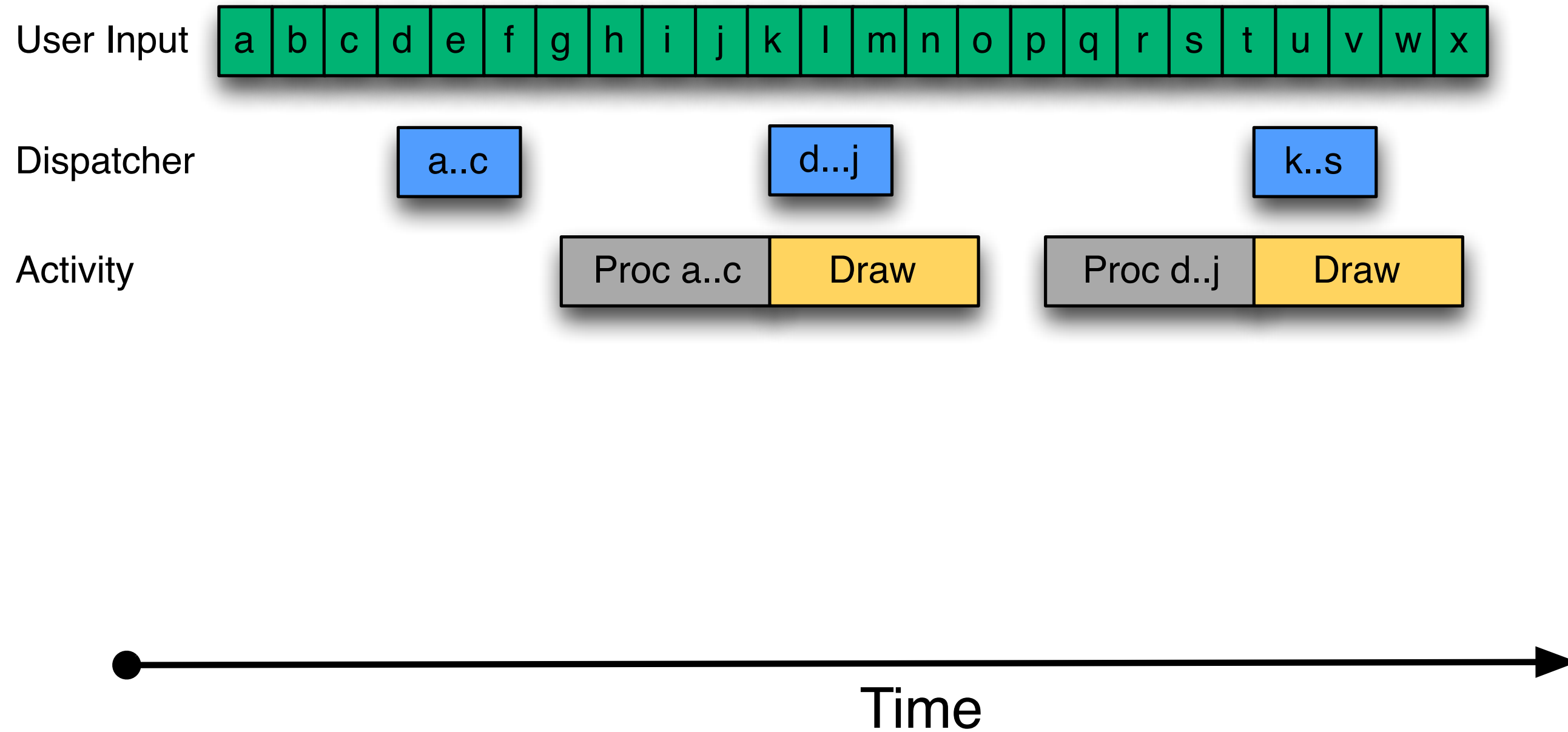
Input Lag (What a drag)



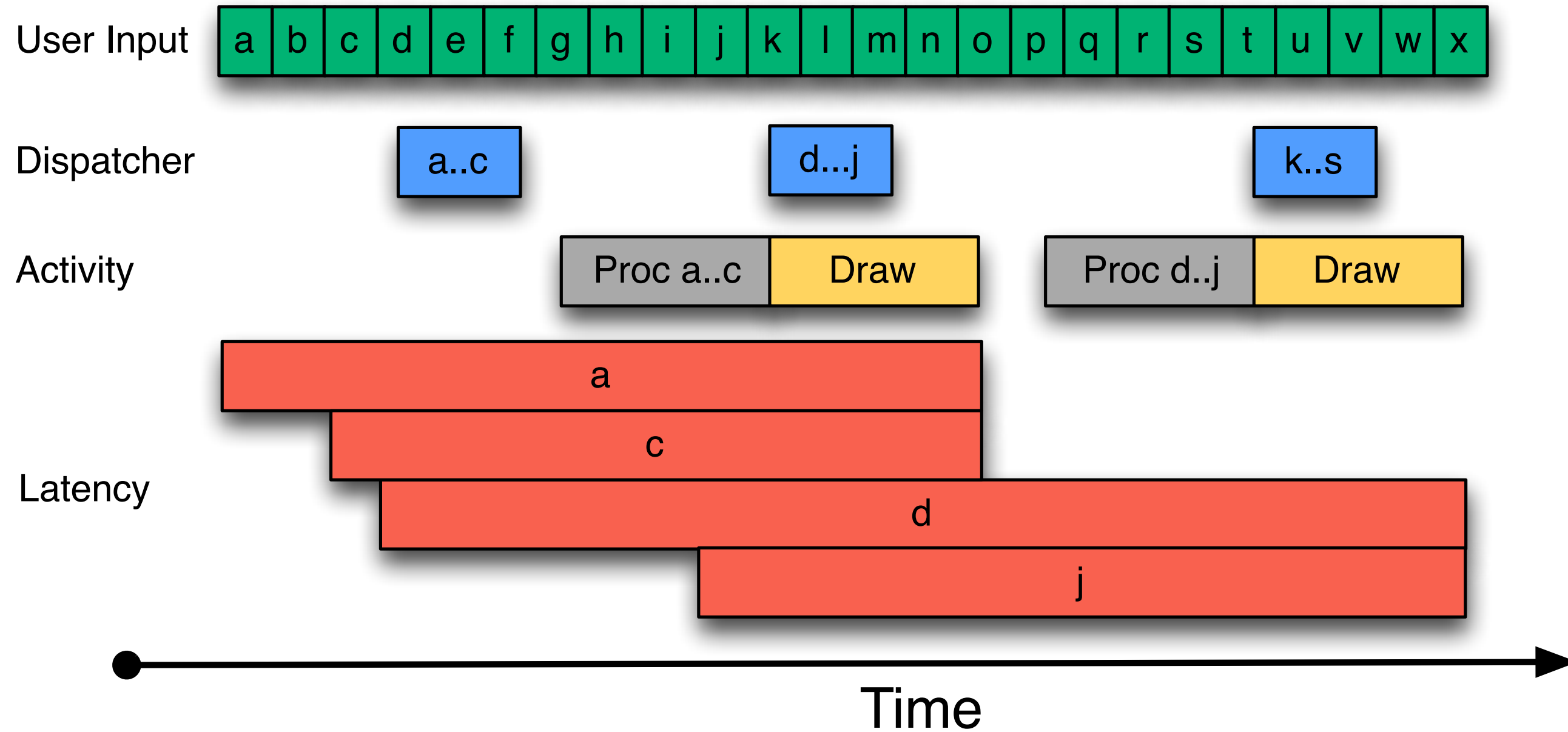
Input Lag (What a drag)



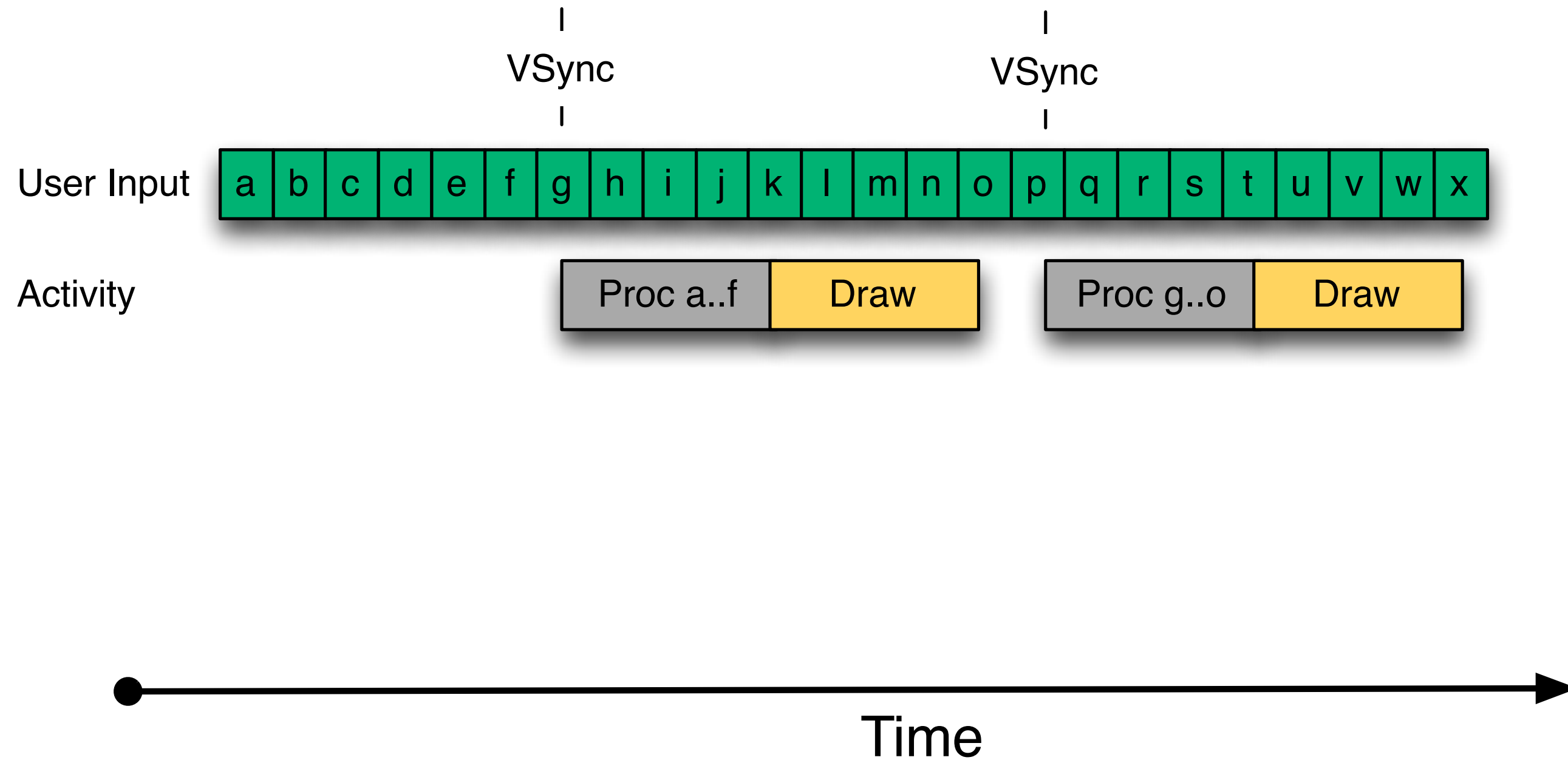
Input Latency



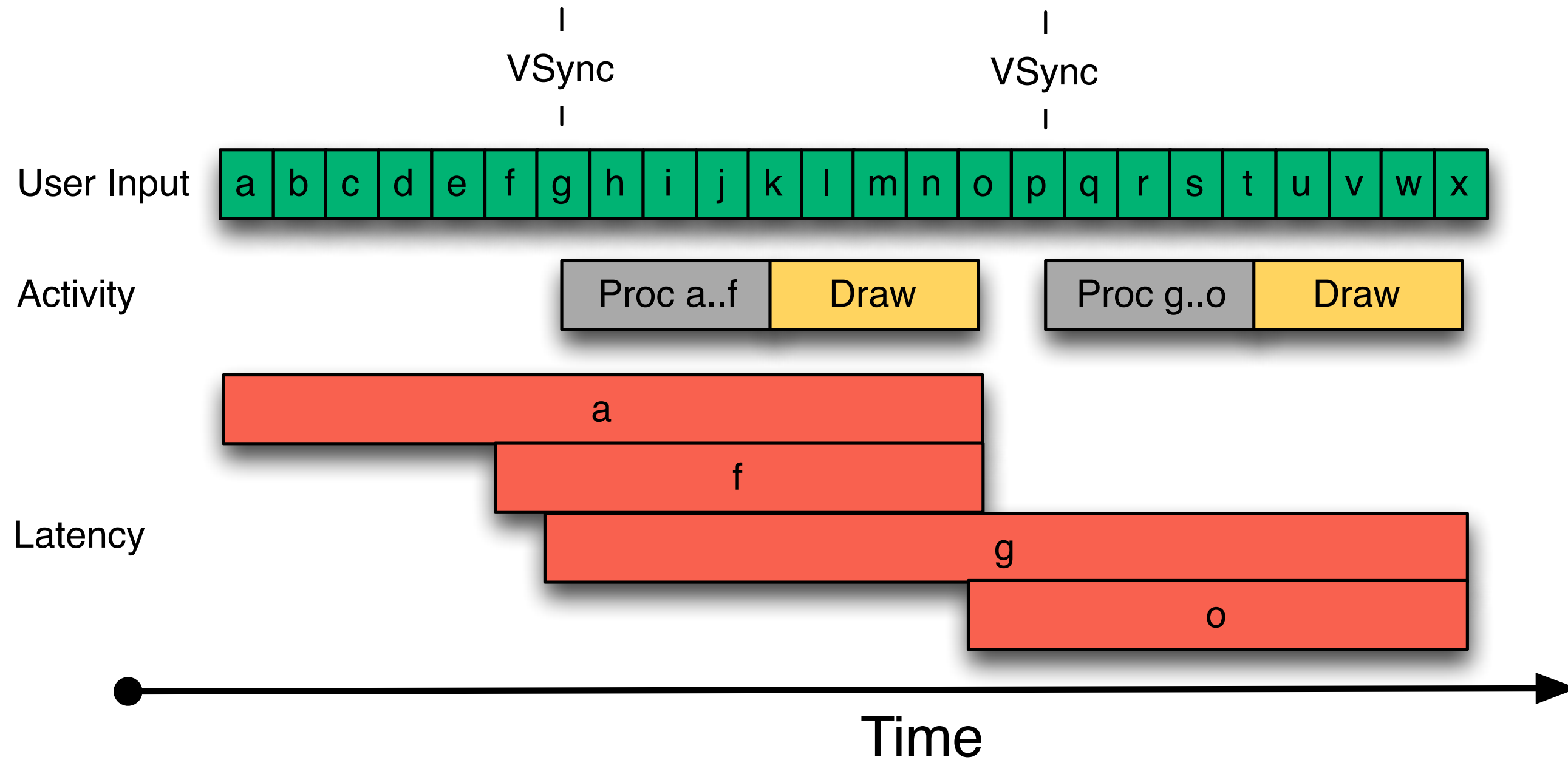
Input Latency



Event Streaming



Event Streaming



Framerate

Drawing faster is a good start

Drawing



Framerate

Drawing faster is a good start

Drawing



Framerate

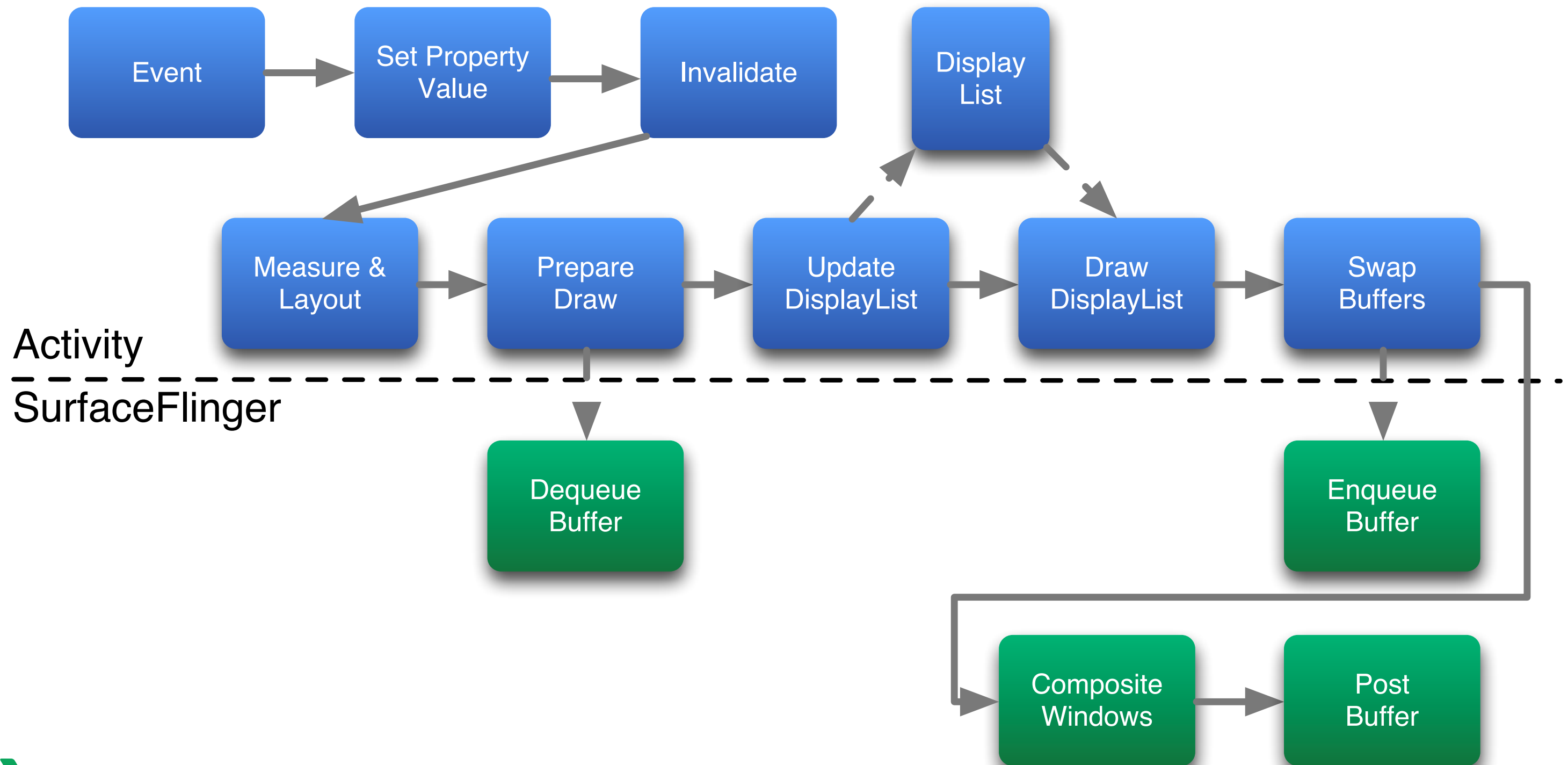
Drawing faster is a good start

Drawing

But that's not quite all there is to it...



Drawing: The Big Picture



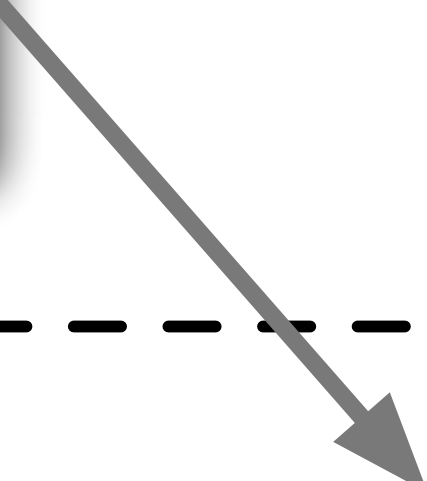
Something Happens

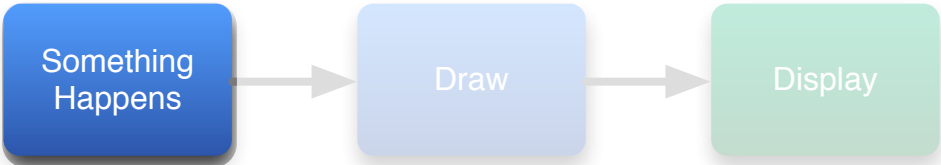
Draw

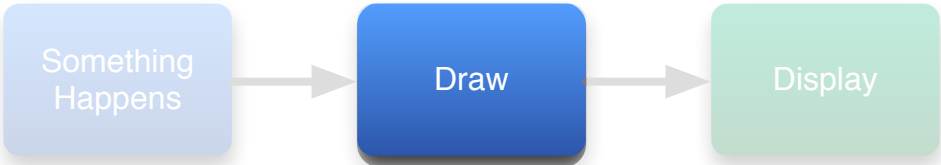
Display

Activity

SurfaceFlinger

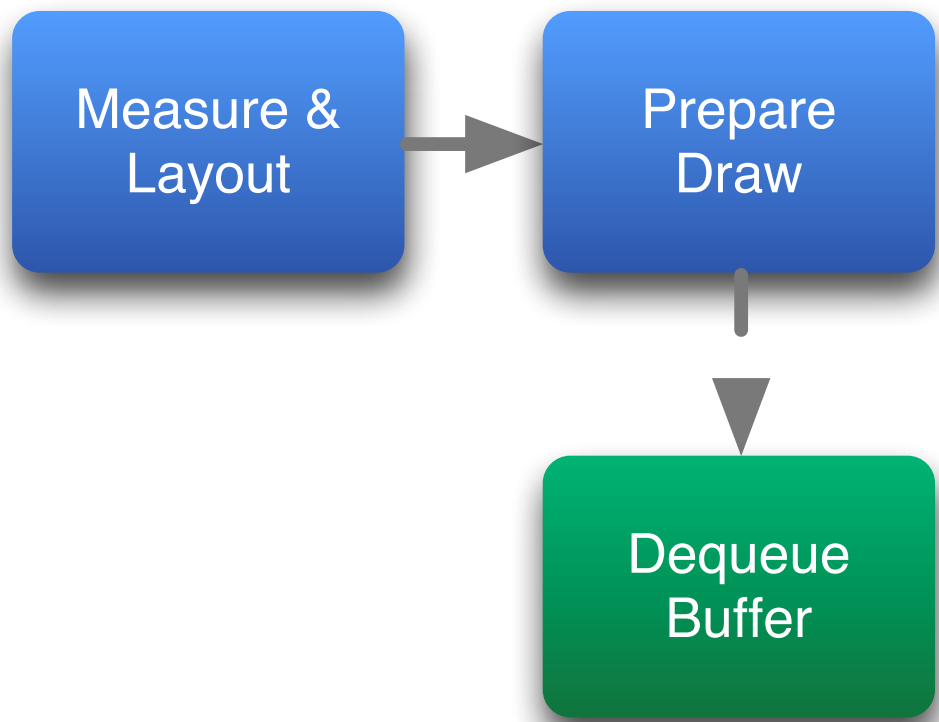
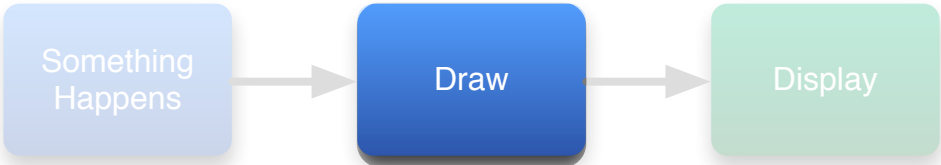


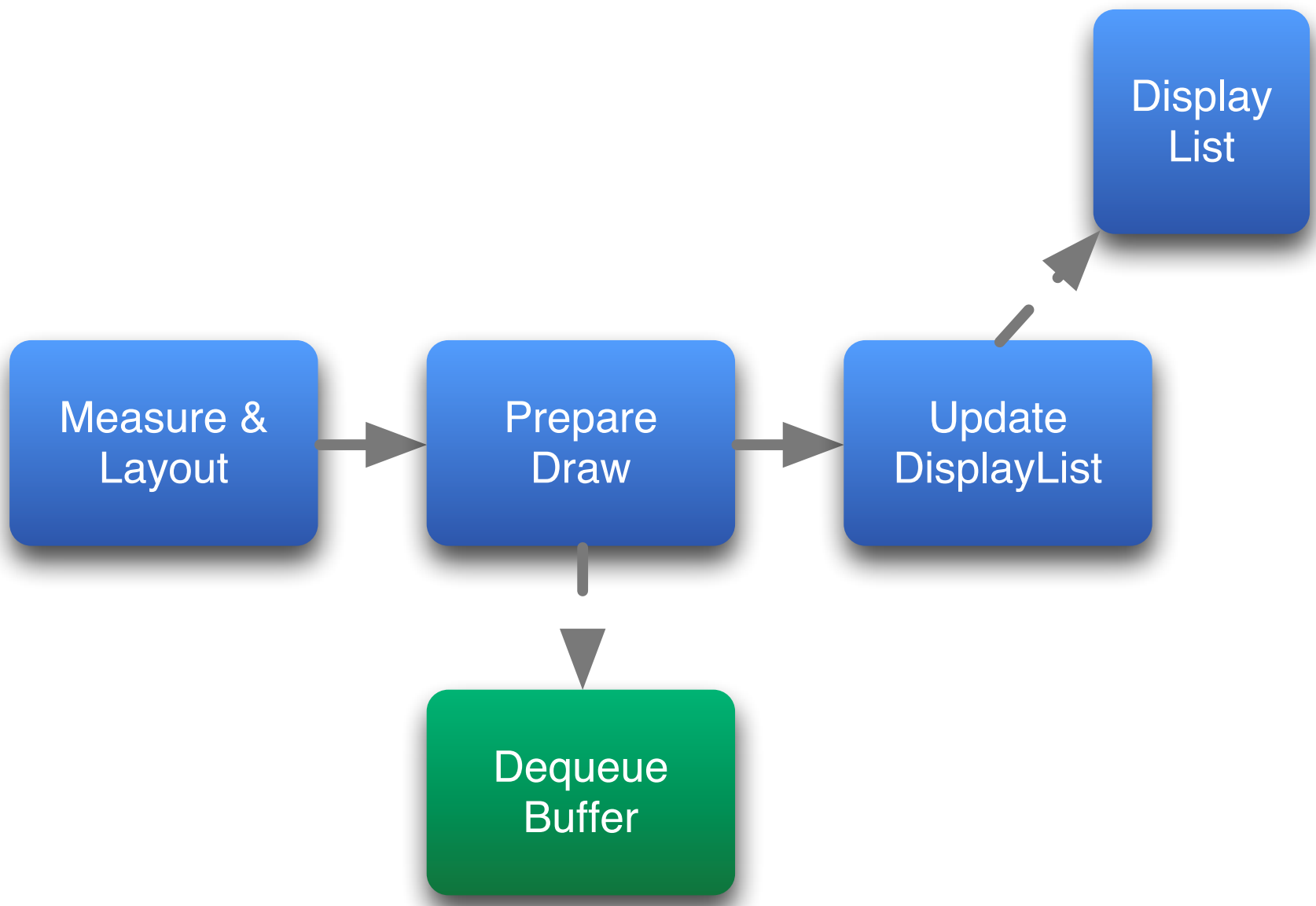
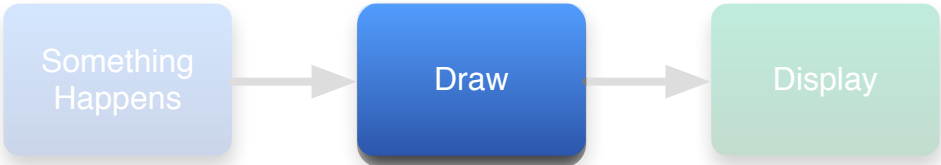


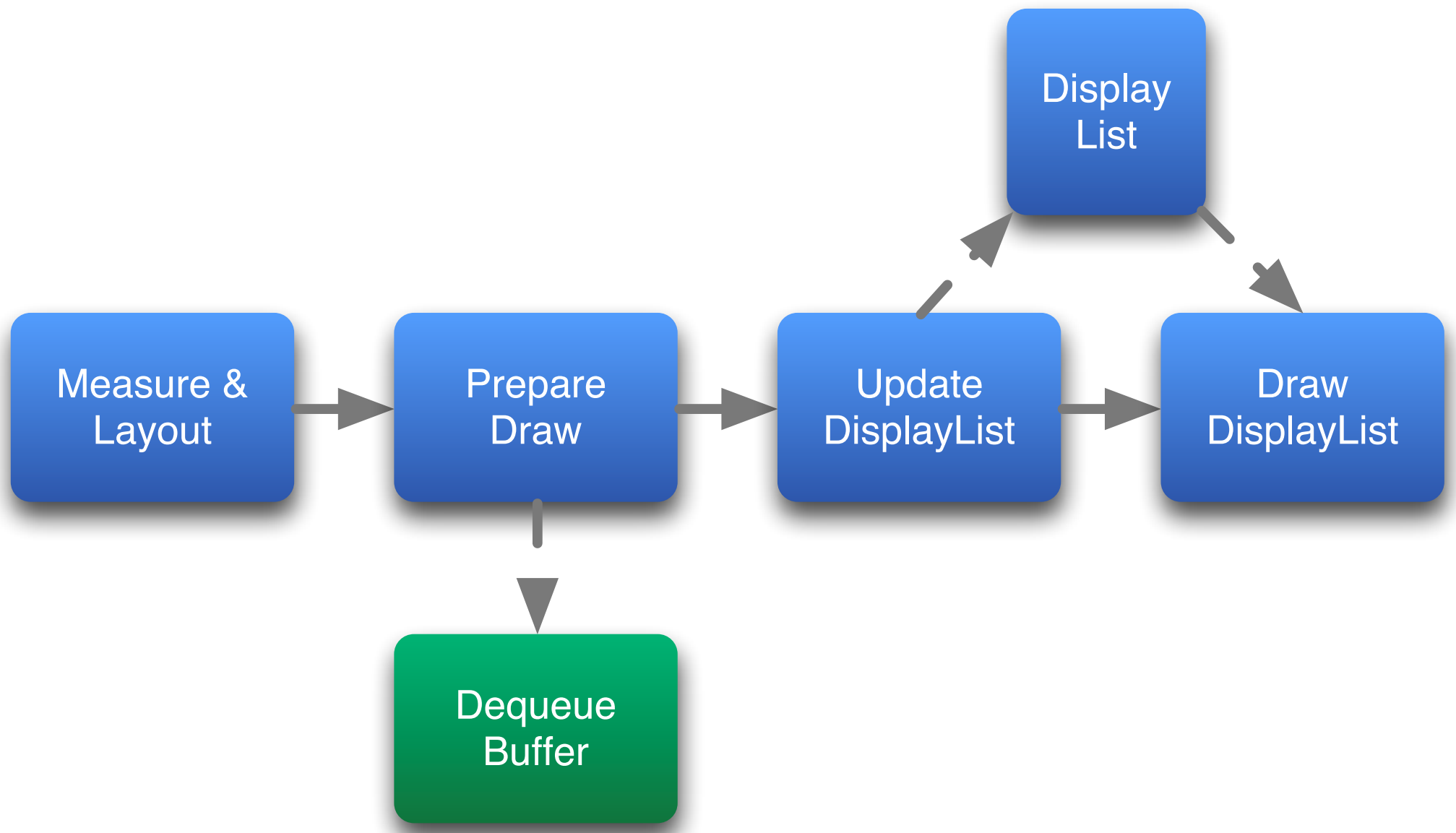
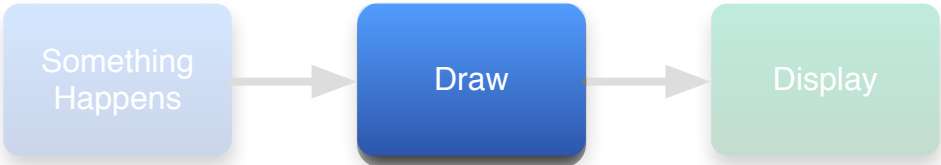


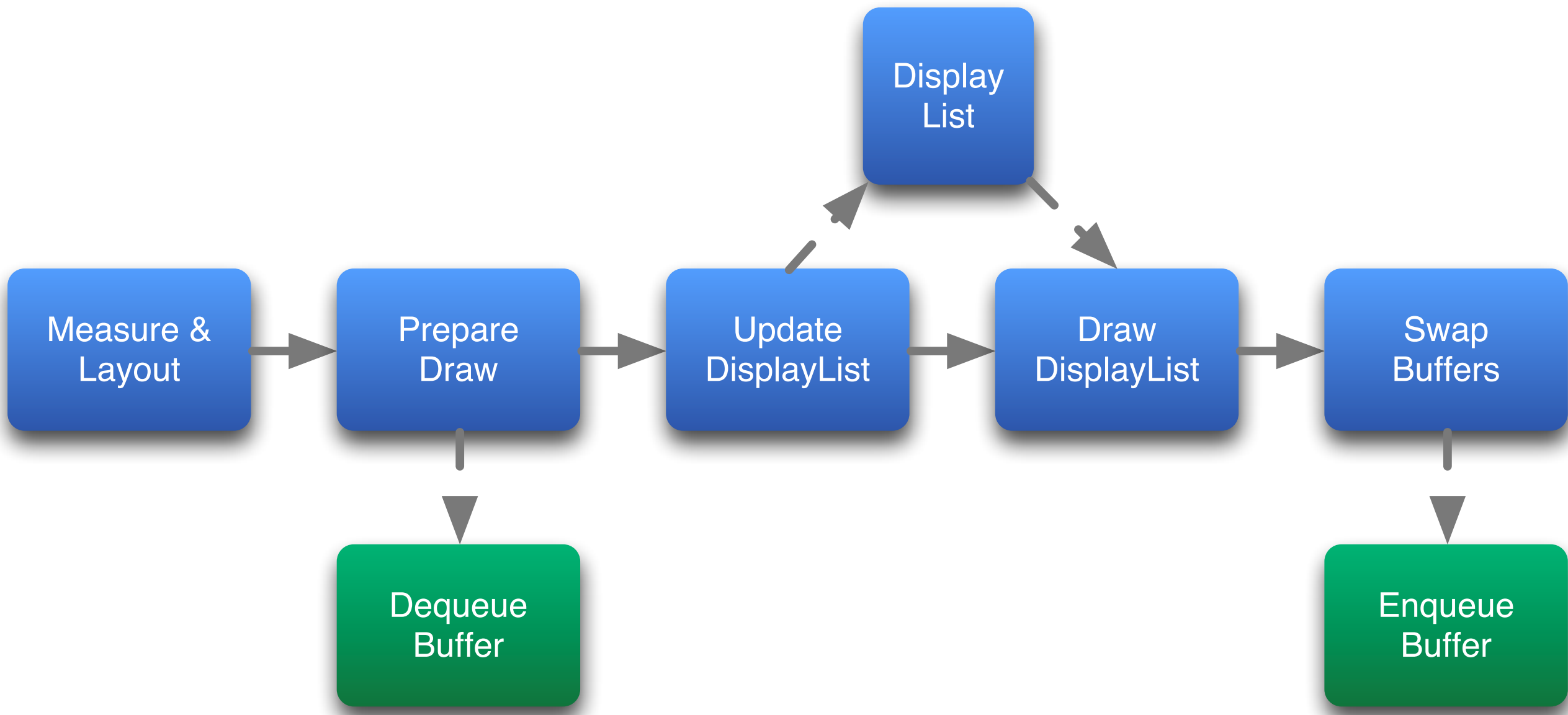
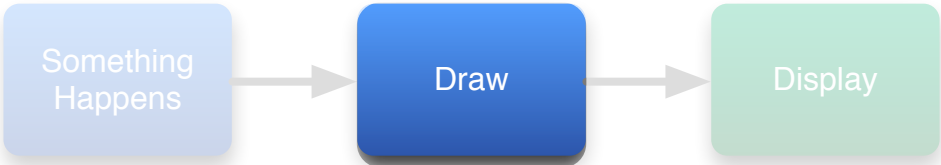
Measure &
Layout

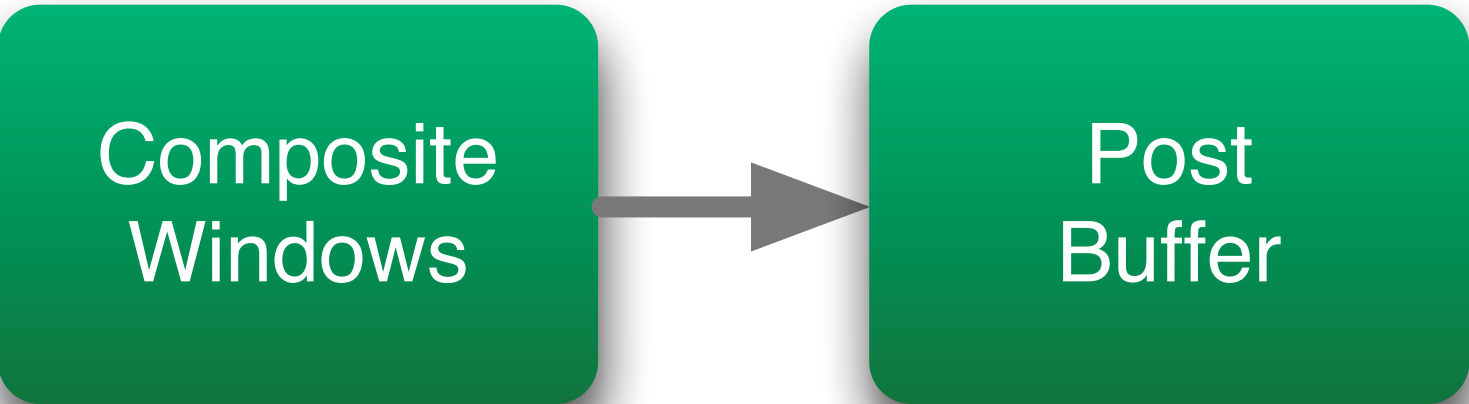
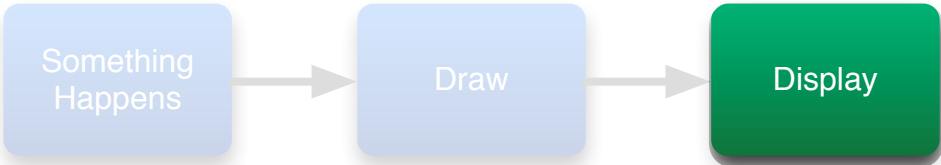


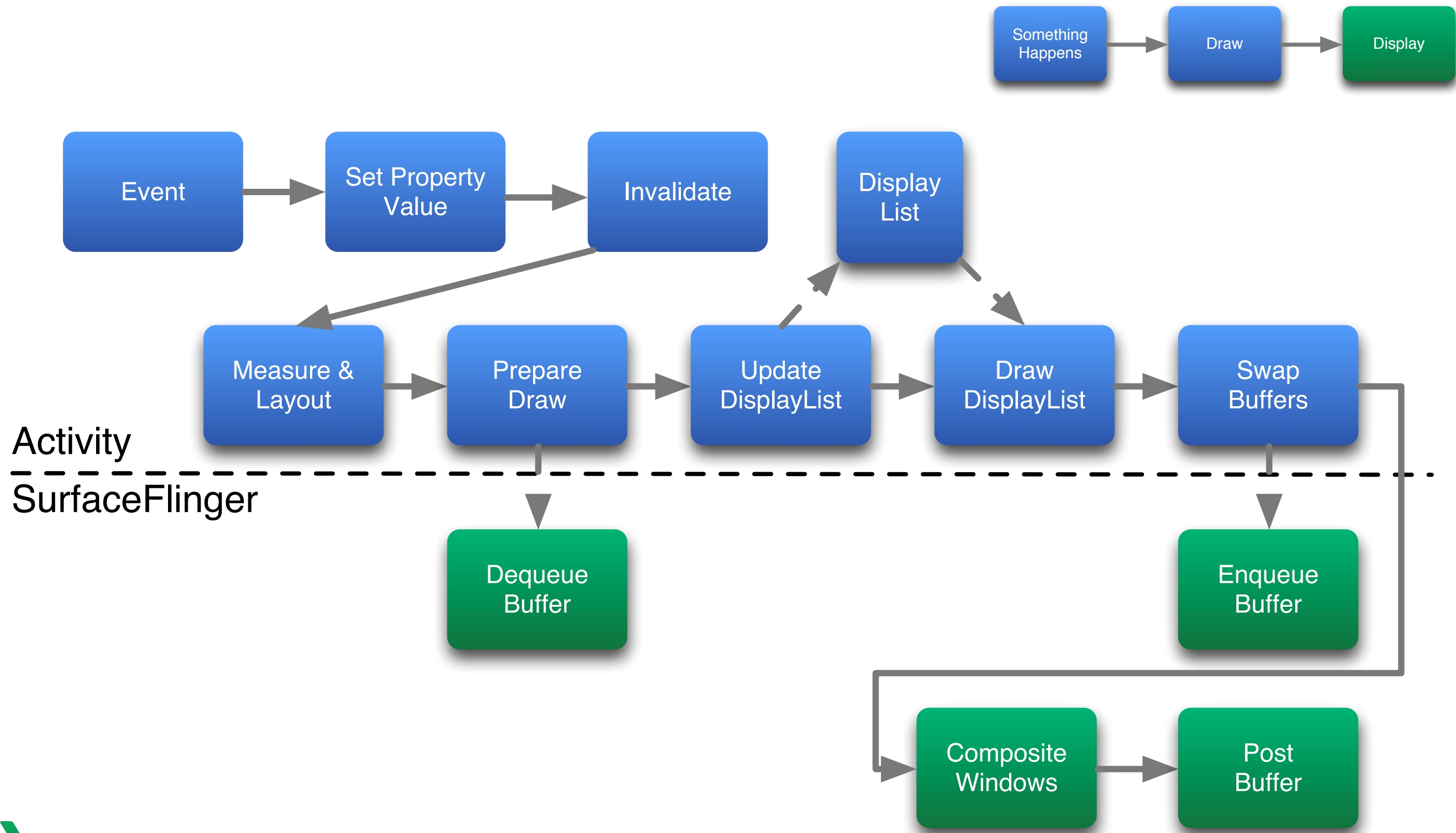














VSync

Synchronizing rendering with the display refresh

60 fps

Displays refresh at 60 Hz, allowing apps to render at 60fps



60 Hz



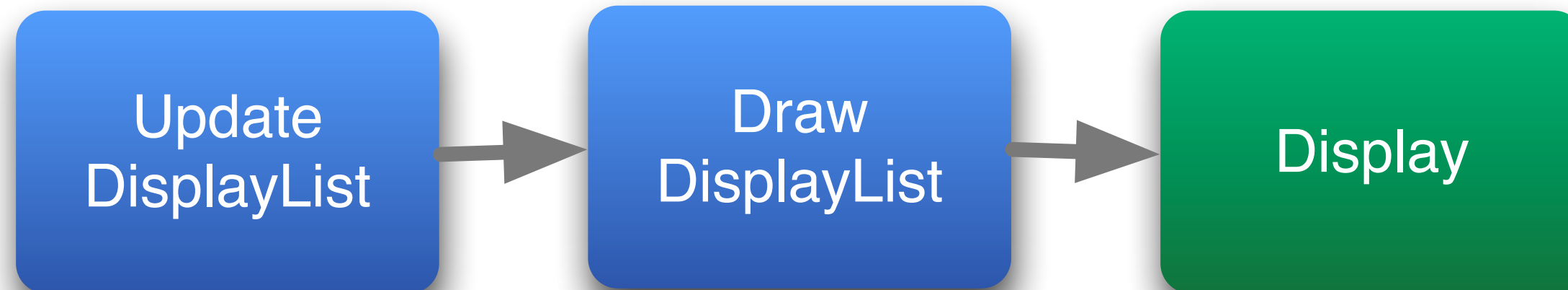


60 Hz

Tearing



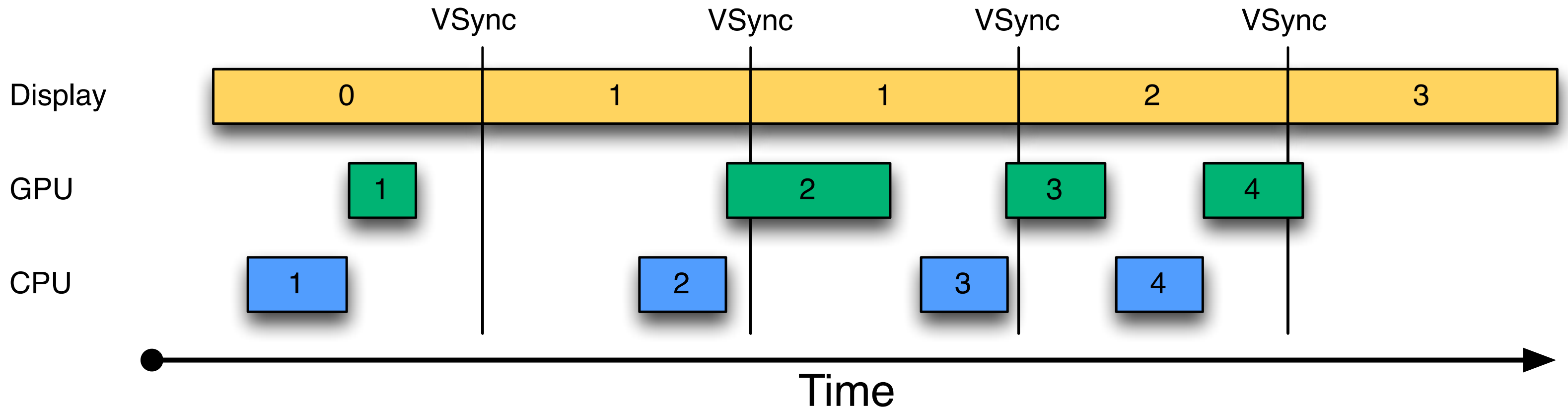
Getting the Pixels to the Screen



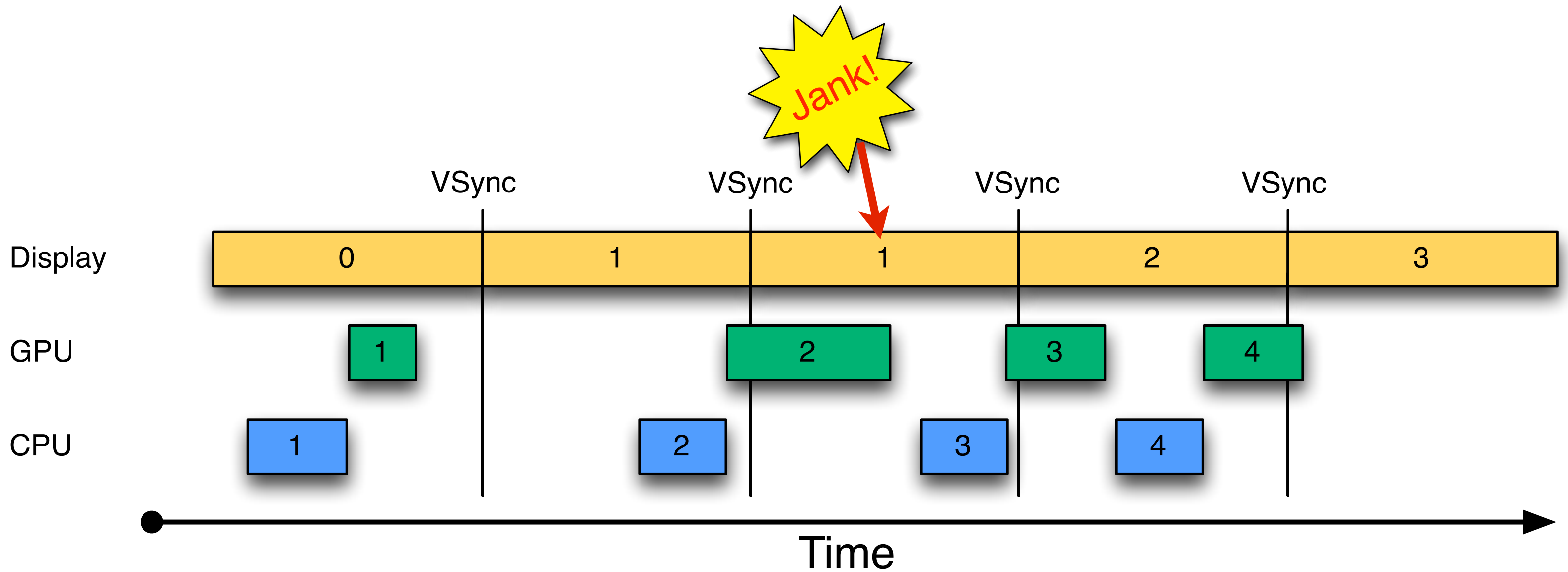
Time



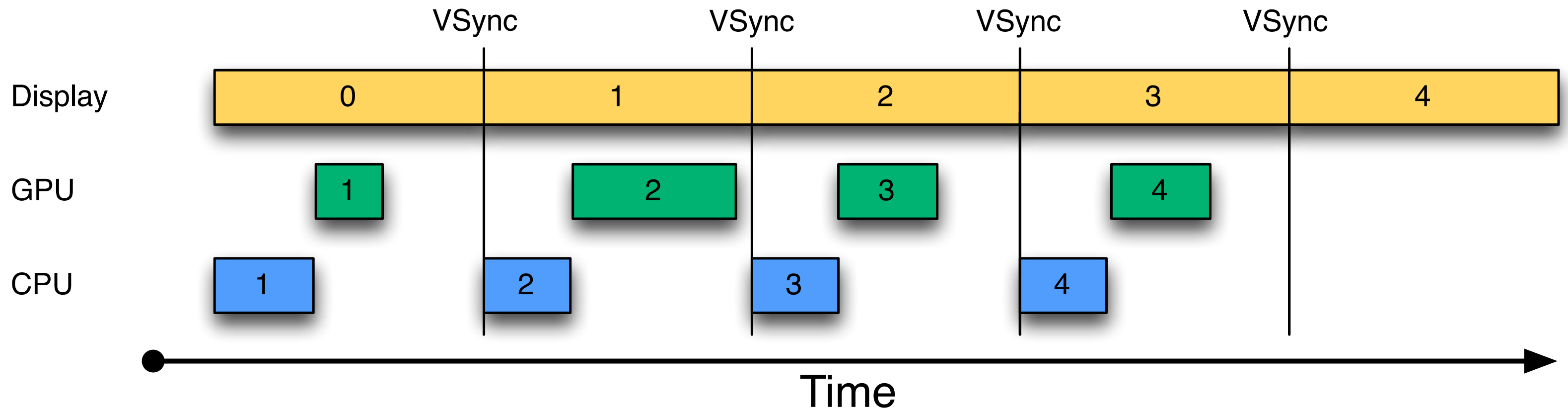
Drawing without VSync



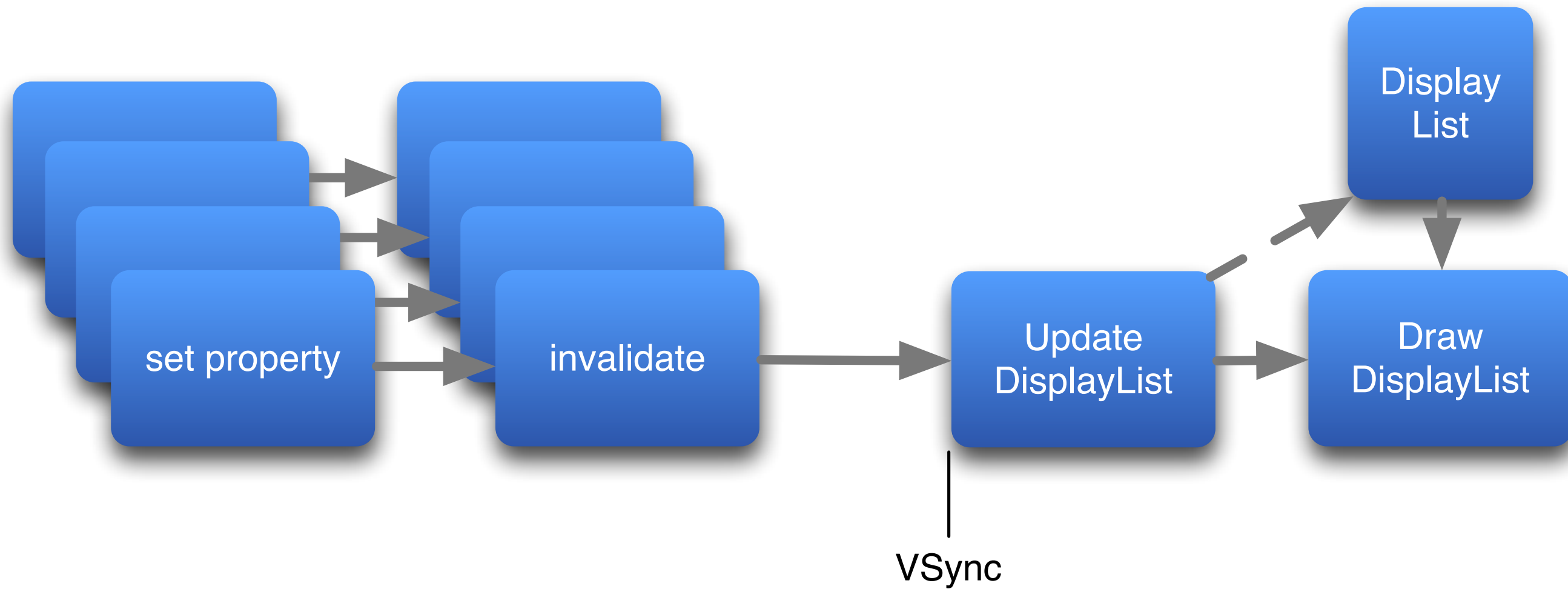
Drawing without VSync



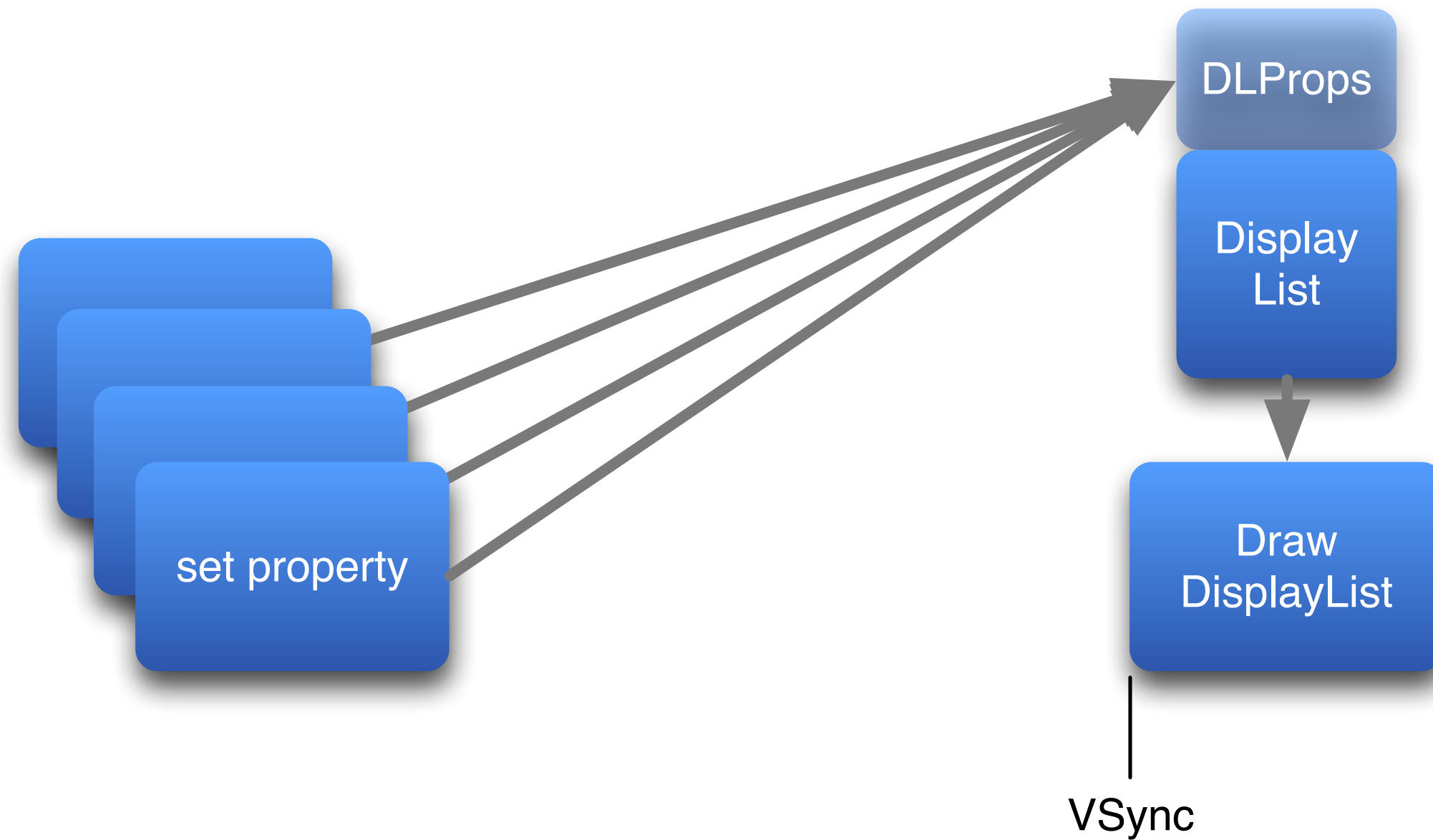
Drawing with VSync



Drawing with DisplayLists



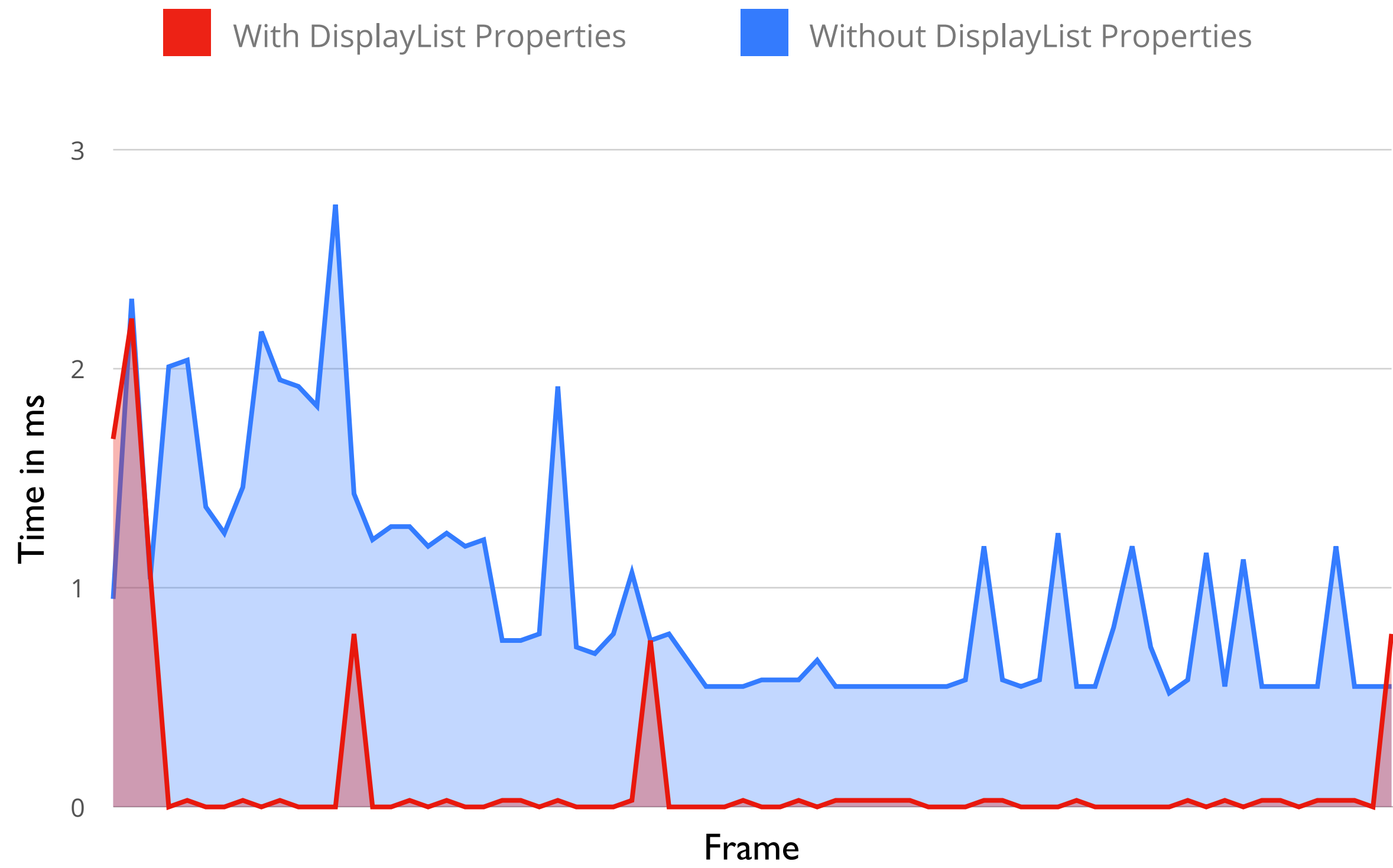
Drawing with DisplayList Properties



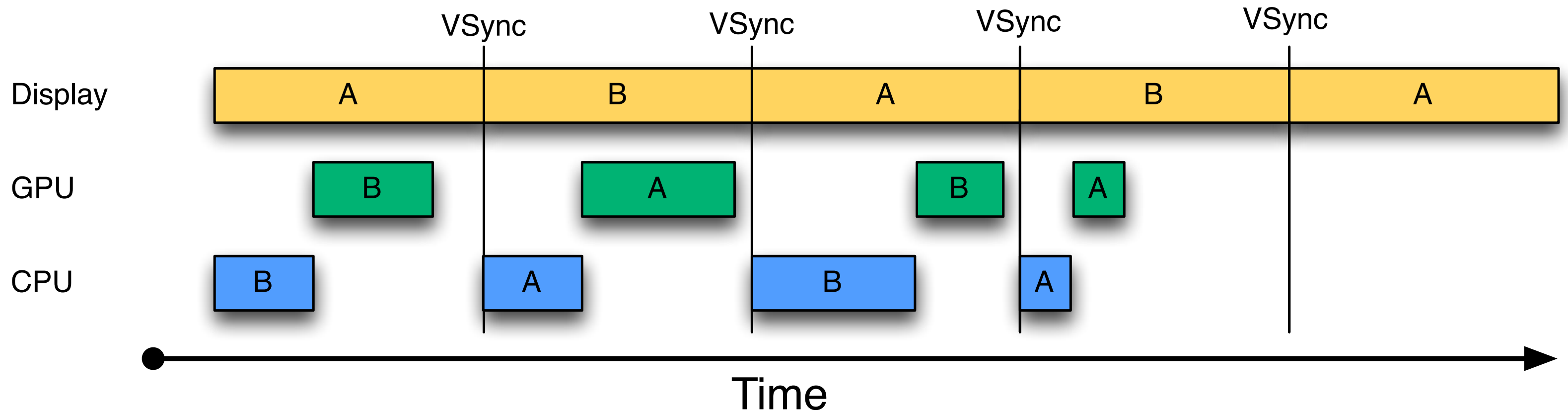
DisplayList Properties

- Free with ViewPropertyAnimator
- Or use ObjectAnimator with View properties
 - alpha
 - translationX/Y
 - scaleX/Y
 - rotation/X/Y

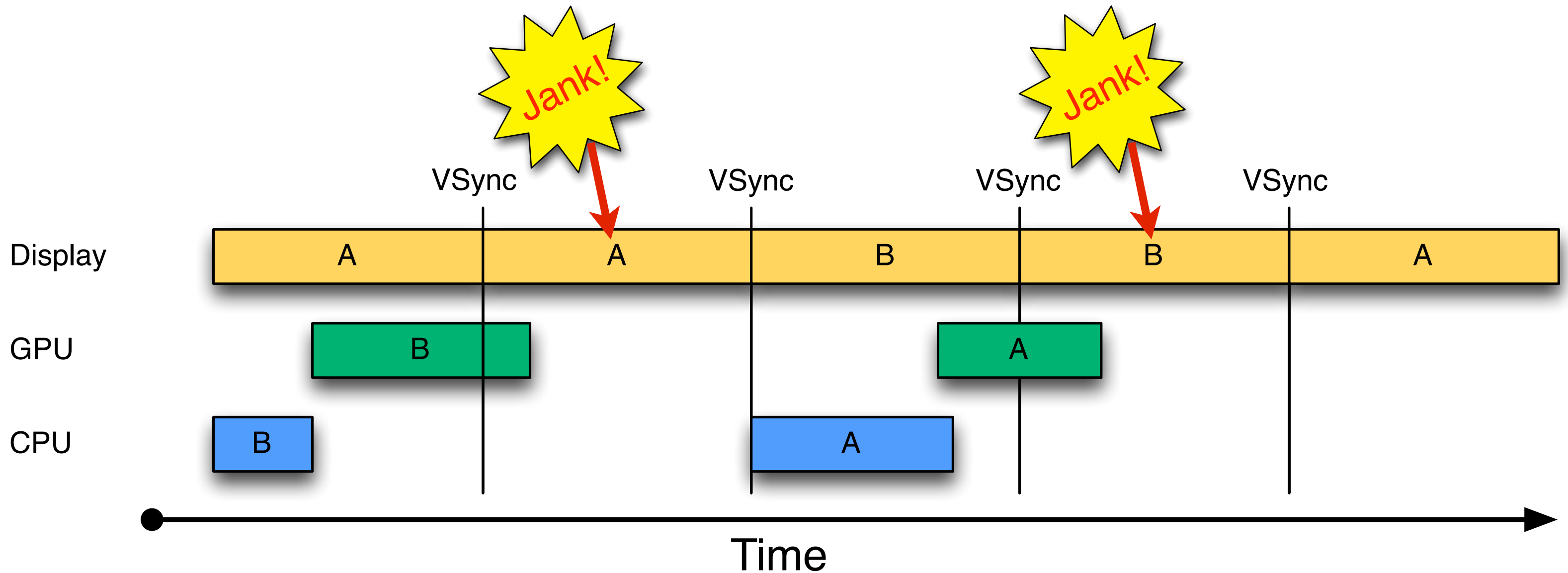




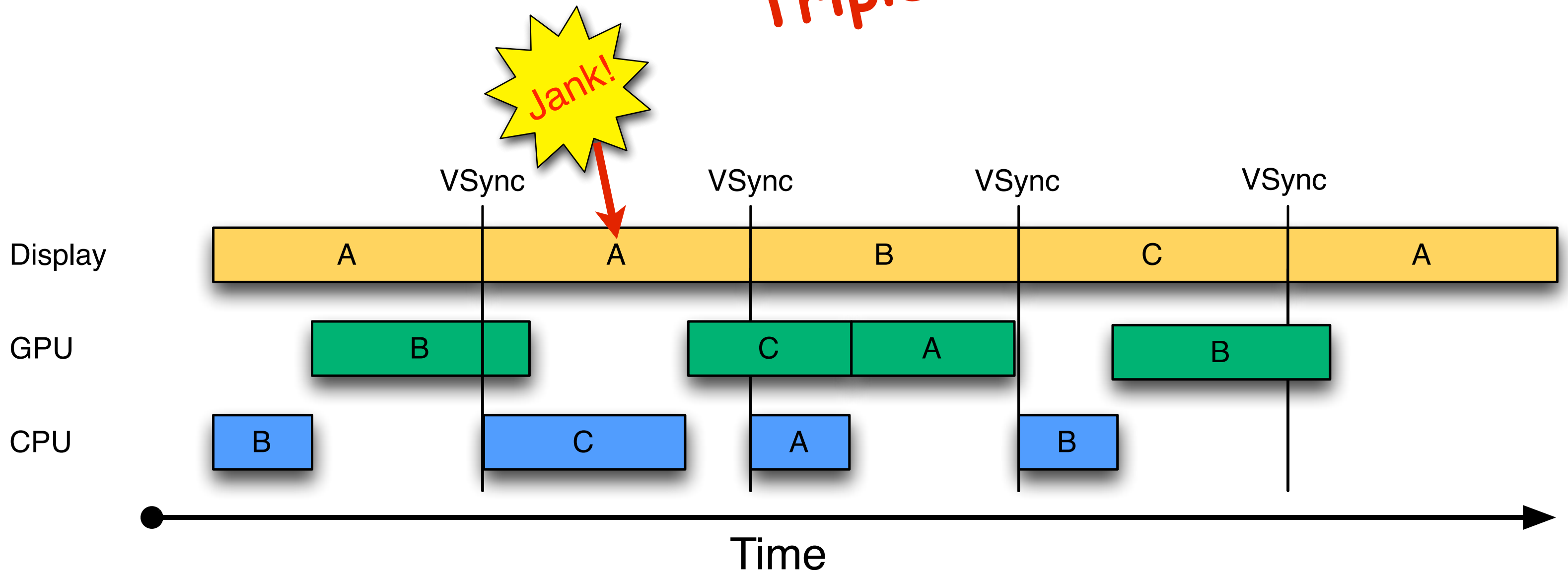
Parallel Processing and Double Buffering



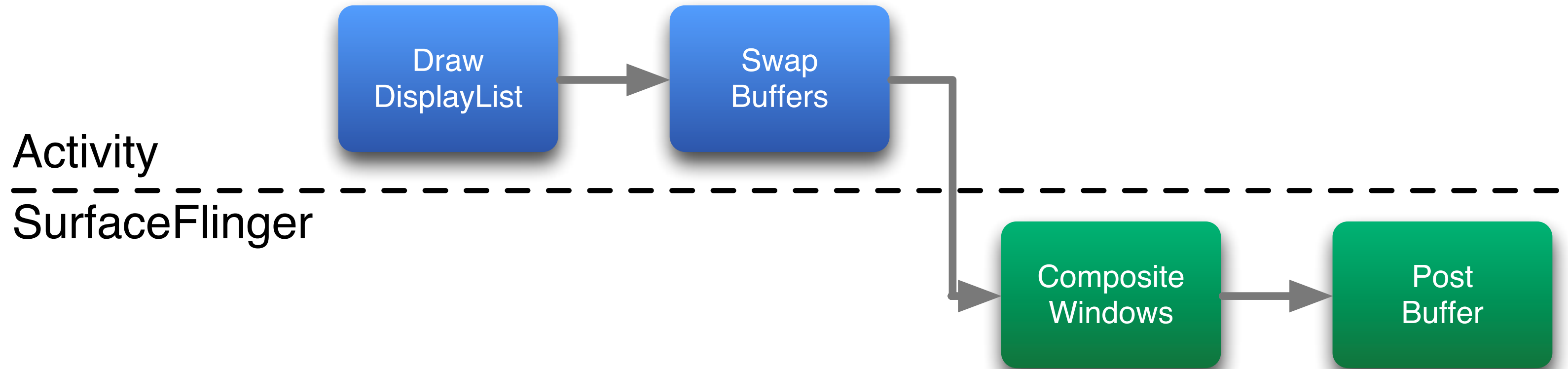
Parallel Processing and Double Buffering



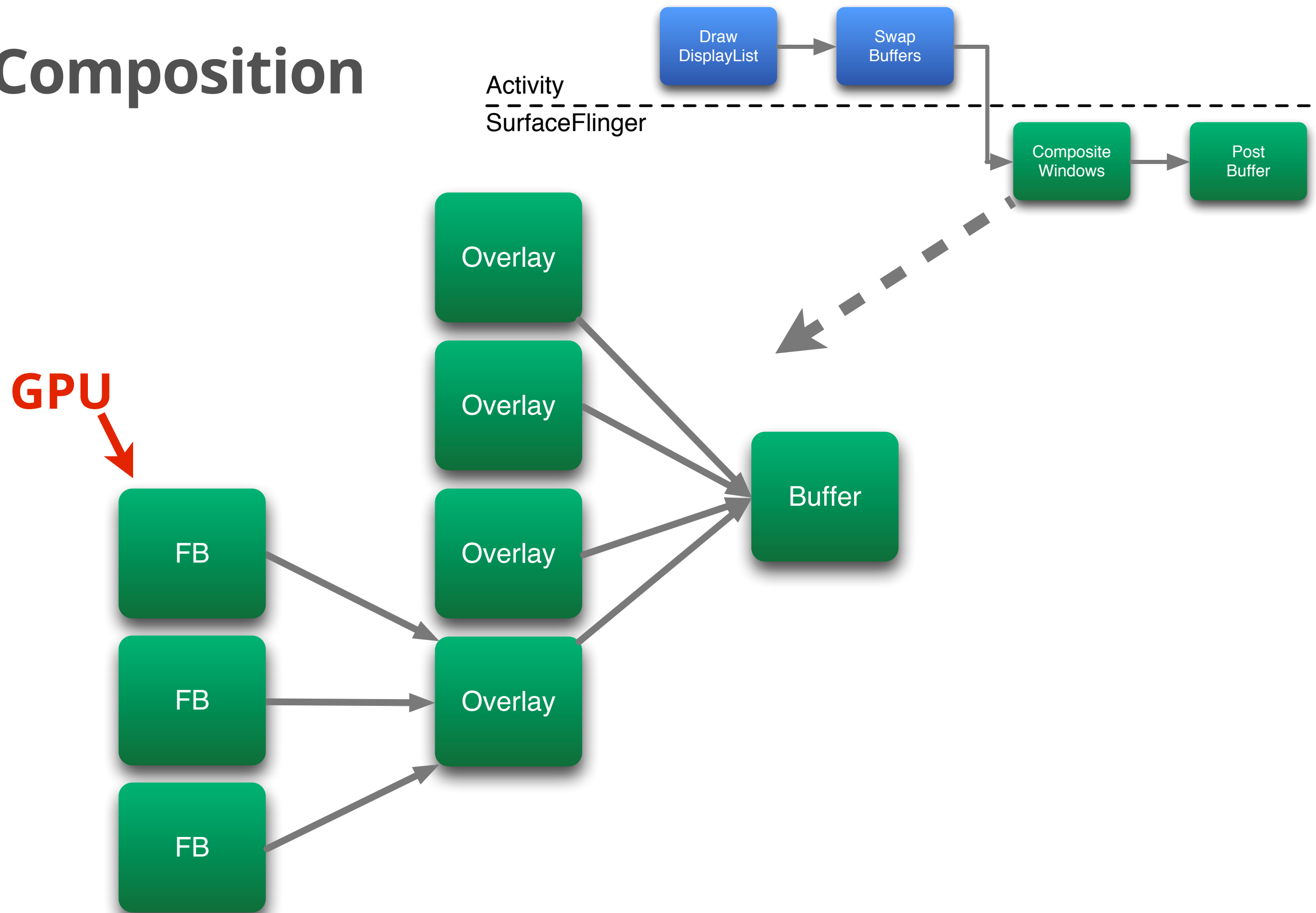
Parallel Processing and ~~Double~~ Triple Buffering



Window Composition



Window Composition

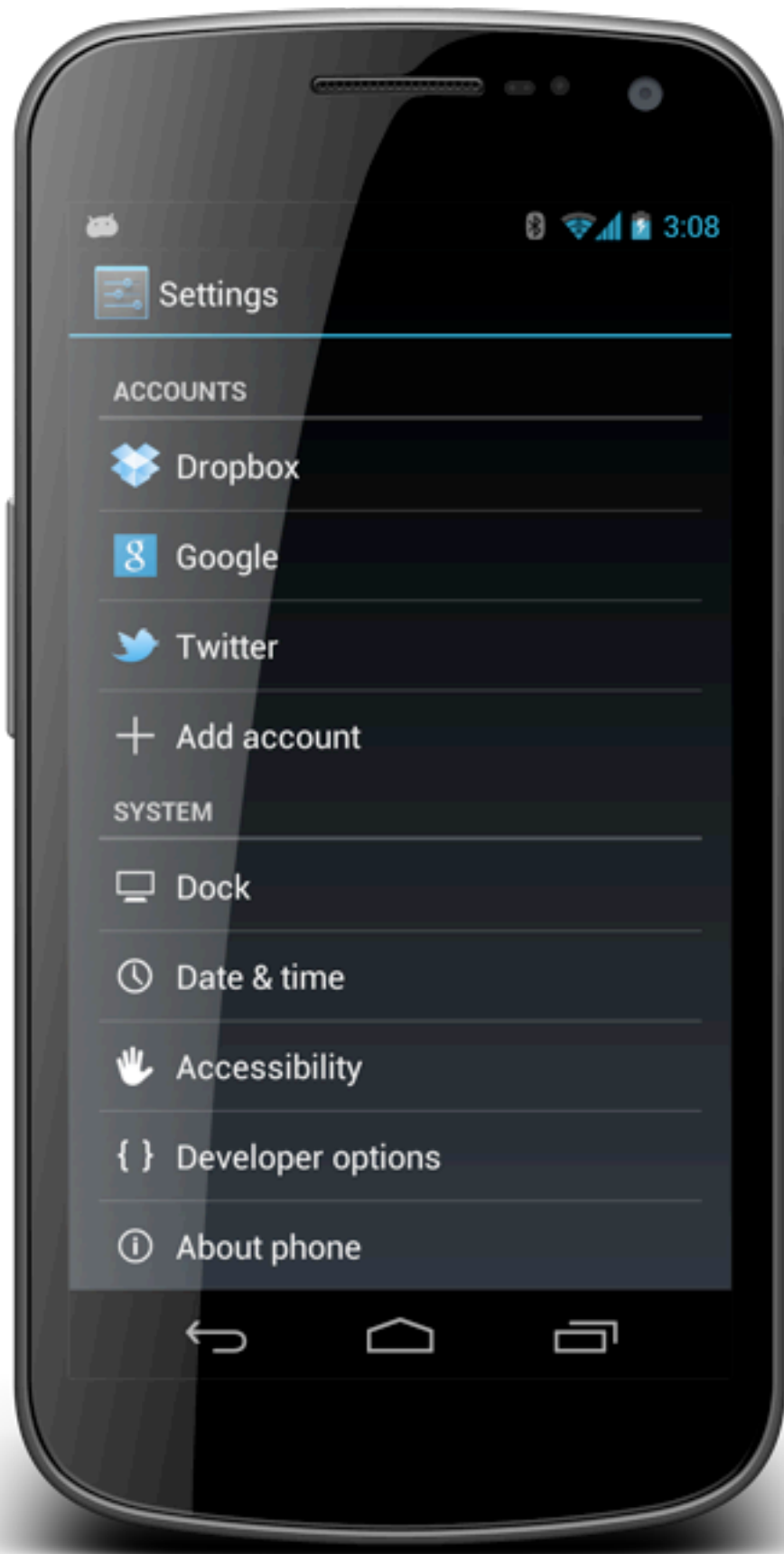


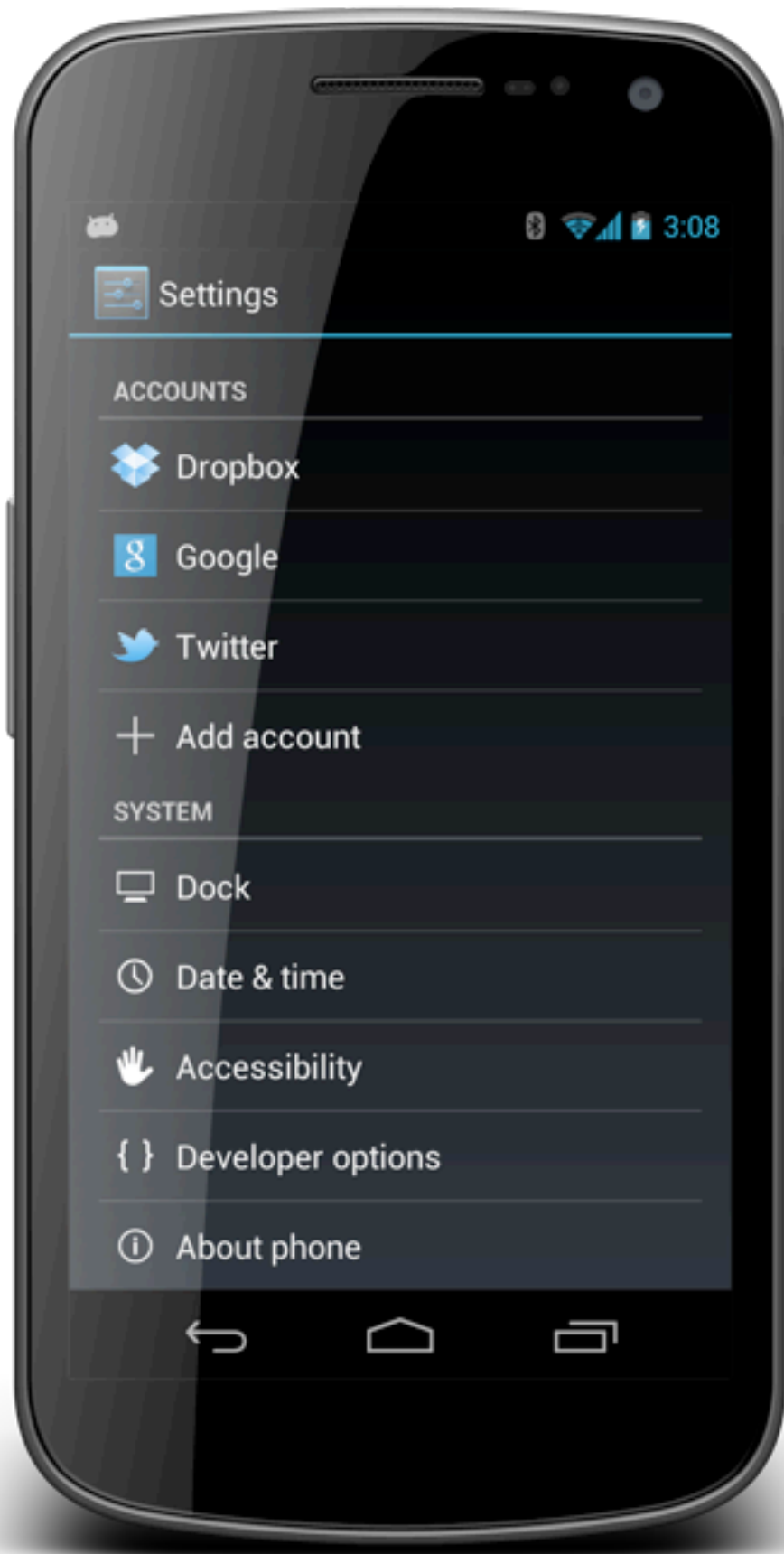
```
android:sdk $ cd platform-tools/
```

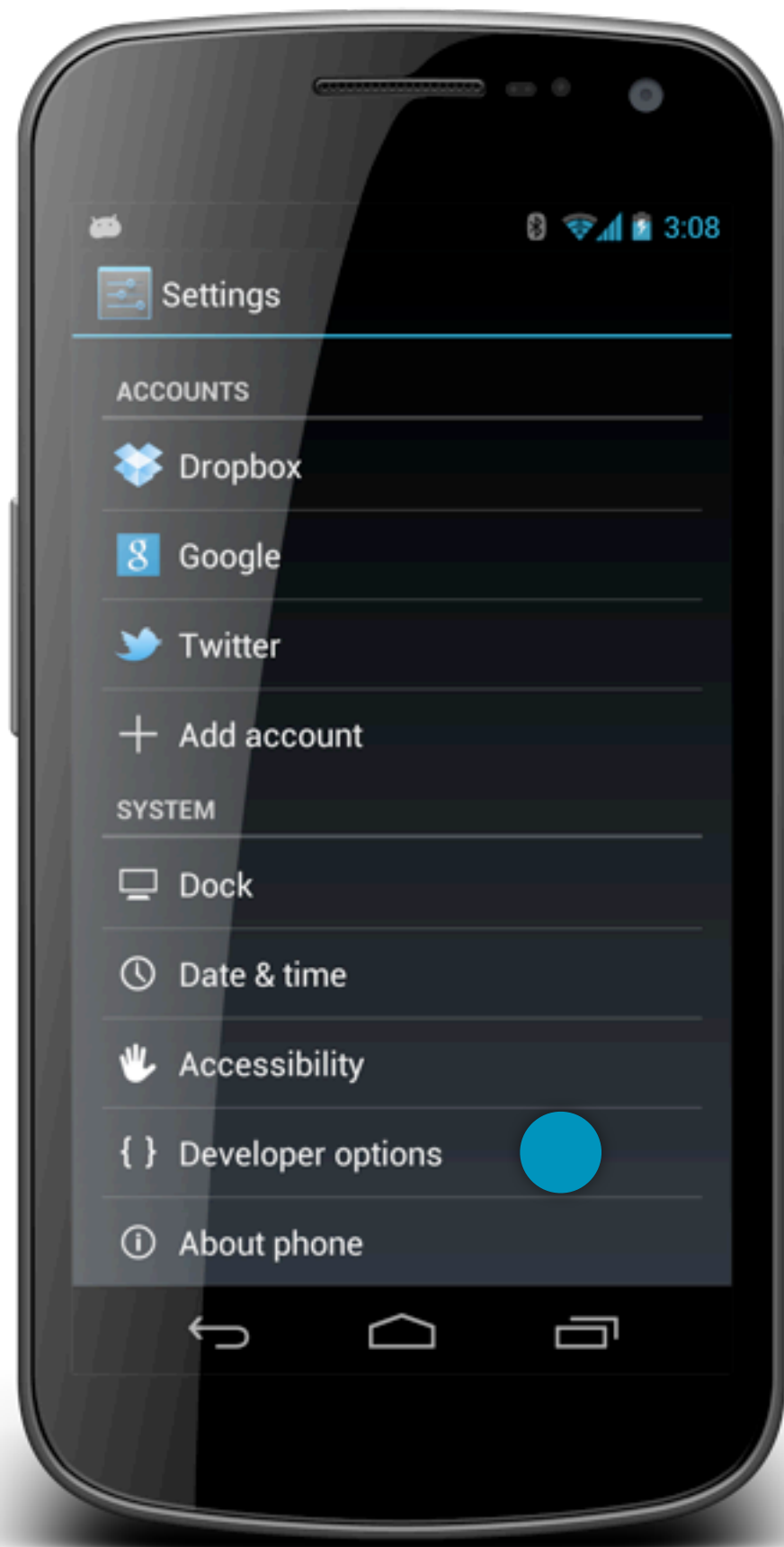
```
android:platform-tools $ adb shell ↵
```

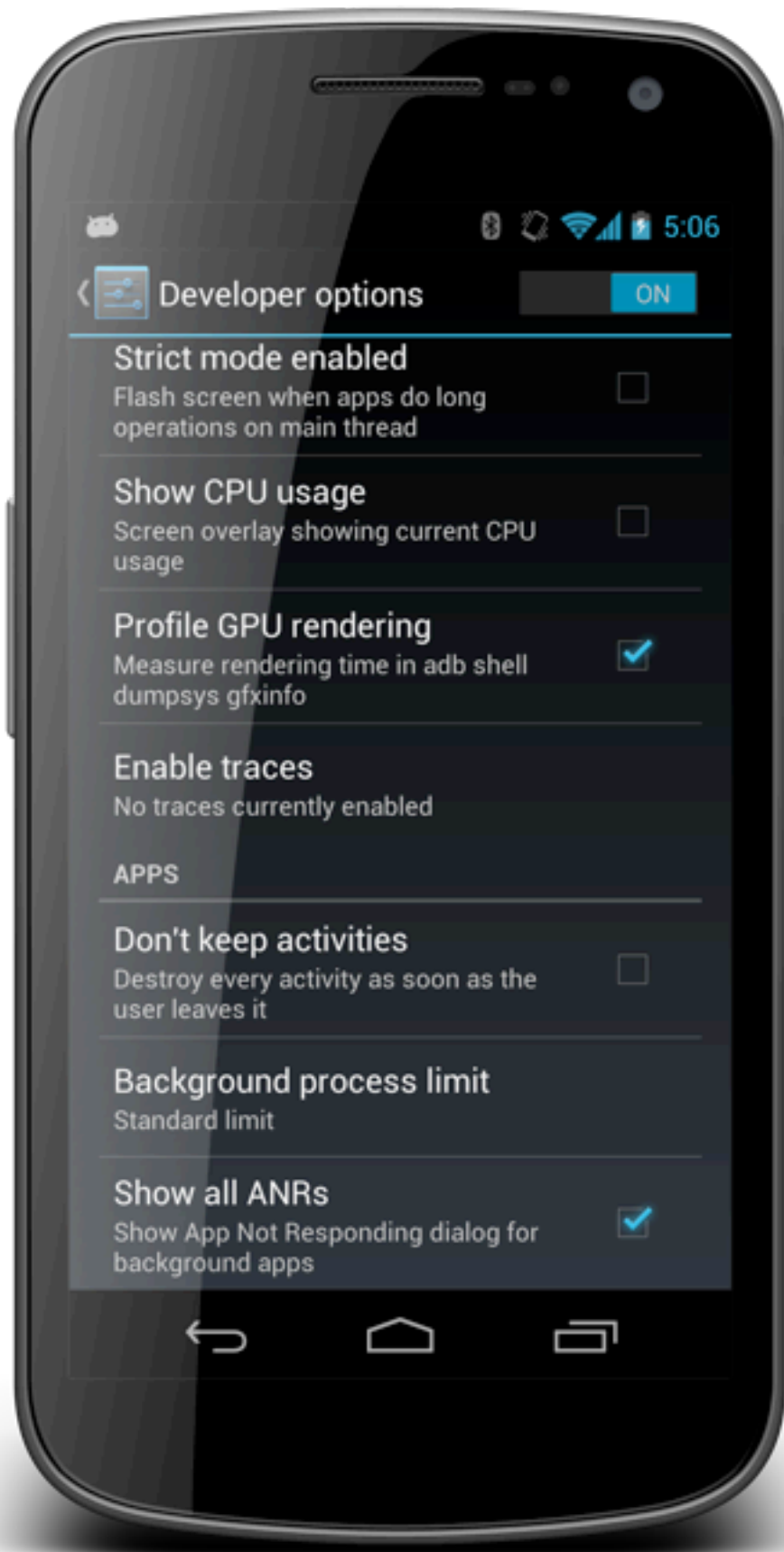
```
    dumpsys gfxinfo █
```

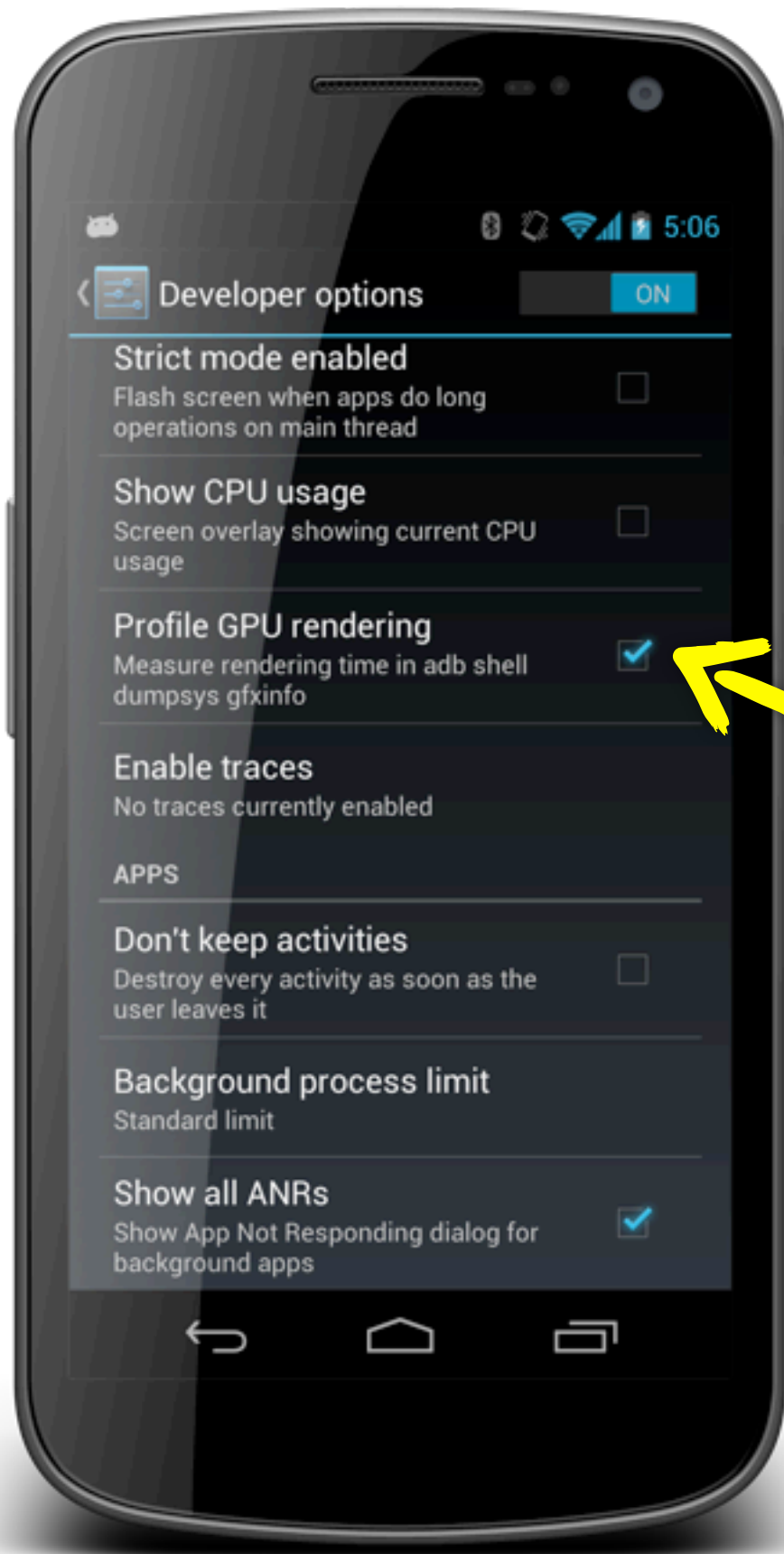






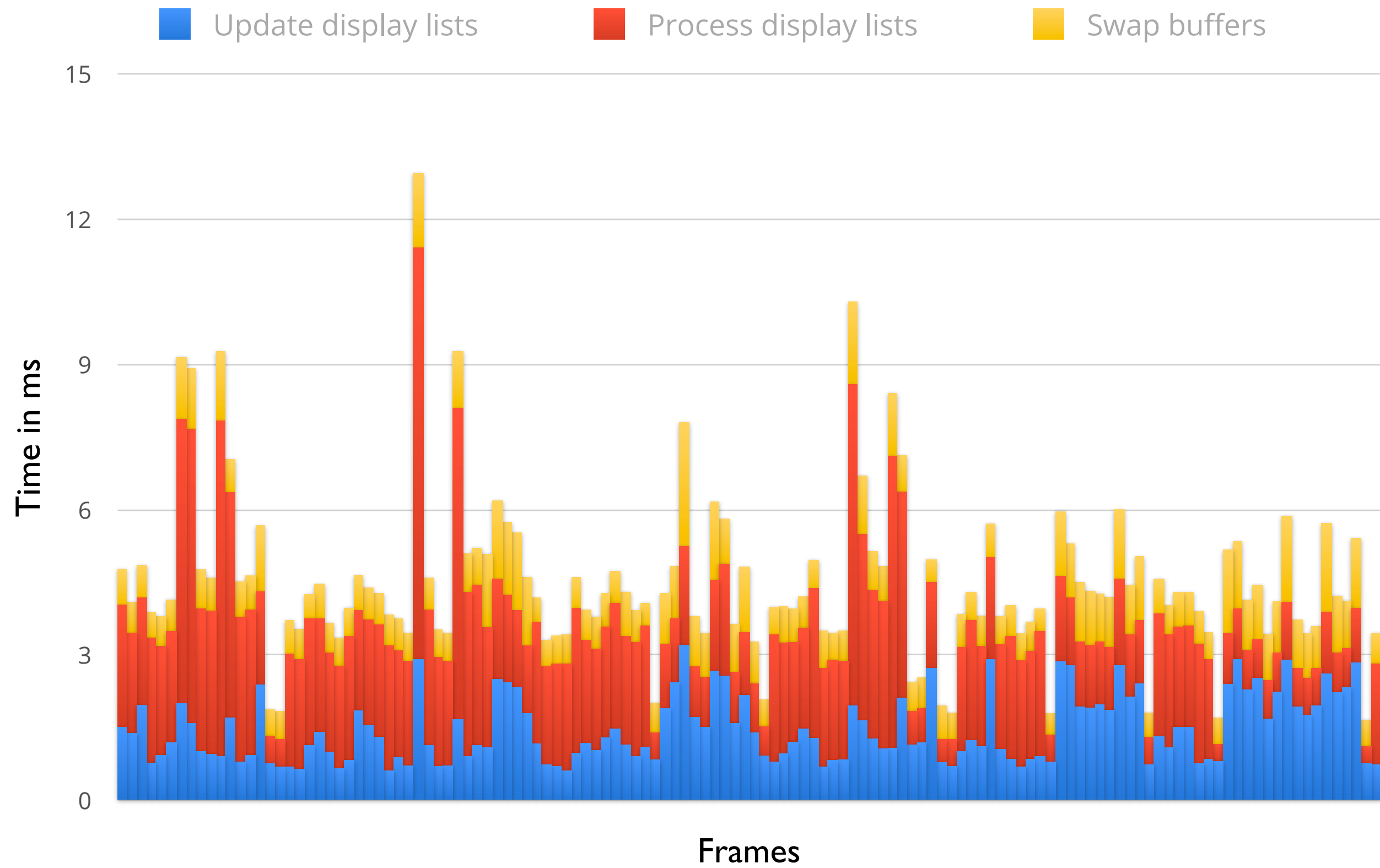








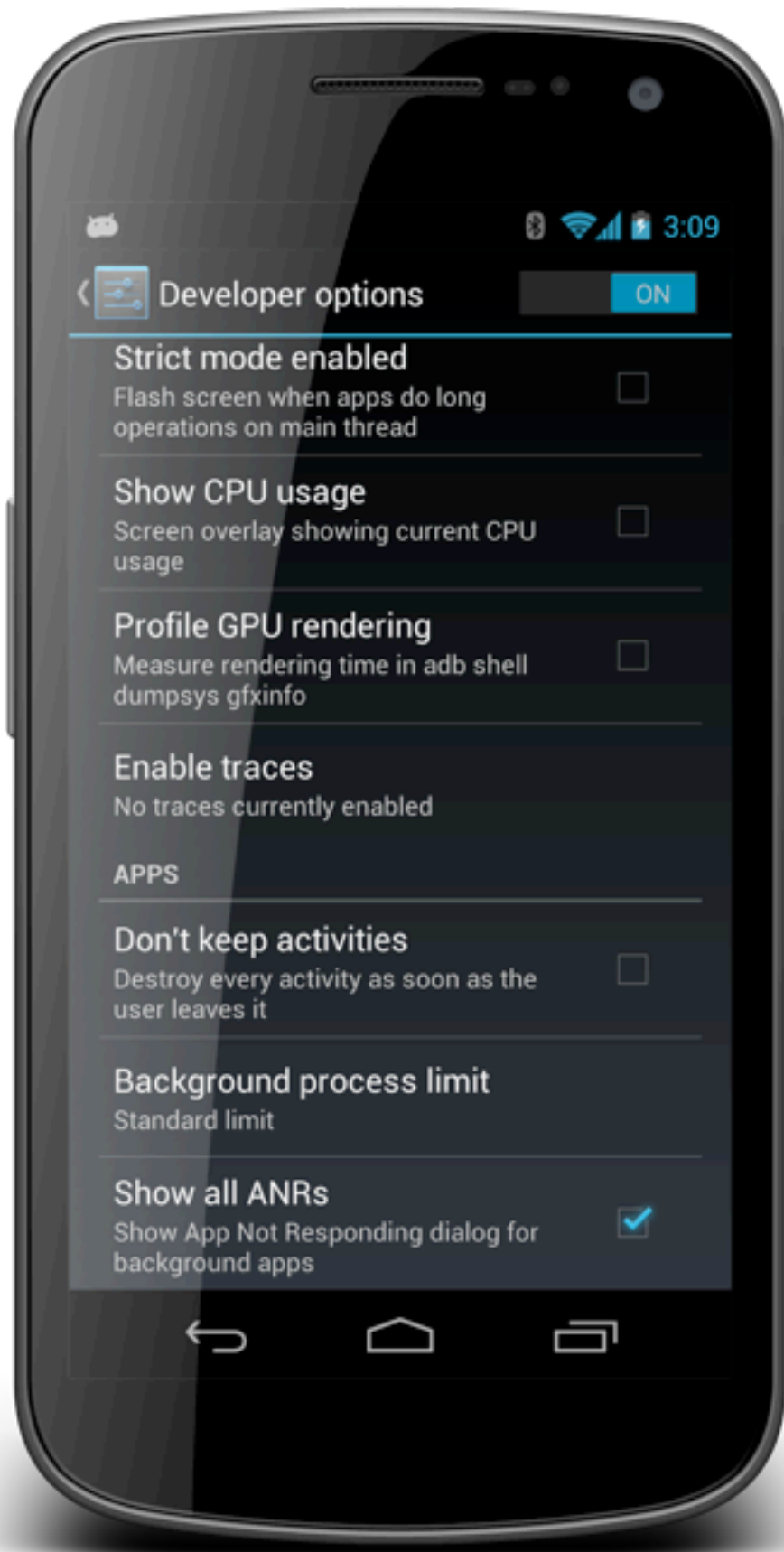
Demo

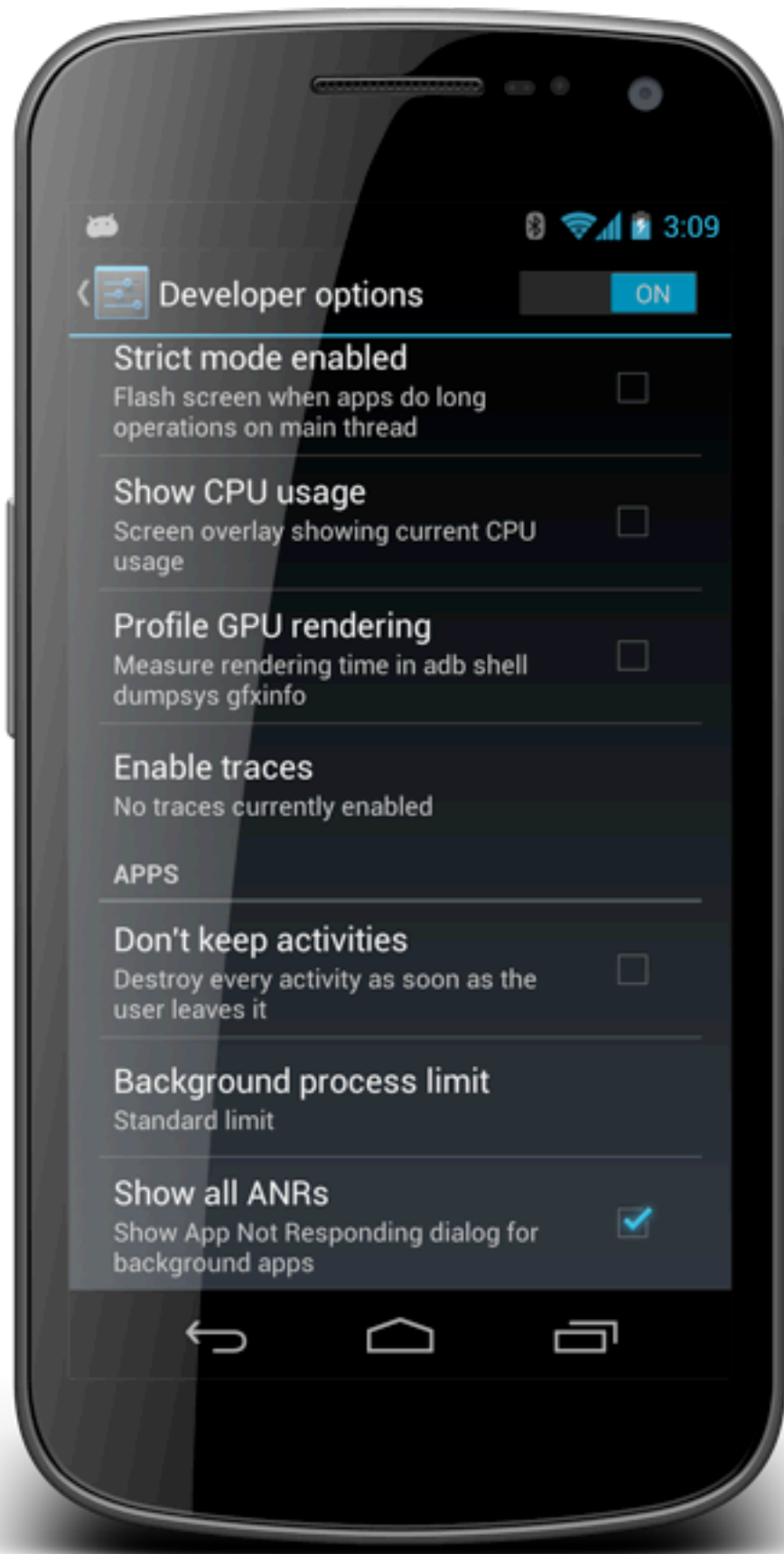


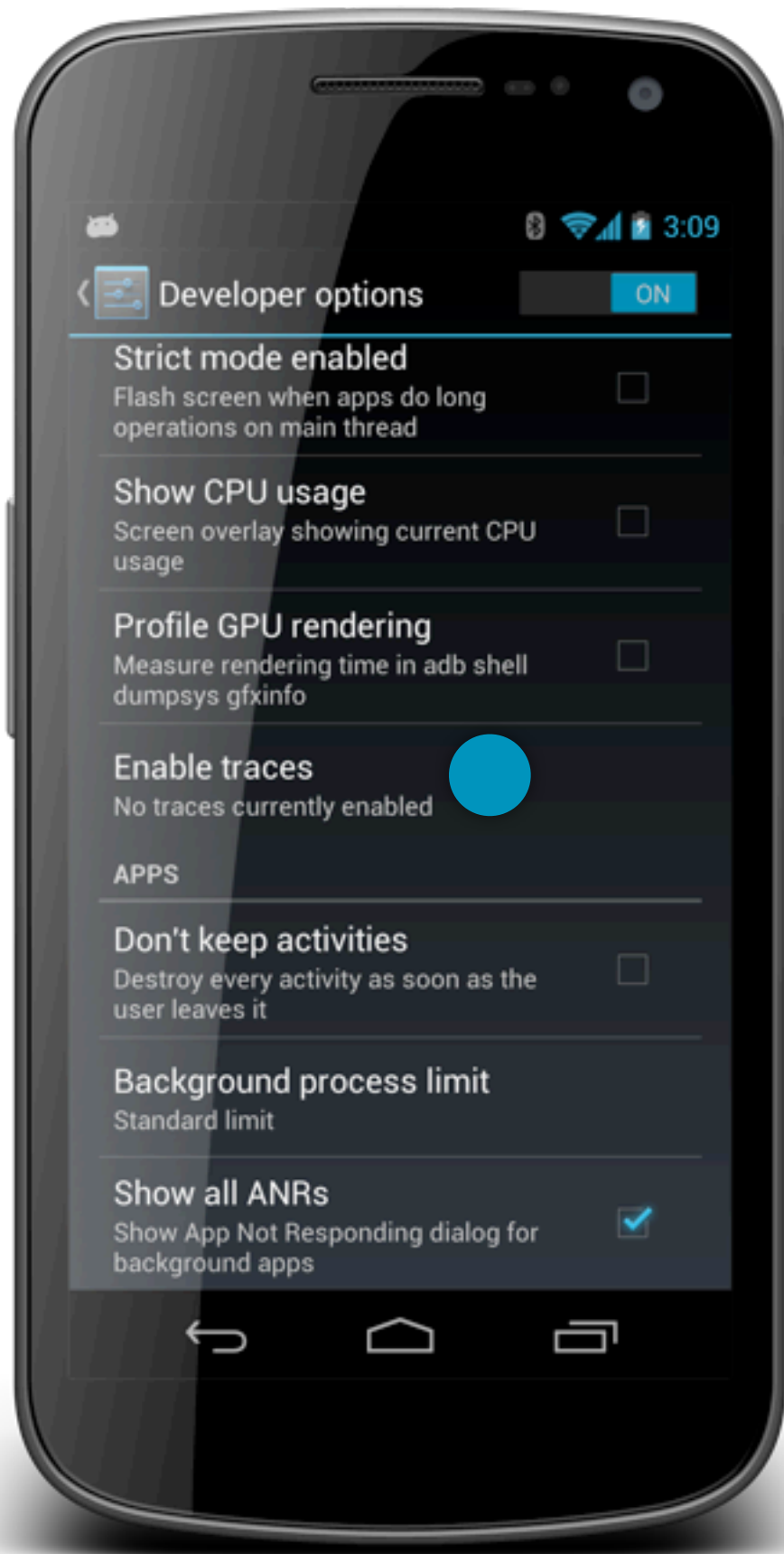
```
android:sdk $ cd tools/systrace
```

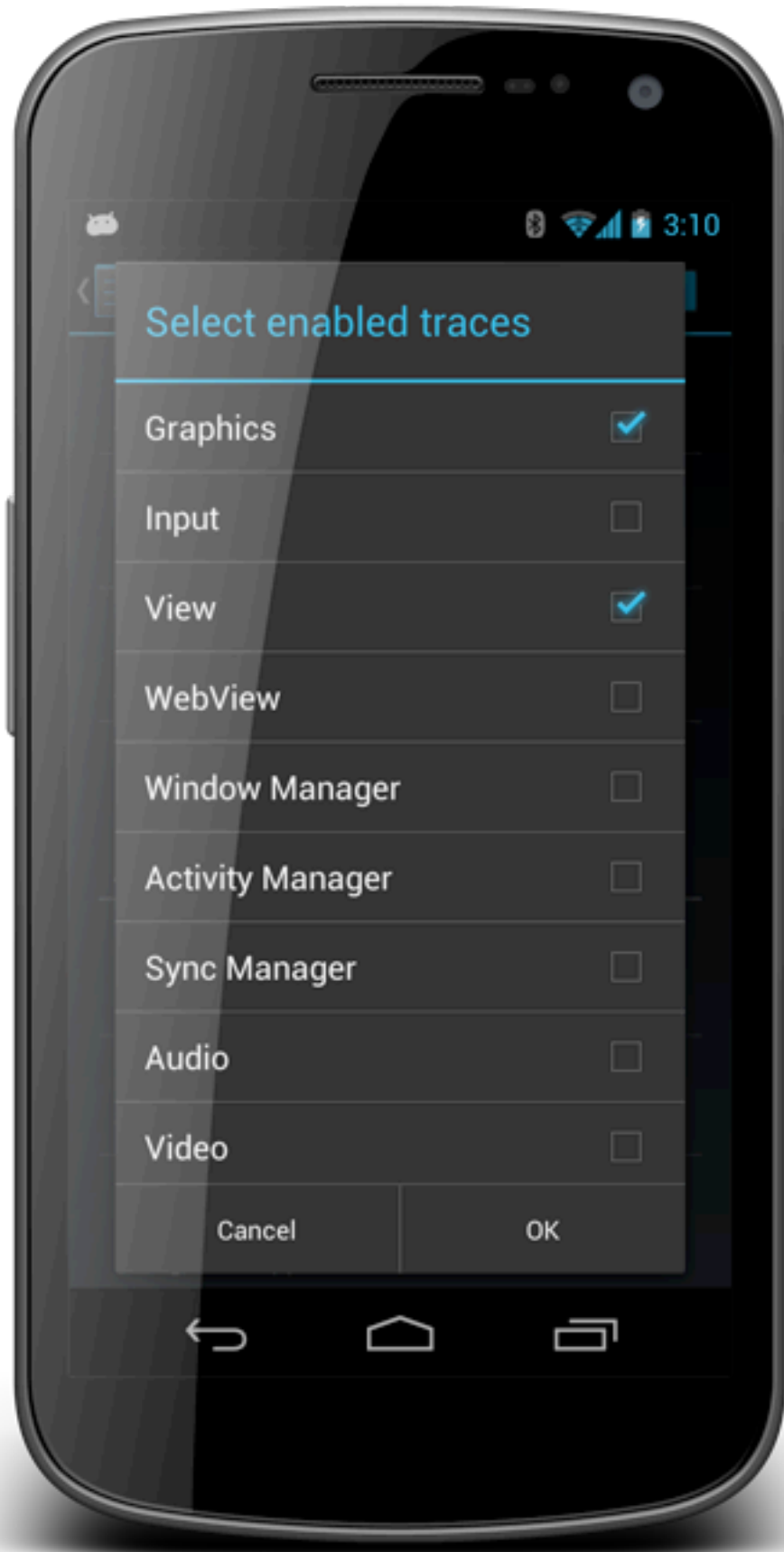
```
android:systrace $ ./systrace ■
```

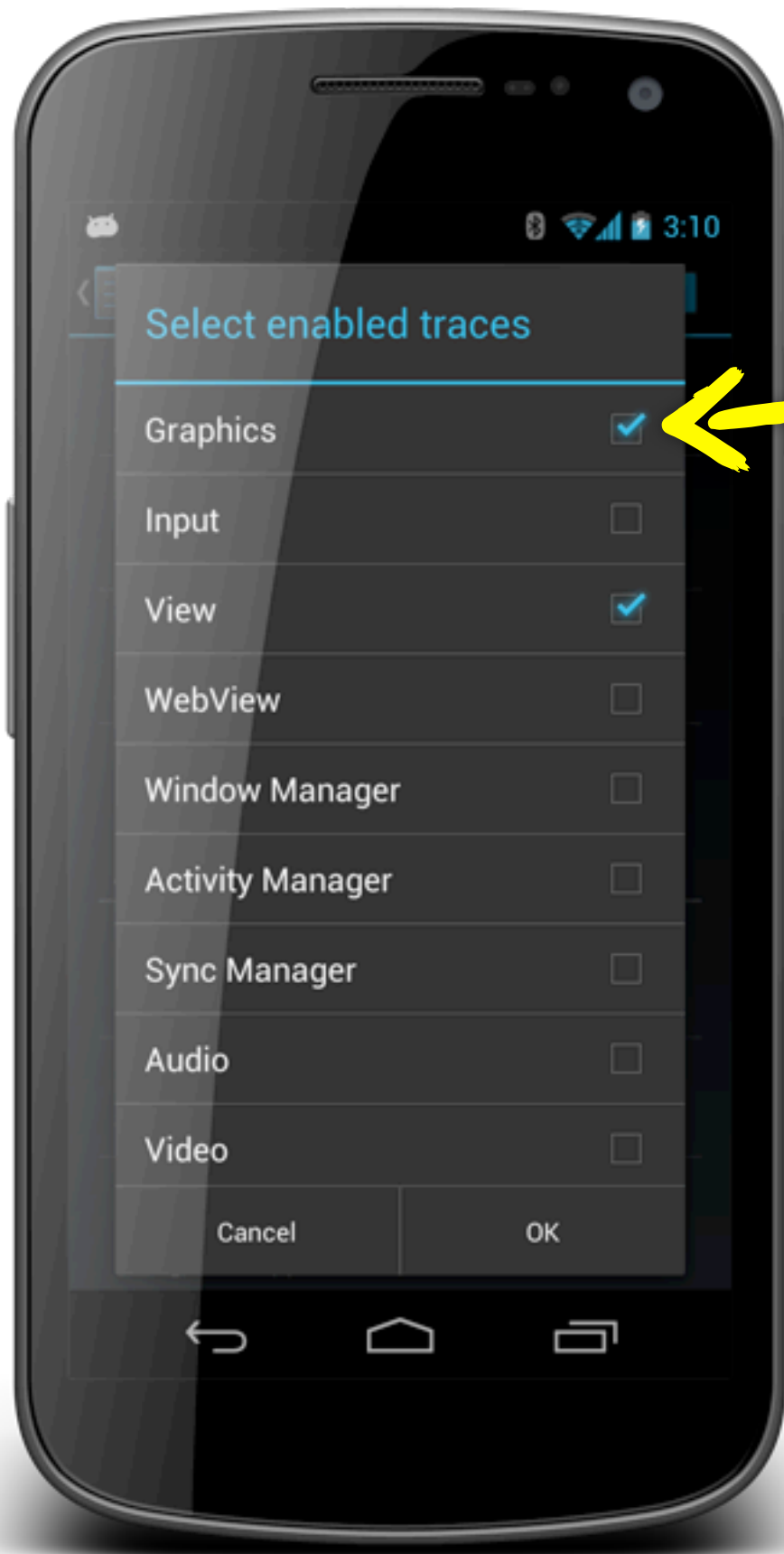


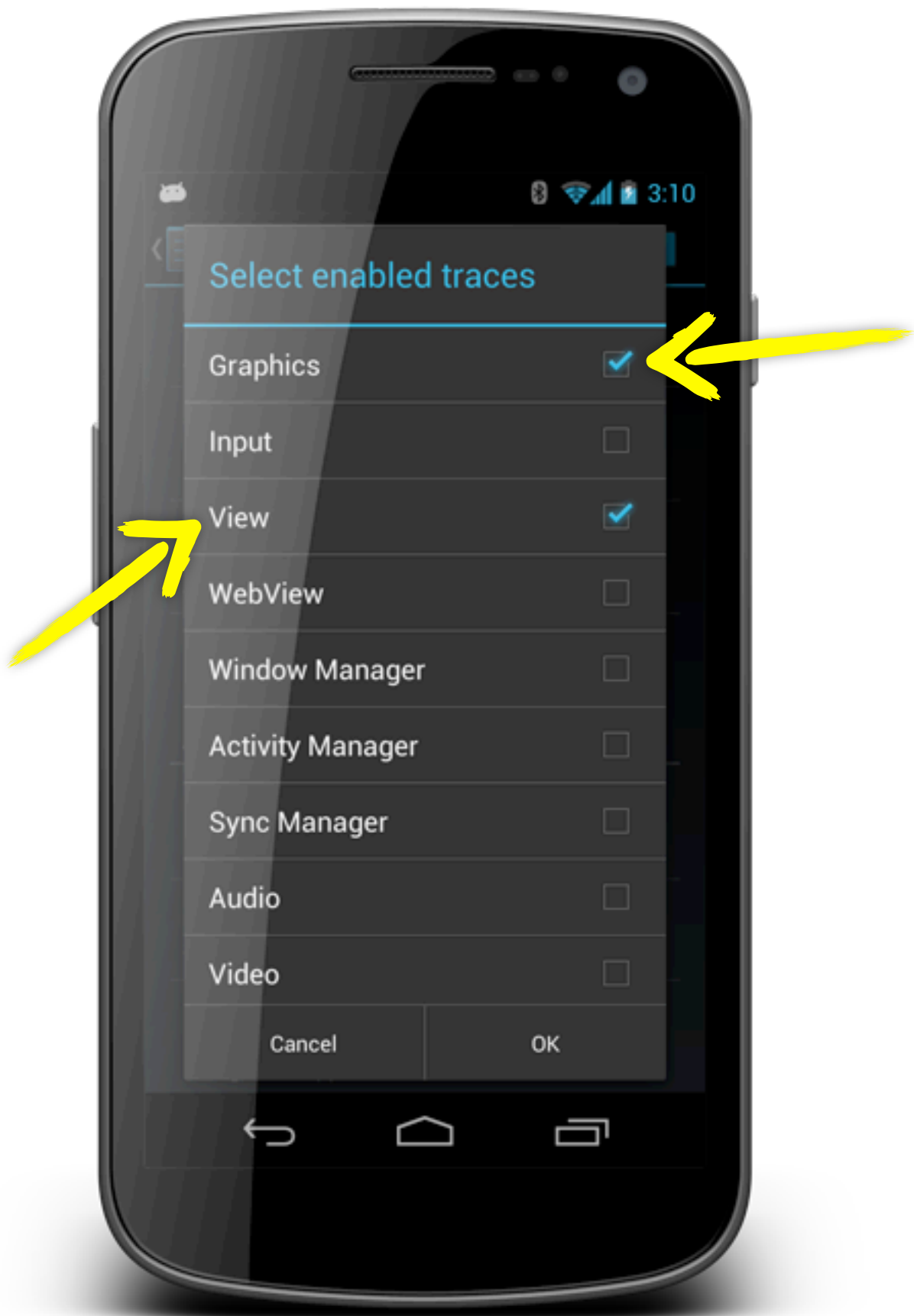














Demo



Captures 5 seconds



HTML



Stand-alone HTML output



```
android:sdk $ cd platform-tools/
```

```
android:platform-tools $ adb shell ↵
```

```
    dumpsys SurfaceFlinger ■
```





Demo

type		...		name
-----+		...		+-----
OVERLAY		...		com...SlowListActivity
OVERLAY		...		StatusBar
OVERLAY		...		NavigationBar



	type		...		name
	-----+		...		+-----
✓	OVERLAY		...		com...SlowListActivity
✓	OVERLAY		...		StatusBar
✓	OVERLAY		...		NavigationBar



```

      type      | ... | name
-----+-----+-----
OVERLAY | ... | com...SlowListActivity
      FB | ... | PopupWindow:424d4cc8
      FB | ... | StatusBar
      FB | ... | NavigationBar

```

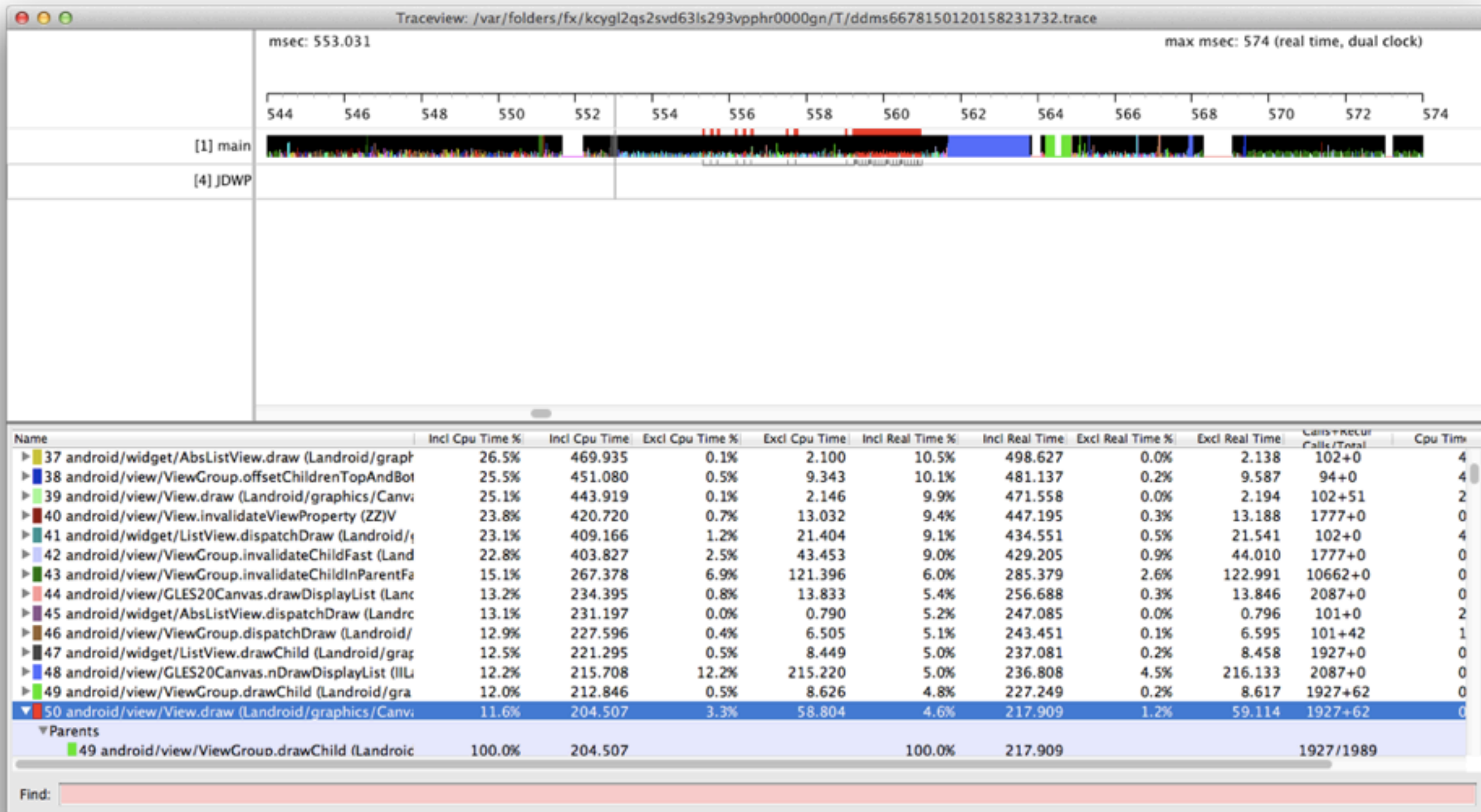


	type		...		name
	-----+		...		+-----
✓	OVERLAY		...		com...SlowListActivity
✗	FB		...		PopupWindow:424d4cc8
✗	FB		...		StatusBar
✗	FB		...		NavigationBar



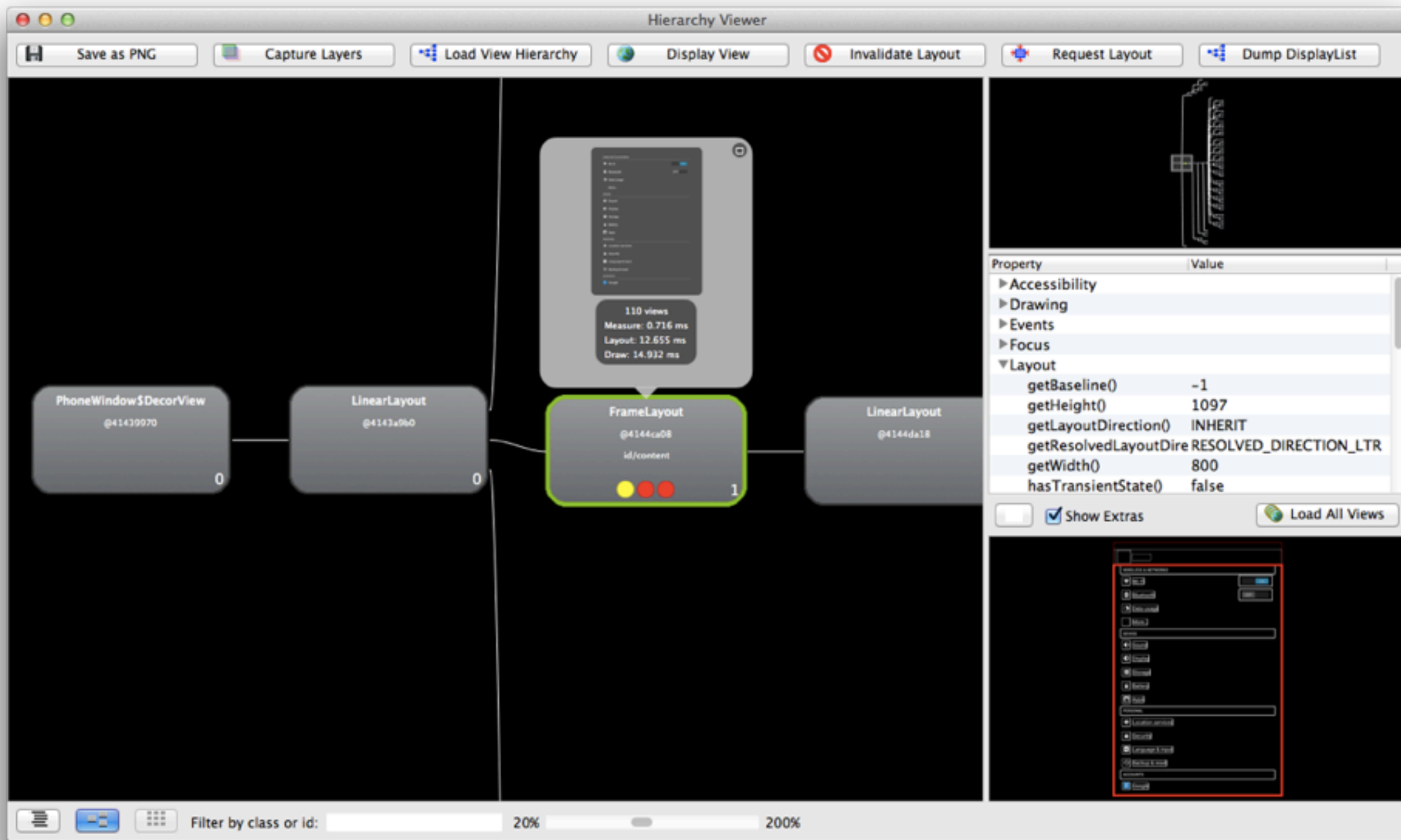
Caveats





traceview





HierarchyViewer



<https://github.com/romainguy/ViewServer>



The screenshot shows the Android Debug Monitor (ADM) interface with the OpenGL Tracer active. The main window displays a list of OpenGL calls with columns for Function, Wall Time (ns), and Thread Time (r). The 'GL State' panel on the right shows the current state of the OpenGL context, including Vertex Array Data, Transformation State, Rasterization State, Pixel Operations, and Texture State. The 'Frame Sum...' panel on the far right shows a list of frame IDs and their corresponding times.

Function	Wall Time (ns)	Thread Time (r)
glVertexAttribPointer(indx = 0, size = 2, type = GL_FLOAT, normalized = false, stride = 0, offset = 0)	0	0
glVertexAttribPointer(indx = 1, size = 2, type = GL_FLOAT, normalized = false, stride = 0, offset = 0)	0	0
glDrawElements(mode = GL_TRIANGLES, count = 36, type = GL_UNSIGNED_SHORT, offset = 0)	305,176	305,176
LinearLayout	0	0
ImageView	0	0
glScissor(x = 44, y = 0, width = 64, height = 9)	0	0
glUseProgram(program = 210003)	0	0
glUniformMatrix4fv(location = 1, count = 1, transpose = false, value = [0.0, 0.0, 0.0, 0.0], offset = 0)	30,518	30,518
glBindTexture(target = GL_TEXTURE_2D, texture = 2100030)	0	0
glVertexAttribPointer(indx = 0, size = 2, type = GL_FLOAT, normalized = false, stride = 0, offset = 0)	0	0
glVertexAttribPointer(indx = 1, size = 2, type = GL_FLOAT, normalized = false, stride = 0, offset = 0)	0	0
glBindBuffer(target = GL_ELEMENT_ARRAY_BUFFER, buffer = 0)	0	0
glVertexAttribPointer(indx = 0, size = 2, type = GL_FLOAT, normalized = false, stride = 0, offset = 0)	0	0
glVertexAttribPointer(indx = 1, size = 2, type = GL_FLOAT, normalized = false, stride = 0, offset = 0)	0	0
glDrawArrays(mode = GL_TRIANGLE_STRIP, first = 0, count = 4)	457,763	91,526
RelativeLayout	0	0
TextView	0	0
glScissor(x = 124, y = 0, width = 81, height = 2)	0	0
glUseProgram(program = 420006)	0	0
glUniformMatrix4fv(location = 1, count = 1, transpose = false, value = [0.0, 0.0, 0.0, 0.0], offset = 0)	0	0
glBindTexture(target = GL_TEXTURE_2D, texture = 210003)	0	0
glUniform4f(location = 3, x = 0.952941, y = 0.952941, z = 0.952941, w = 0.952941, offset = 0)	457,764	457,764
glGenBuffers(n = 1, buffers = [770011])	0	0
glBindBuffer(target = GL_ARRAY_BUFFER, buffer = 770011)	0	0
glBufferData(target = GL_ARRAY_BUFFER, size = 864, data = [864 bytes], offset = 0)	701,904	244,128
glScissor(x = 0, y = 0, width = 720, height = 1038)	0	0
glUseProgram(program = 210003)	0	0
glUniformMatrix4fv(location = 1, count = 1, transpose = false, value = [0.0, 0.0, 0.0, 0.0], offset = 0)	0	0
glBindTexture(target = GL_TEXTURE_2D, texture = 1680024)	0	0
glVertexAttribPointer(indx = 0, size = 2, type = GL_FLOAT, normalized = false, stride = 0, offset = 0)	30,517	30,517
glVertexAttribPointer(indx = 1, size = 2, type = GL_FLOAT, normalized = false, stride = 0, offset = 0)	457,764	457,764
glDrawArrays(mode = GL_TRIANGLES, first = 0, count = 54)	305,175	305,175
glGetError(void) = (GLenum) GL_NO_ERROR	0	0
eglSwapBuffers	0	0

Tracer for OpenGL ES



Dalvik Debug Monitor

Info Threads VM Heap Allocation Tracker Sysinfo Network Emulator Control Event Log

Start Tracking Get Allocations Filter: Inc. trace

Alloc Order	Allocation Size	Allocated Class	Thread	Allocated in	Allocated in
128	12	java.util.concurrent.locks.ReentrantReadWriteLock	1	java.util.concurrent.locks.ReentrantReadWriteLock	<init>
111	12	java.util.concurrent.locks.ReentrantReadWriteLock	1	java.util.concurrent.locks.ReentrantReadWriteLock	<init>
94	12	java.util.concurrent.locks.ReentrantReadWriteLock	1	java.util.concurrent.locks.ReentrantReadWriteLock	<init>
216	40	java.util.concurrent.locks.ReentrantReadWriteLock	1	java.util.concurrent.locks.ReentrantReadWriteLock	<init>
199	40	java.util.concurrent.locks.ReentrantReadWriteLock	1	java.util.concurrent.locks.ReentrantReadWriteLock	<init>
182	40	java.util.concurrent.locks.ReentrantReadWriteLock	1	java.util.concurrent.locks.ReentrantReadWriteLock	<init>
131	40	java.util.concurrent.locks.ReentrantReadWriteLock	1	java.util.concurrent.locks.ReentrantReadWriteLock	<init>
114	40	java.util.concurrent.locks.ReentrantReadWriteLock	1	java.util.concurrent.locks.ReentrantReadWriteLock	<init>
97	40	java.util.concurrent.locks.ReentrantReadWriteLock	1	java.util.concurrent.locks.ReentrantReadWriteLock	<init>

Class	Method	File	Line	Native
java.util.concurrent.locks.ReentrantReadWriteLock	<init>	ReentrantReadWriteLock.java	214	false
java.util.concurrent.locks.ReentrantReadWriteLock	<init>	ReentrantReadWriteLock.java	204	false
android.animation.PropertyValuesHolder	<init>	PropertyValuesHolder.java	106	false
android.animation.PropertyValuesHolder	<init>	PropertyValuesHolder.java	35	false
android.animation.PropertyValuesHolder\$FloatPropertyValuesHolder	<init>	PropertyValuesHolder.java	909	false
android.animation.PropertyValuesHolder	ofFloat	PropertyValuesHolder.java	175	false
com.android.launcher2.Folder	animateOpen	Folder.java	421	false
com.android.launcher2.Launcher	openFolder	Launcher.java	2200	false
com.android.launcher2.Launcher	handleFolderClick	Launcher.java	2059	false
com.android.launcher2.Launcher	onClick	Launcher.java	1889	false
android.view.View	performClick	View.java	4078	false
android.view.View\$PerformClick	run	View.java	1695	false
android.os.Handler	handleCallback	Handler.java	615	false

Saved Filters

All messages (no filters)

Search for messages. Accepts Java regexes. Prefix with pid:, app:, tag: or text: to limit scope. verbose

Le Time	PID	TID	Application	Tag	Text
D 06-21 01:31:30.274383	4385		com.example.andr	dalvikvm	GC_CONCURRENT freed 384K, 12% free 7690K/8711K, paused 21 ms+6ms, total 55ms

Allocation Tracker



Tips & Tricks



- ✓ Consistent frame-rate
- ✓ Lower latency
- ✓ Faster display list drawing
- ✓ Faster display list updates
- ✓ GPU-free window composition



new



~~now~~



✓ Consistent frame-rate



```

public void bindView(View view, Context context, Cursor c) {
    BookViewHolder holder = (BookViewHolder) view.getTag();
    String bookId = c.getString(mInternalIdIndex);
    holder.bookId = bookId;
    holder.sortTitle = c.getString(mSortTitleIndex);

    final ShelvesActivity activity = mActivity;
    if (activity.getScrollState() == SCROLL_STATE_FLING ||
        activity.isPendingCoversUpdate()) {
        holder.title.setCompoundDrawablesWithIntrinsicBounds(
            null, null, null, mDefaultCover);
        holder.queryCover = true;
    } else {
        holder.title.setCompoundDrawablesWithIntrinsicBounds(
            null, null, null,
            ImageUtilities.getCachedCover(bookId, mDefaultCover));
        holder.queryCover = false;
    }

    final CharArrayBuffer buffer = holder.buffer;
    c.copyStringToBuffer(mTitleIndex, buffer);
    final int size = buffer.sizeCopied;
    if (size != 0) {
        holder.title.setText(buffer.data, 0, size);
    }
}
}

```



// Do Less!



- ✓ Consistent frame-rate
- ✓ Lower latency



android.view.Choreographer



```
// Invalidates at the next v-sync event  
myView.postInvalidateOnAnimation();
```



```
callback = new Runnable() {  
    public void run() {  
        setupAndStartAnimation();  
    }  
}
```

```
myView.postOnAnimation(callback);
```




```
callback = new Choreographer.FrameCallback() {  
    public void doFrame(long frameTime) {  
        render(frameTime);  
    }  
};
```

```
Choreographer c = Choreographer.getInstance();  
c.postFrameCallback(callback);
```



- ✓ Consistent frame-rate
- ✓ Lower latency



```
view.setLayerType(View.LAYER_TYPE_HARDWARE, null);
```



```
view.animate().alpha(0).withLayer();
```



- ✓ Consistent frame-rate
- ✓ Faster display list drawing



CLIP PING

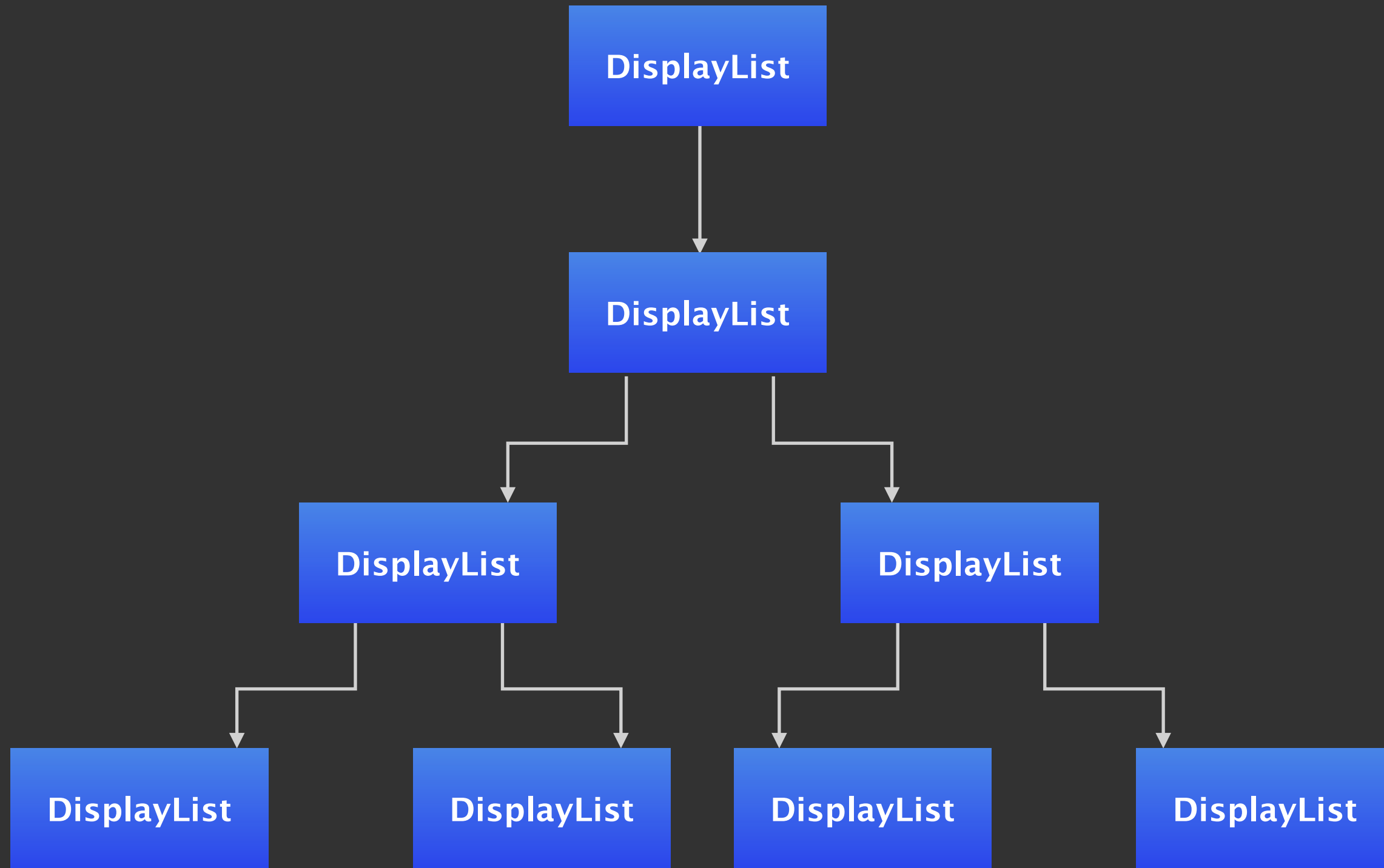


```
view.invalidate();
```

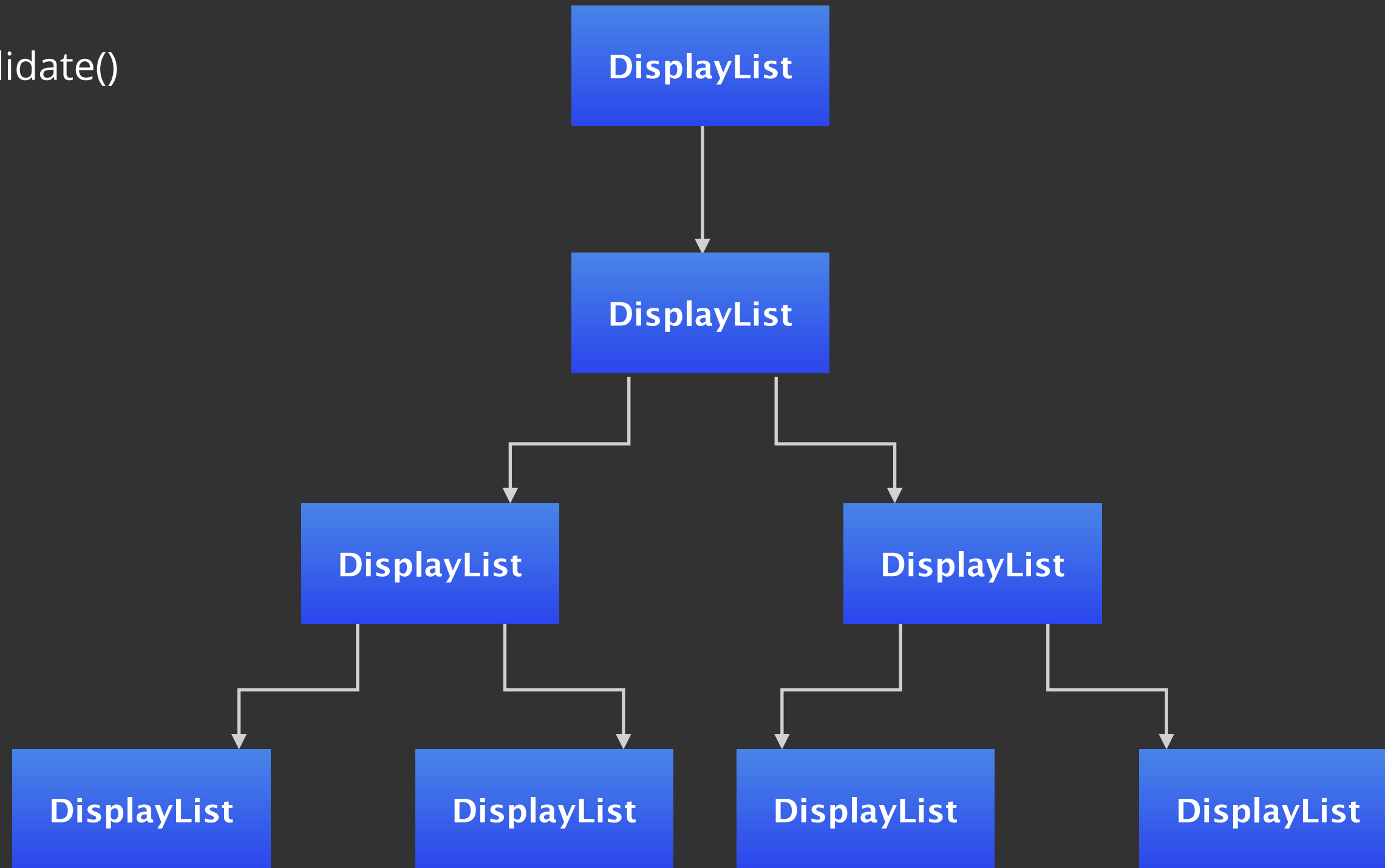


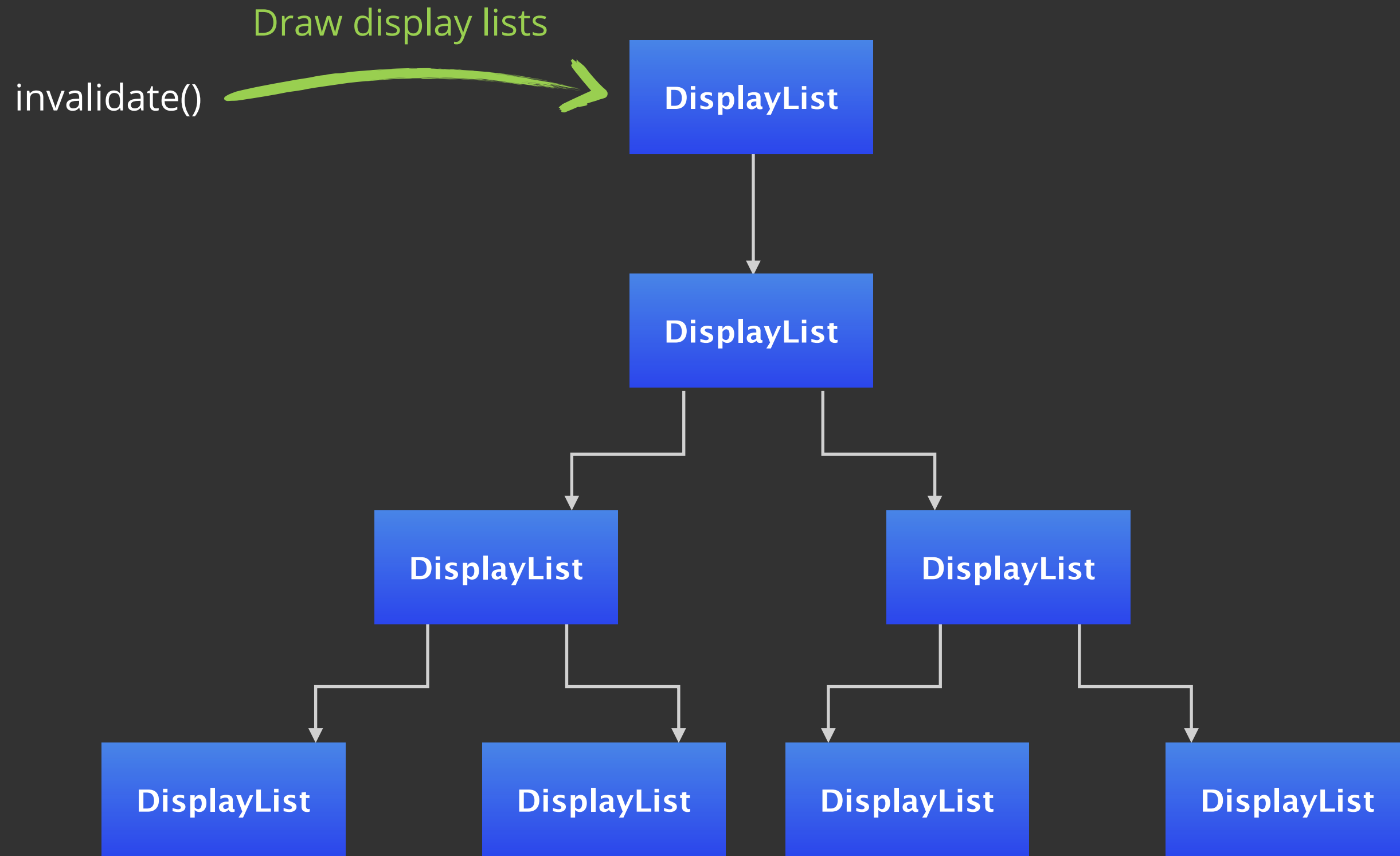
```
view.invalidate(left, top, right, bottom);
```

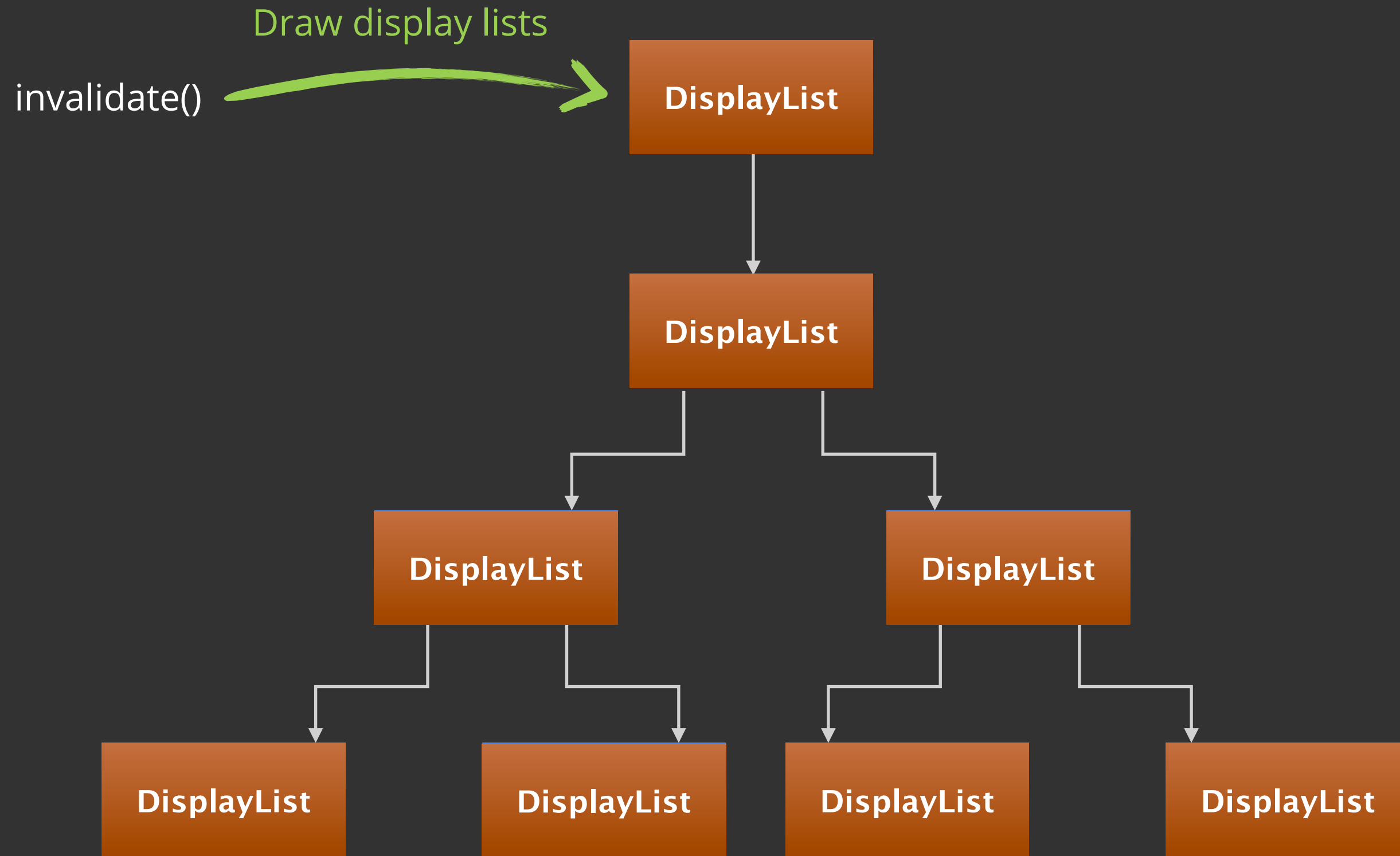


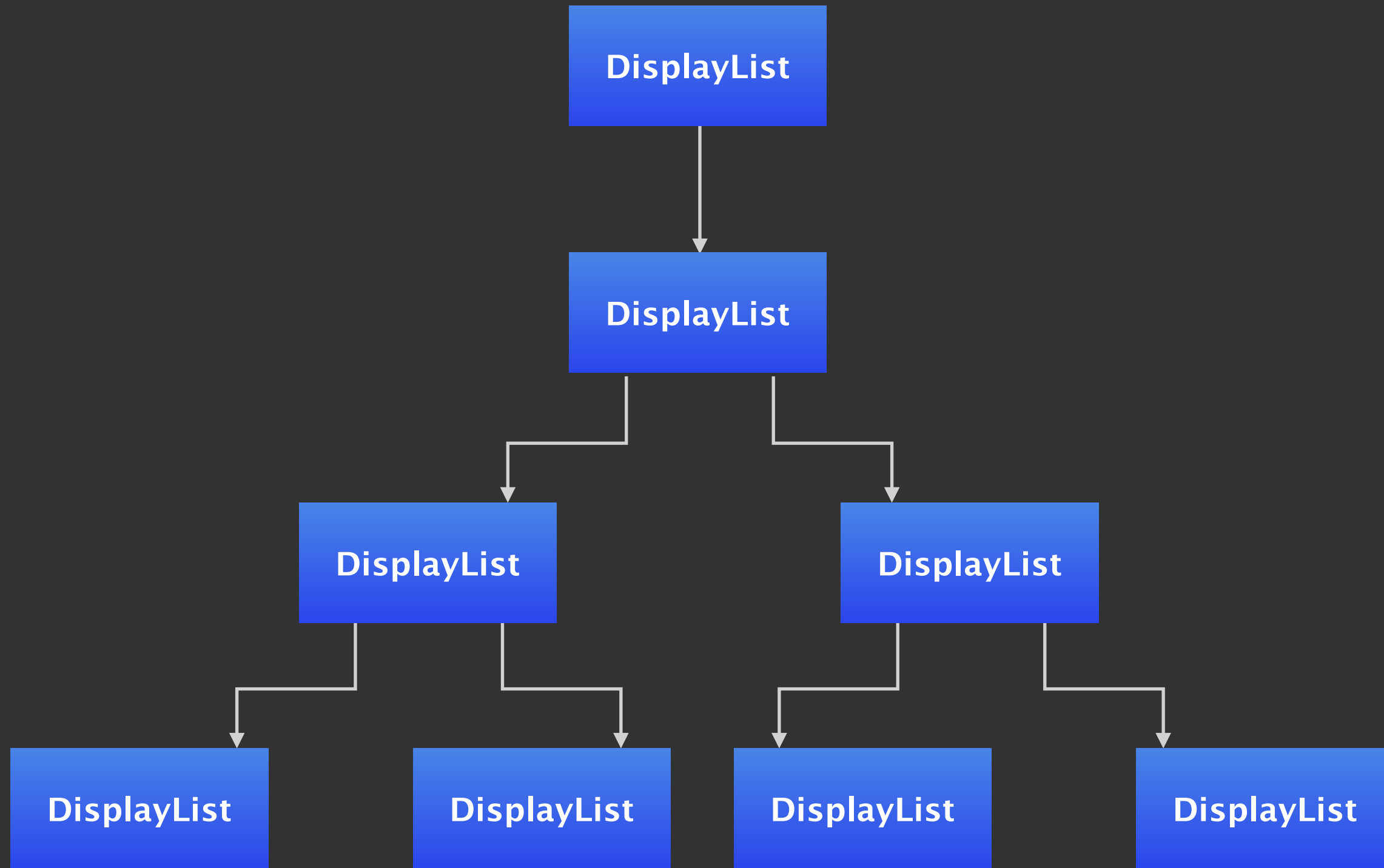


invalidate()

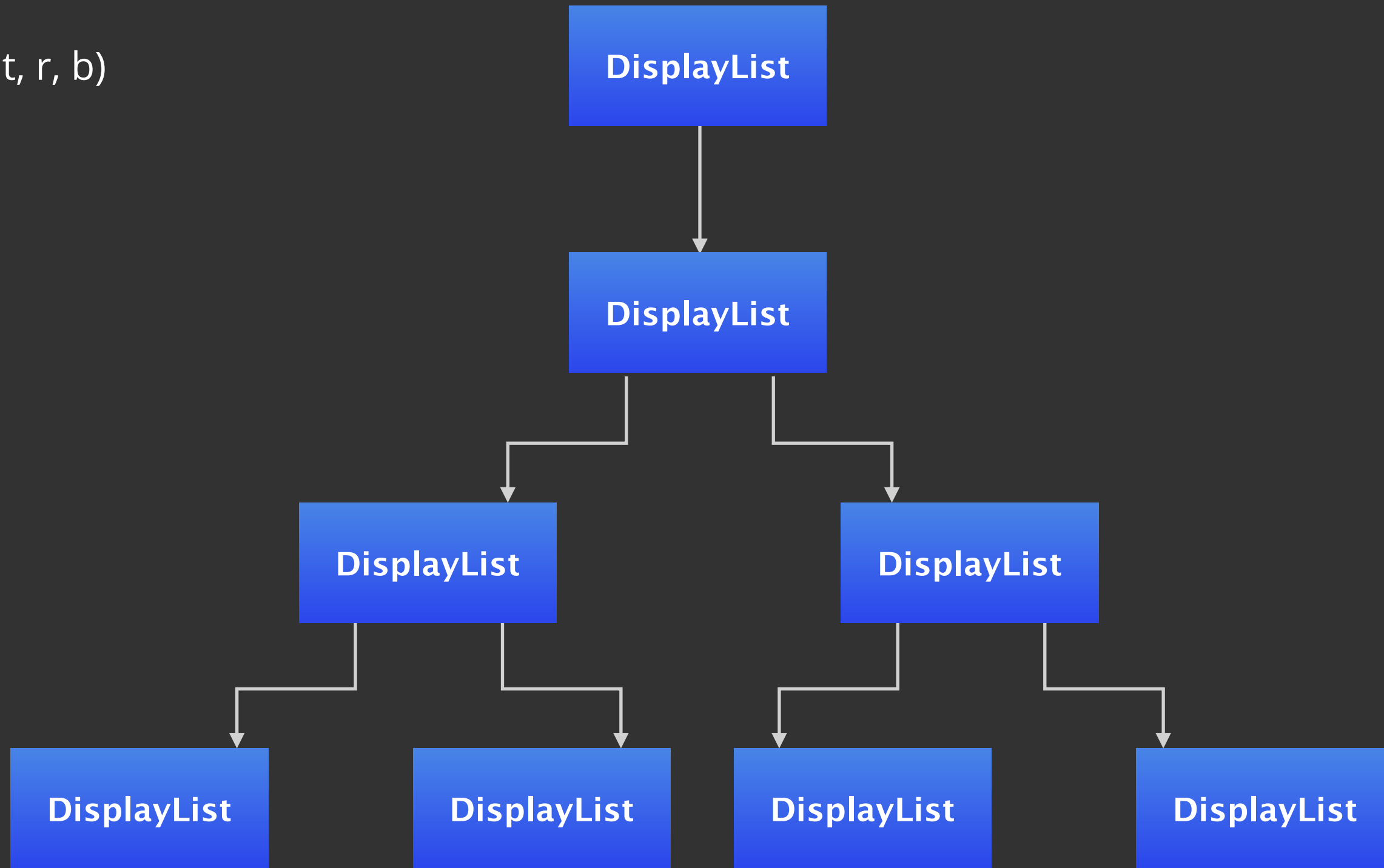


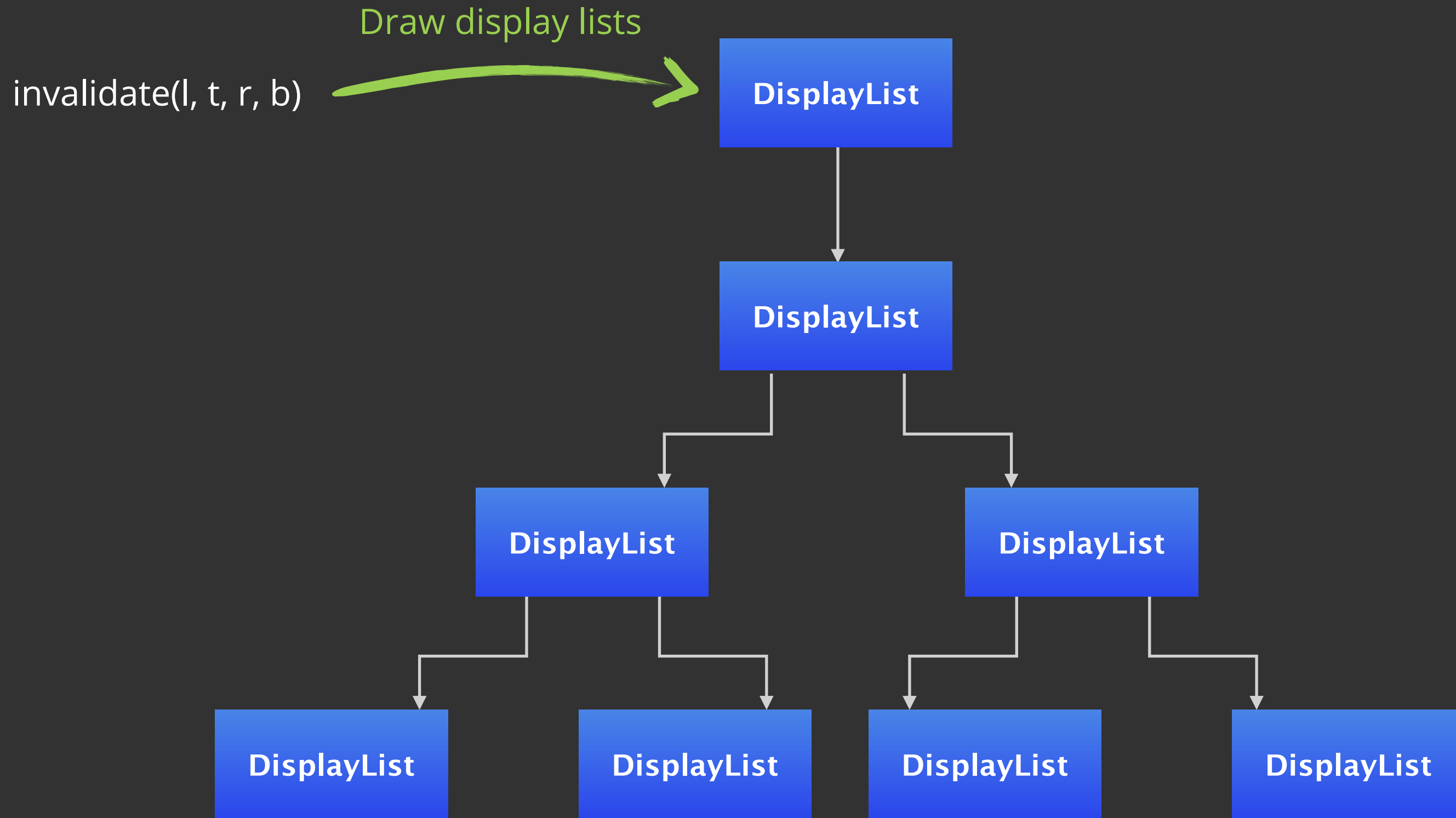


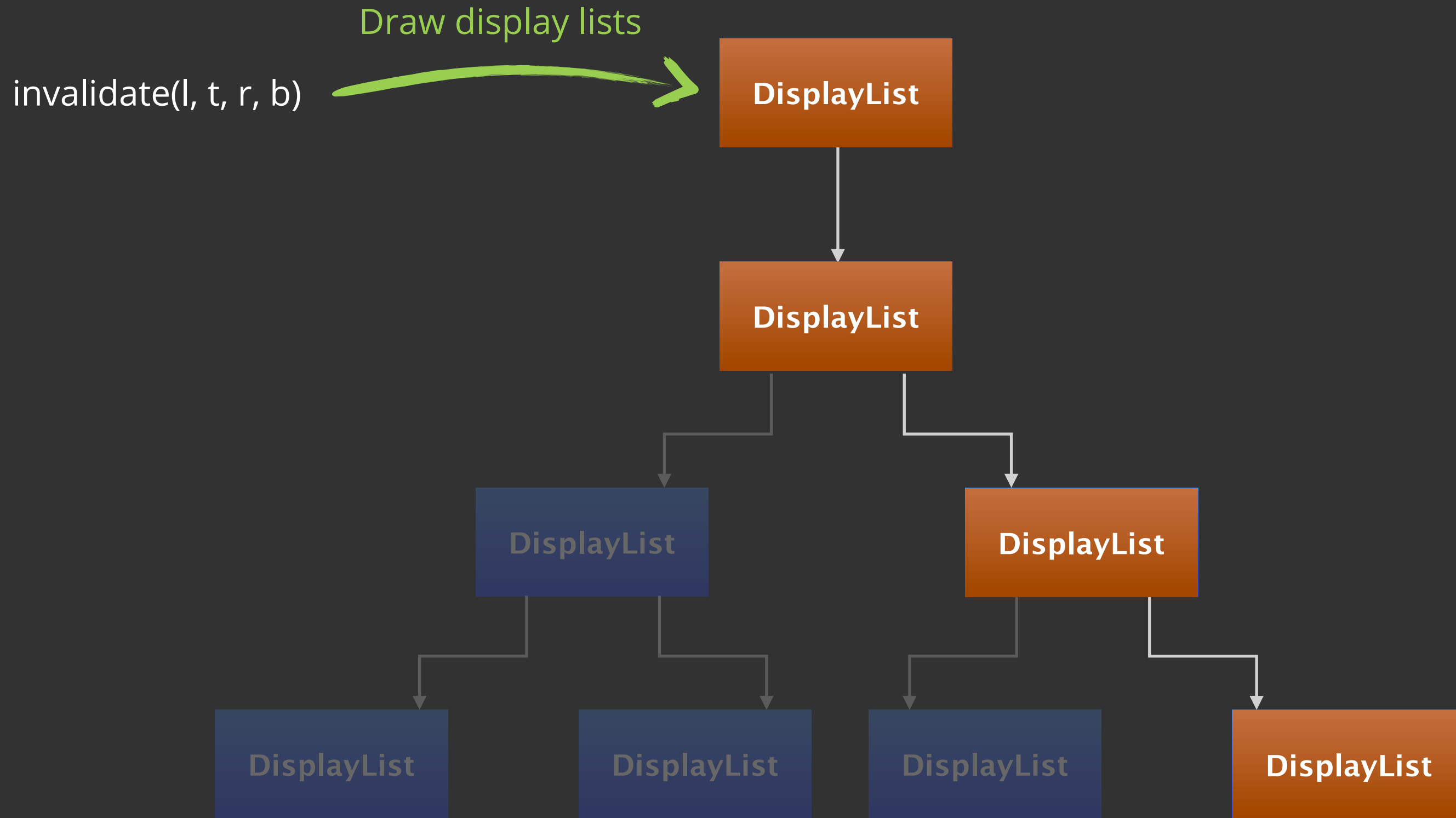




`invalidate(l, t, r, b)`

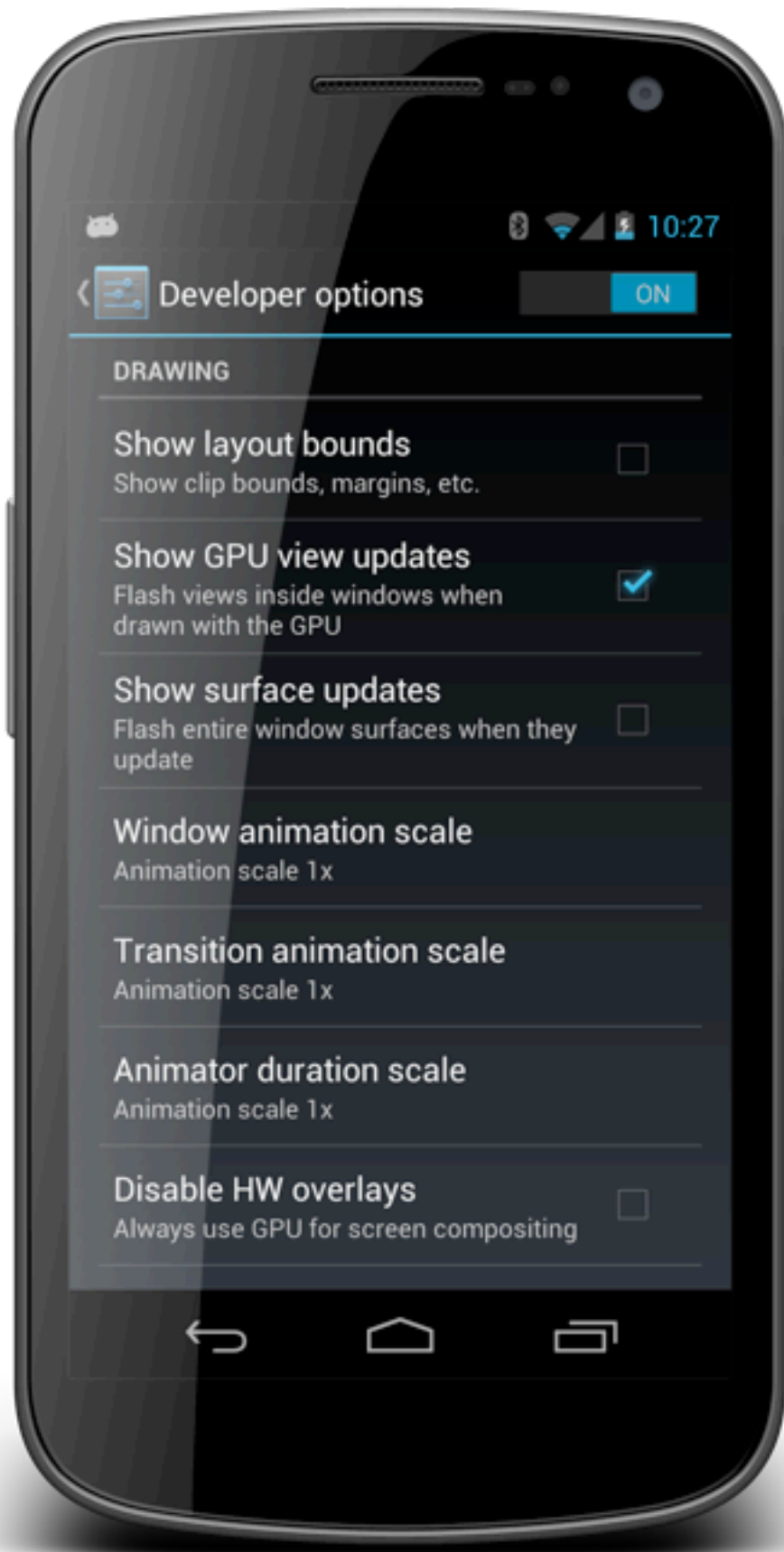


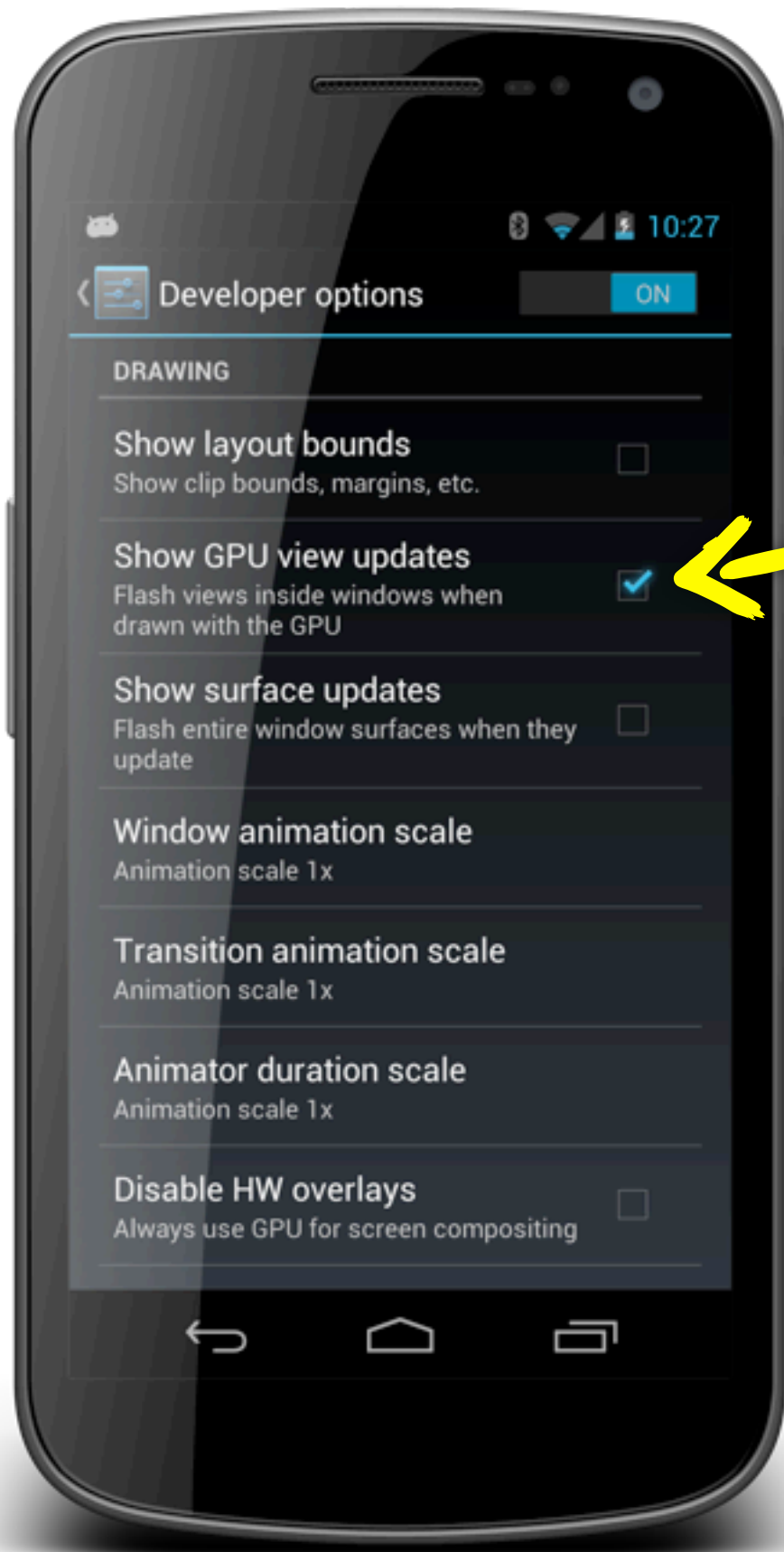




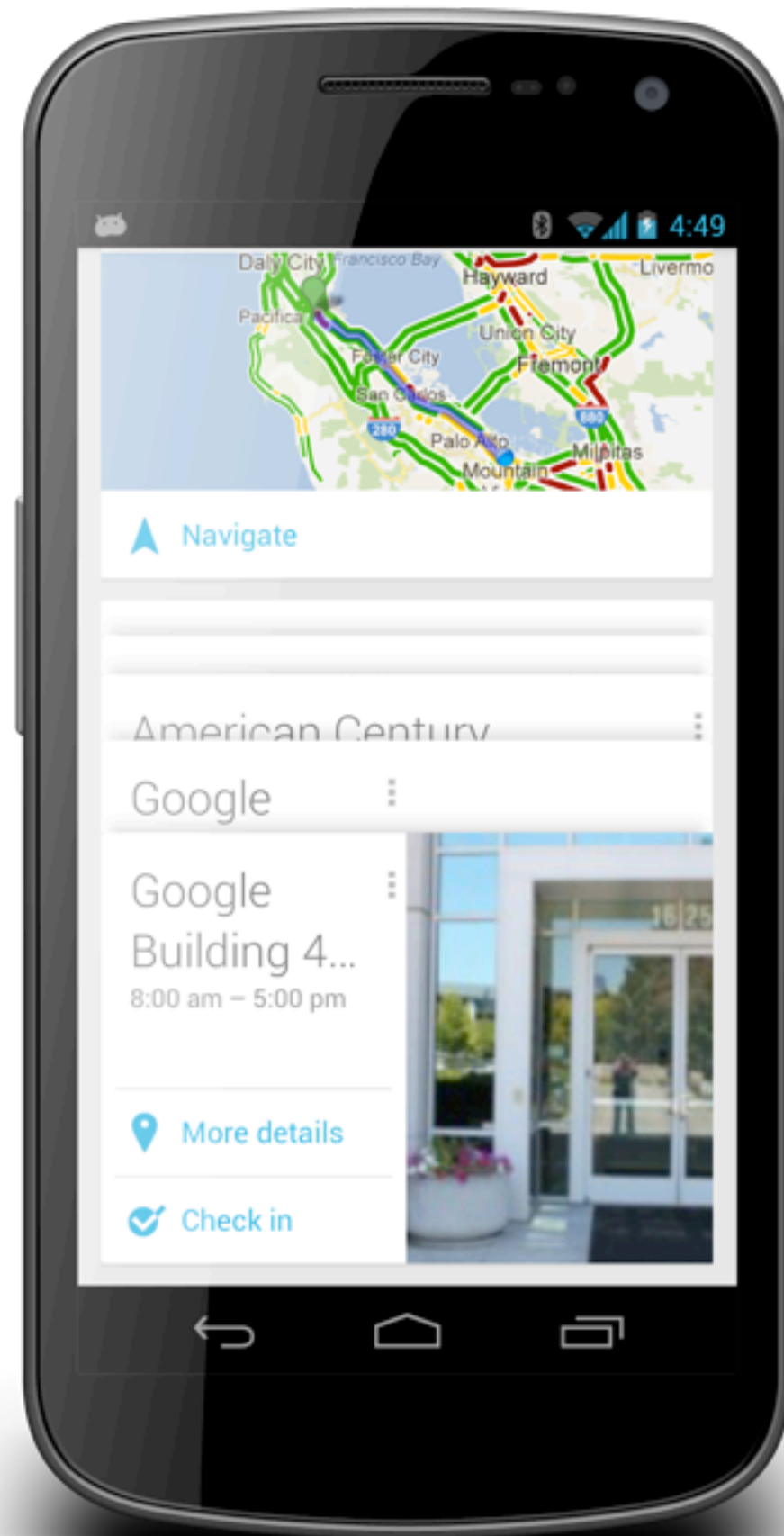
✓ Faster display list drawing

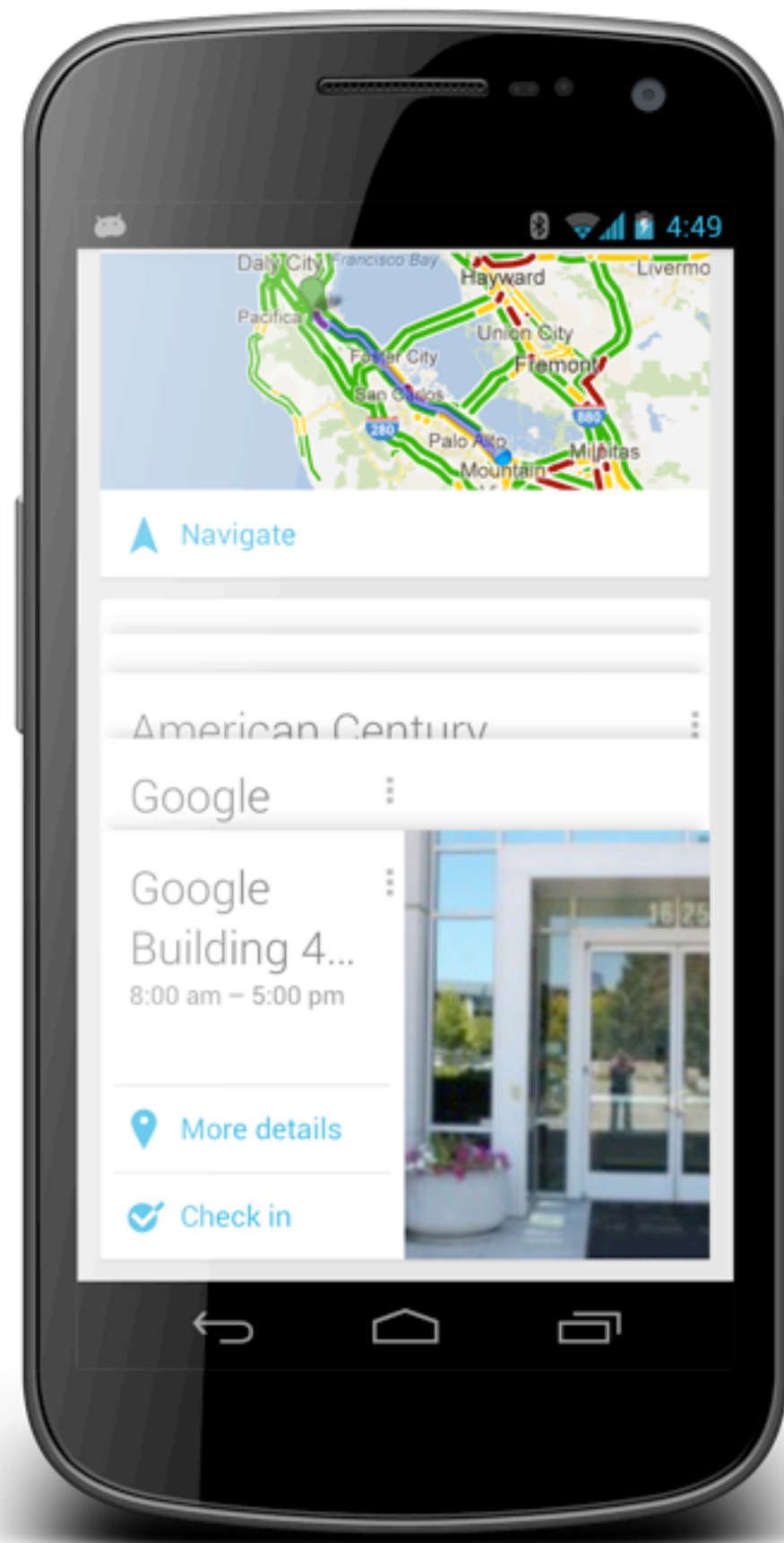


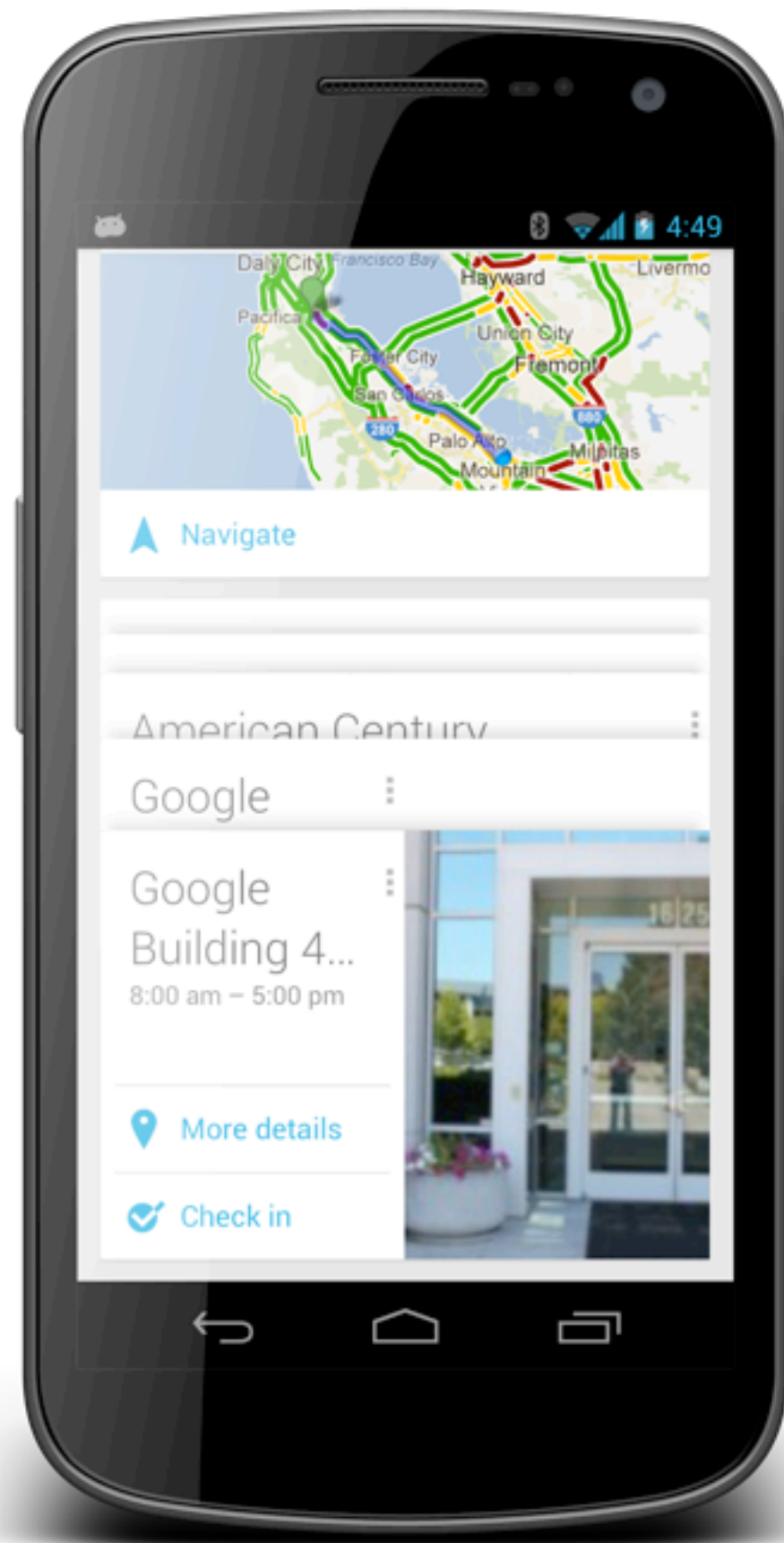




Don't draw invisible items







Google



Google
Building 4...
8:00 am – 5:00 pm



 More details

 Check in





Google



search
engine · customer
support · tech
support · customer...



More details



Check in



Google



Google
Building 4...



8:00 am – 5:00 pm

 More details

 Check in



Google



Google



Building 4...

8:00 am – 5:00 pm



More details



Check in



```
@Override
public void onDraw(Canvas c) {
    c.save();
    if (stacked) {
        c.clipRect(headerLeft, headerTop,
                  headerRight, headerBottom);
    }
    drawHeader(c);
    drawContent(c);
    c.restore();
}
```





clipRect



Google



Google



Google
Building 4...



8:00 am – 5:00 pm



More details



Check in



✓ Faster display list drawing



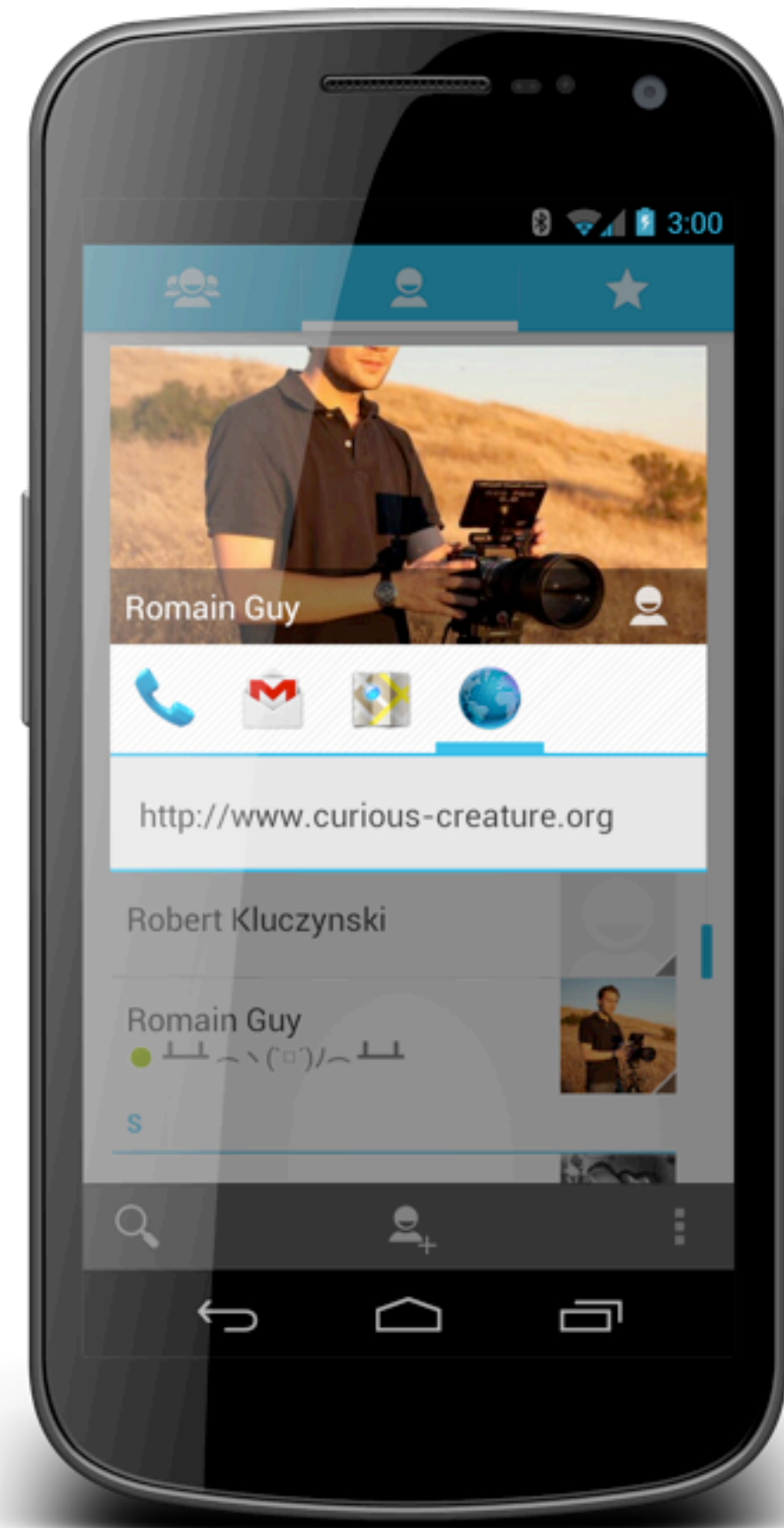
```
private void drawContent(Canvas c) {  
    for (Item item : itemList) {  
        if (!c.quickReject(item.l, item.t, item.r, item.b,  
            Canvas.EdgeType.BW)) {  
            item.draw(c);  
        }  
    }  
}
```



- ✓ Faster display list drawing
- ✓ Faster display list updates



Dim layer



```
getWindow().addFlags(  
    WindowManager.LayoutParams.FLAG_DIM_BEHIND);
```



type		...		name
-----+		...		+-----
OVERLAY		...		com...MyActivity
FB		...		DimAnimator
FB		...		StatusBar
FB		...		NavigationBar



	type		...		name
	-----+		...		+-----
✓	OVERLAY		...		com...MyActivity
✗	FB		...		DimAnimator
✗	FB		...		StatusBar
✗	FB		...		NavigationBar



```
getWindow().addFlags(  
—— WindowManager.LayoutParams.FLAG_DIM_BEHIND);
```




```
getWindow().addFlags(  
—— WindowManager.LayoutParams.FLAG_DIM_BEHIND);  
getWindow().setBackground(  
    new ColorDrawable(0x7f000000));
```



- ✓ Faster display list drawing
- ✓ GPU-free window composition



Users love butter



Users love **butter**

Spread the word



For More Information

Google I/O 2011
Parleys.com

Accelerated Android Rendering
Various Android GUI performance talks



Q&A

+Chet Haase
+Romain Guy

@chethaase
@romainguy





Google
Developers