Google™ 11 I O

Android Rendering

Romain Guy
Chet Haase
May 11, 2011

@romainguy
@chethaase

Feedback goo.gl/wI57L
Hashtags #io2011, #Android

Google™ 11 I|O

Android Accelerated Rendering

Romain Guy
Chet Haase
May 11, 2011

@romainguy
@chethaase

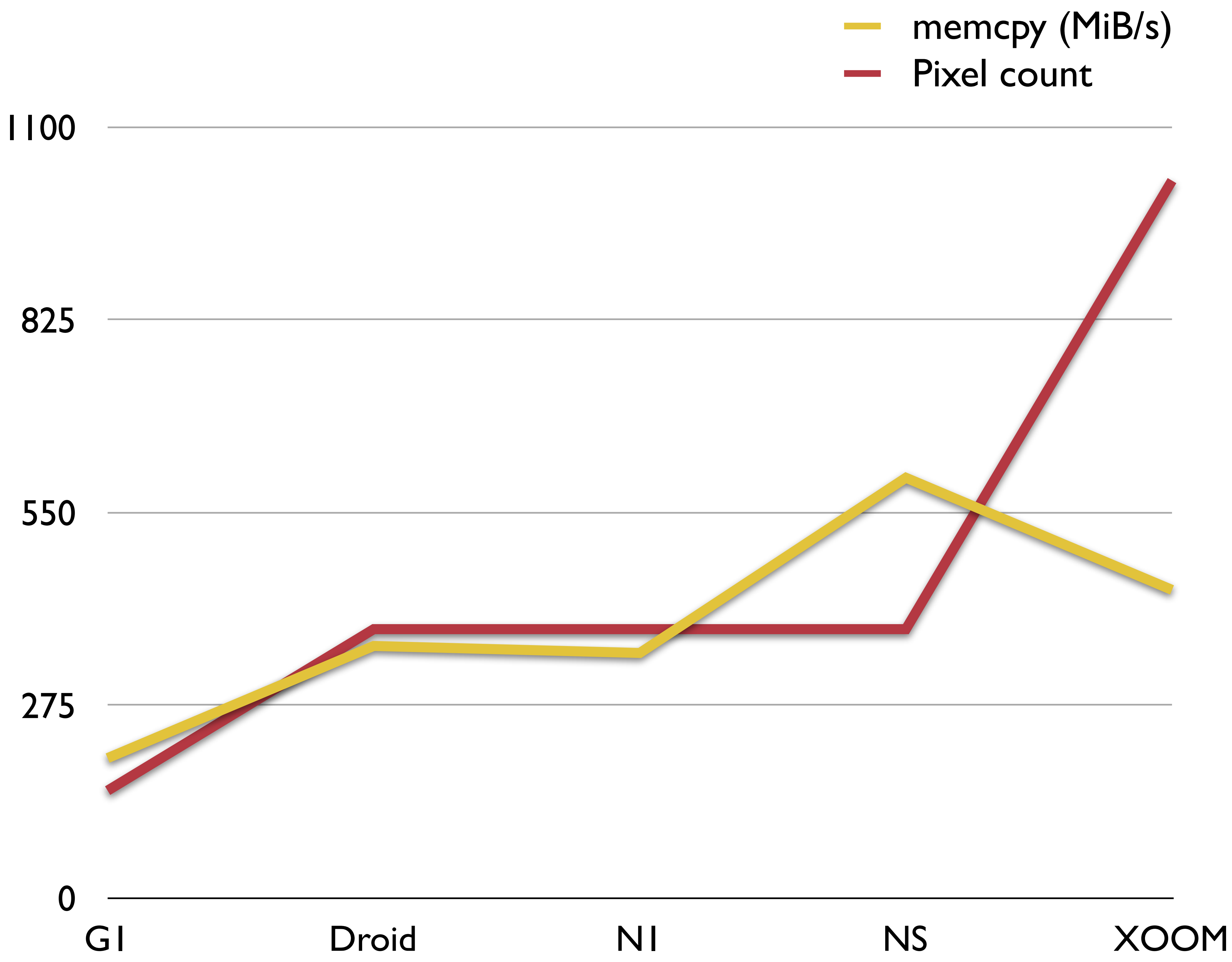
Feedback goo.gl/wI57L
Hashtags #io2011, #Android

Hardware OpenGL

3:05
Google
Gmail
Talk
Docs
Google Re
Books
Contacts
Calendar
Maps

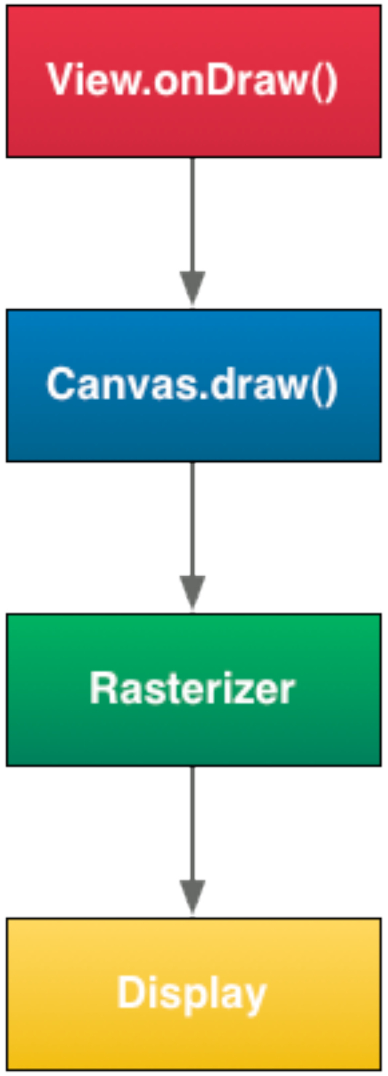
Why now?

Google
Apps
Browser
Calendar
talk
Talk
Gmail
Maps
Blogs
Games
Photo
News
Music
4:25

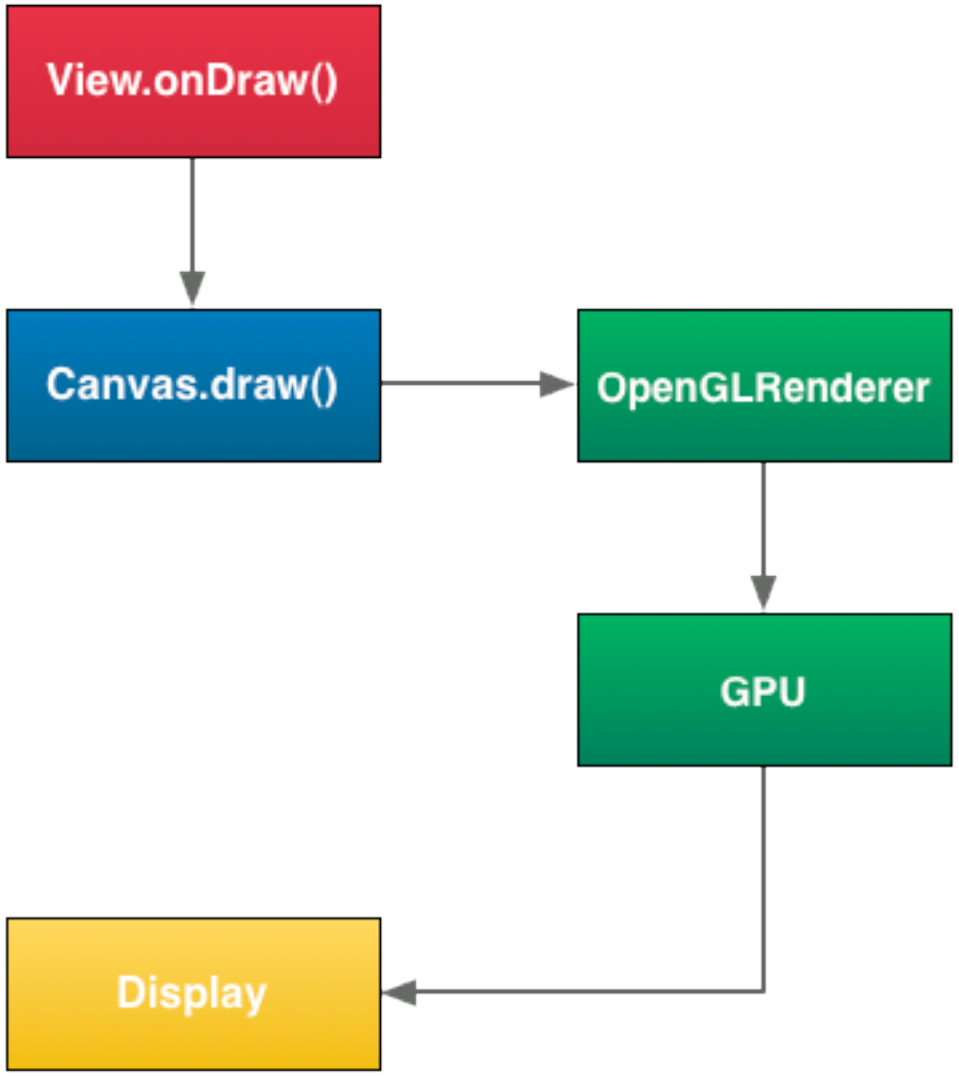

memcpy (MiB/s)
Pixel count
1100
825
550
275
0
G1
Droid
N1
NS
XOOM

UI on the GPU

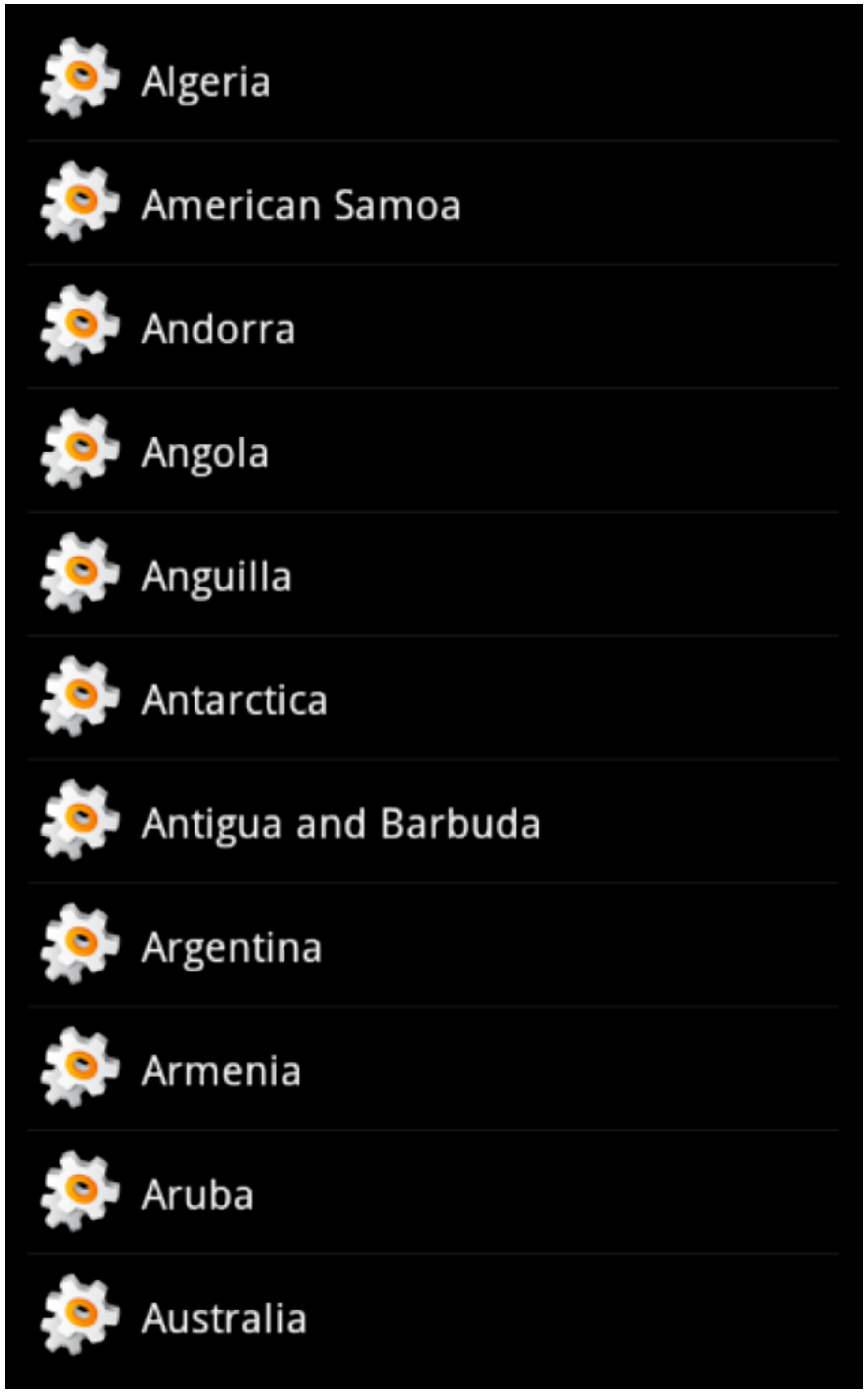
GPUI



Software rendering



Hardware rendering



Algeria

American Samoa

Andorra

Angola

Anguilla

Antarctica

Antigua and Barbuda

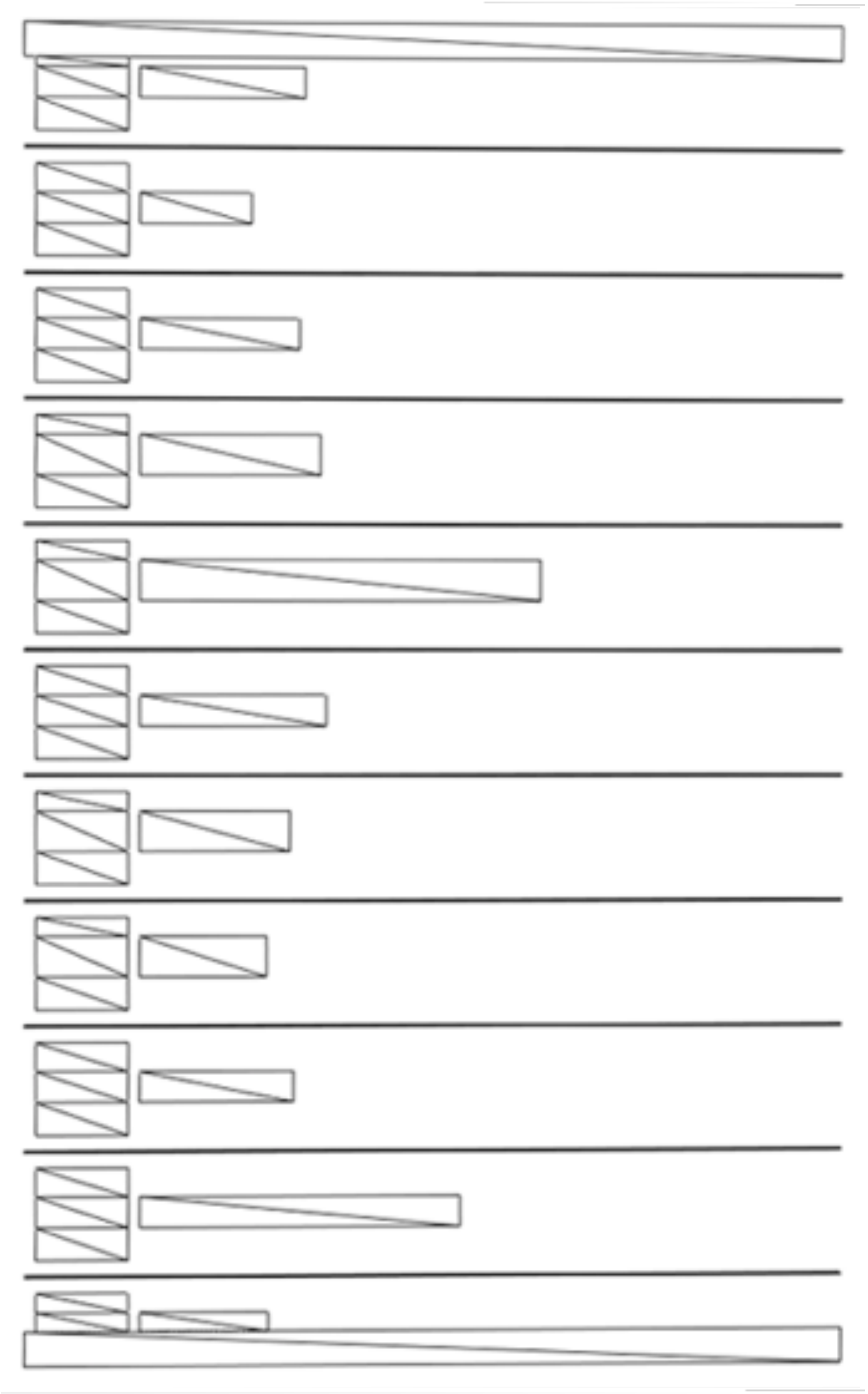
Argentina

Armenia

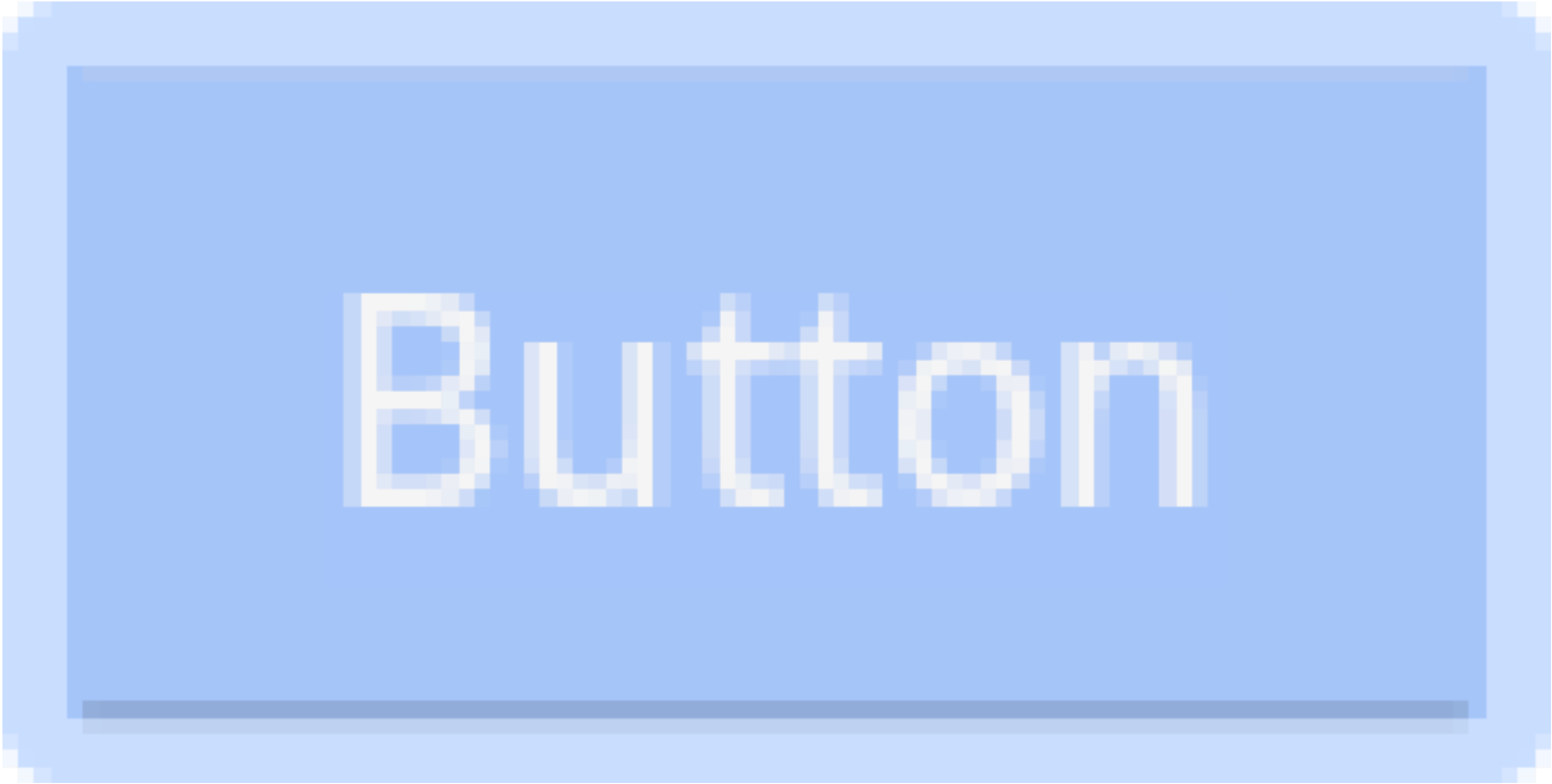
Aruba

Australia

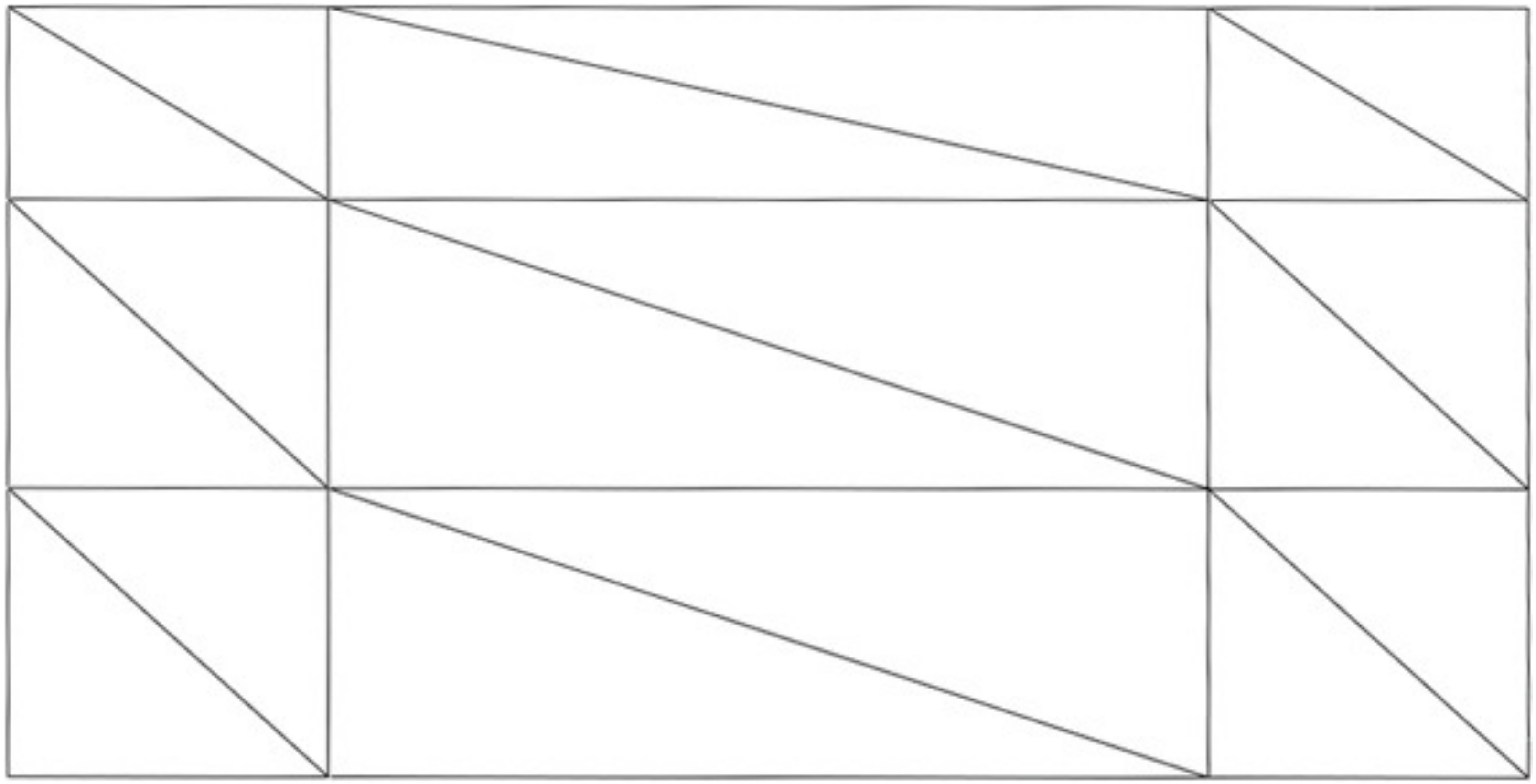
Drawing a list



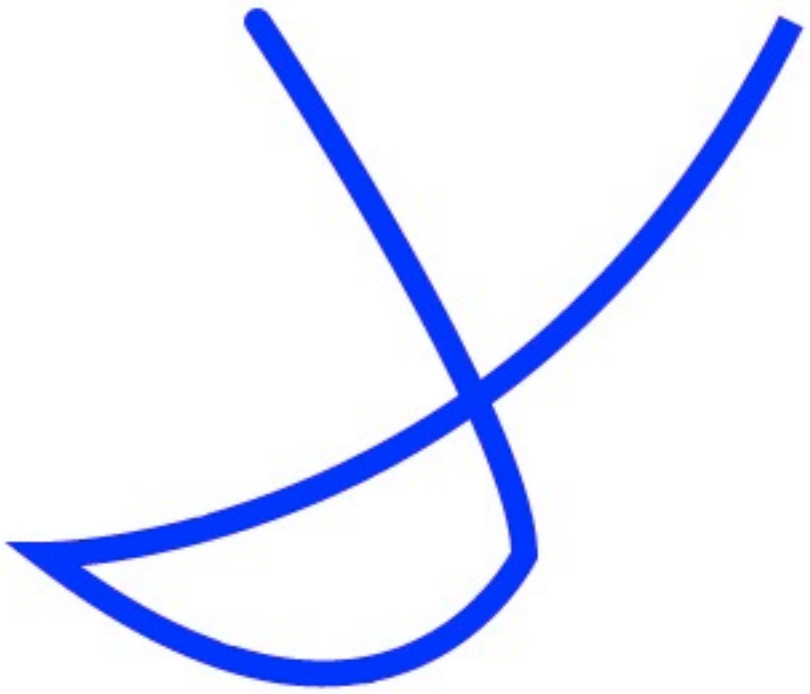
Drawing a list



Drawing a button

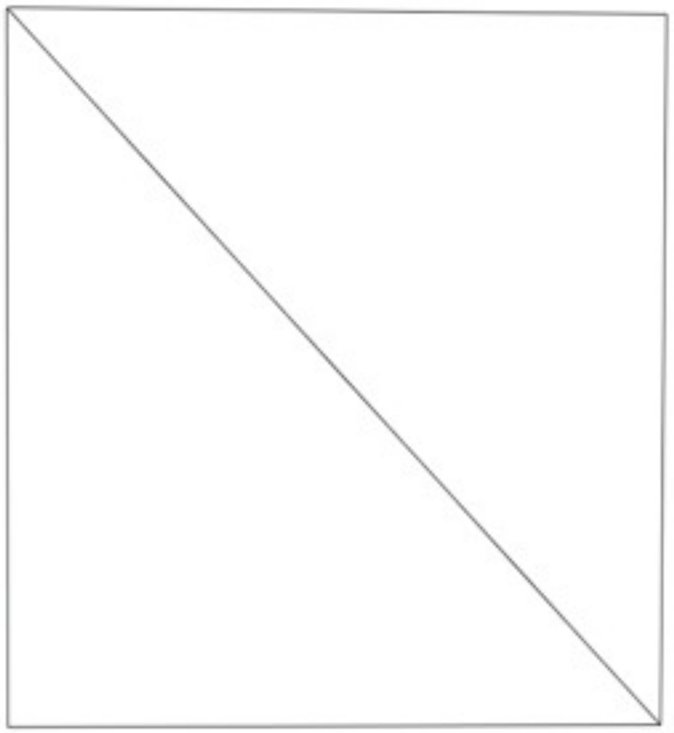


Drawing a button



Drawing a path

Drawing a path



Drawing a path

```
attribute vec4 position;
attribute vec2 texCoords;
uniform mat4 transform;
uniform mat4 screenSpace;
varying vec2 outTexCoords;
varying vec2 linear;

void main(void) {
    outTexCoords = texCoords;
    linear = vec2((screenSpace * position).x, 0.5);
    gl_Position = transform * position;
}
```

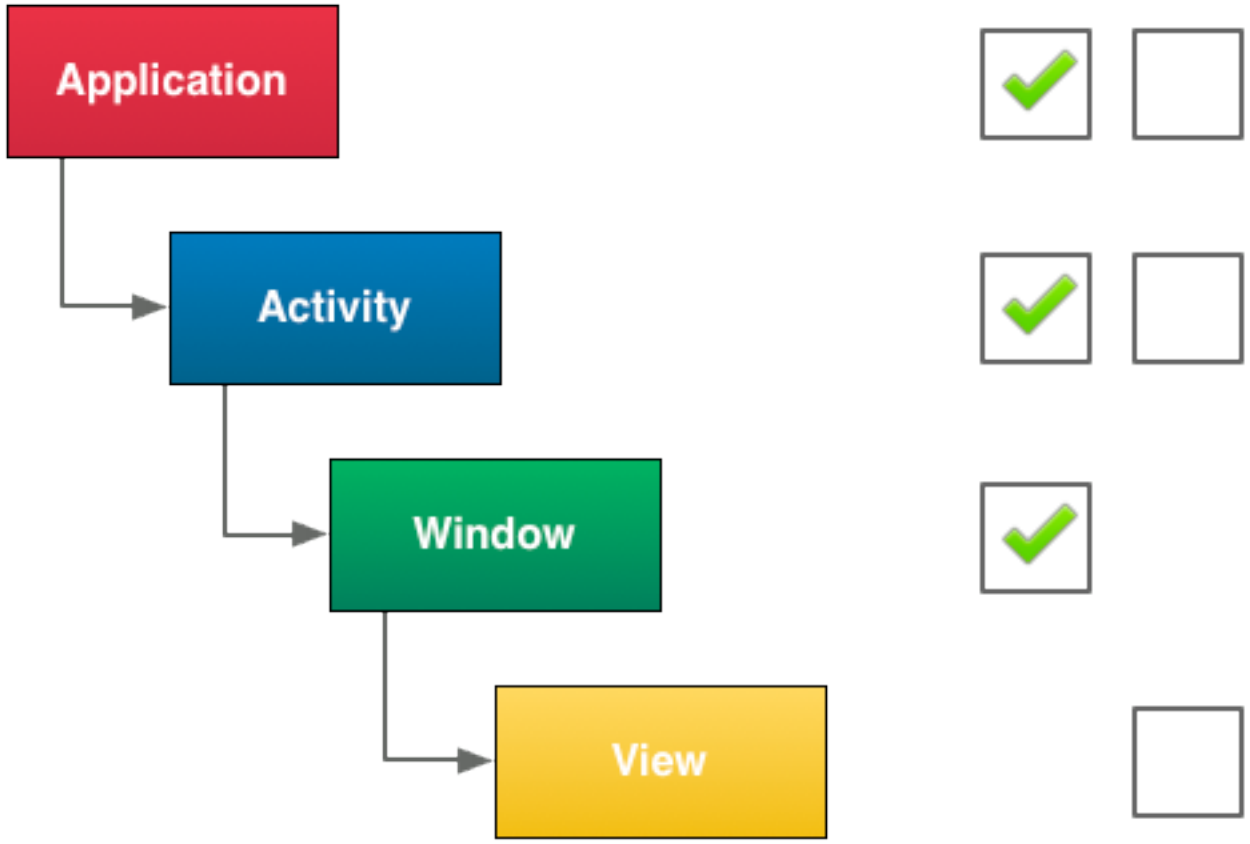
Text+gradient vertex shader

```
precision mediump float;

varying vec2 outTexCoords;
varying vec2 linear;
uniform sampler2D sampler;
uniform sampler2D gradientSampler;

void main(void) {
    lowp vec4 color;
    vec4 gradient = texture2D(gradientSampler, linear);
    color = gradient * texture2D(sampler, outTexCoords).a;
    gl_FragColor = color;
}
```

Text+gradient fragment shader

?

```
android:hardwareAccelerated="true"
```



Hardware acceleration control

```xml
<application android:hardwareAccelerated="true">
    <activity ... />
    <activity android:hardwareAccelerated="false" />
</application>
```

```
getWindow().setFlags(
    WindowManager.LayoutParams.FLAG_HARDWARE_ACCELERATED,
    WindowManager.LayoutParams.FLAG_HARDWARE_ACCELERATED);
```

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

```
View.isHardwareAccelerated();
Canvas.isHardwareAccelerated();
```

GPU aware code

Caveats

Canvas	clipPath
	clipRegion
	drawPicture
	drawPoints
	drawPosText
	drawTextOnPath
	drawVertices
Paint	setLinearText
	setMaskFilter
	setRasterizer

Unsupported operations

Limitations

Canvas	clipRect	3D transforms, XOR, Diff, ReverseDiff ignored
	drawBitmapMesh	Colors array ignored
	drawLines	No anti-aliasing
	setDrawFilter	Ignored
Paint	setDither	Ignored
	setFilterBitmap	Filtering always on
	setShadowLayer	Works with text only

ComposeShader

Cannot contain another ComposeShader

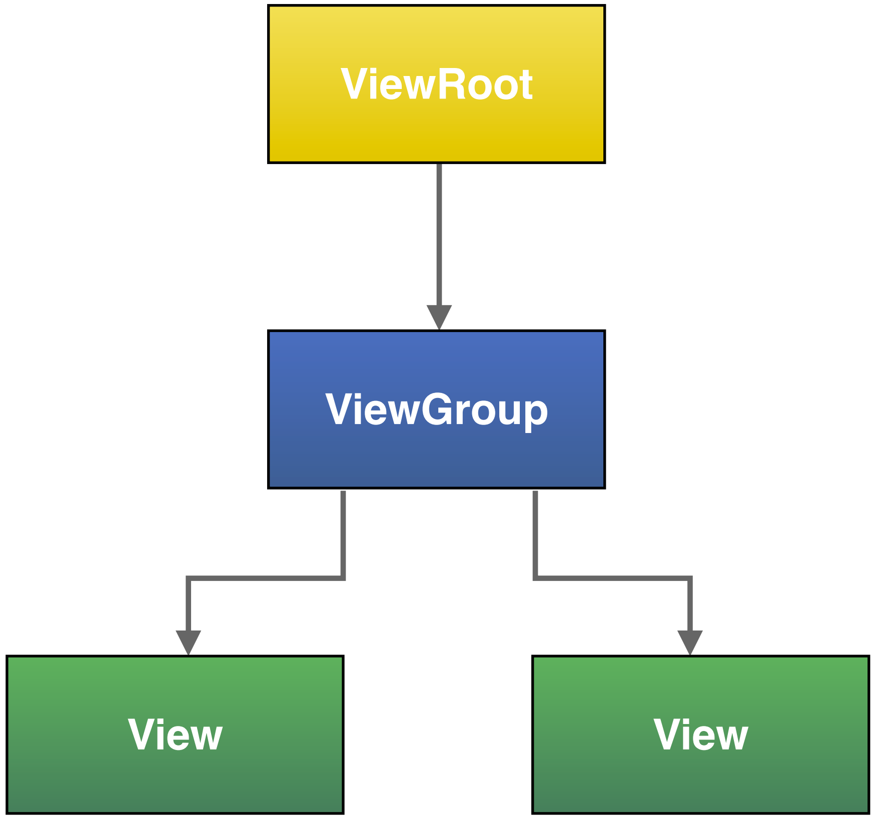
Cannot contain two shaders of the same type

Limitations

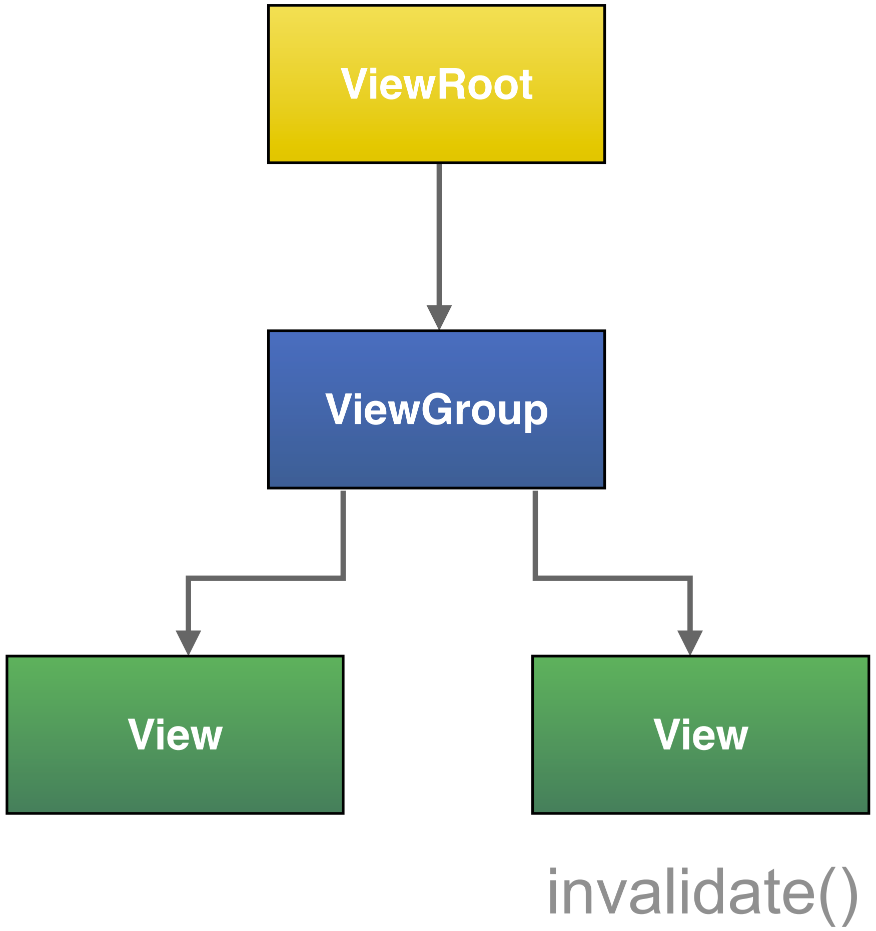
```
view.setLayerType(View.LAYER_TYPE_SOFTWARE, null);
```

Workaround

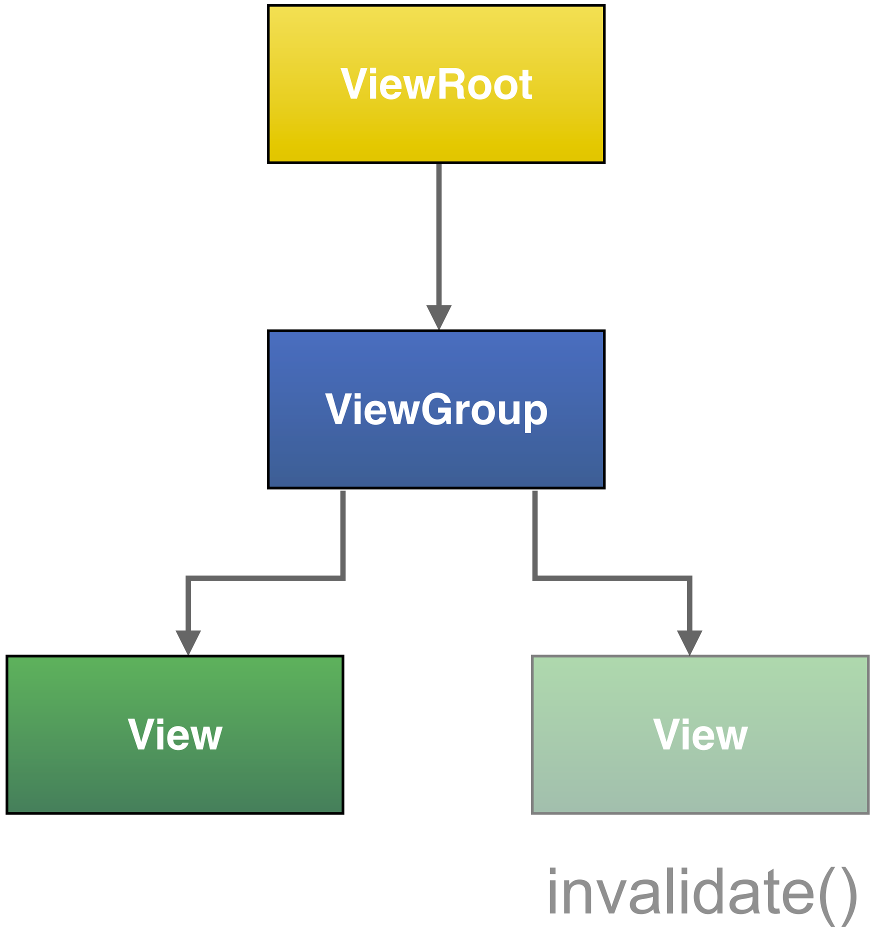
New drawing model



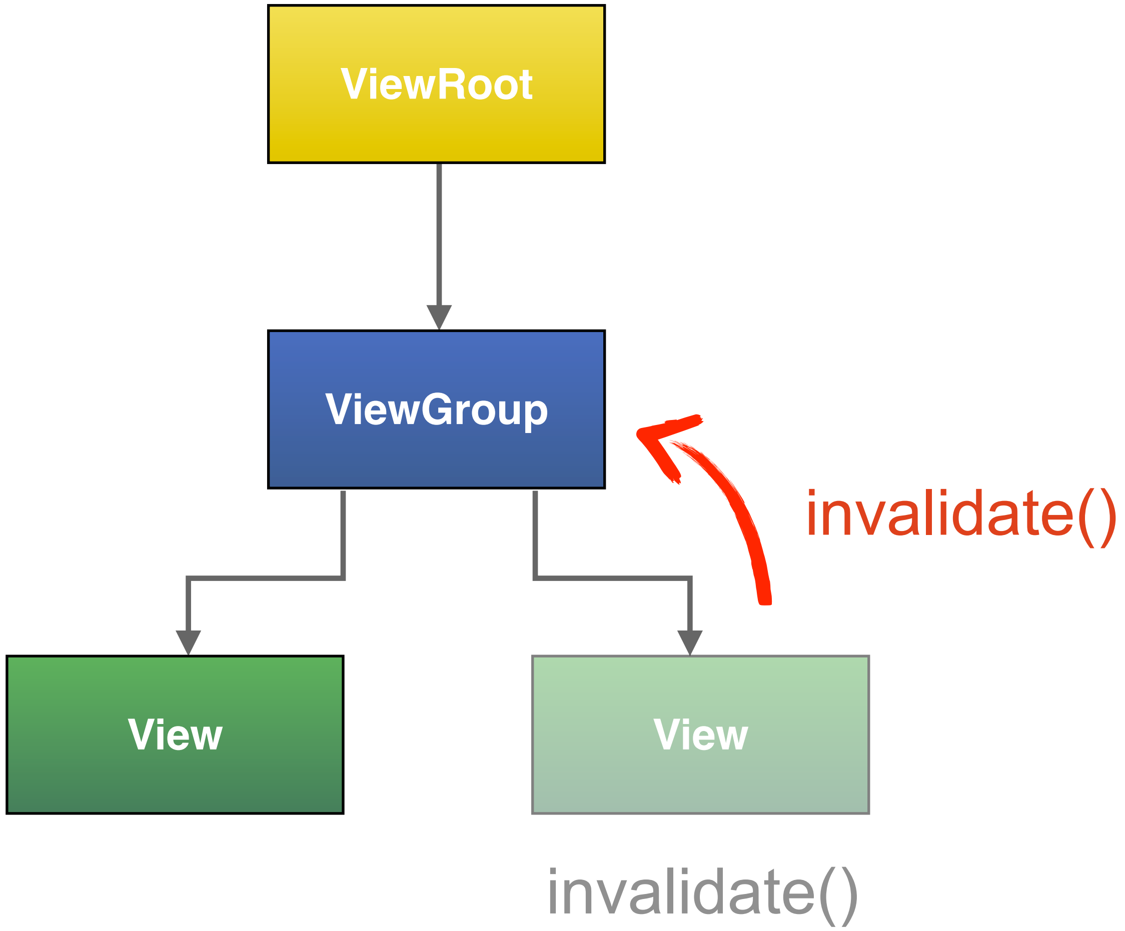
Old model



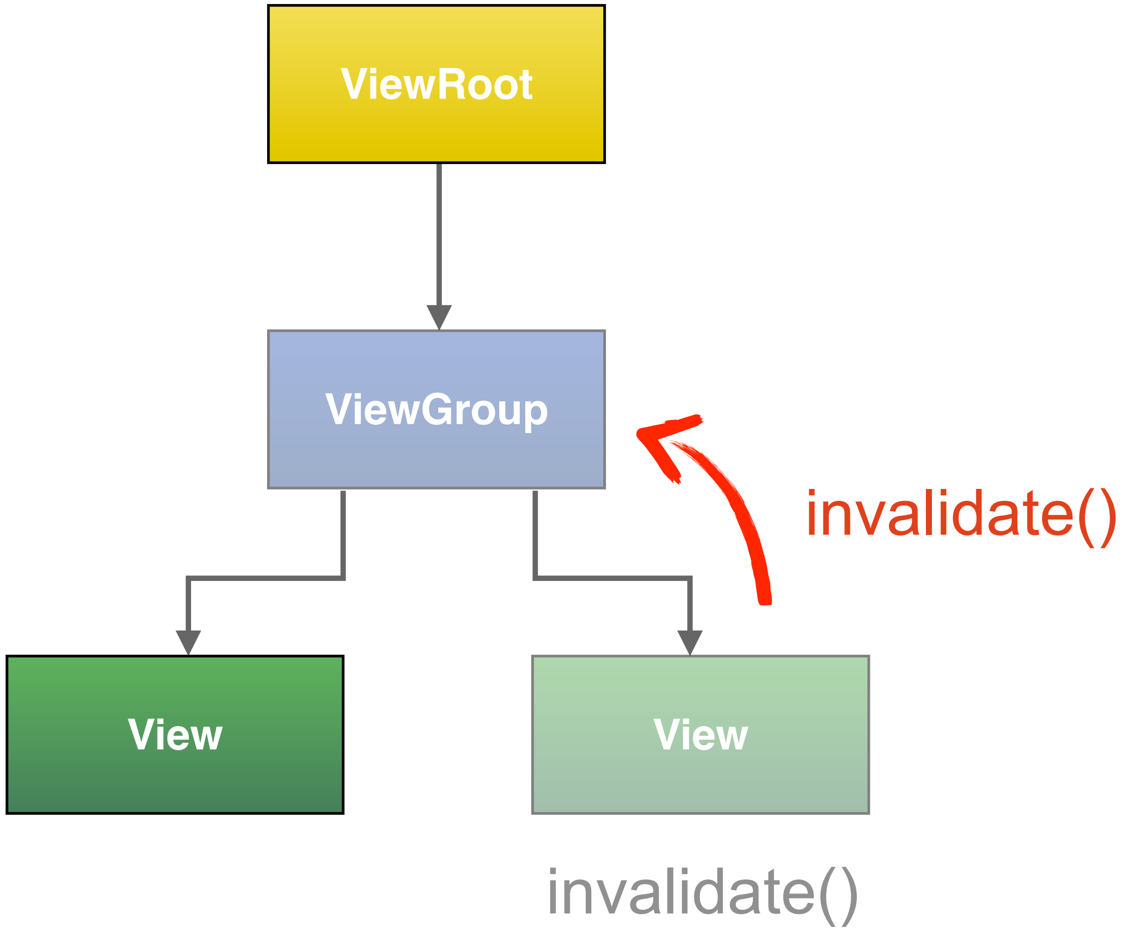
Old model



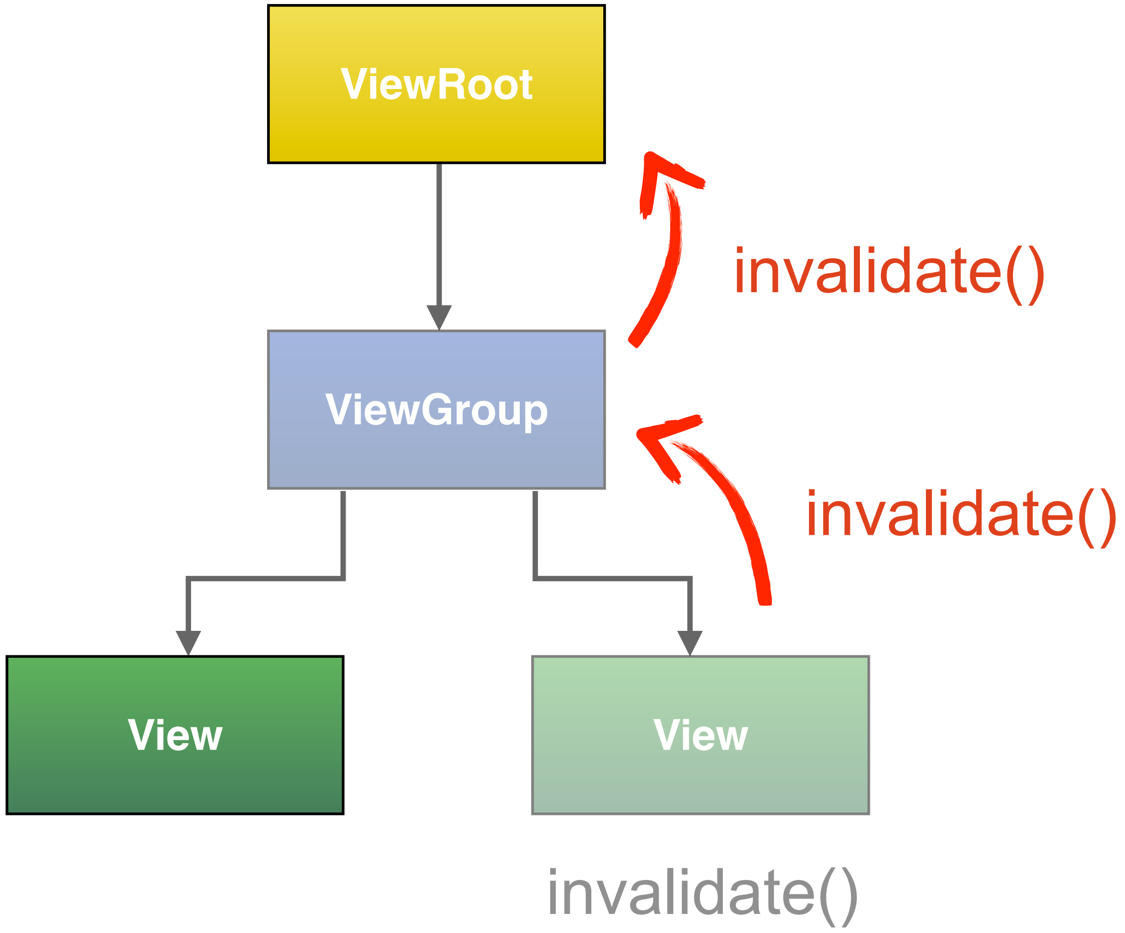
Old model



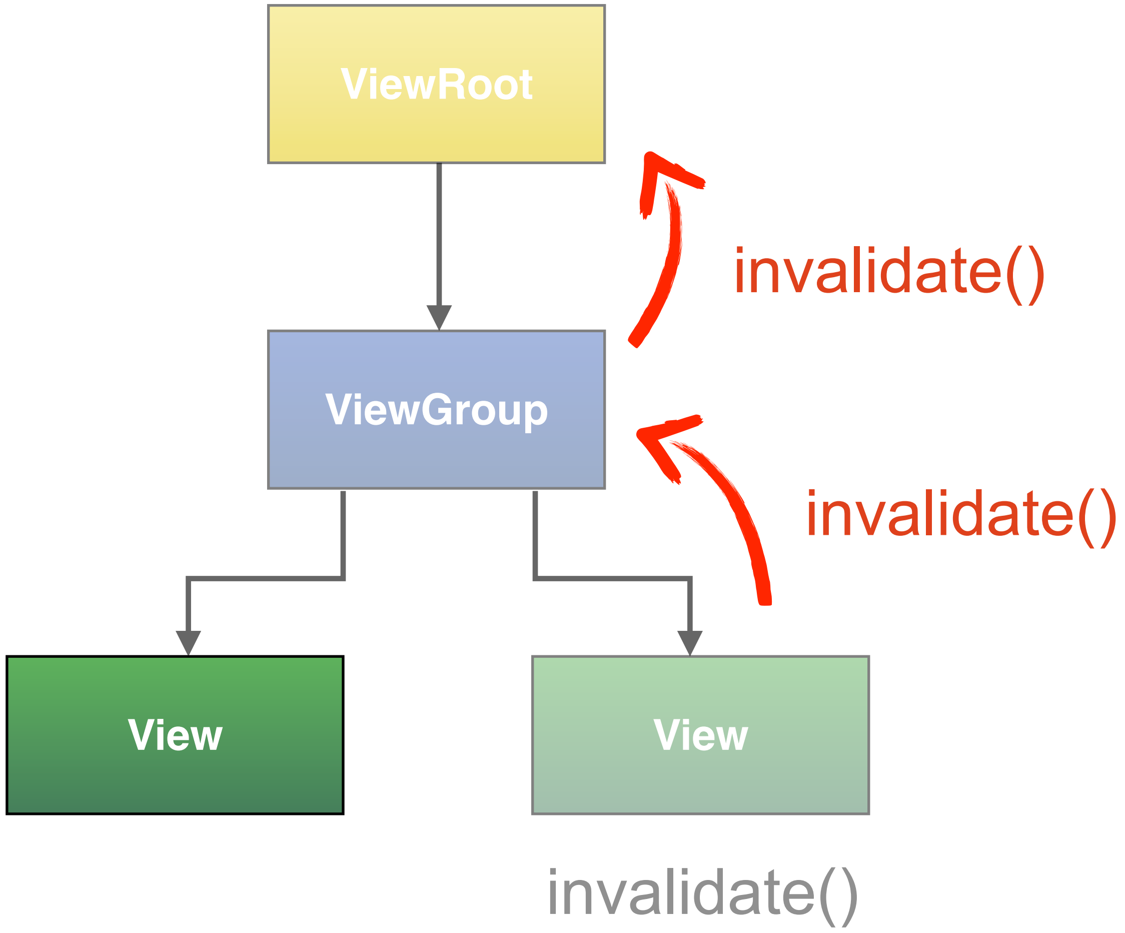
Old model



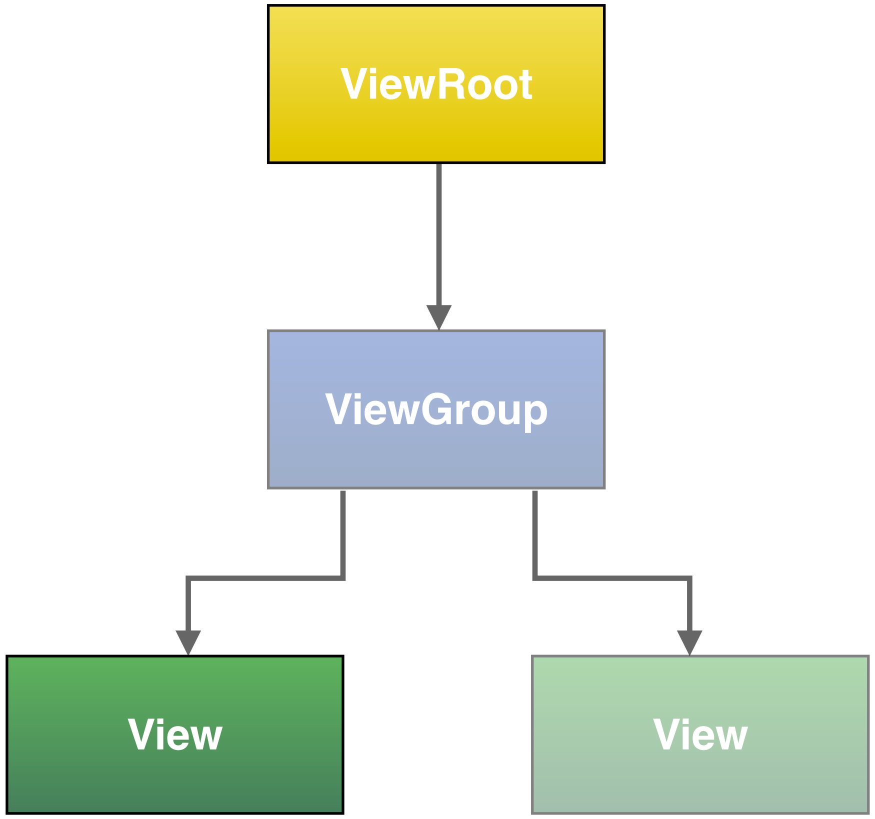
Old model


ViewRoot
invalidate()
ViewGroup
invalidate()
View
View
invalidate()

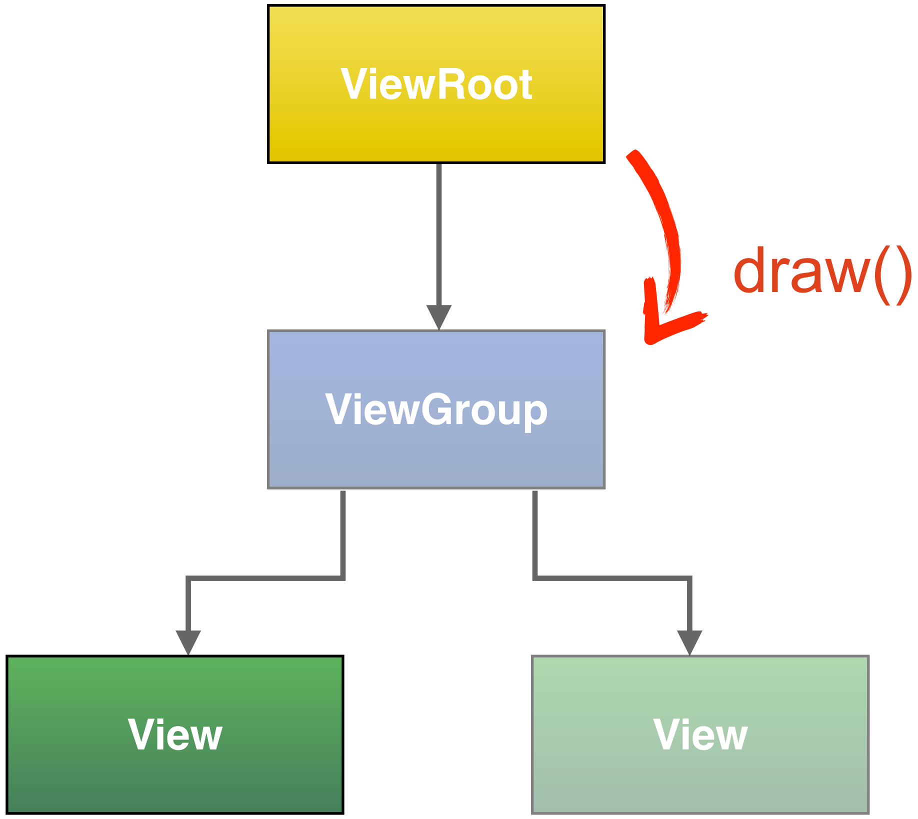
Old model


ViewRoot
invalidate()
ViewGroup
invalidate()
View
View
invalidate()

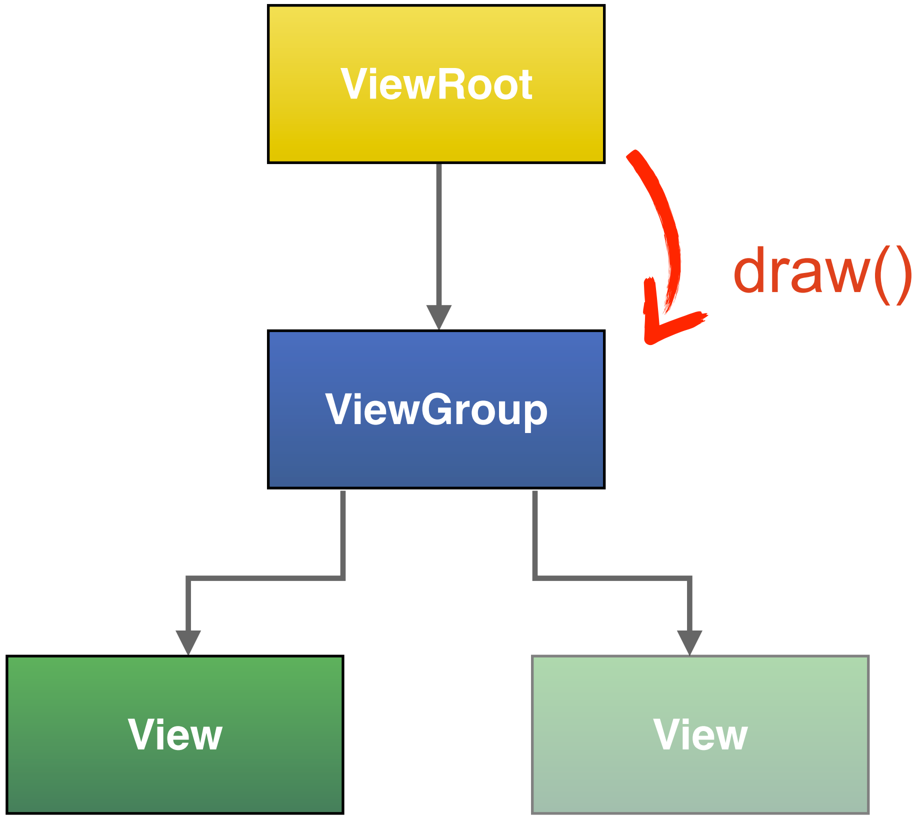
Old model



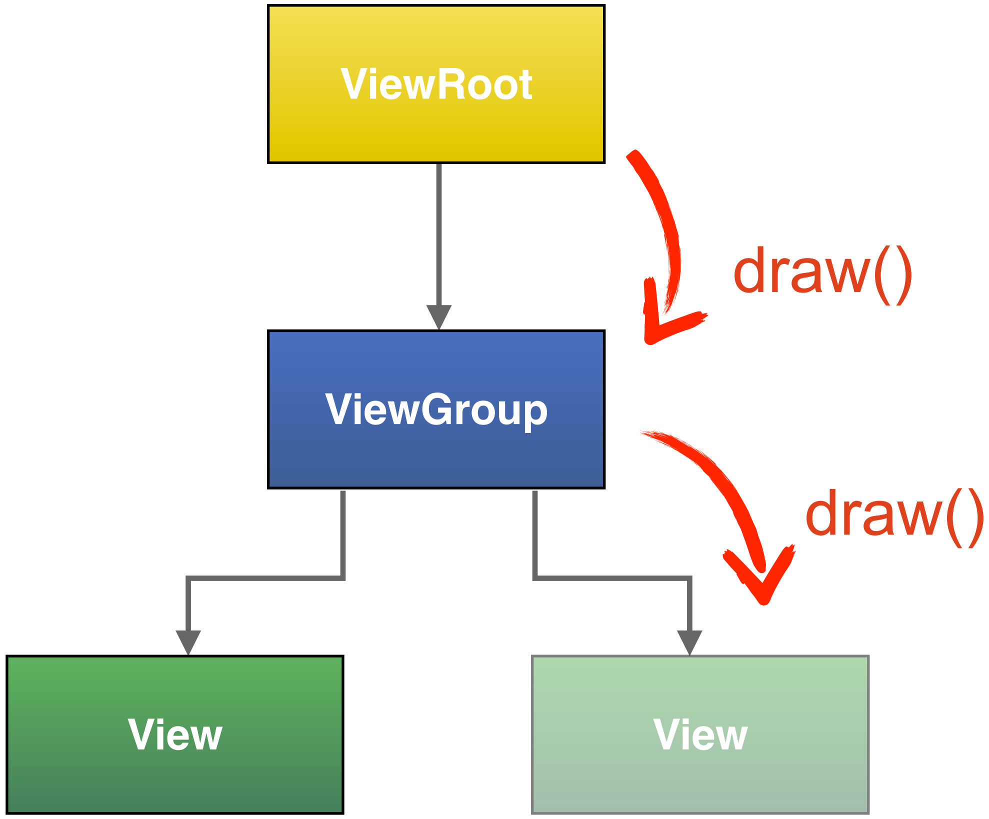
Old model



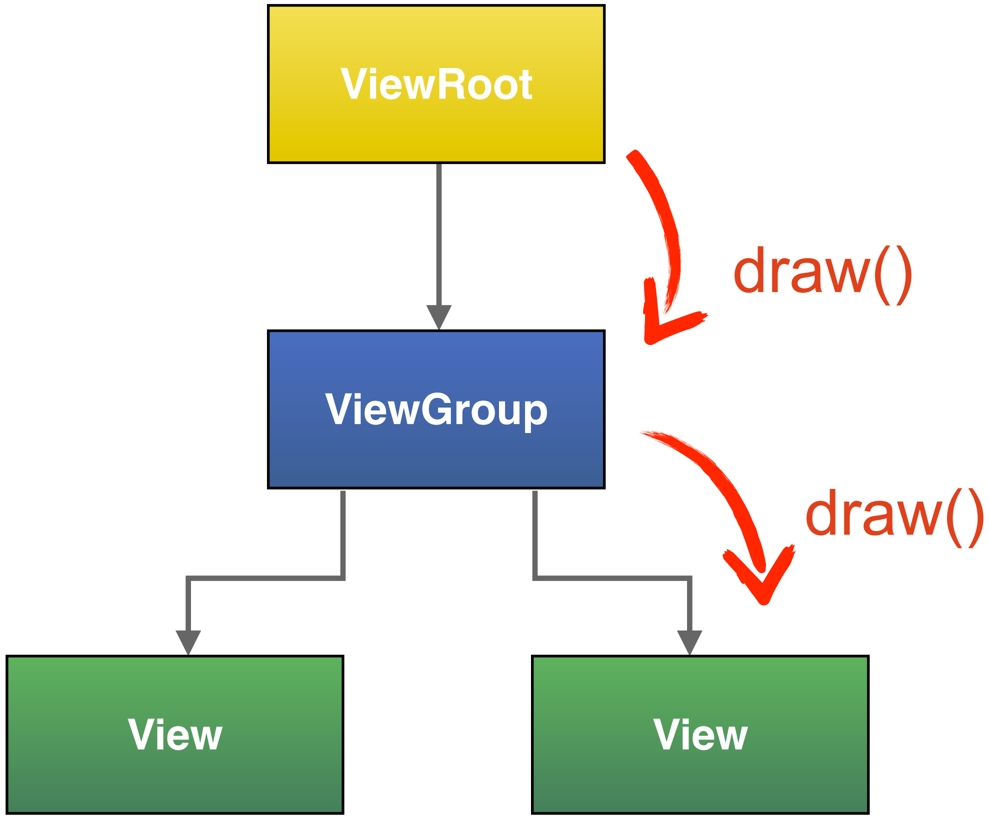
Old model



Old model



Old model



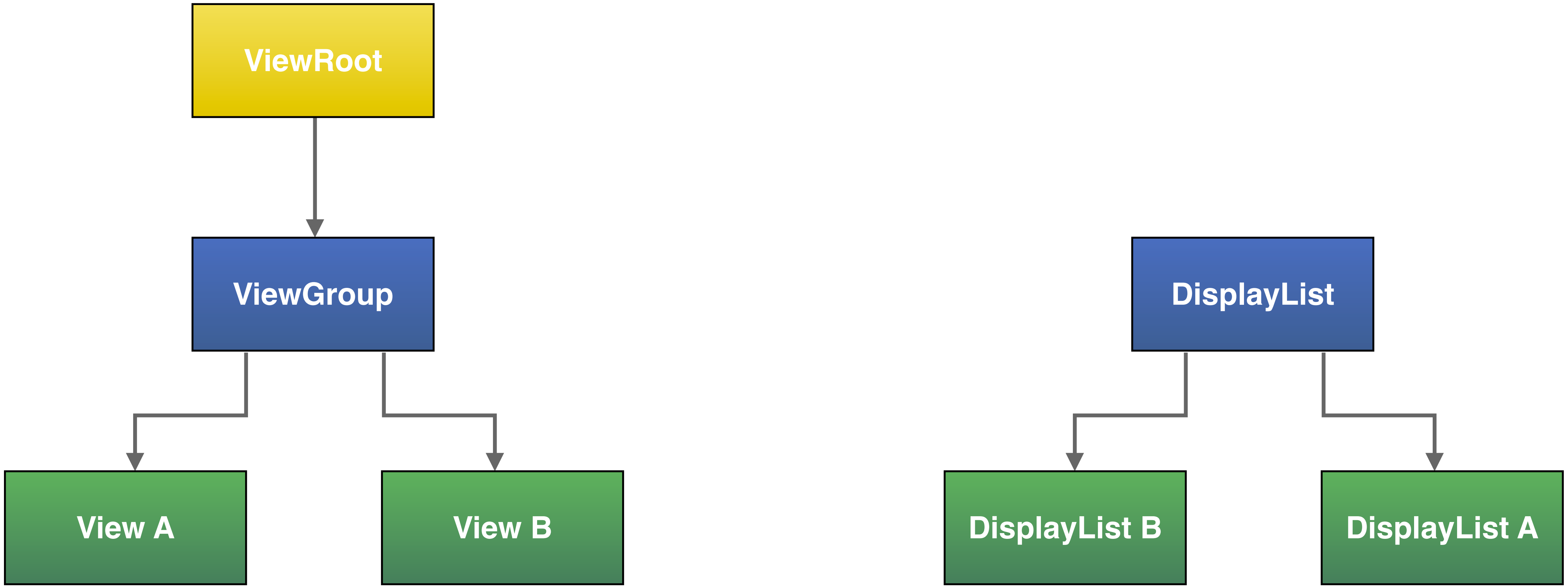
Old model

- No logic!
- List of drawing commands
- Cached when a View changes

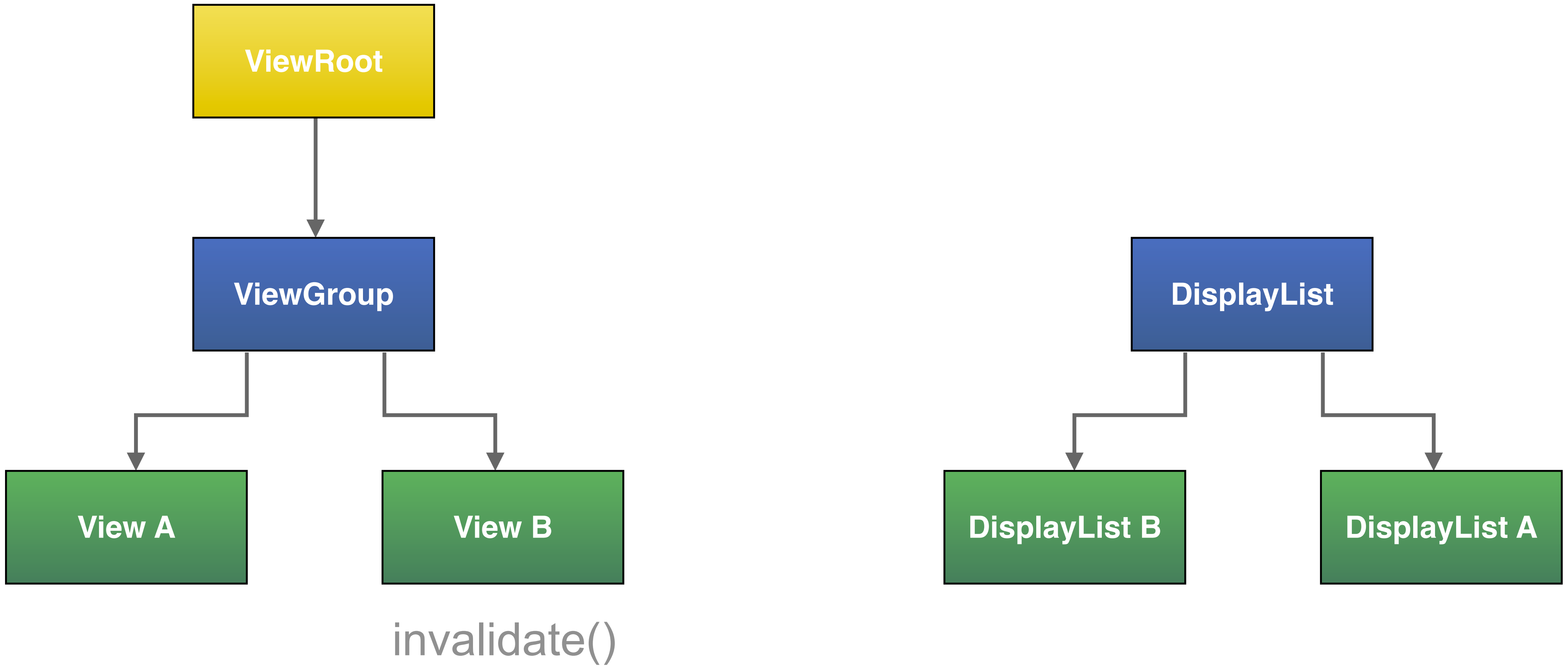
DisplayList

```
Save 3
DrawPatch
Save 3
ClipRect 20.00, 4.00, 99.00, 44.00, 1
Translate 20.00, 12.00
DrawText 9, 18, 9, 0.00, 19.00, 0x17e898
Restore
RestoreToCount 0
```

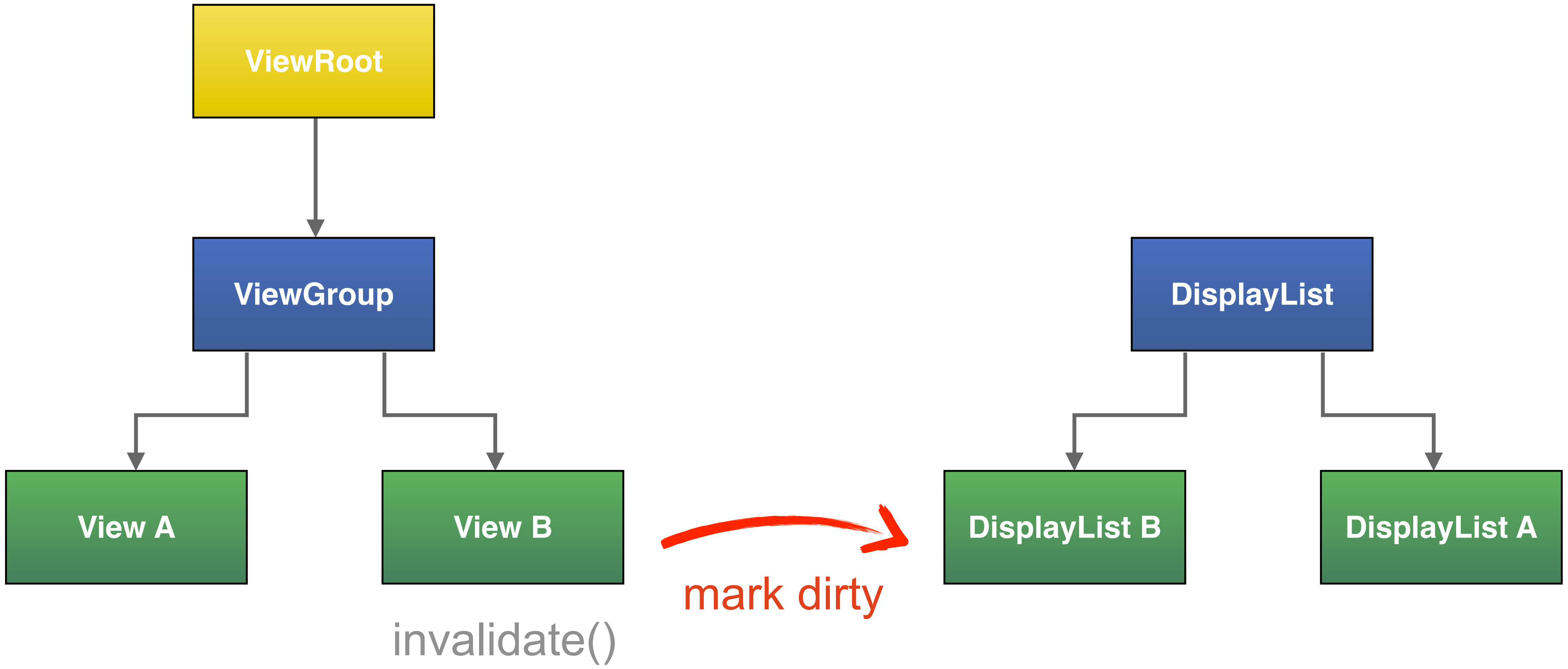
DisplayList to draw a Button


ViewRoot
ViewGroup
View A
View B
DisplayList
DisplayList B
DisplayList A

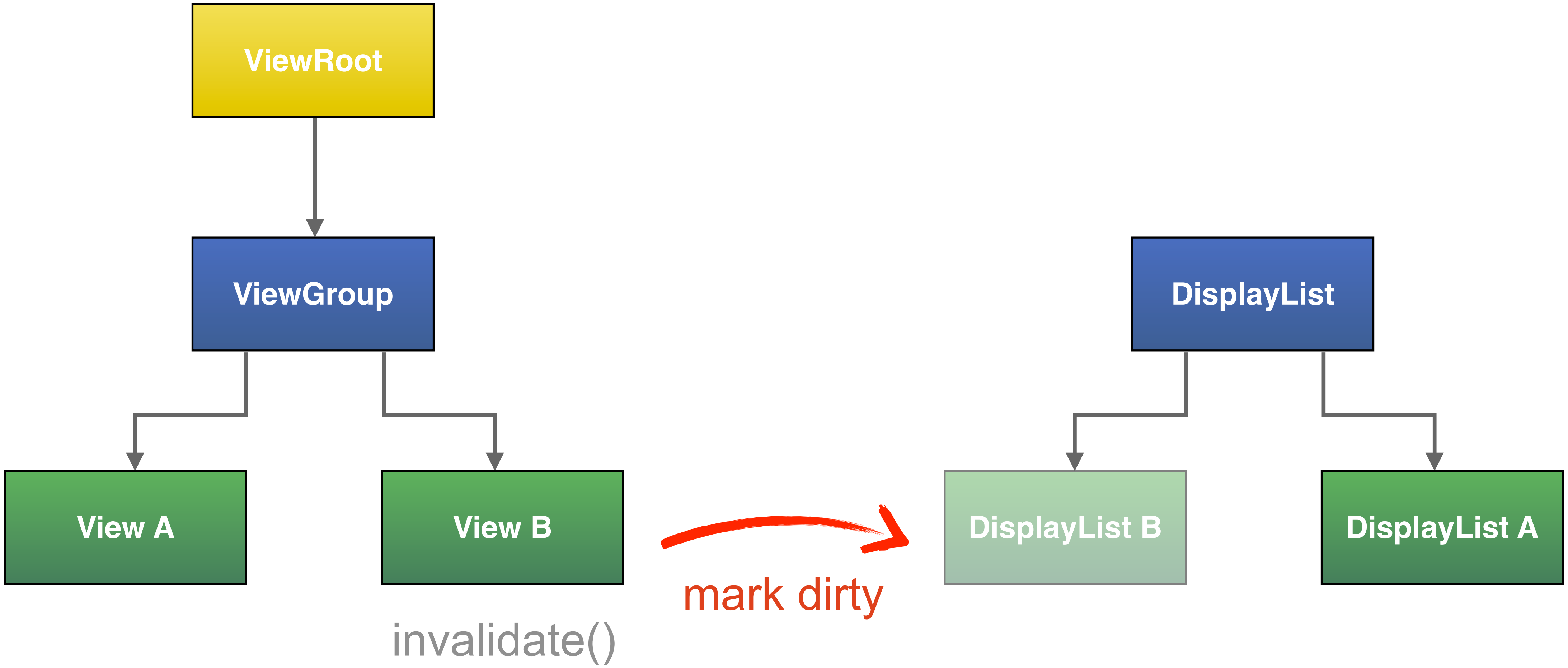
New model


ViewRoot
ViewGroup
View A
View B
invalidate()
DisplayList
DisplayList B
DisplayList A

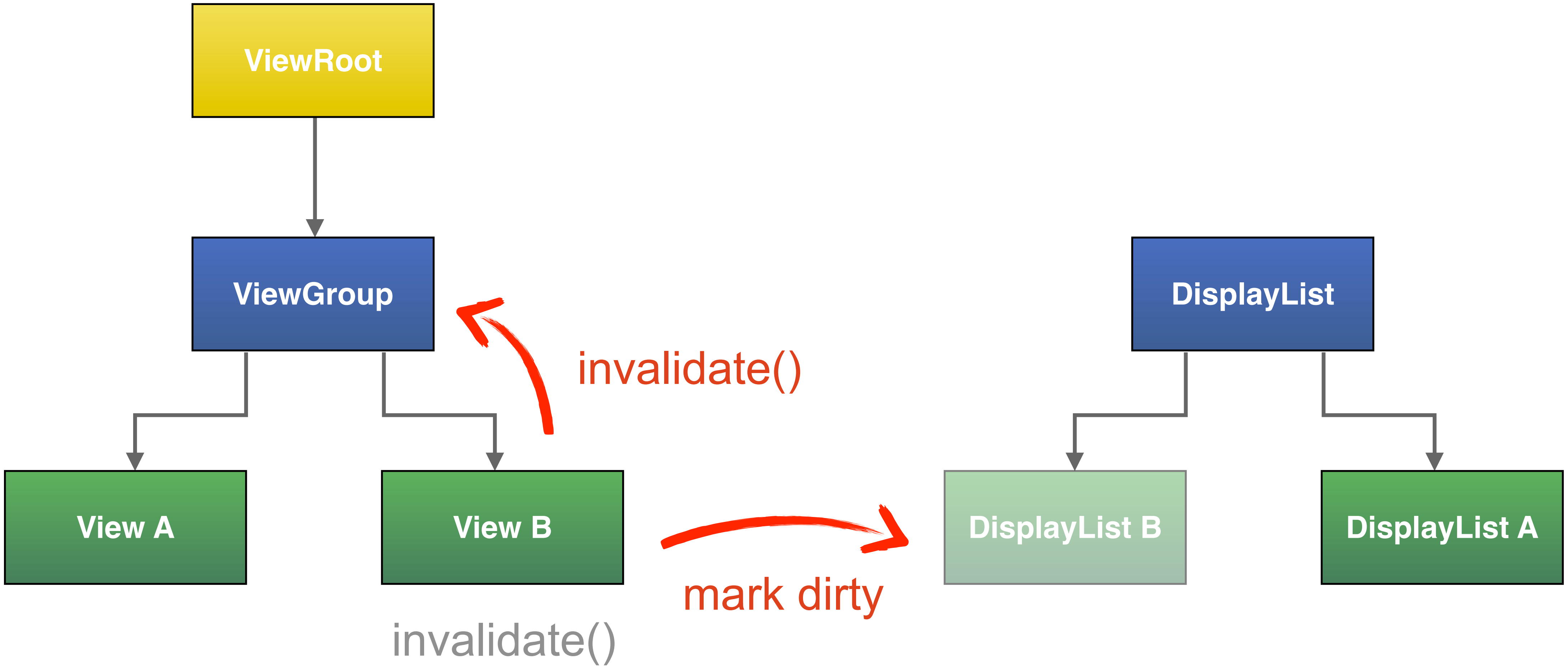
New model


ViewRoot
ViewGroup
View A
View B
invalidate()
mark dirty
DisplayList
DisplayList B
DisplayList A

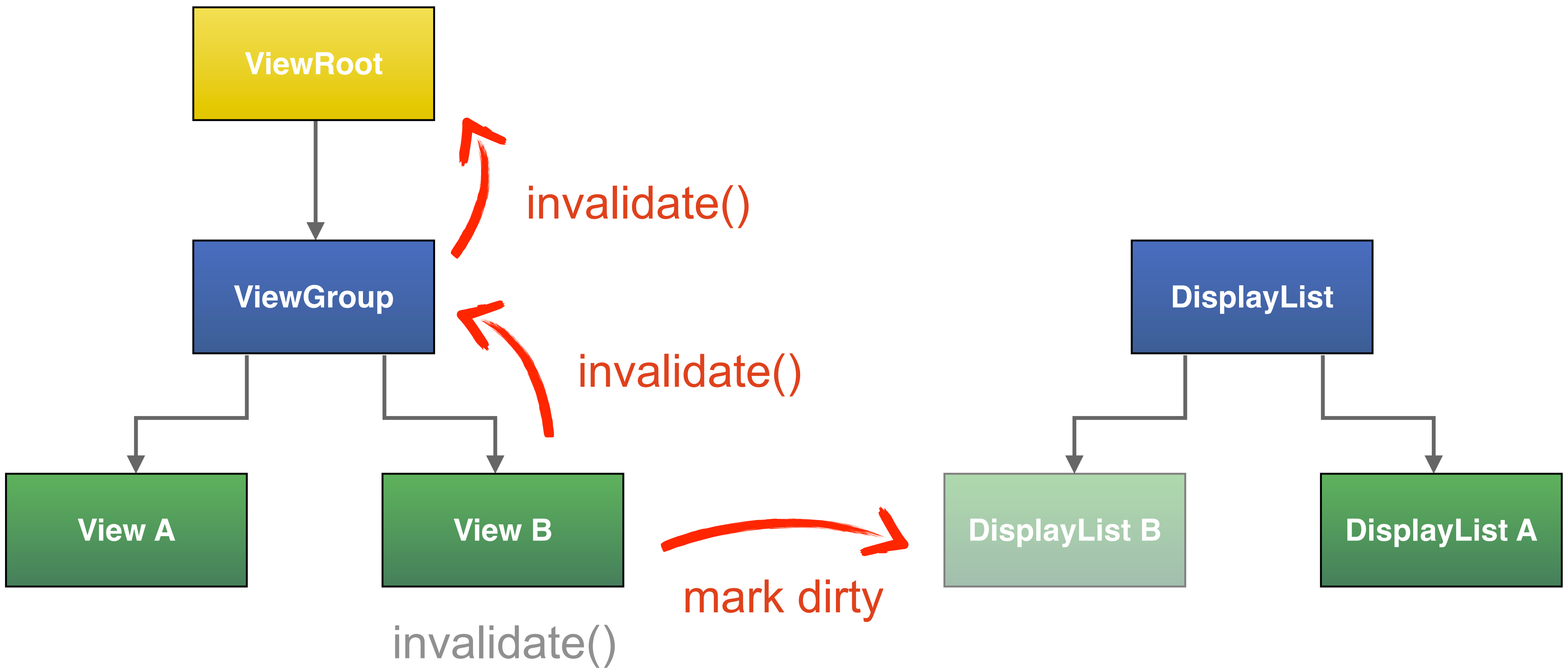
New model


ViewRoot
ViewGroup
View A
View B
invalidate()
mark dirty
DisplayList
DisplayList B
DisplayList A

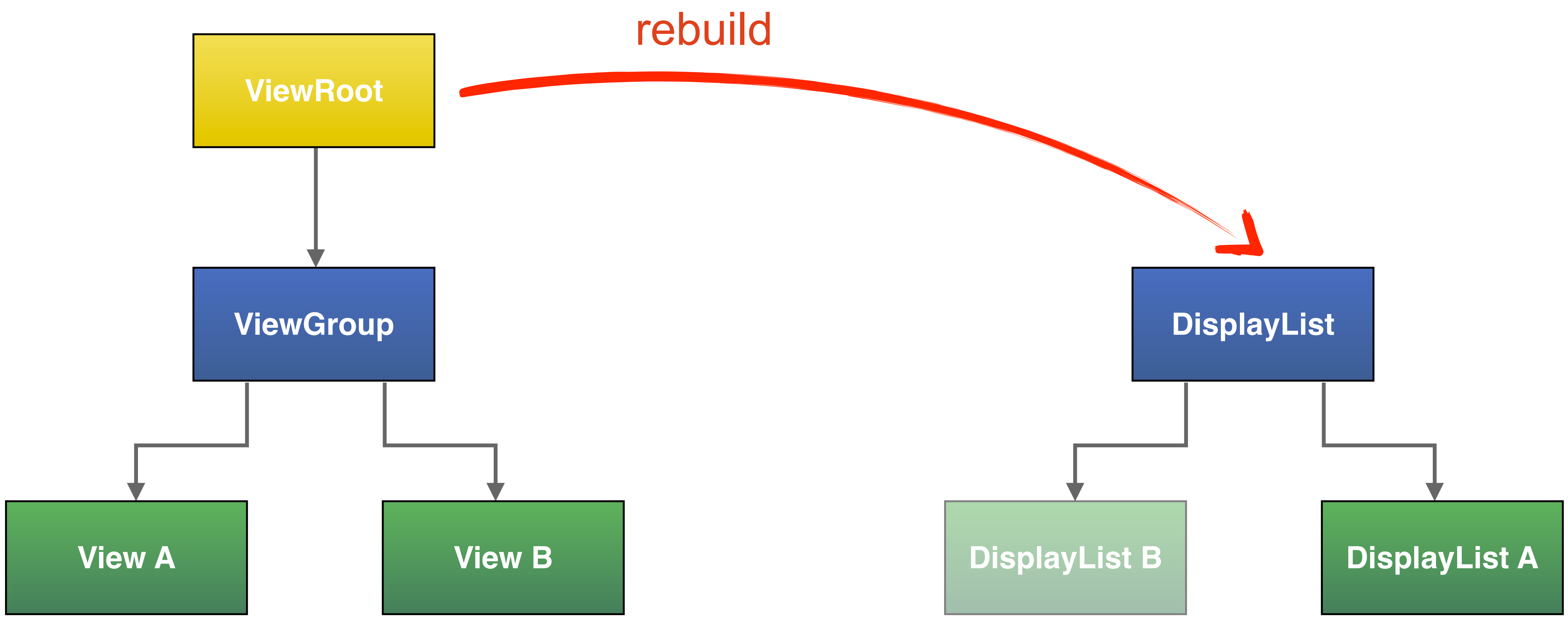
New model


ViewRoot
ViewGroup
invalidate()
View A
View B
invalidate()
mark dirty
DisplayList
DisplayList B
DisplayList A

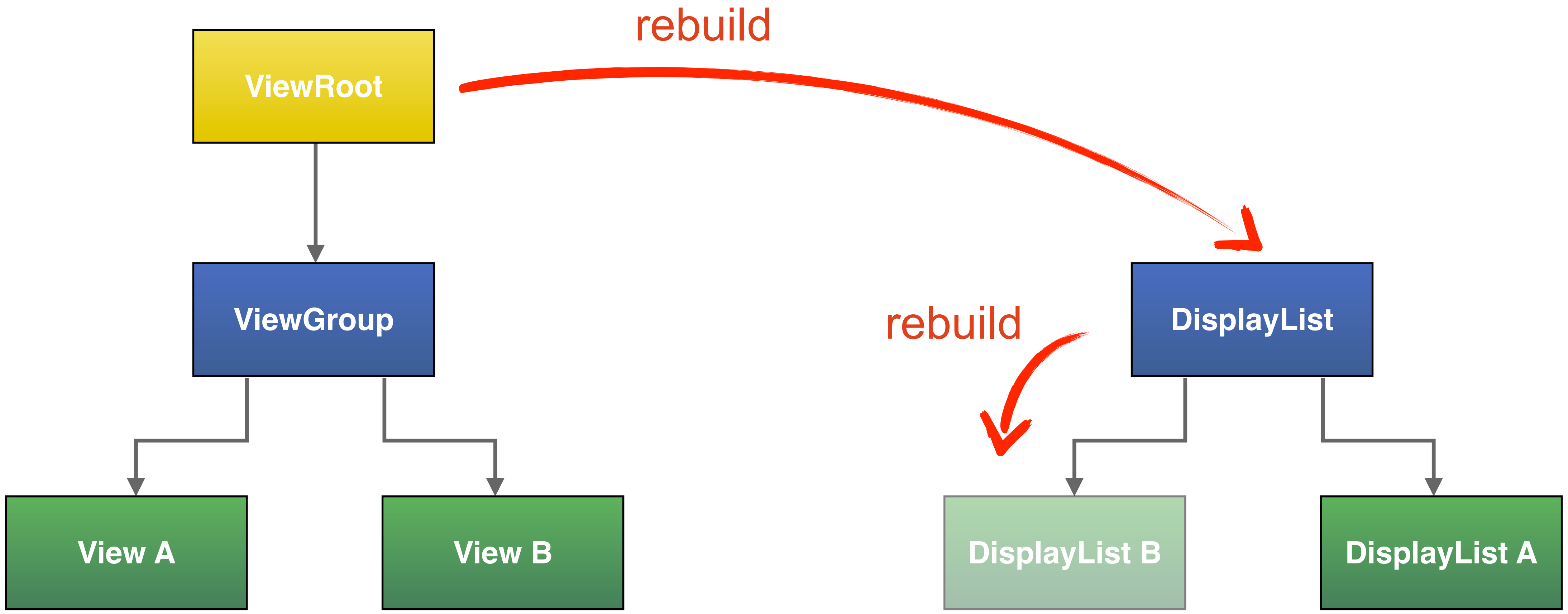
New model


ViewRoot
invalidate()
ViewGroup
invalidate()
View A
View B
invalidate()
mark dirty
DisplayList
DisplayList B
DisplayList A

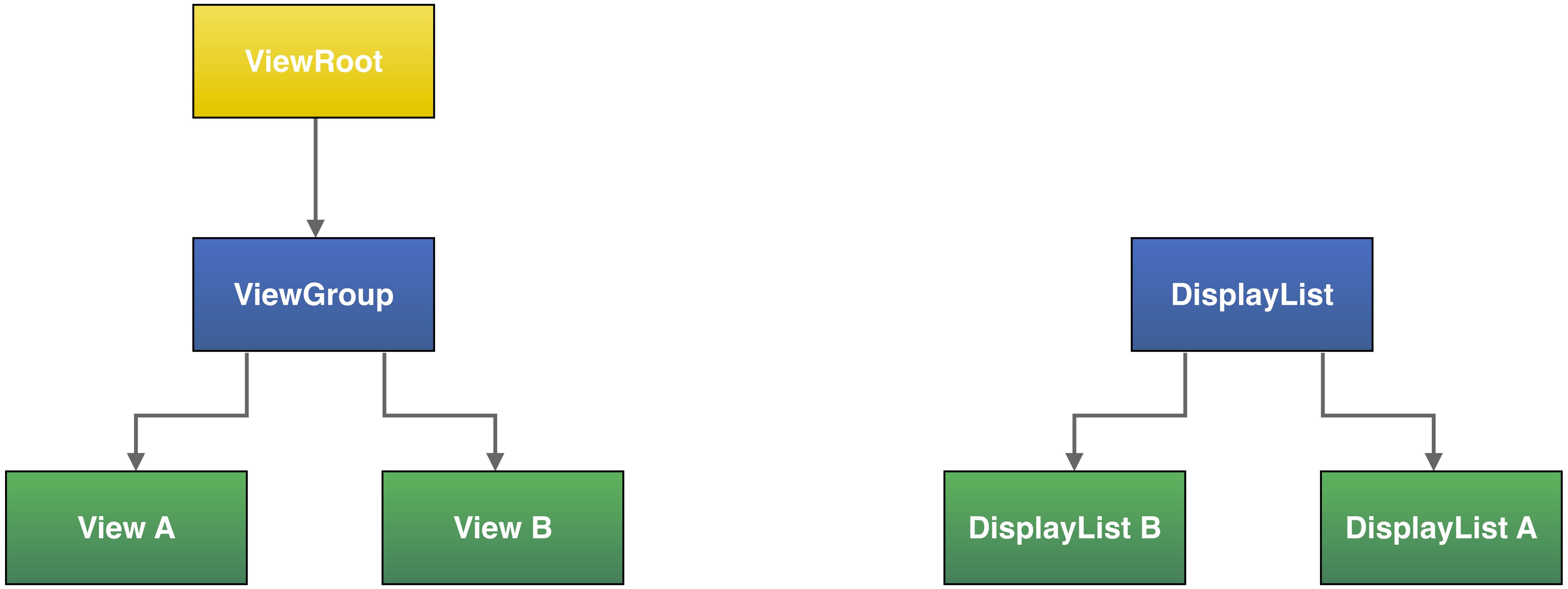
New model


rebuild
ViewRoot
ViewGroup
View A
View B
DisplayList
DisplayList B
DisplayList A

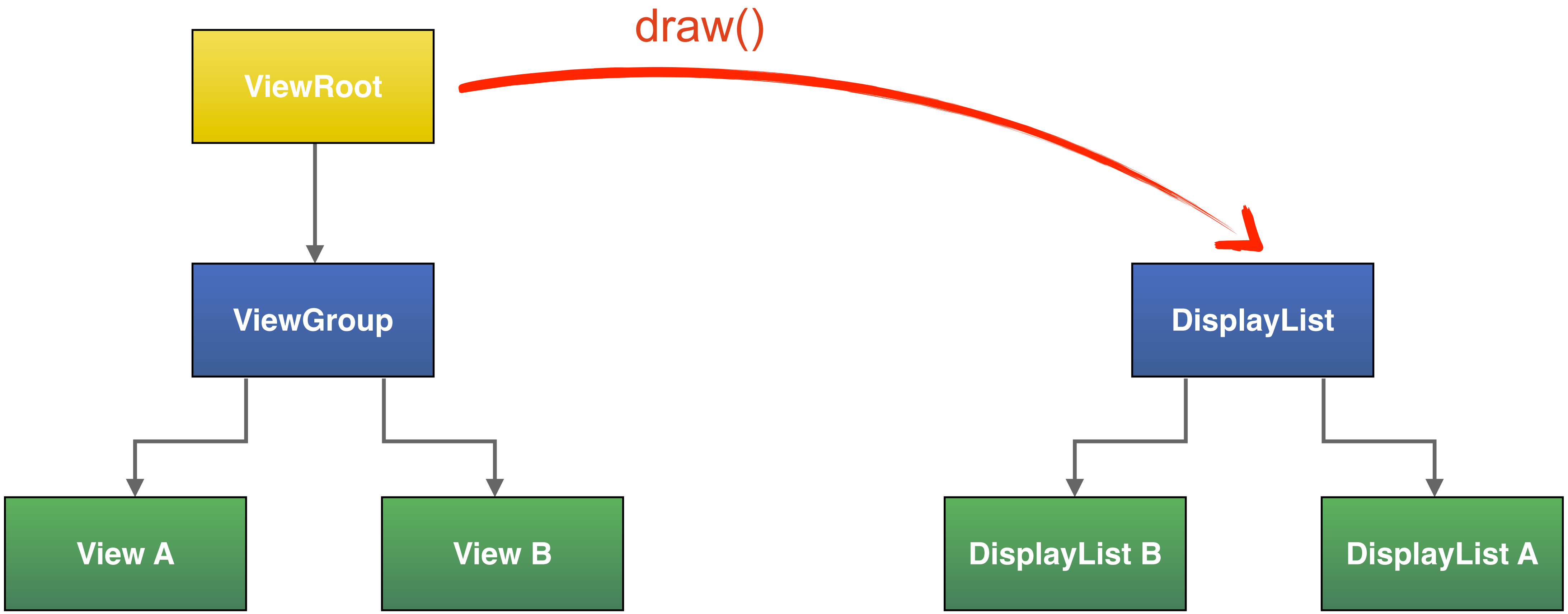
New model


rebuild
ViewRoot
ViewGroup
View A
View B
rebuild
DisplayList
DisplayList B
DisplayList A

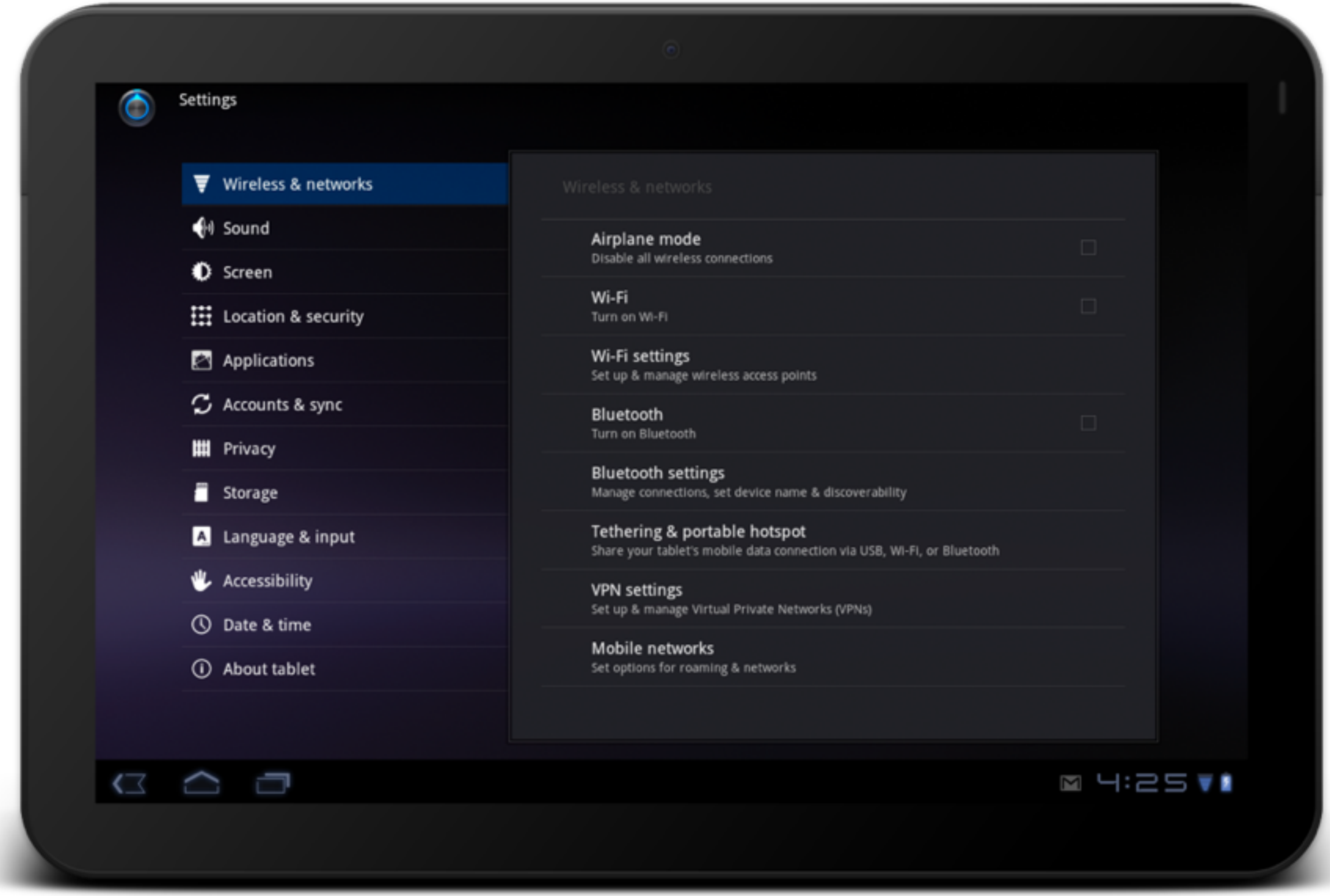
New model


ViewRoot
ViewGroup
View A
View B
DisplayList
DisplayList B
DisplayList A

New model


draw()
ViewRoot
ViewGroup
View A
View B
DisplayList
DisplayList B
DisplayList A

New model



Settings

- Wireless & networks
- Sound
- Screen
- Location & security
- Applications
- Accounts & sync
- Privacy
- Storage
- Language & input
- Accessibility
- Date & time
- About tablet

Wireless & networks

Airplane mode □
Disable all wireless connections

Wi-Fi □
Turn on Wi-Fi

Wi-Fi settings
Set up & manage wireless access points

Bluetooth □
Turn on Bluetooth

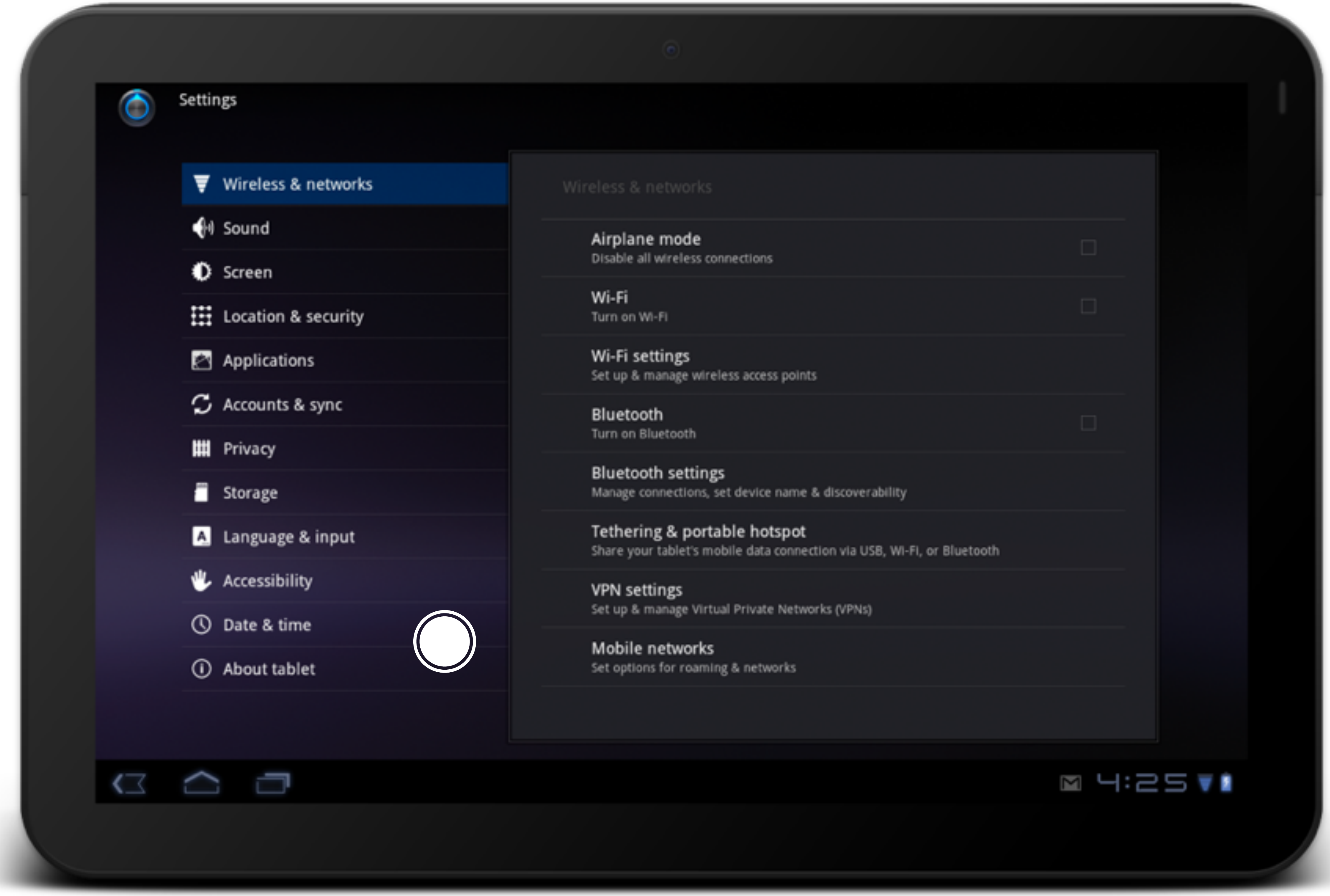
Bluetooth settings
Manage connections, set device name & discoverability

Tethering & portable hotspot
Share your tablet's mobile data connection via USB, Wi-Fi, or Bluetooth

VPN settings
Set up & manage Virtual Private Networks (VPNs)

Mobile networks
Set options for roaming & networks

4:25



Wireless & networks

Airplane mode □
Disable all wireless connections

Wi-Fi □
Turn on Wi-Fi

Wi-Fi settings
Set up & manage wireless access points

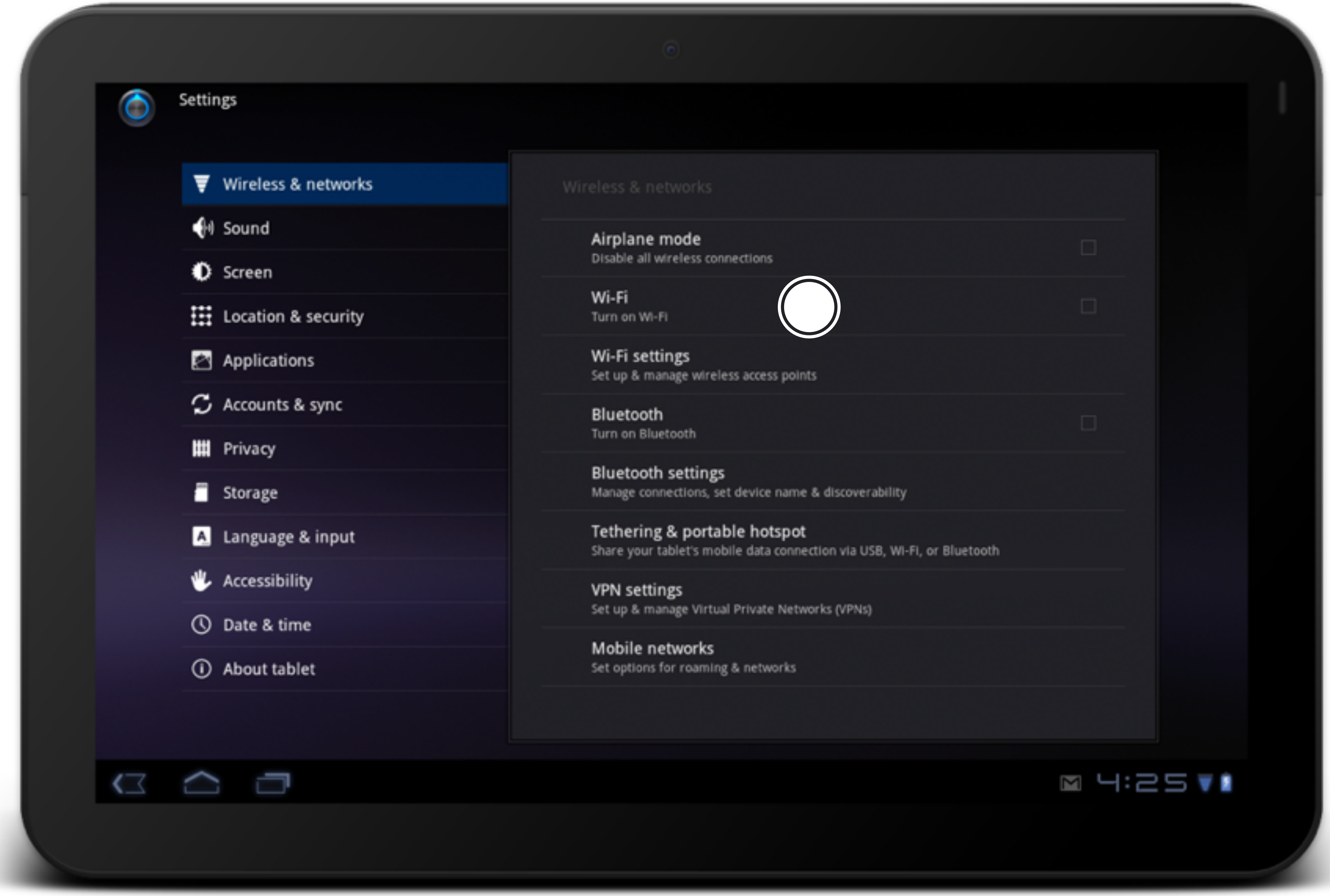
Bluetooth □
Turn on Bluetooth

Bluetooth settings
Manage connections, set device name & discoverability

Tethering & portable hotspot
Share your tablet's mobile data connection via USB, Wi-Fi, or Bluetooth

VPN settings
Set up & manage Virtual Private Networks (VPNs)

Mobile networks
Set options for roaming & networks



Wireless & networks

Airplane mode □
Disable all wireless connections

Wi-Fi □
Turn on Wi-Fi

Wi-Fi settings
Set up & manage wireless access points

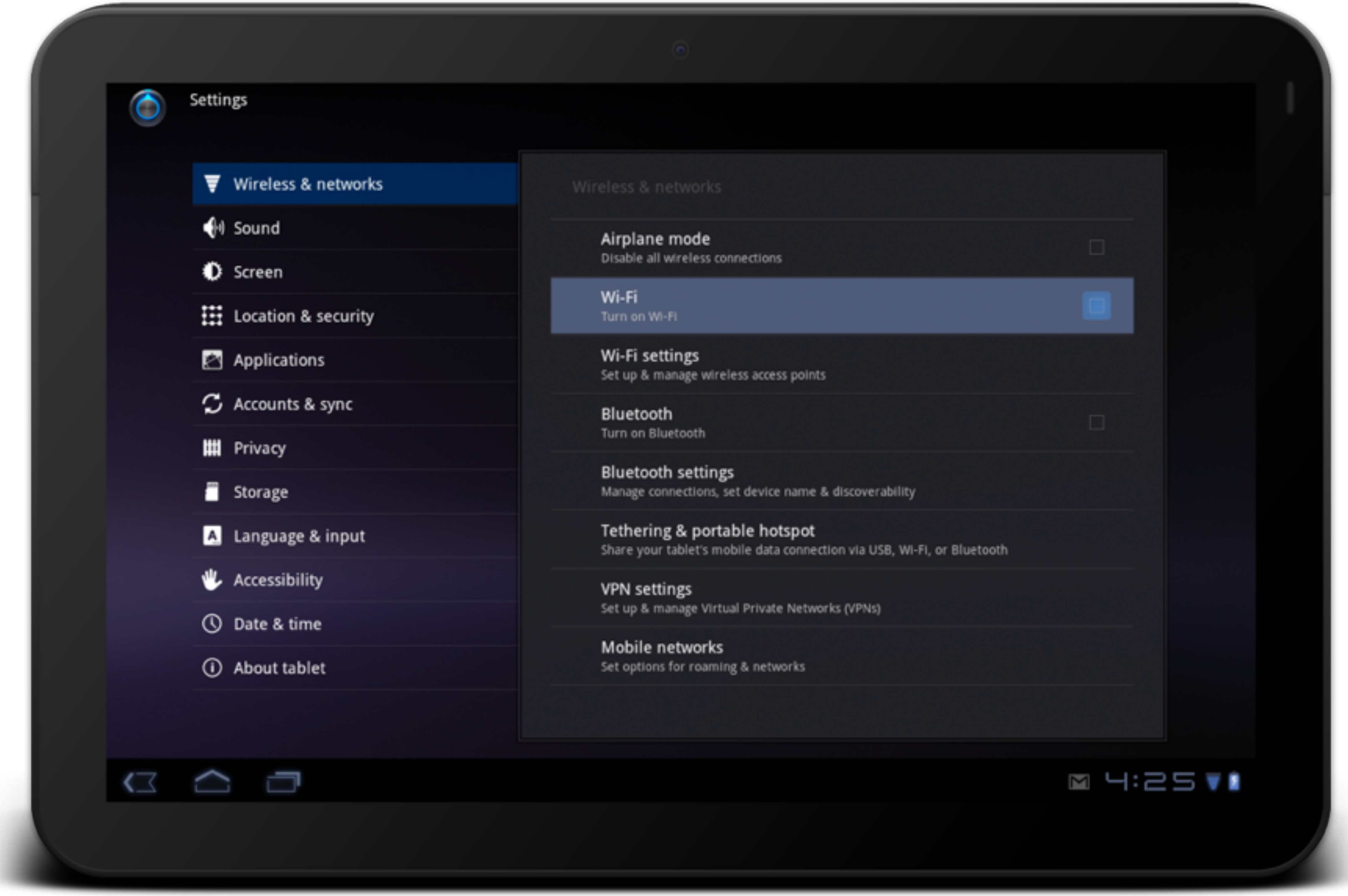
Bluetooth □
Turn on Bluetooth

Bluetooth settings
Manage connections, set device name & discoverability

Tethering & portable hotspot
Share your tablet's mobile data connection via USB, Wi-Fi, or Bluetooth

VPN settings
Set up & manage Virtual Private Networks (VPNs)

Mobile networks
Set options for roaming & networks



Wireless & networks

Airplane mode ☐
Disable all wireless connections

Wi-Fi ☐
Turn on Wi-Fi

Wi-Fi settings
Set up & manage wireless access points

Bluetooth ☐
Turn on Bluetooth

Bluetooth settings
Manage connections, set device name & discoverability

Tethering & portable hotspot
Share your tablet's mobile data connection via USB, Wi-Fi, or Bluetooth

VPN settings
Set up & manage Virtual Private Networks (VPNs)

Mobile networks
Set options for roaming & networks

```
background.draw();
panel.draw();
selector.draw();
wifi.draw();
turnOnWifi.draw();
checkBox.draw();
```

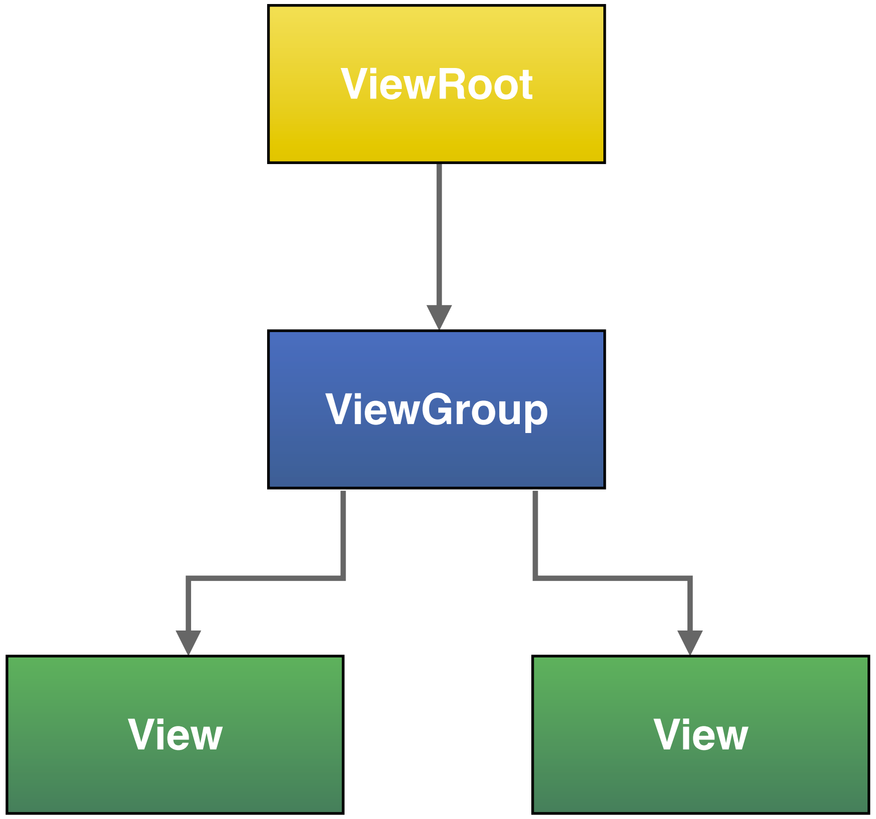
Old model

```
selector.draw();
drawDisplayLists();
```

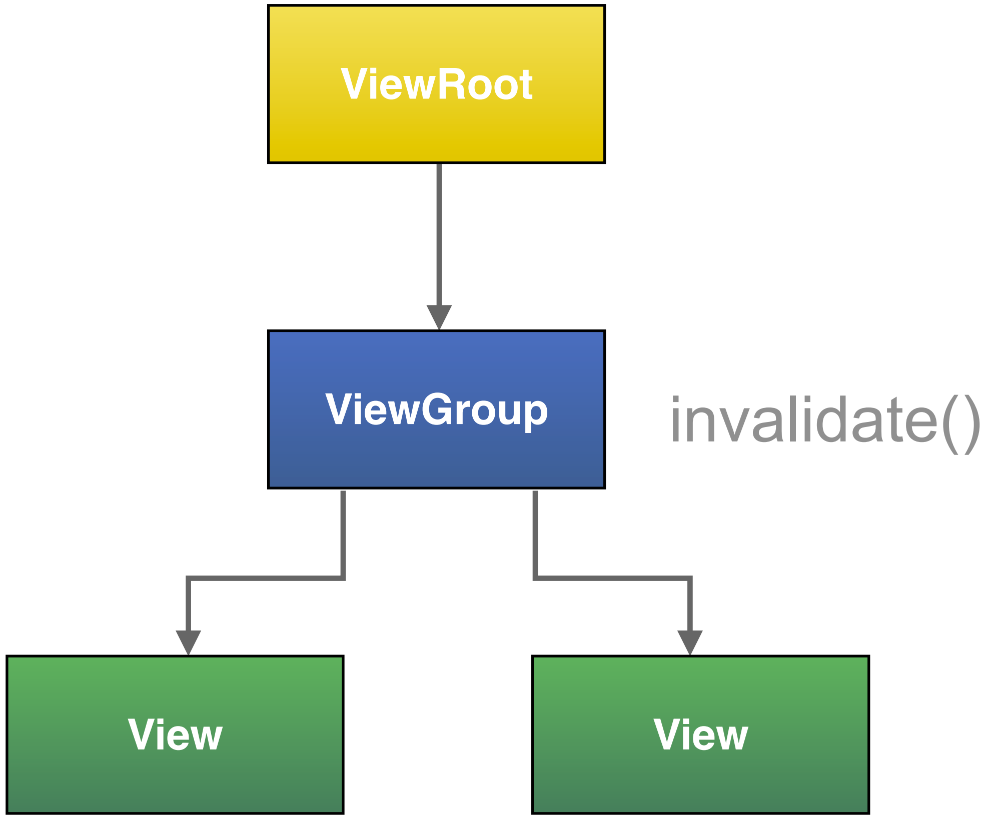
New model

	Old model	New model
View.invalidate()	☑	☑
Intersects a dirty View	☑	□

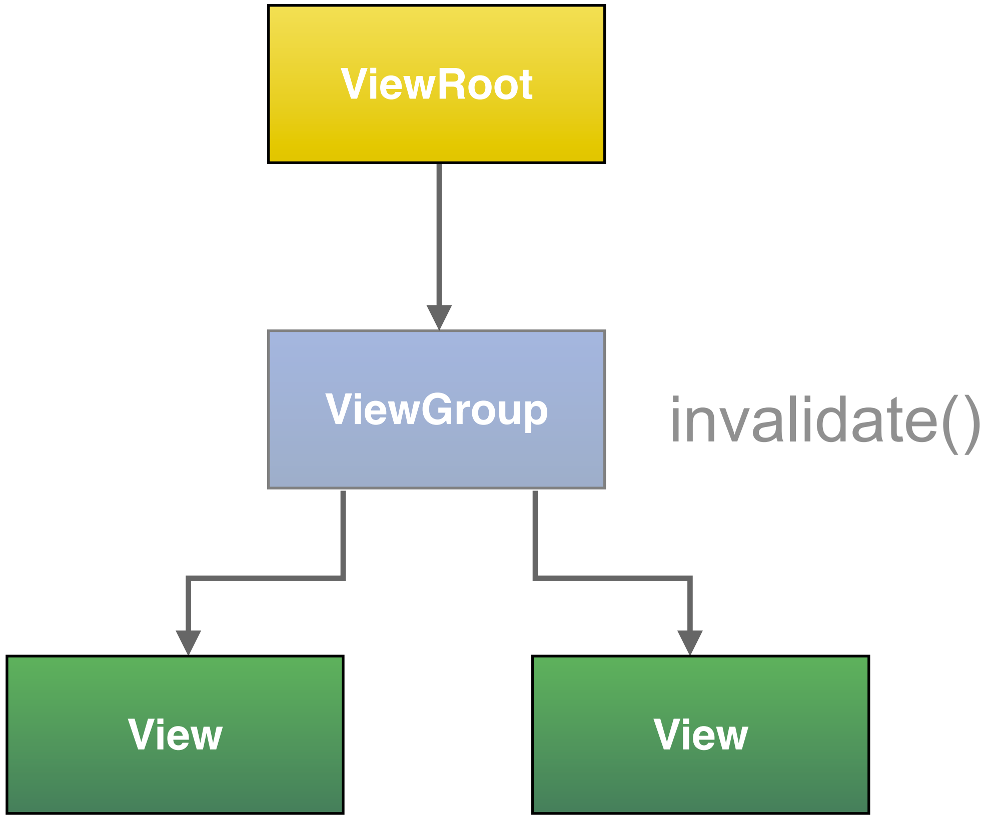
When does View.draw() run?



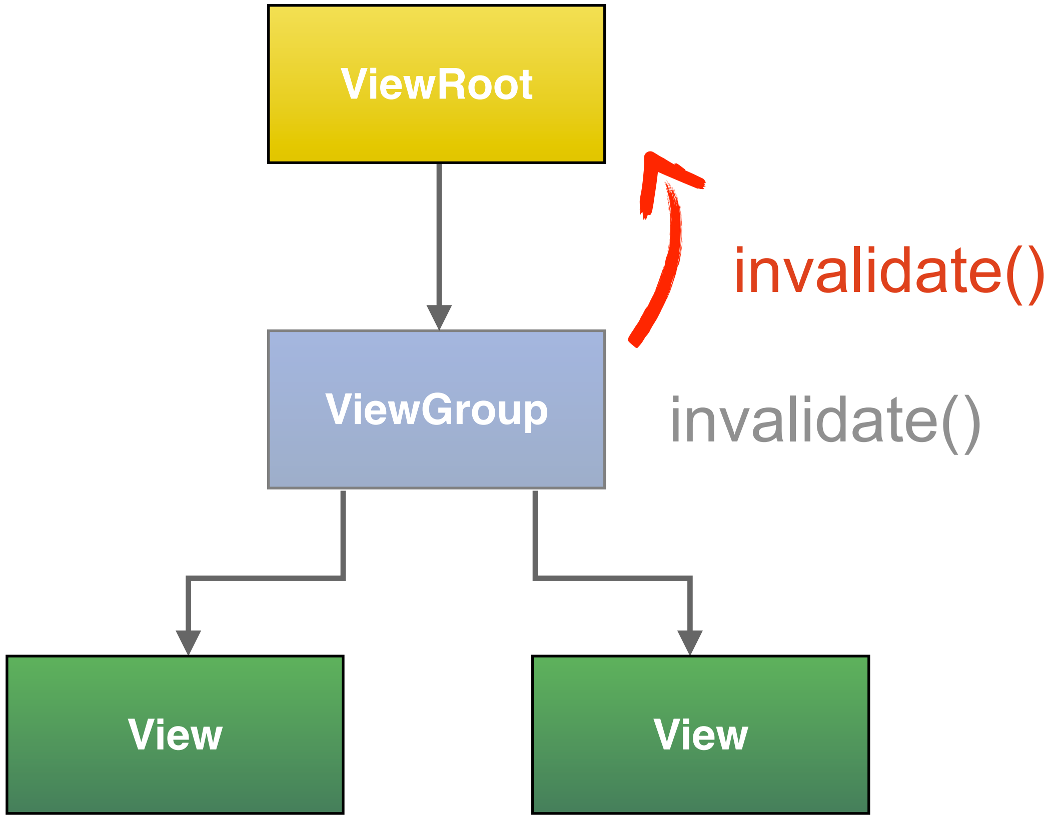
Old model



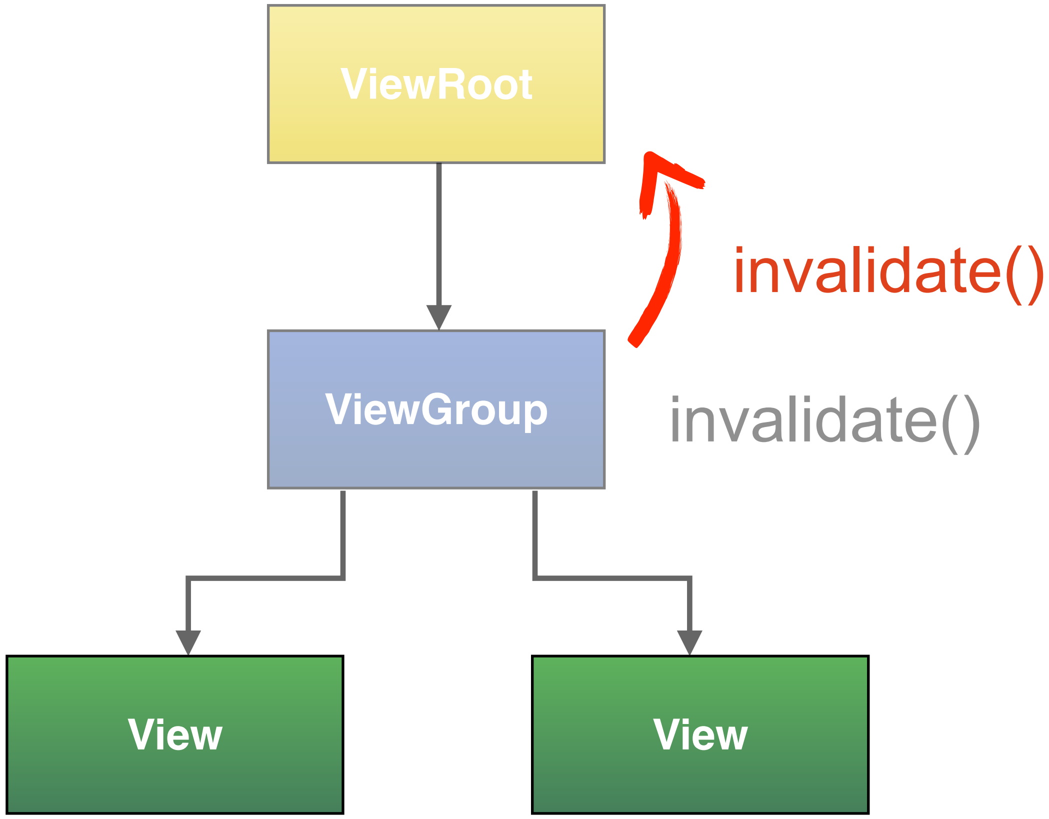
Old model



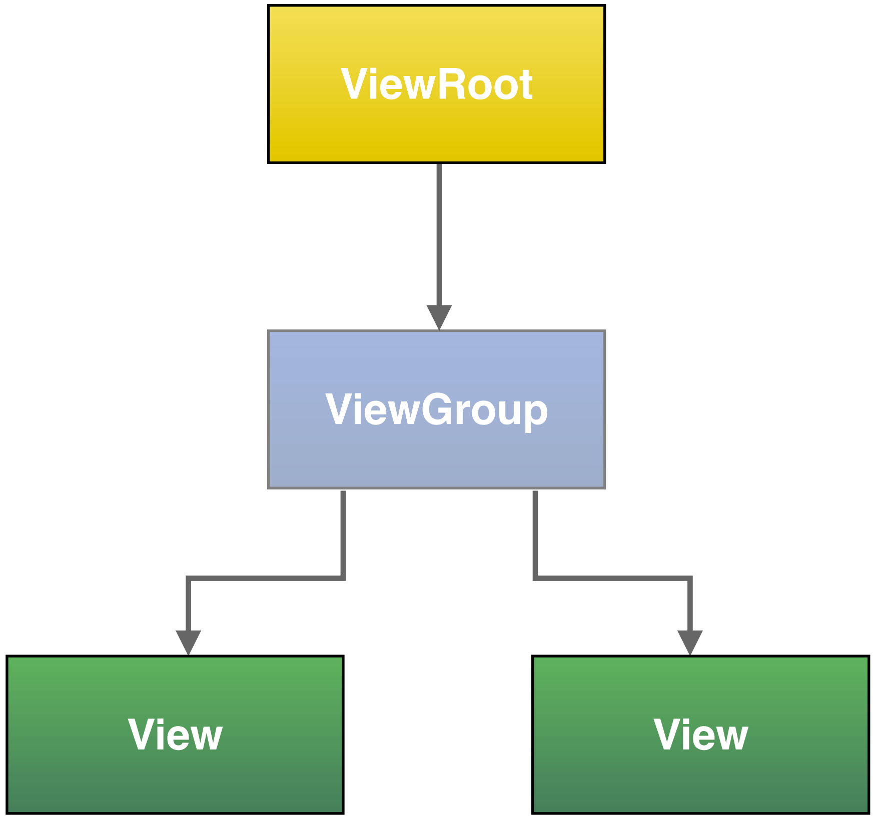
Old model



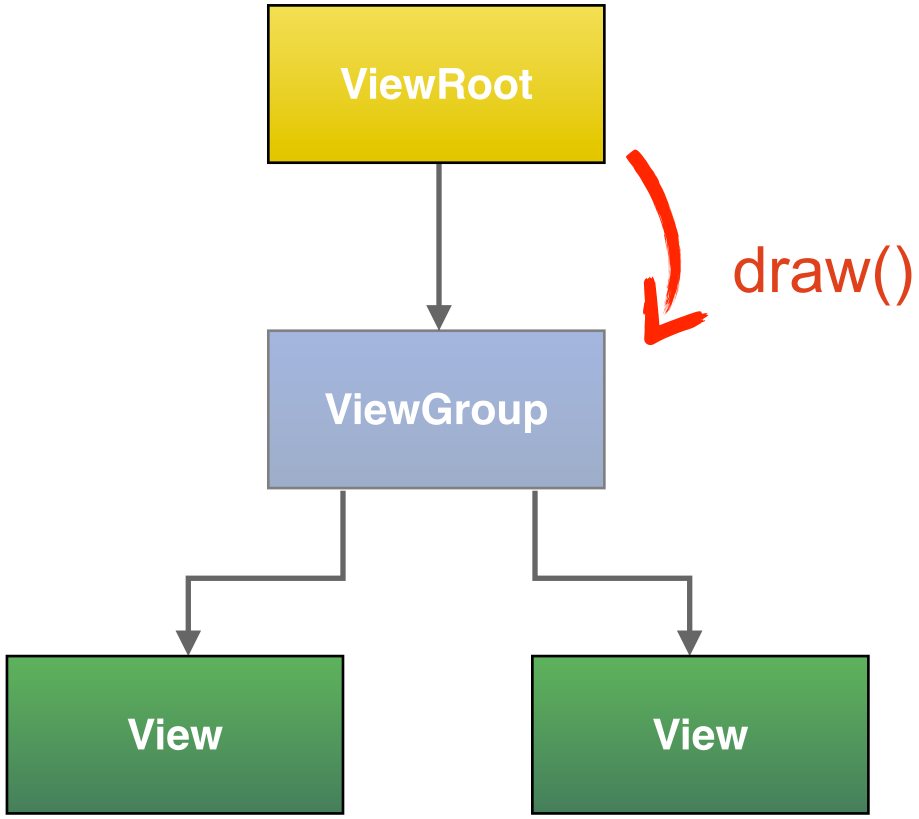
Old model



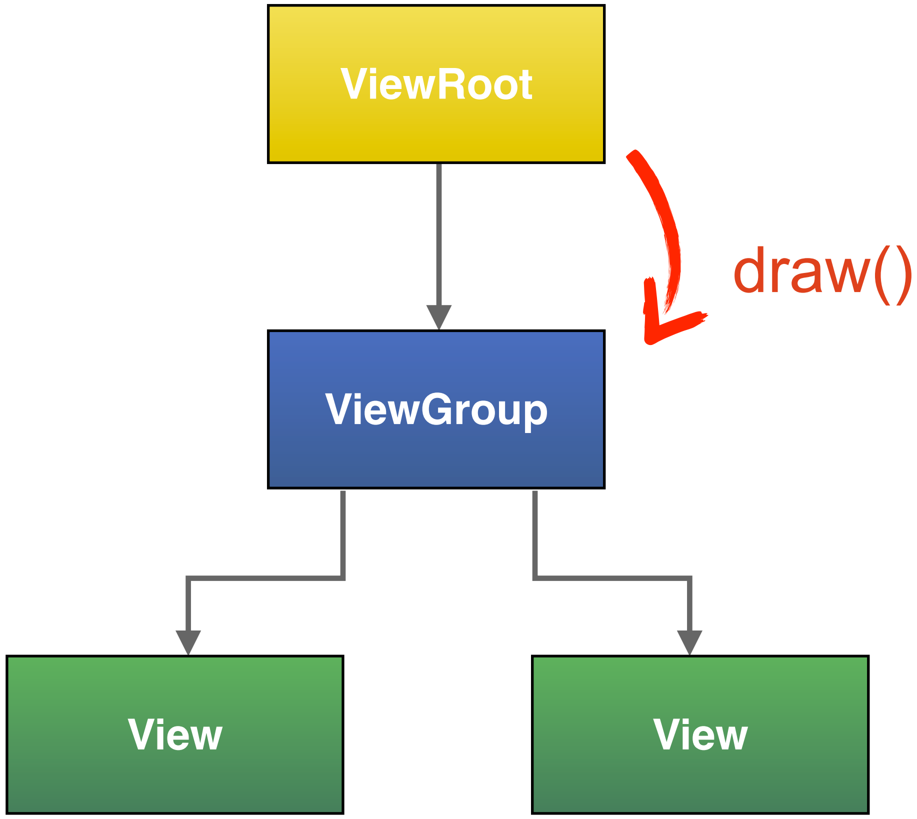
Old model



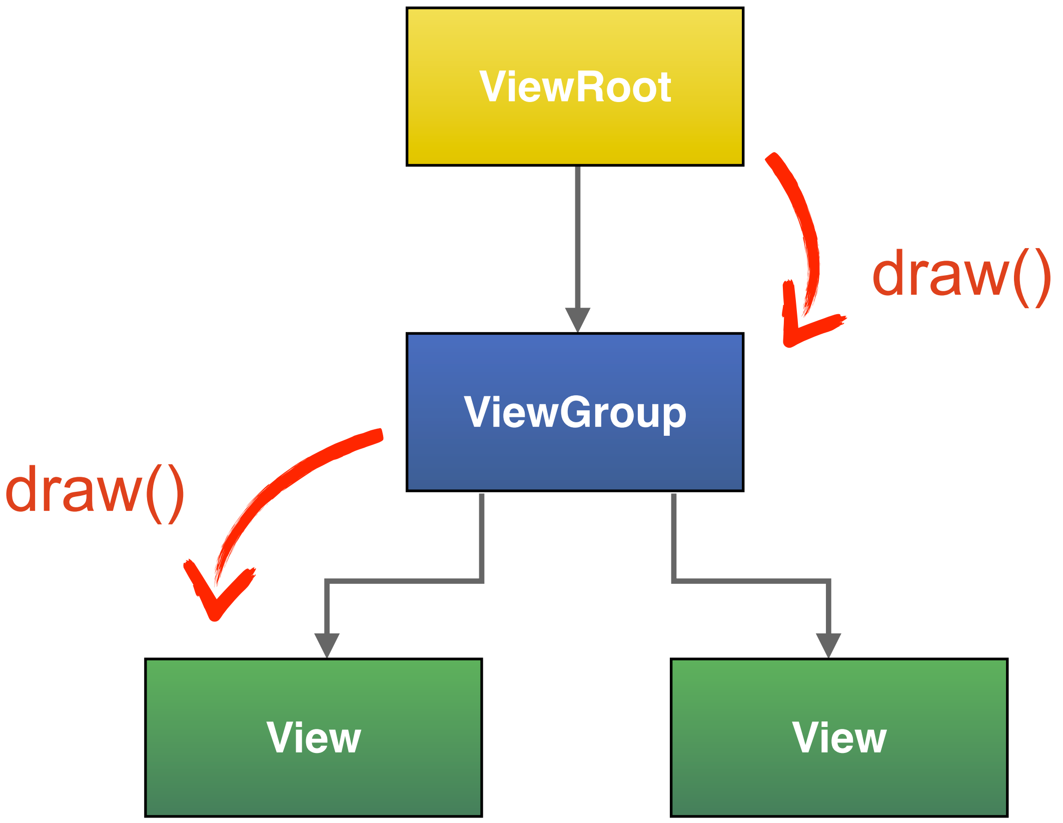
Old model



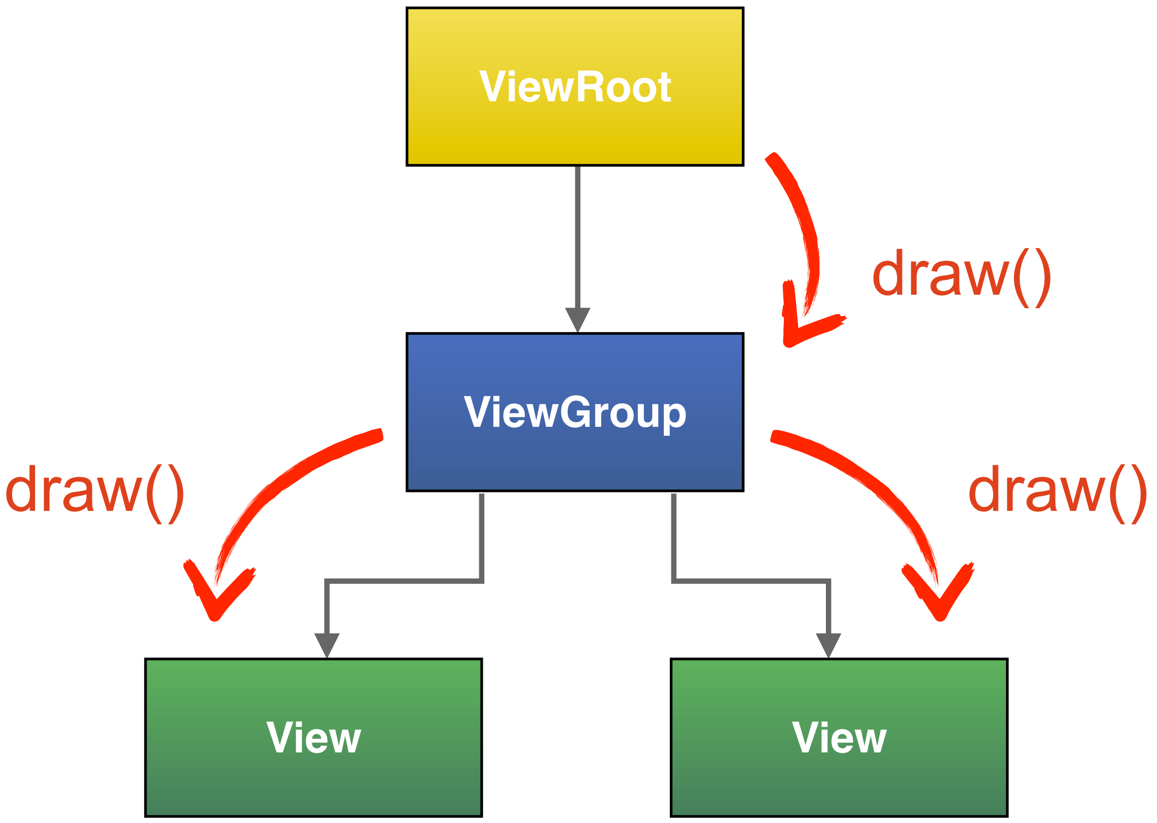
Old model



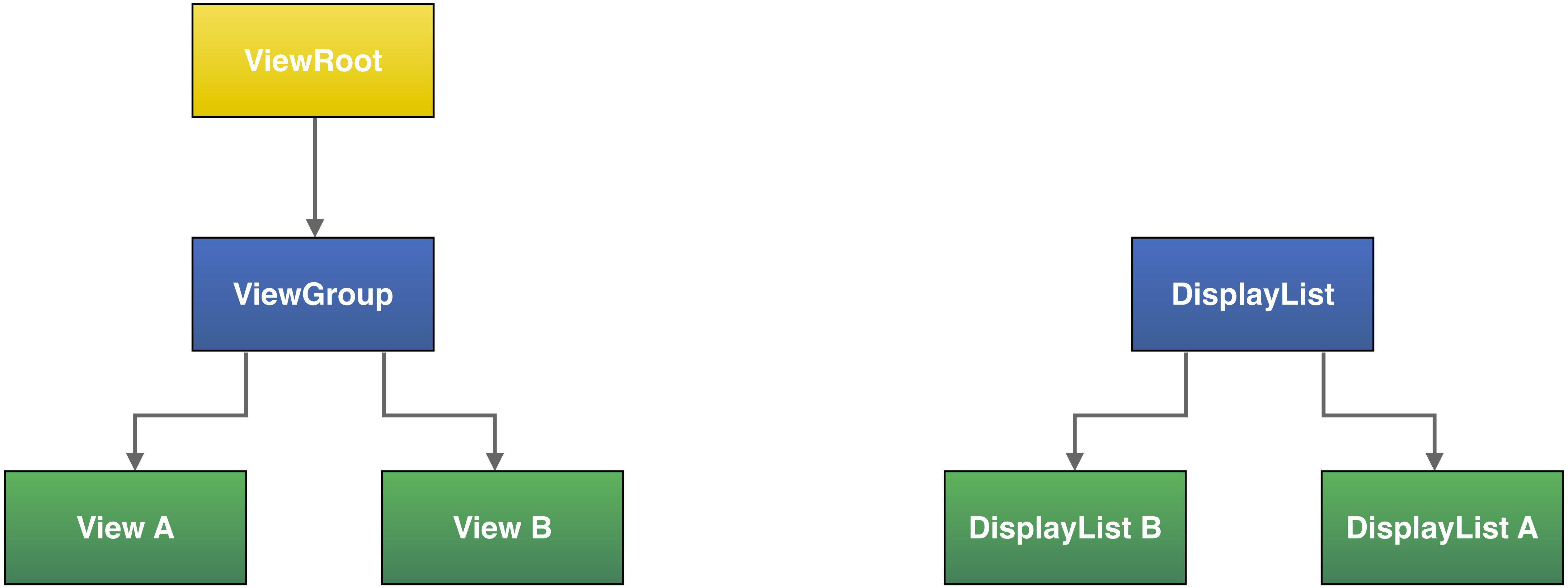
Old model



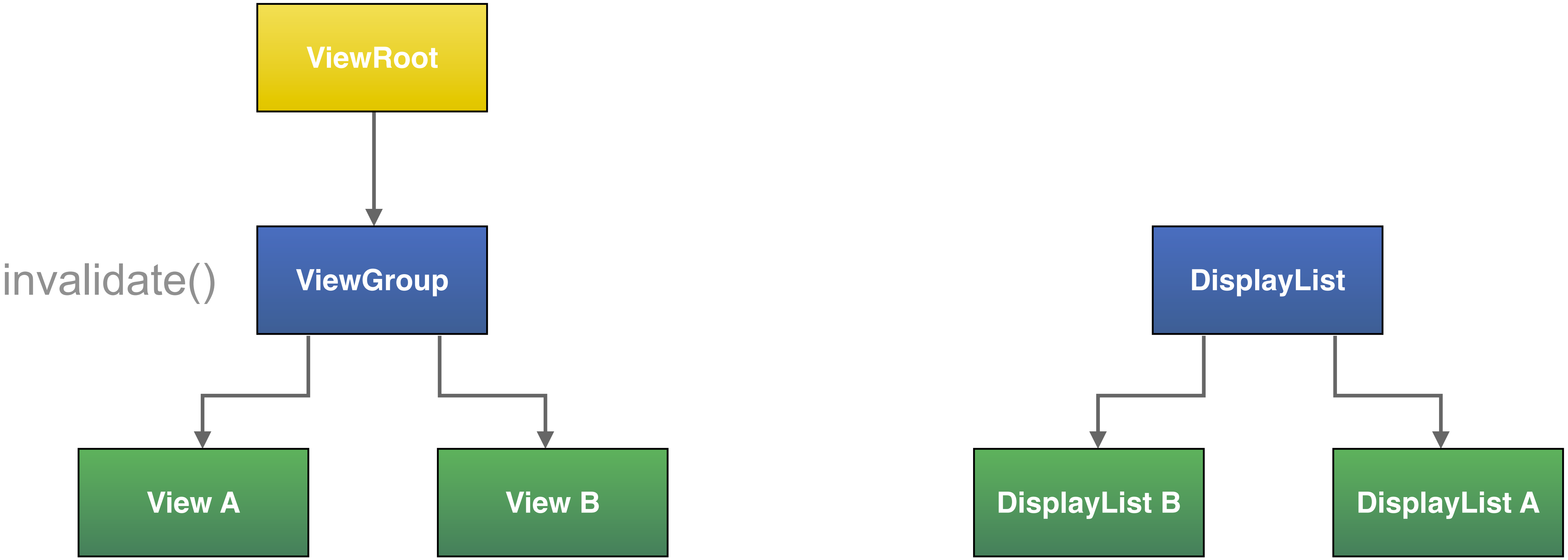
Old model



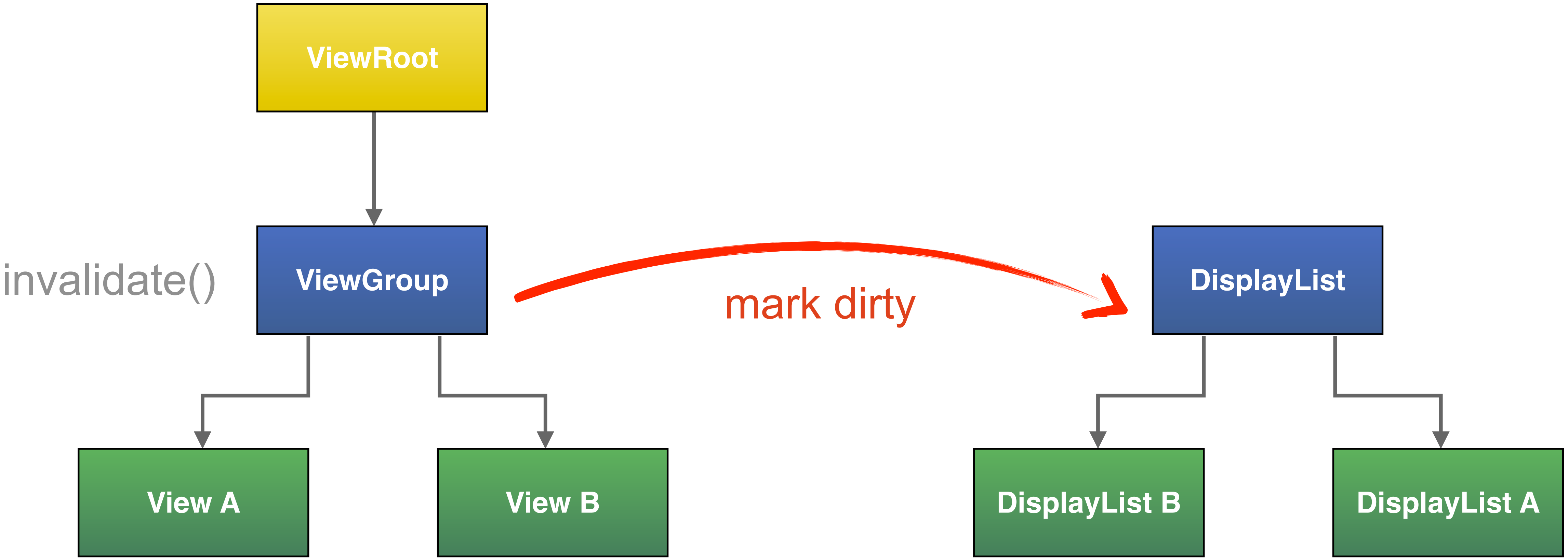
Old model


ViewRoot
ViewGroup
View A
View B
DisplayList
DisplayList B
DisplayList A

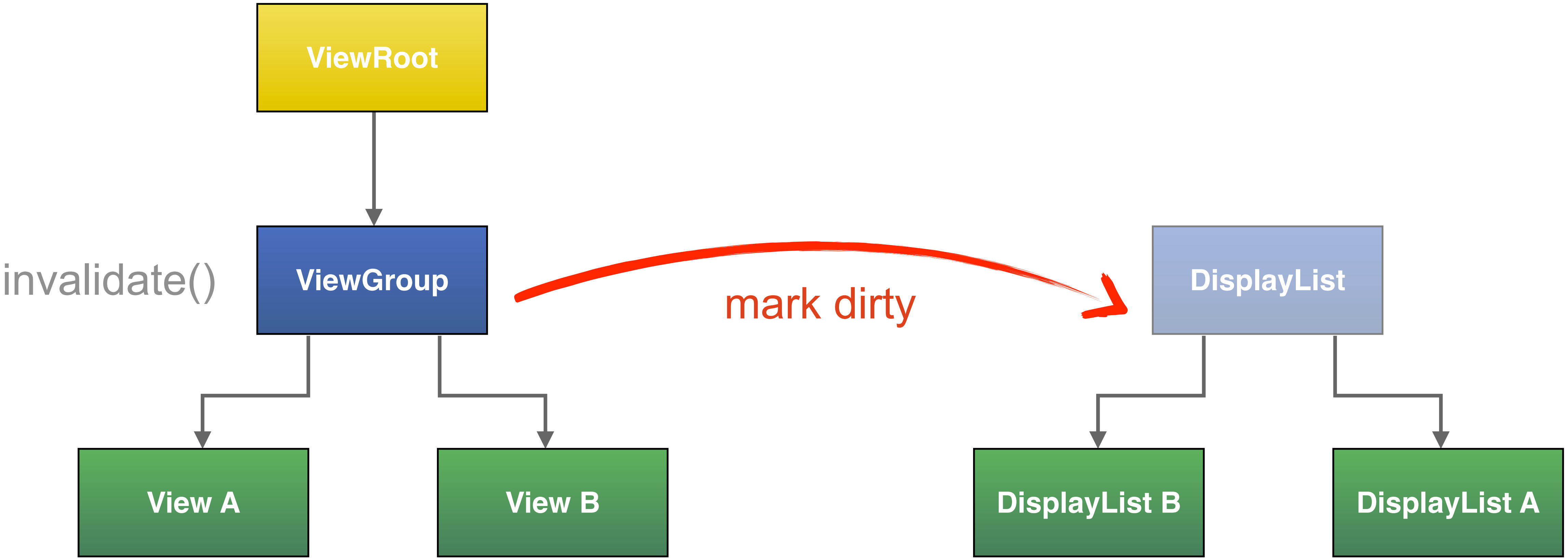
New model


ViewRoot
invalidate()
ViewGroup
View A
View B
DisplayList
DisplayList B
DisplayList A

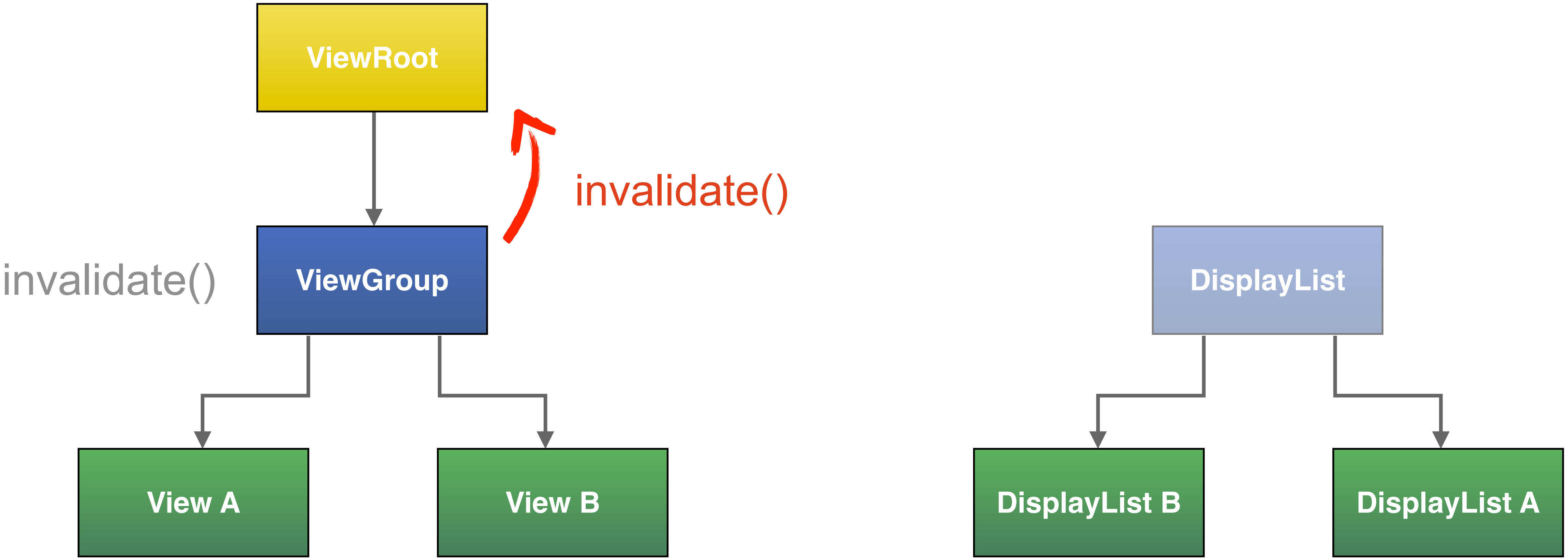
New model


ViewRoot
invalidate()
ViewGroup
mark dirty
DisplayList
View A
View B
DisplayList B
DisplayList A

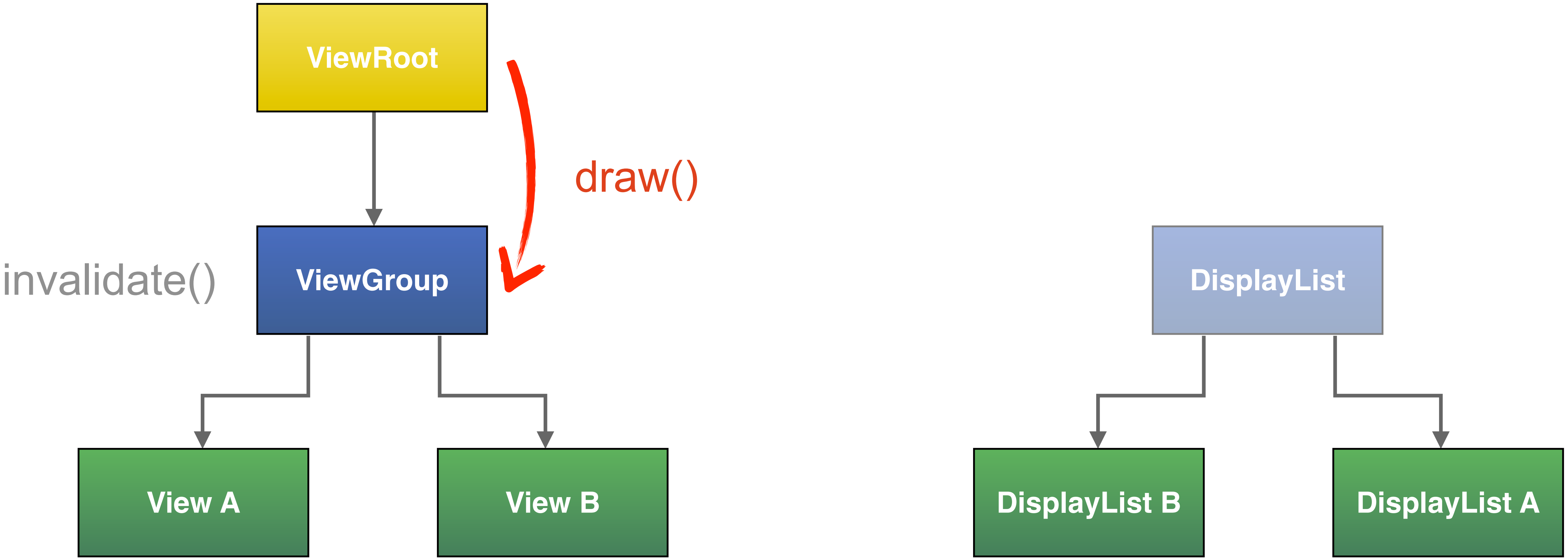
New model


ViewRoot
invalidate()
ViewGroup
mark dirty
DisplayList
View A
View B
DisplayList B
DisplayList A

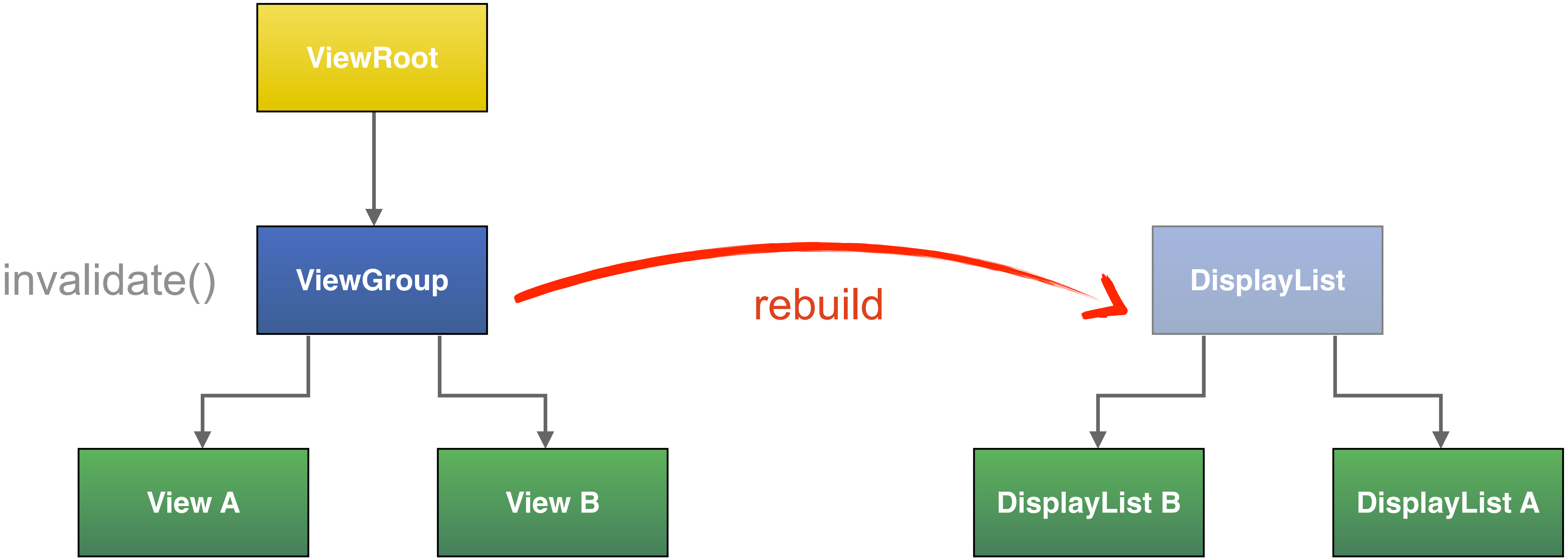
New model


ViewRoot
invalidate()
invalidate()
ViewGroup
View A
View B
DisplayList
DisplayList B
DisplayList A

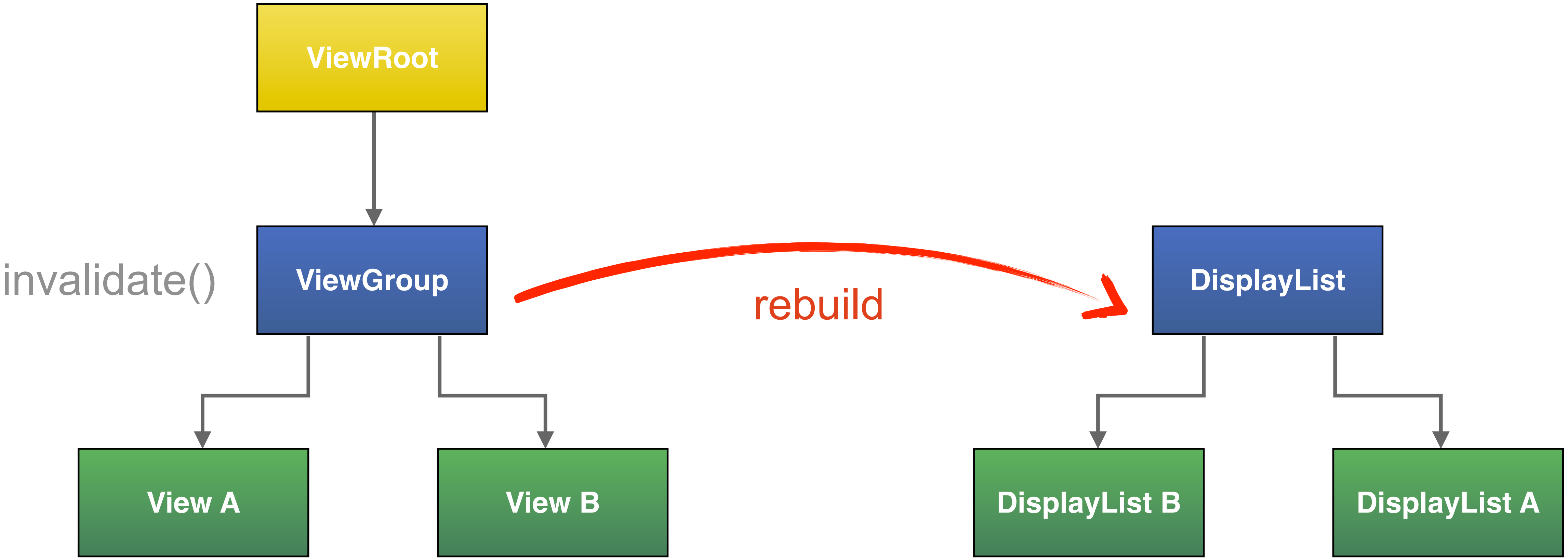
New model


ViewRoot
draw()
invalidate()
ViewGroup
View A
View B
DisplayList
DisplayList B
DisplayList A

New model

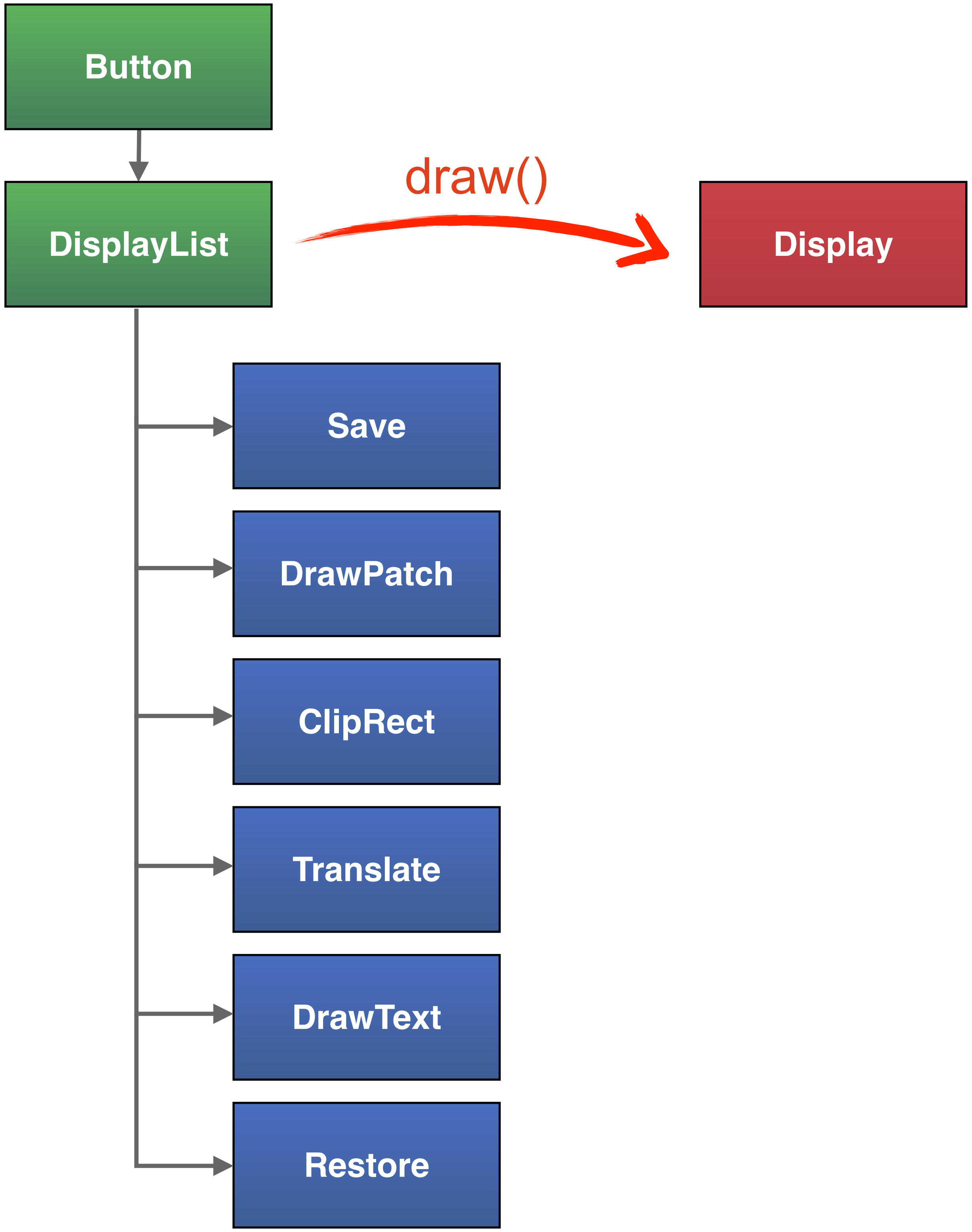

ViewRoot
invalidate()
ViewGroup
rebuild
DisplayList
View A
View B
DisplayList B
DisplayList A

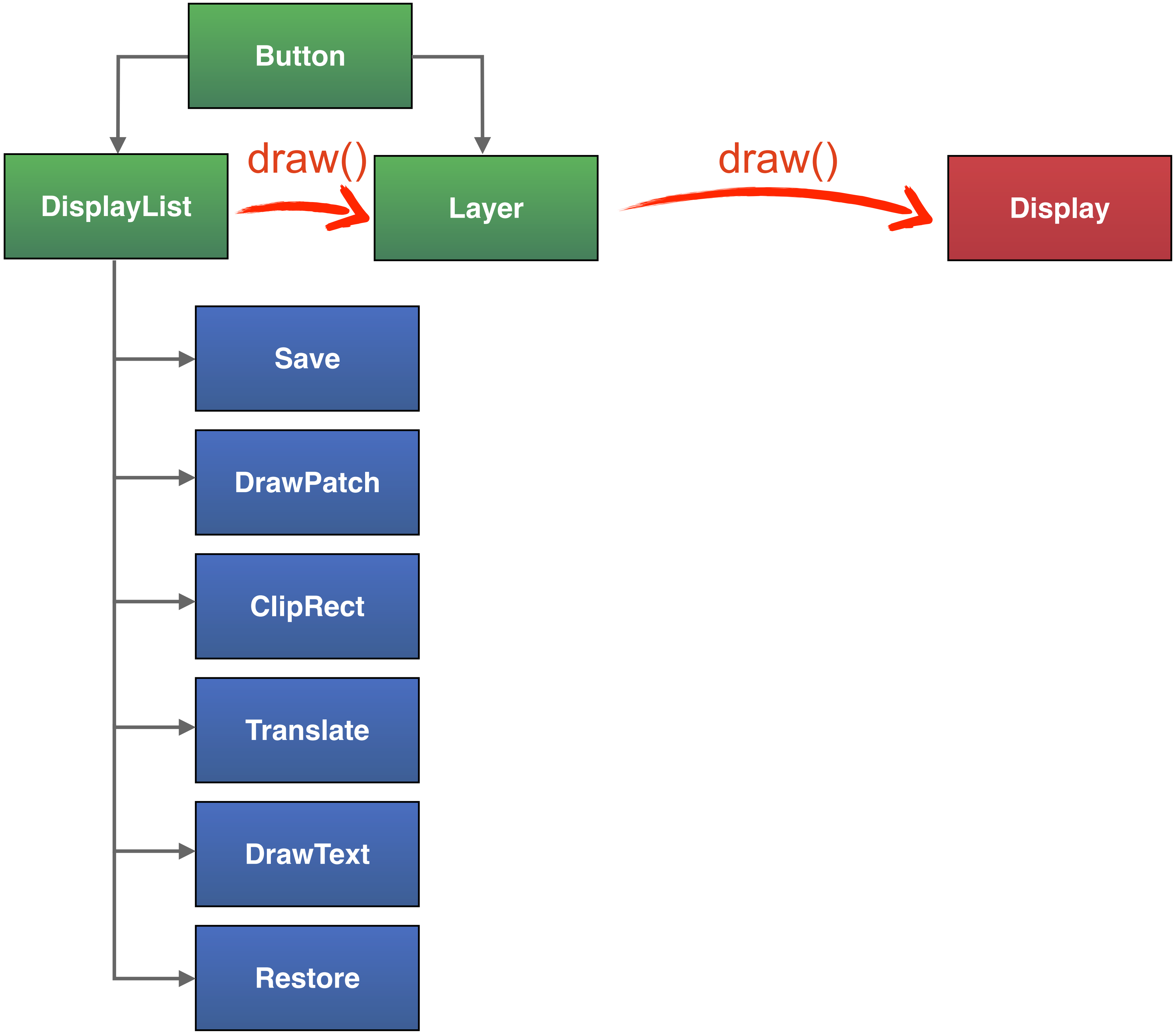
New model


ViewRoot
invalidate()
ViewGroup
rebuild
DisplayList
View A
View B
DisplayList B
DisplayList A

New model

```
View.setLayerType(int type, Paint p)
```


Button
DisplayList
draw()
Display
Save
DrawPatch
ClipRect
Translate
DrawText
Restore


Button
DisplayList
draw()
Layer
draw()
Display
Save
DrawPatch
ClipRect
Translate
DrawText
Restore

```
view.setLayerType(View.LAYER_TYPE_NONE, null)
```

3 types of layers

```
view.setLayerType(View.LAYER_TYPE_SOFTWARE, null)
```

Software layer

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

Hardware layer

1. Performance

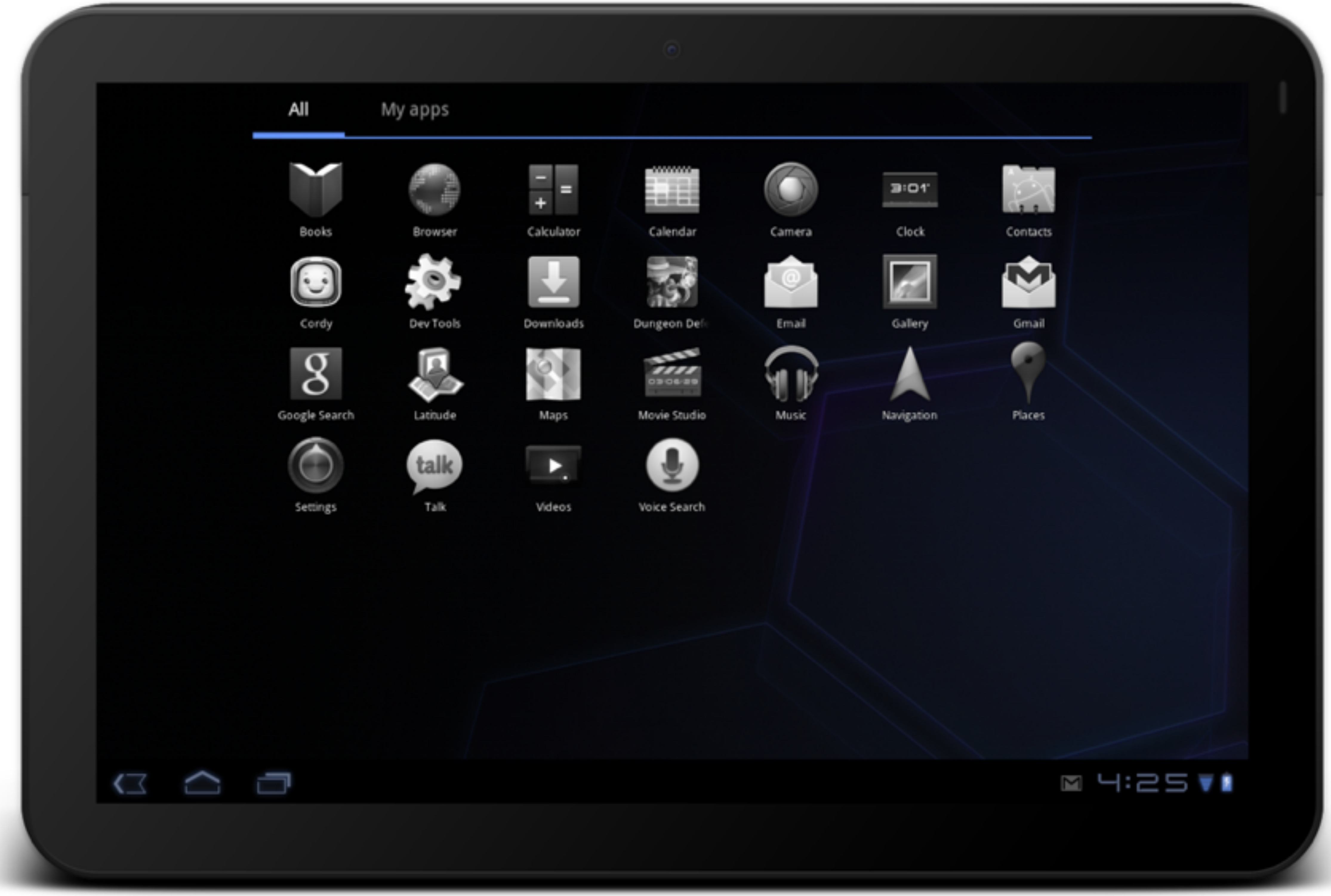
Hardware layers

	Drawing a ListView		
	Hardware layer	DisplayList	Software
Time in ms	0.009	2.1	10.3

Measured when drawing a ListView with Android 3.0 on a Motorola XOOM

2. Visual effects

Hardware and software layers


All
My apps
Books
Browser
Calculator
Calendar
Camera
Clock
Contacts
Cordy
Dev Tools
Downloads
Dungeon Def
Email
Gallery
Gmail
Google Search
Latitude
Maps
Movie Studio
Music
Navigation
Places
Settings
Talk
Videos
Voice Search
4:25

```
ColorMatrix m = new ColorMatrix();
m.setSaturation(0.0f);

Paint p = new Paint();
p.setColorFilter(new ColorMatrixColorFilter(m));

page.setLayerType(LAYER_TYPE_HARDWARE, p);
```

3. Compatibility

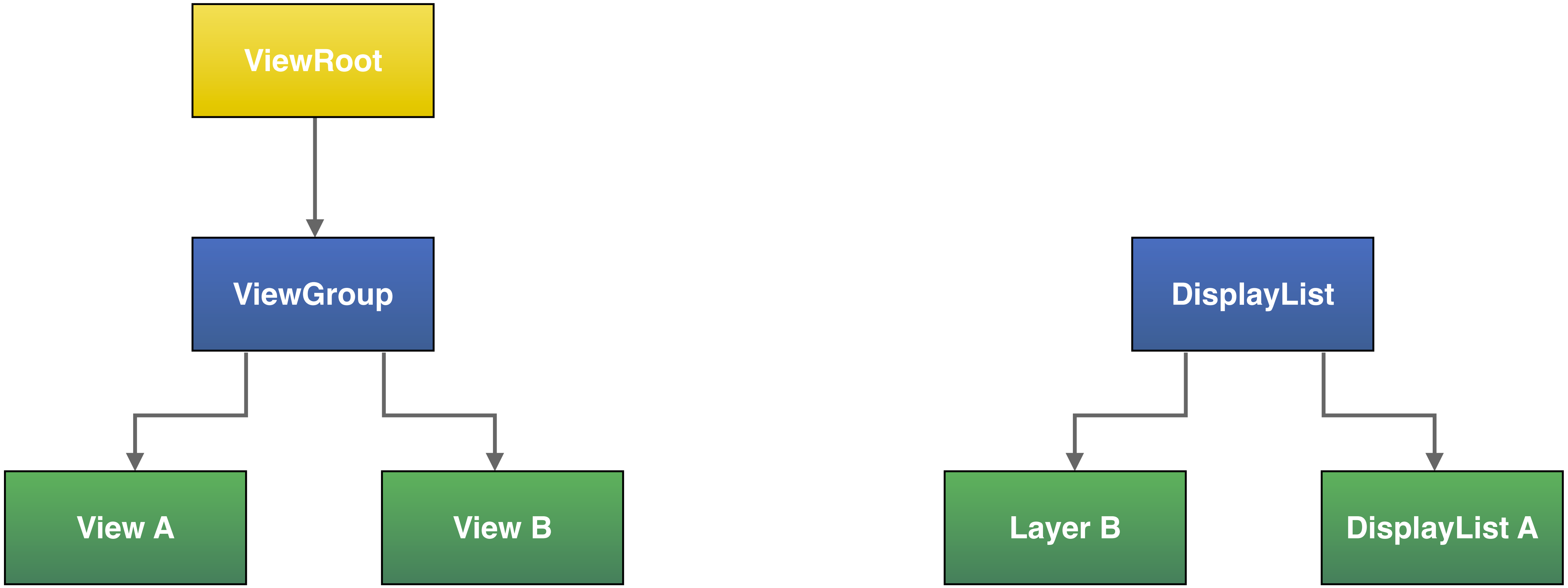
Software layers

4. Animations

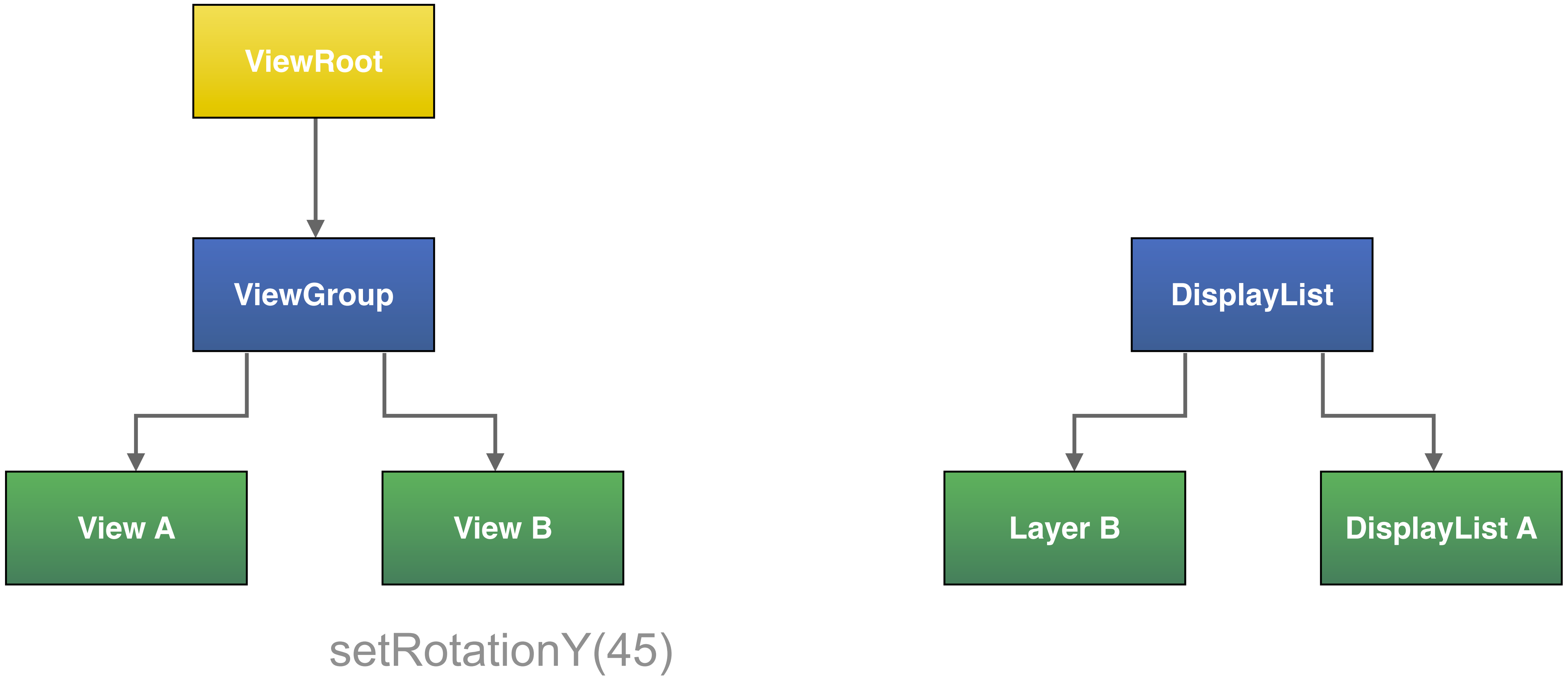
Hardware and software layers

Opacity	Position	Size	Orientation	Origin
alpha	x	scaleX	rotation	pivotX
	y	scaleY	rotationX	pivotY
	translationX		rotationY	
	translationY			

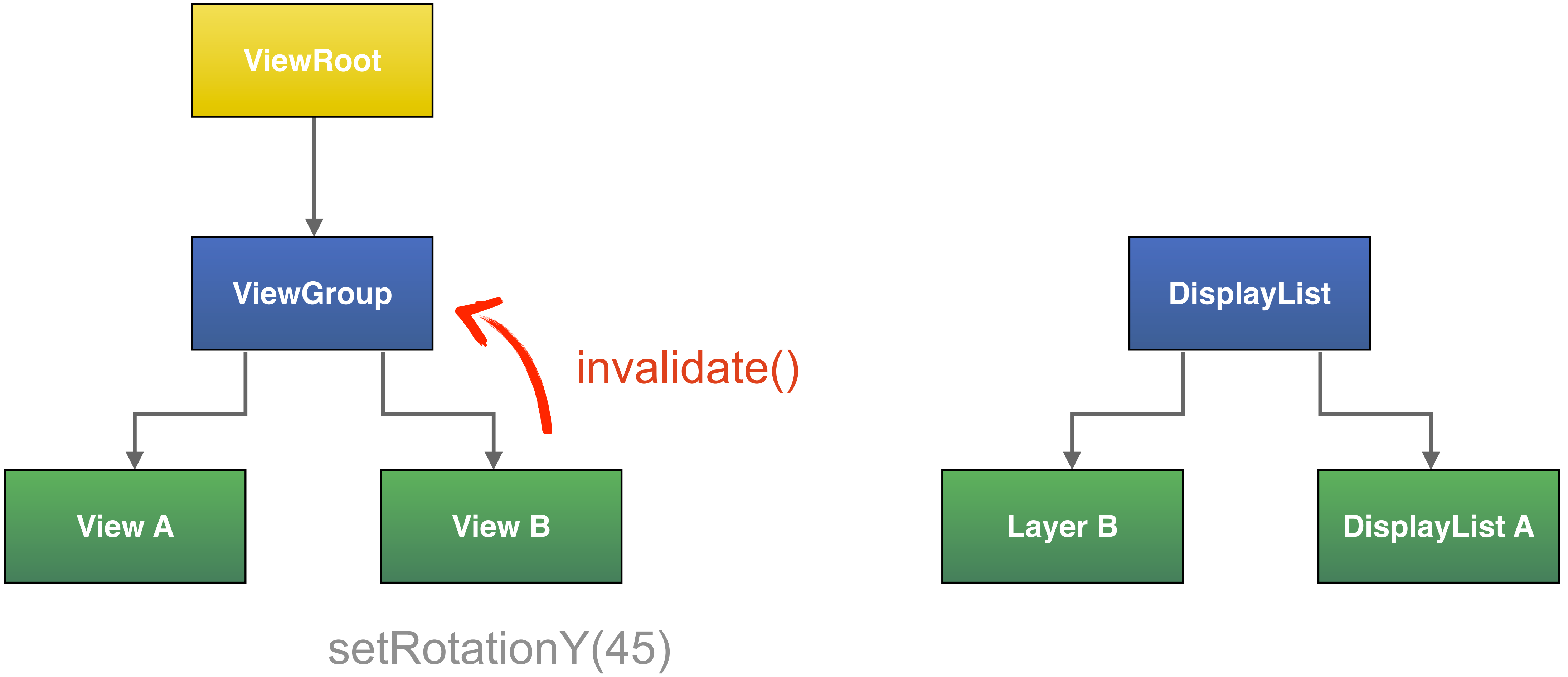
Layers-friendly properties



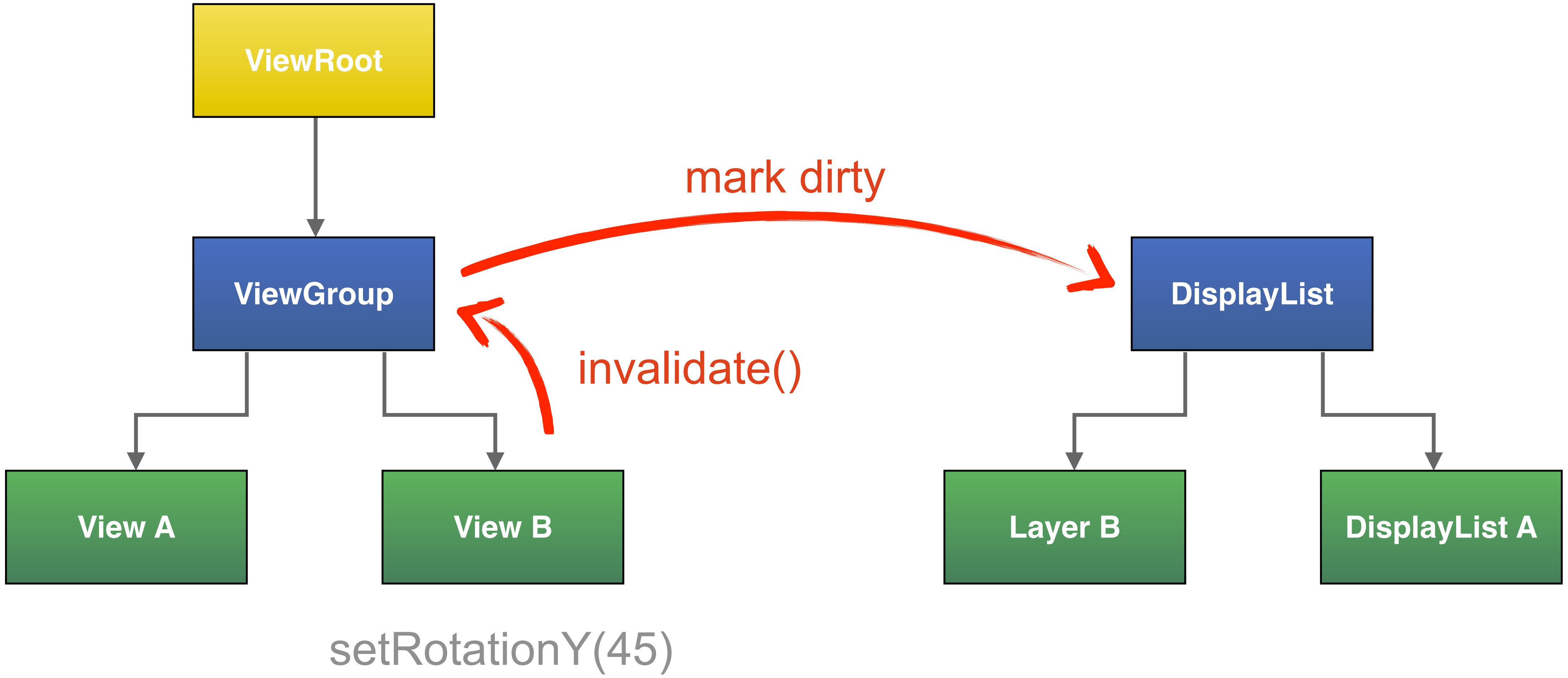
Layers-friendly properties



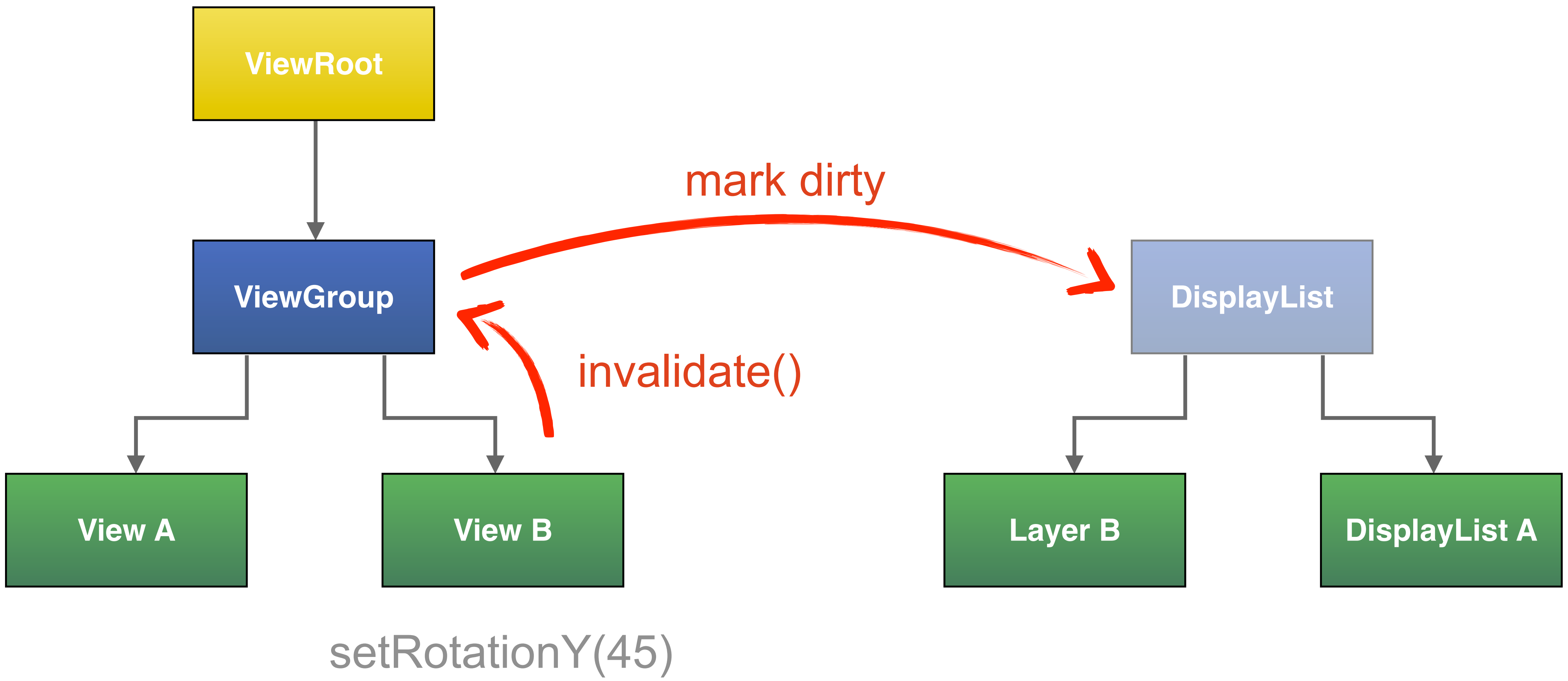
Layers-friendly properties



Layers-friendly properties



Layers-friendly properties



Layers-friendly properties

```
Save 3
DrawDisplayList A
DrawLayer B
Restore
```

Parent DisplayList

viewB.setRotationY(45)

```
Save 3
DrawDisplayList A
DrawLayer B
Restore
```

Parent DisplayList

viewB.setRotationY(45)

```
Save 3
DrawDisplayList A
DrawLayer B
Restore
```

```
Save 3
DrawDisplayList A
Rotate 0.0, 45.0, 0.0
DrawLayer B
Restore
```

Parent DisplayList

```
view.setLayerType(View.LAYER_TYPE_HARDWARE, null);
ObjectAnimator.ofFloat(view, "rotationY", 180).start();
```

Animating a complex View efficiently

```
view.setLayerType(View.LAYER_TYPE_HARDWARE, null);
ObjectAnimator animator = ObjectAnimator.ofFloat(
    view, "rotationY", 180);

animator.addListener(new AnimatorListenerAdapter() {
    @Override
    public void onAnimationEnd(Animator animation) {
        view.setLayerType(View.LAYER_TYPE_NONE, null);
    }
});

animator.start();
```

Tips & tricks

1. Don’t use too many views

Keep your hierarchy flat

2. Be careful of setAlpha()

Without hardware layers, it costs 2x the fill-rate

3. Reuse rendering objects

Don’t create new Paints, Bitmaps, etc. in draw()

4. Don’t modify Bitmaps often

Every change causes a texture upload

5. Don’t modify Paths often

Every change causes a new rasterization

6. Avoid overdraw

GPUs are fill-rate limited

7. Profile

DDMS and traceview are your friends

Android blog	d.android.com
Romain’s blog	curious-creature.org
Chet’s blog	graphics-geek.blogspot.com

Feedback http://goo.gl/wI57L
Hashtags #io2011, #Android

Q & A

Feedback http://goo.gl/wI57L
Hashtags #io2011, #Android

Google™ 11 IO