



UI Guide

Giftbot

Original iPhone
320 x 480 points

iPhone 4
320 x 480 points

iPhone 5
320 x 568 points

iPhone 8
375 x 667 points

iPhone 8 Plus
414 x 736 points

iPhone X
375 x 812 points

@1x



320 x 480 pixels

@2x



640 x 960 pixels

@2x



640 x 1136 pixels

@2x



750 x 1334 pixels

@3x



Downsampled (87%)
1080 x 1920 pixels

1242 x 2208 pixels

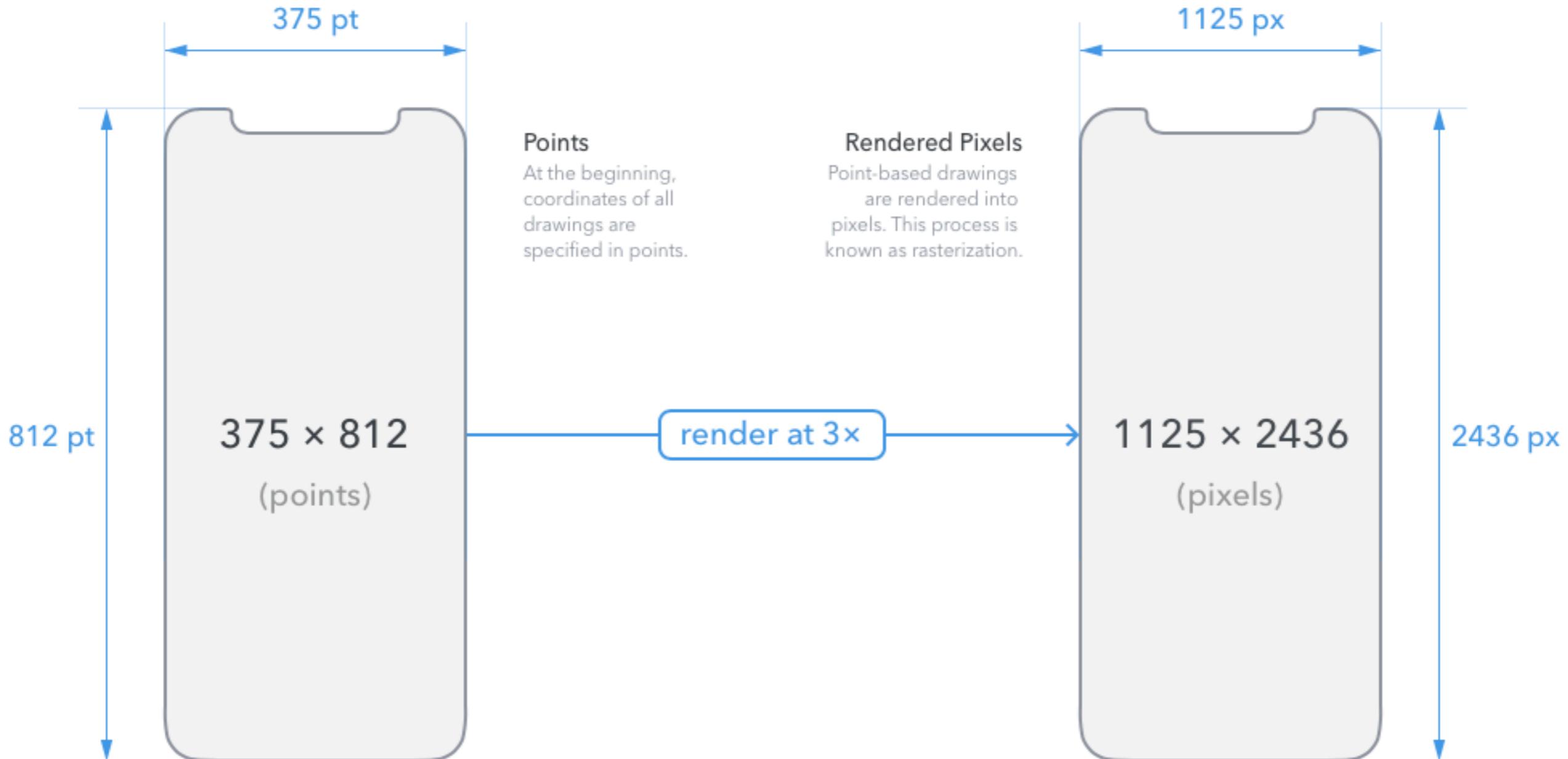
@3x



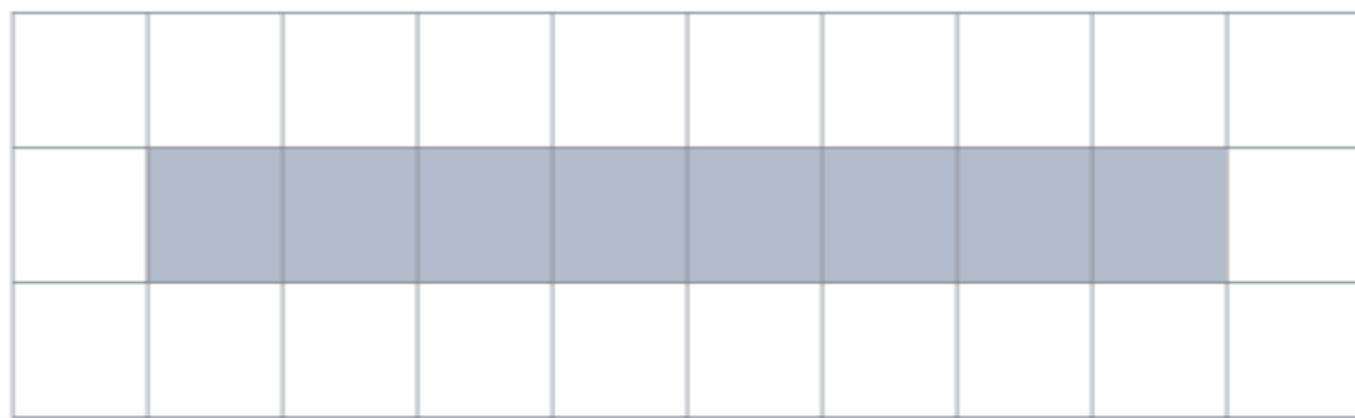
1125 x 2436 pixels

Device	Retina	Portrait (px)	Landscape (px)
iPhone XS Max	🕒	1242 x 2688	2688 x 1242
iPhone XR	🕒	828 x 1792	1792 x 828
iPhone X, XS	🕒	1125 x 2436	2436 x 1125
iPhone 6+, 6S+, 7+, 8+	🕒	1080 x 1920	1920 x 1080
iPhone 6, 6S, 7, 8	🕒	750 x 1334	1334 x 750
iPhone 5, 6SE 5, 5S, 5C, 6SE	🕒	640 x 1136	1136 x 640
iPhone 4 4, 4S	🕒	640 x 960	960 x 640
iPhone 1st, 2nd & 3rd Generation	🕒	320 x 480	480 x 320
iPad Air / Retina iPad 1st & 2nd Generation / 3rd & 4th	🕒	1536 x 2048	2048 x 1536
iPad Pro	🕒	2048 x 2732	2732 x 2048
iPad Mini 2nd, 3rd & 4th Generation	🕒	1536 x 2048	2048 x 1536
iPad Mini, 1st & 2nd Generation	🕒	768 x 1024	1024 x 768

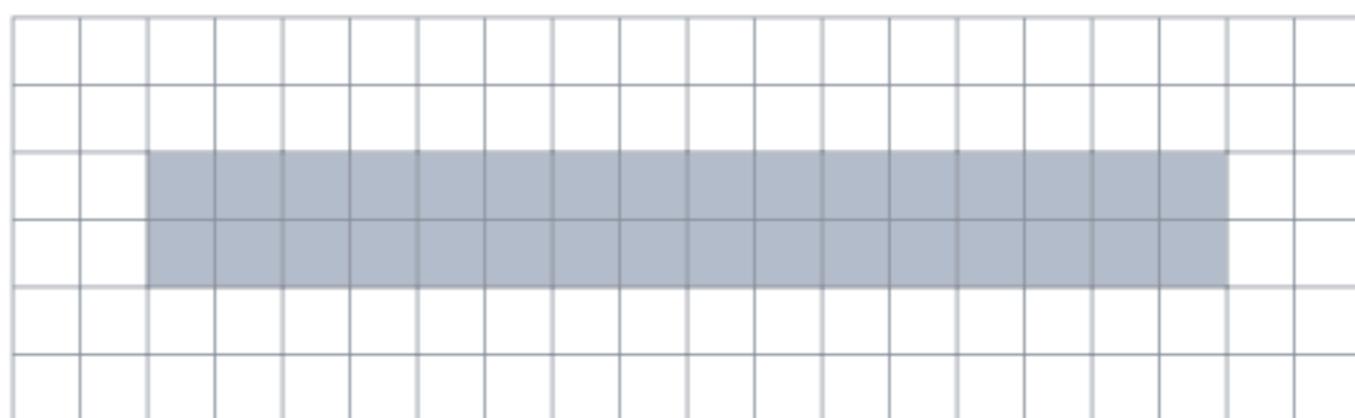
iPhone X Resolution



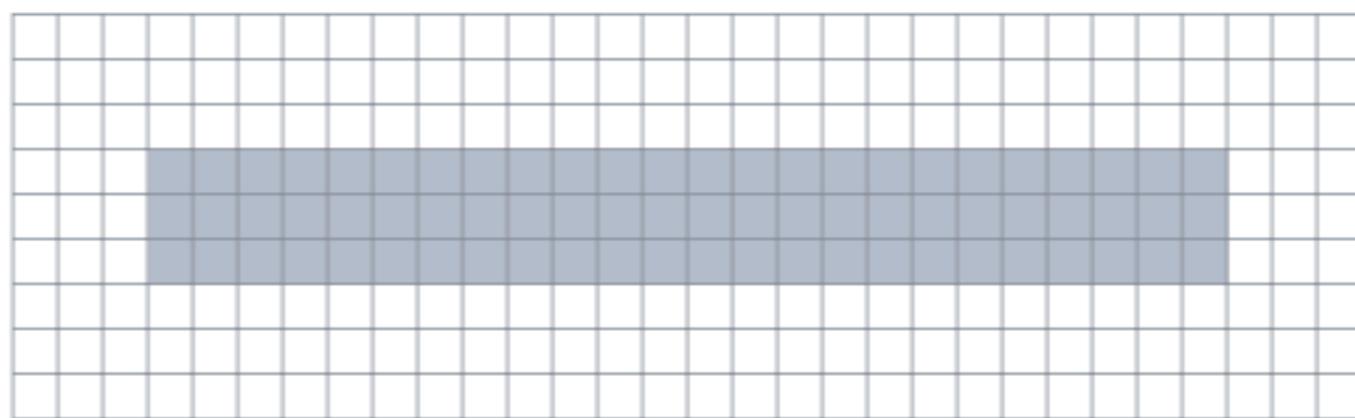
Original iPhone



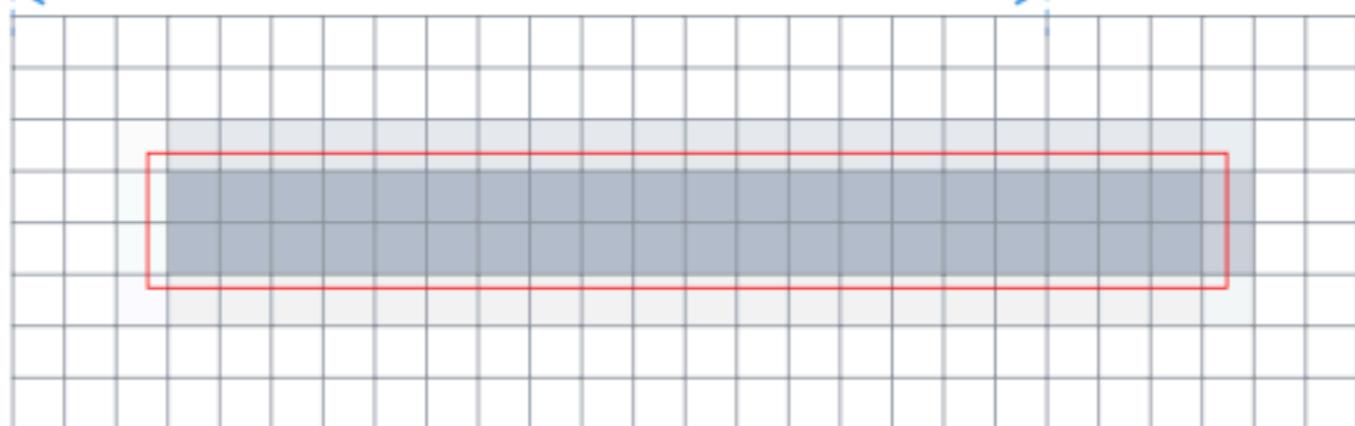
iPhone 5



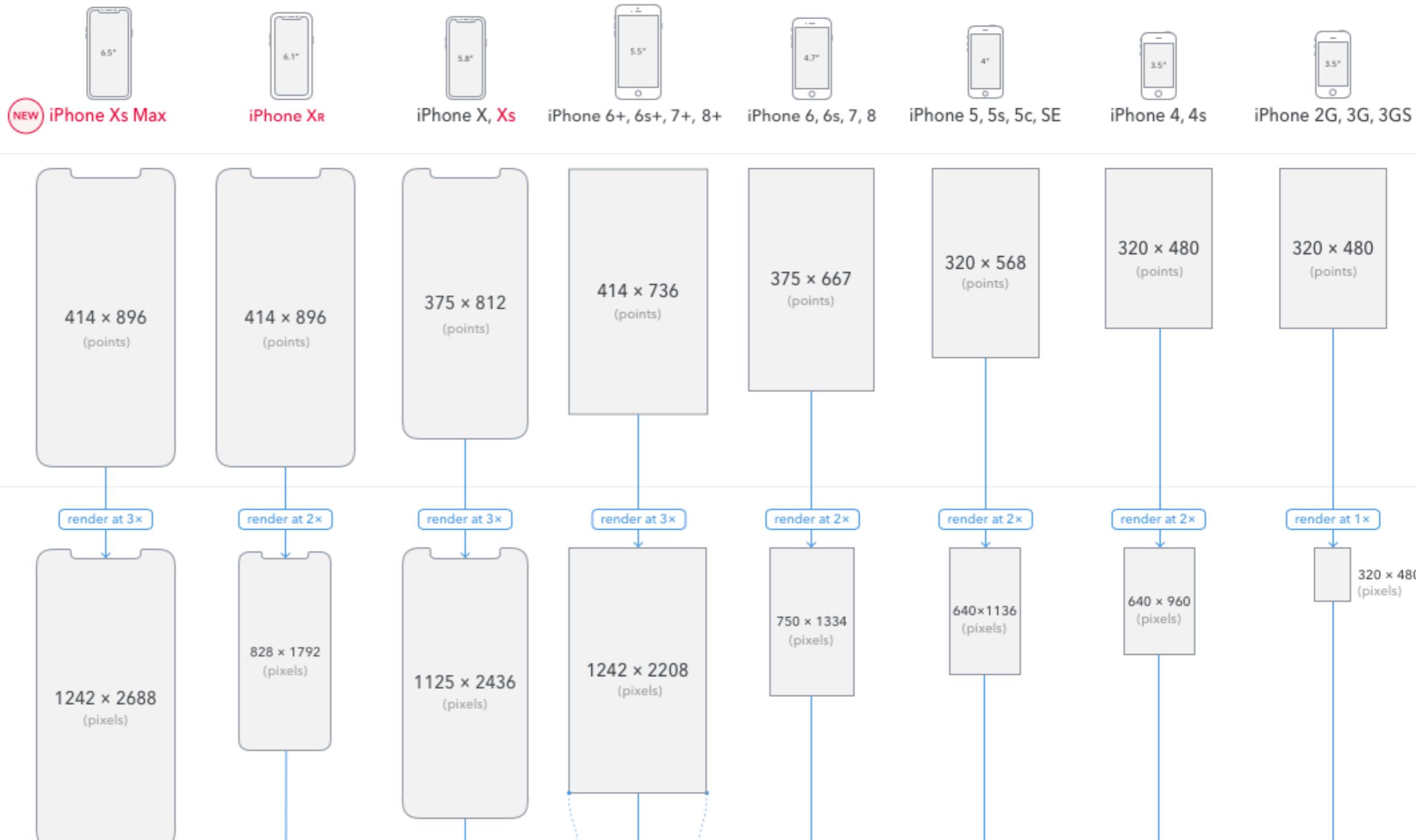
Hypothetical
Perfect 3x Display

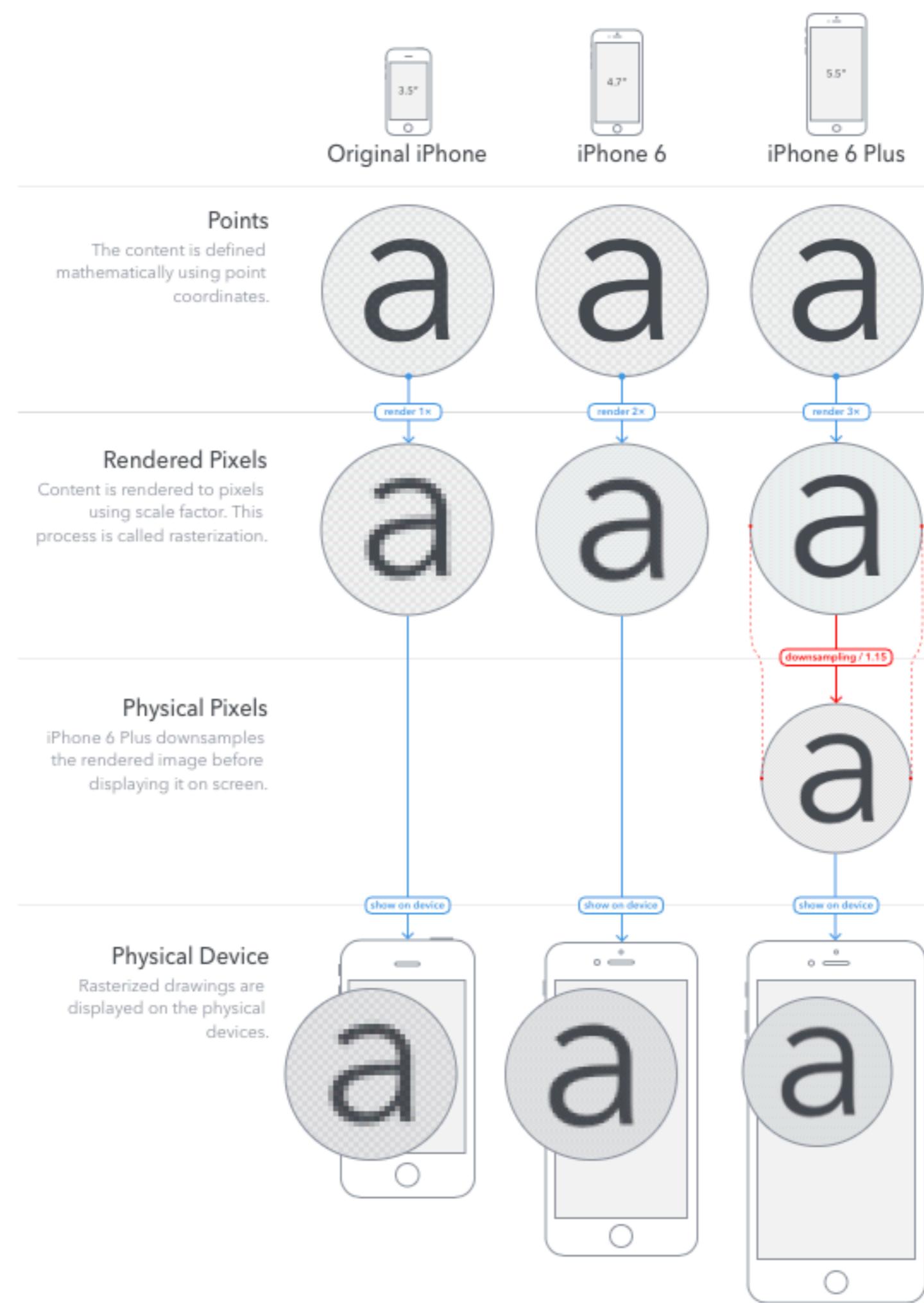


iPhone 6 Plus



iPhone Resolutions



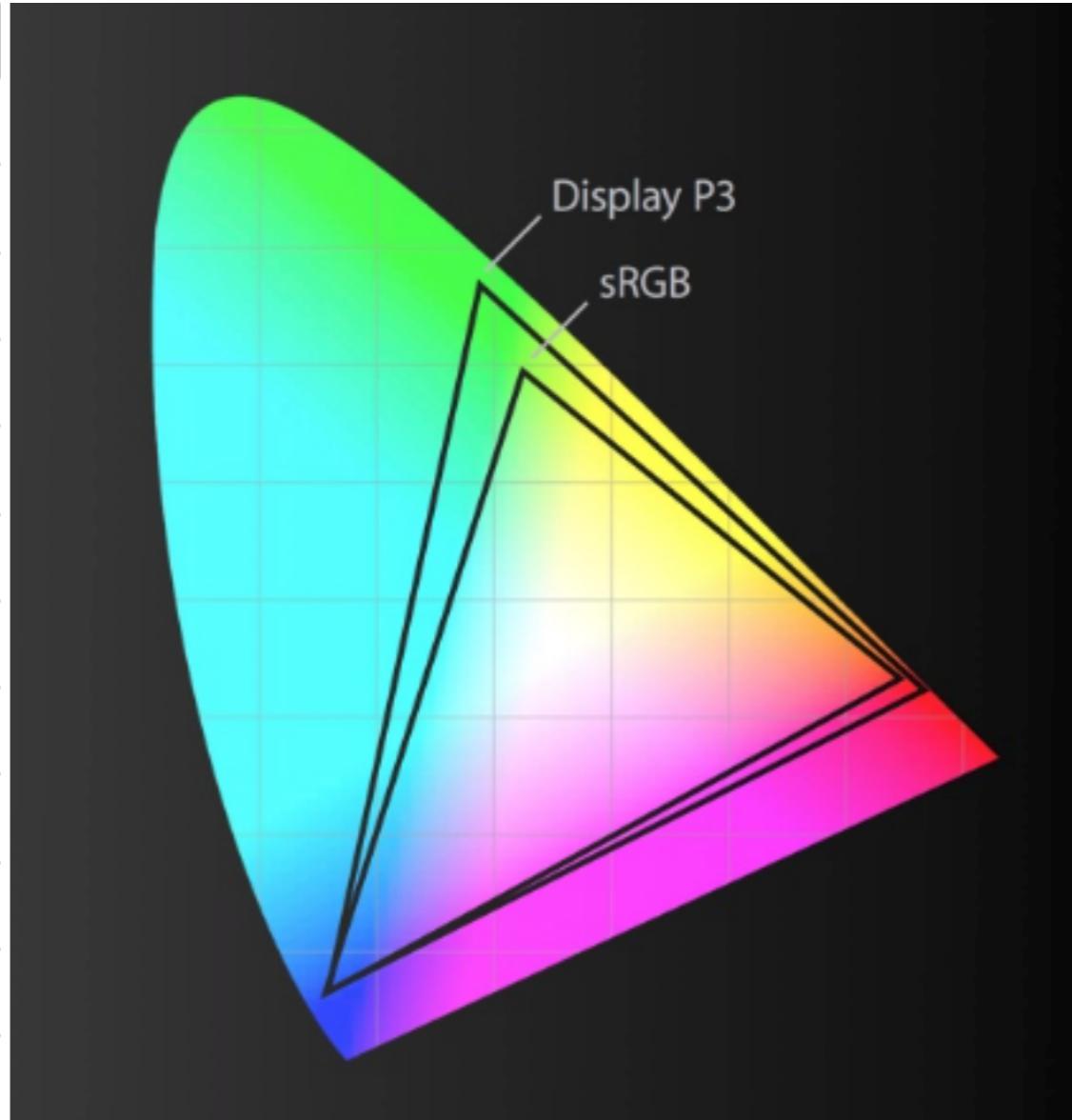


iOS Device Screen Geometry

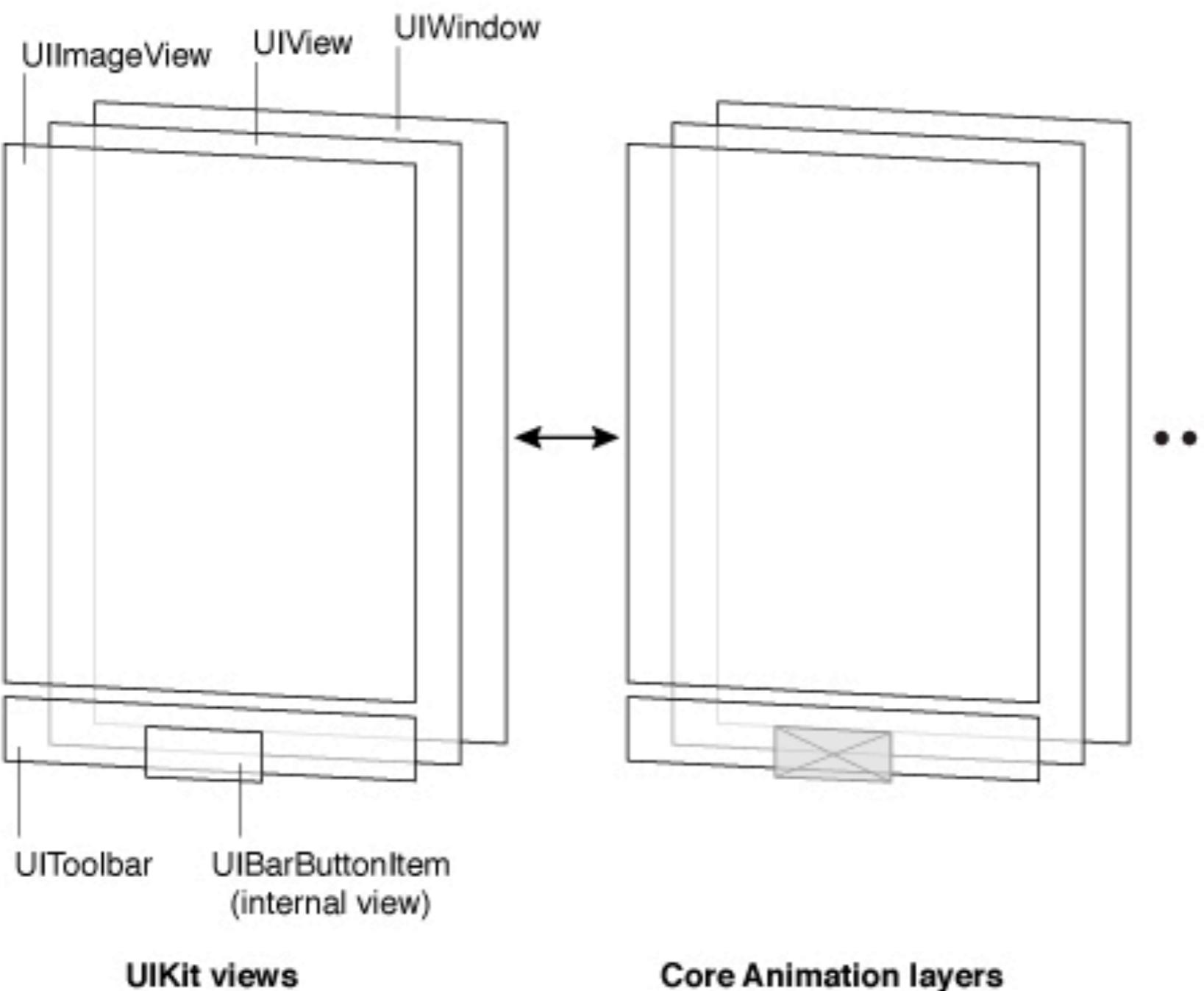
Device	Native Resolution (Pixels)	UIKit Size (Points)	Native Scale factor	UIKit Scale factor
iPhone X	1125 x 2436	375 x 812	3.0	3.0
iPhone 8 Plus	1080 x 1920	414 x 736	2.608	3.0
iPhone 8	750 x 1334	375 x 667	2.0	2.0
iPhone 7 Plus	1080 x 1920	414 x 736	2.608	3.0
iPhone 6s Plus	1080 x 1920	375 x 667	2.608	3.0
iPhone 6 Plus	1080 x 1920	375 x 667	2.608	3.0
iPhone 7	750 x 1334	375 x 667	2.0	2.0
iPhone 6s	750 x 1334	375 x 667	2.0	2.0
iPhone 6	750 x 1334	375 x 667	2.0	2.0
iPhone SE	640 x 1136	320 x 568	2.0	2.0
iPad Pro 12.9-inch (2nd generation)	2048 x 2732	1024 x 1366	2.0	2.0
iPad Pro 10.5-inch	2224 x 1668	1112 x 834	2.0	2.0
iPad Pro (12.9-inch)	2048 x 2732	1024 x 1366	2.0	2.0
iPad Pro (9.7-inch)	1536 x 2048	768 x 1024	2.0	2.0
iPad Air 2	1536 x 2048	768 x 1024	2.0	2.0
iPad Mini 4	1536 x 2048	768 x 1024	2.0	2.0

Color Reproduction

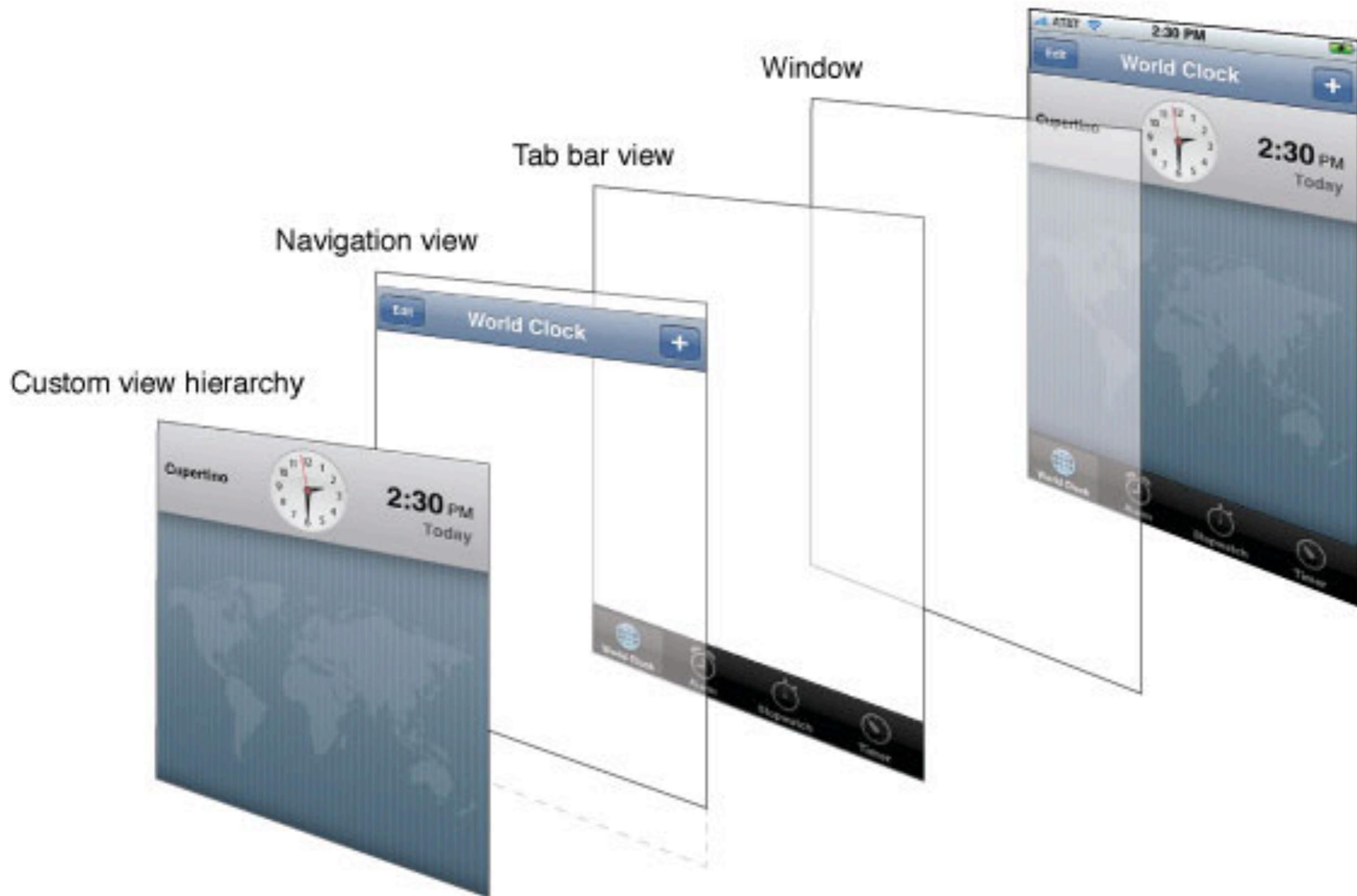
Device	Native Color Space	TrueTone Display
iPhone X	Display P3	Yes
iPhone 8 Plus	Display P3	Yes
iPhone 8	Display P3	Yes
iPhone 7 Plus	Display P3	No
iPhone 7	Display P3	No
iPhone 6s Plus	sRGB	No
iPhone 6s	sRGB	No
iPhone SE	sRGB	No
iPhone 6 Plus	sRGB	No
iPhone 6	sRGB	No
iPad Mini 4	sRGB	No
iPad Air 2	sRGB	No
iPad Pro 12.9-inch (2nd generation)	Display P3	Yes
iPad Pro 10.5-inch	Display P3	Yes
iPad Pro (12.9-inch)	sRGB	No
iPad Pro (9.7-inch)	Display P3	Yes



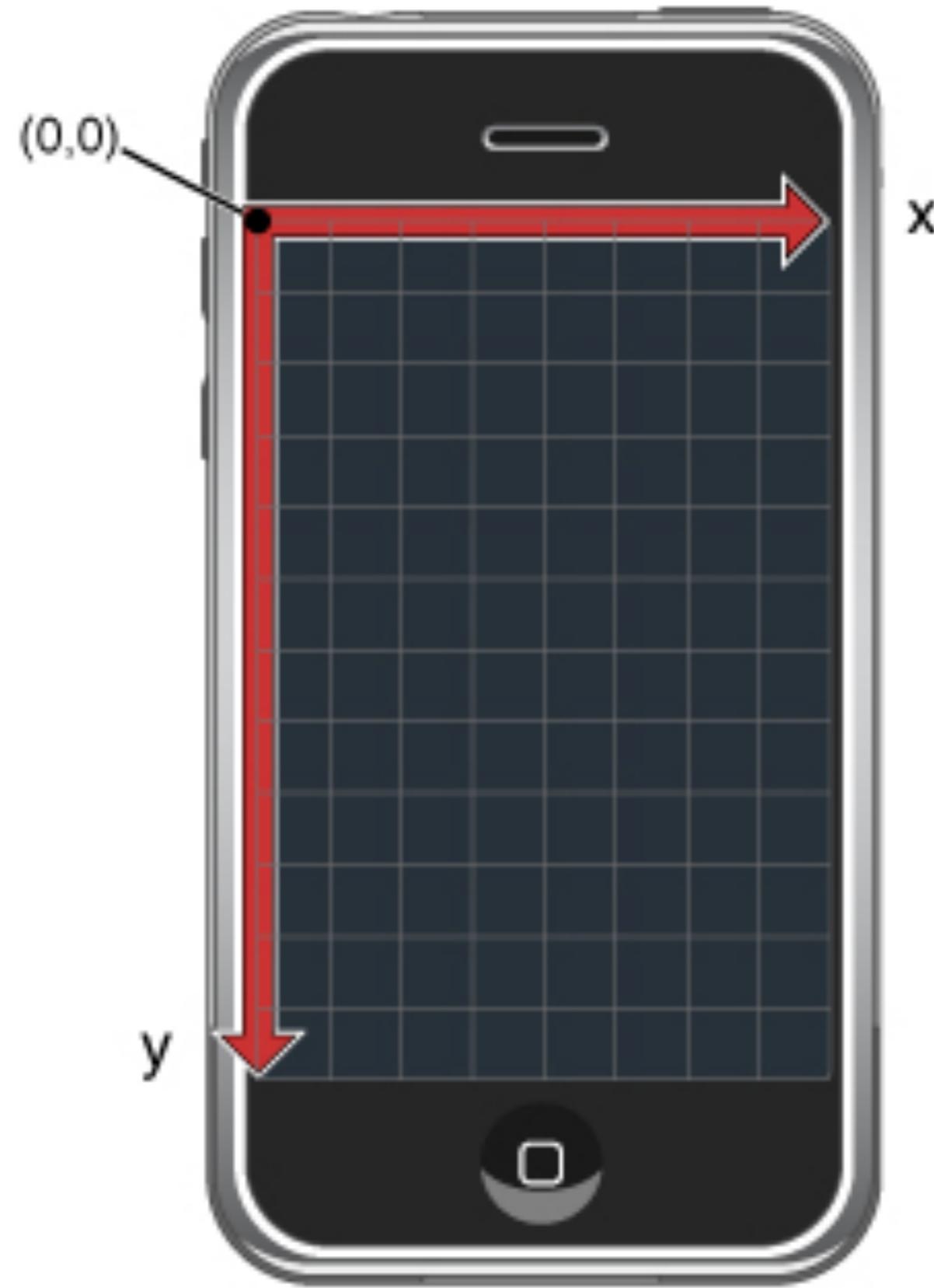
Architecture of the views



Layered views

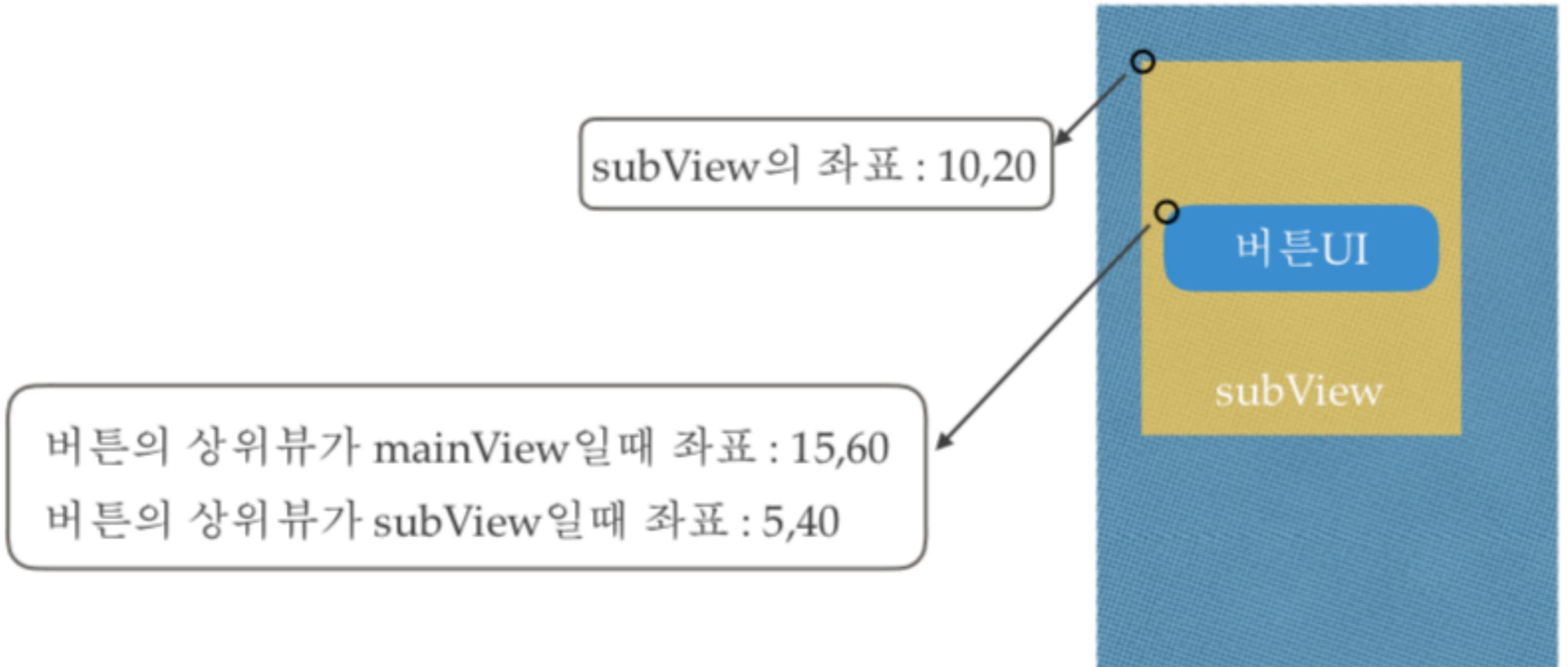


Coordinate system orientation



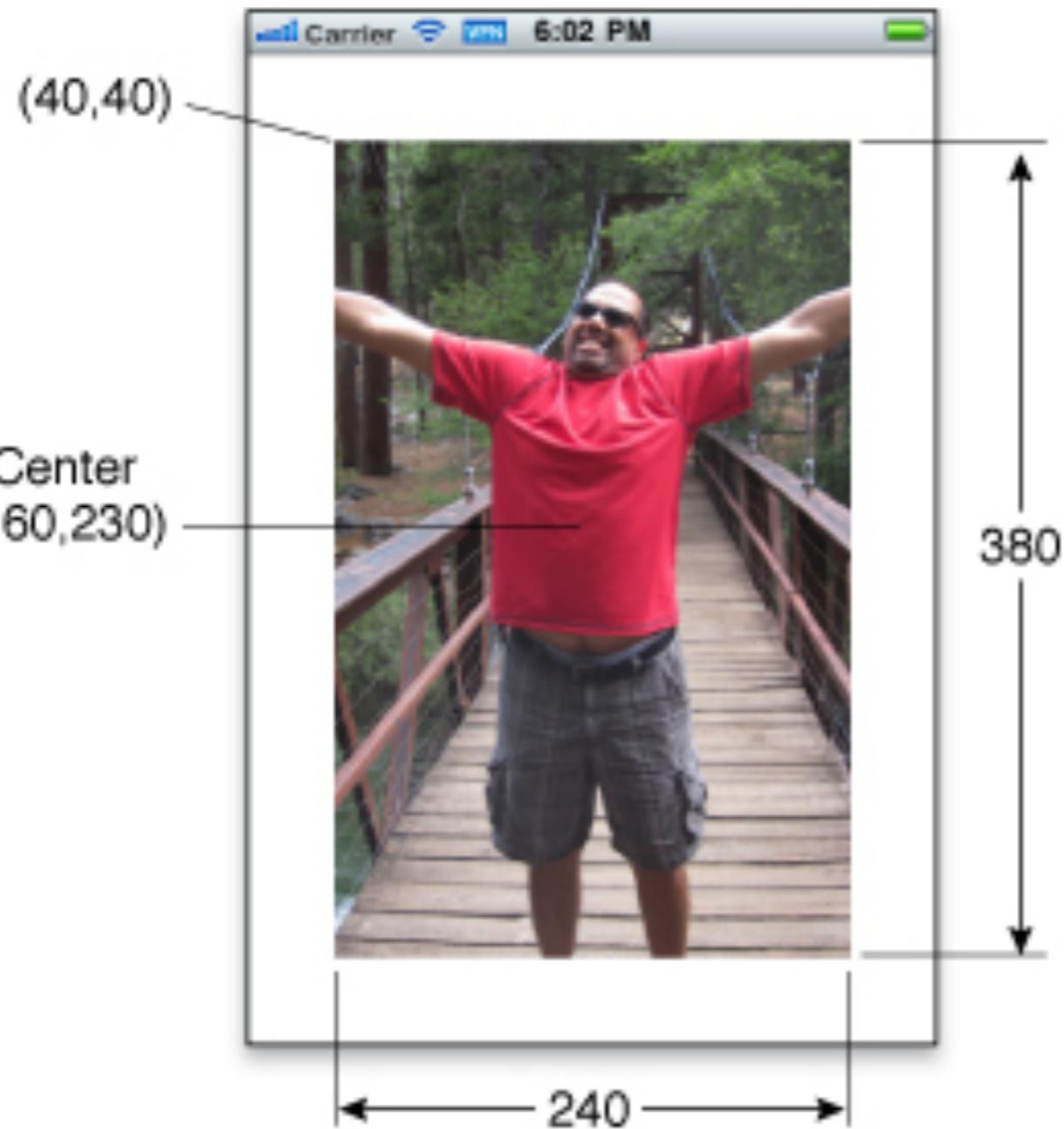
View Frame

View Frame 의 좌표는 상위뷰를 기준으로 결정

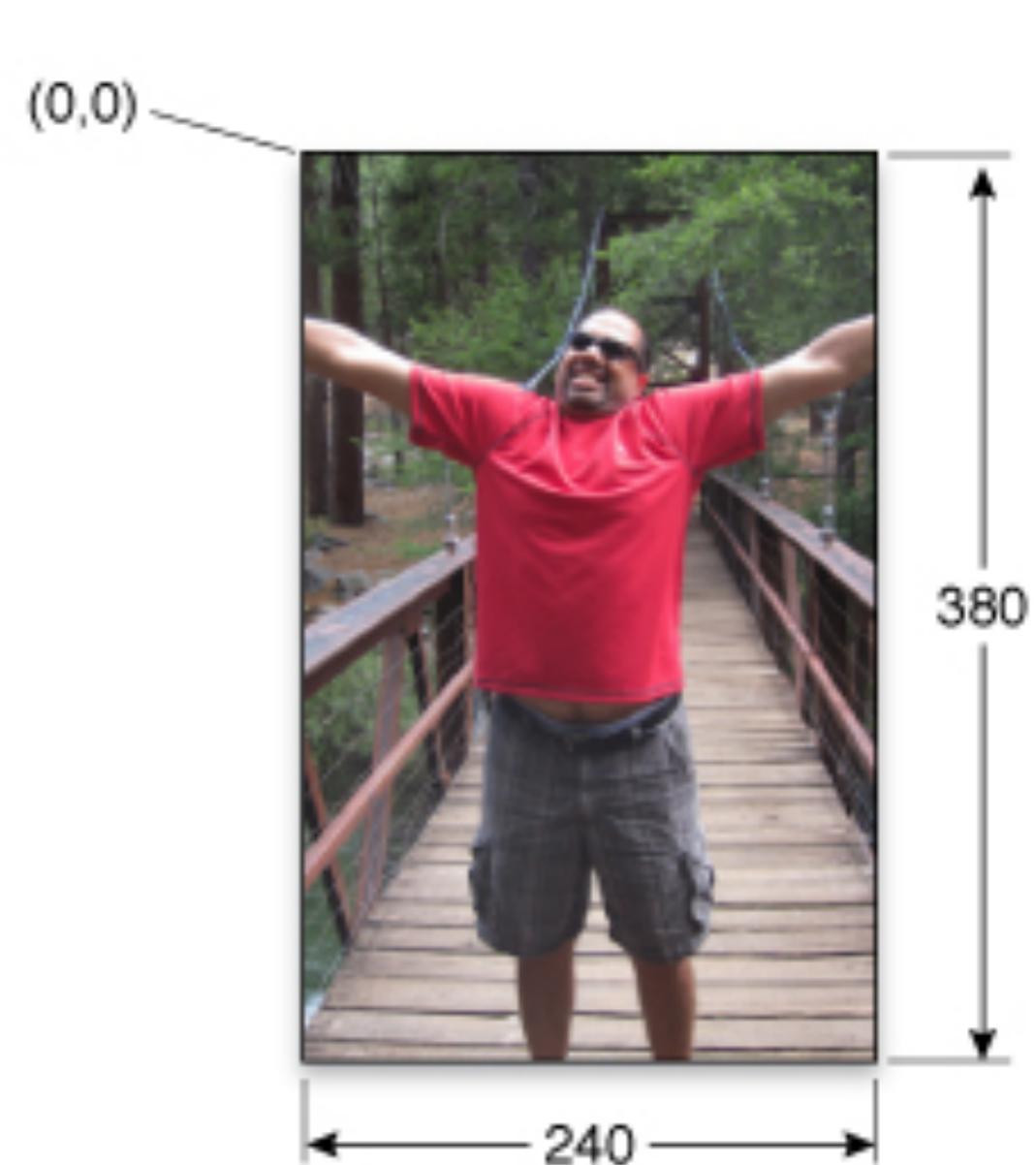


Frame and Bounds

Frame rectangle



Bounds rectangle



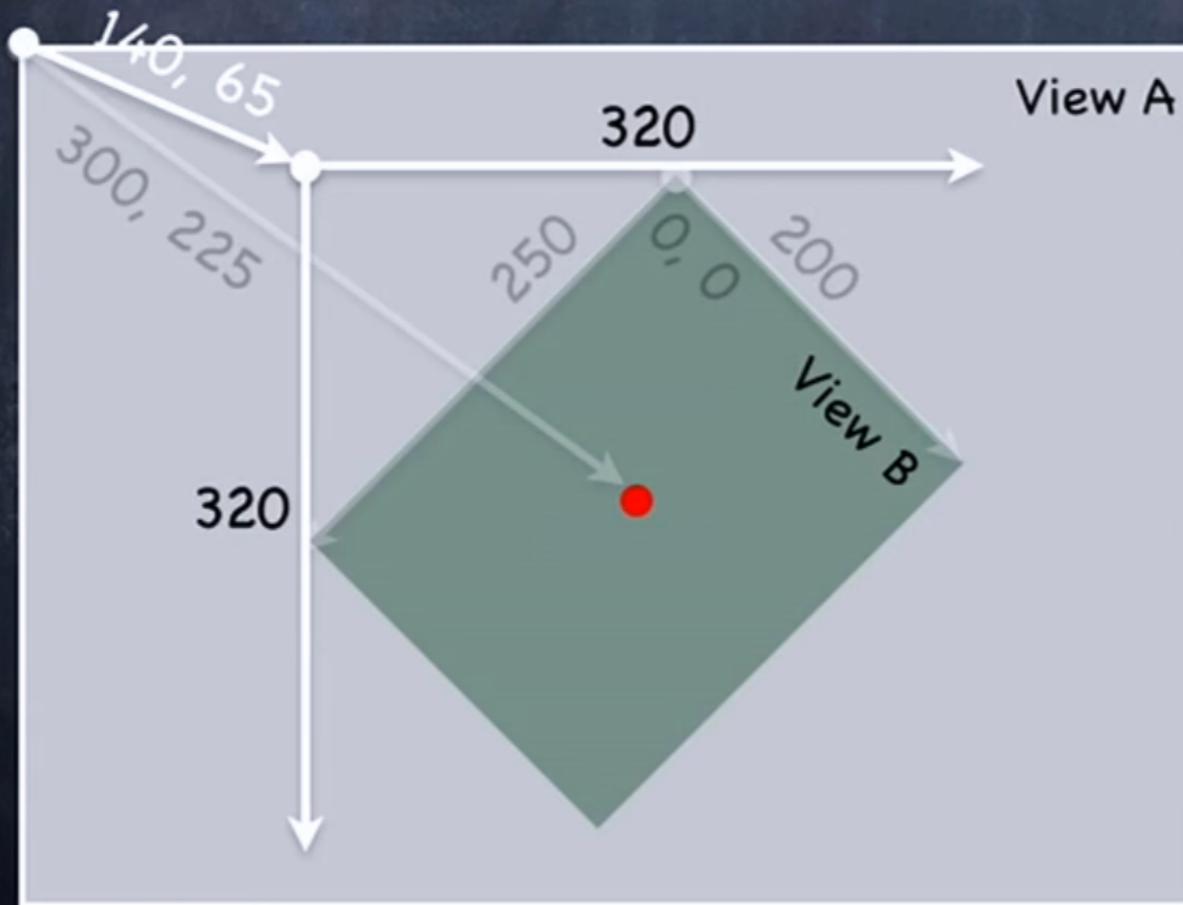
Frame and Bounds

bounds vs frame

- Use **frame** and/or **center** to position a `UIView`

These are never used to draw inside a view's coordinate system

You might think `frame.size` is always equal to `bounds.size`, but you'd be wrong ...



Views can be rotated (and scaled and translated)

View B's bounds = $((0,0),(200,250))$

View B's frame = $((140,65),(320,320))$

View B's center = $(300,225)$

View B's middle in its own coordinates is ...

(bounds.midX, bounds.midY) = (100, 125)

Views are rarely rotated, but don't misuse frame or center anyway by assuming that.



Super View

frame (0, 0, 375, 667)
bounds (0, 0, 375, 667)

(0, 0)

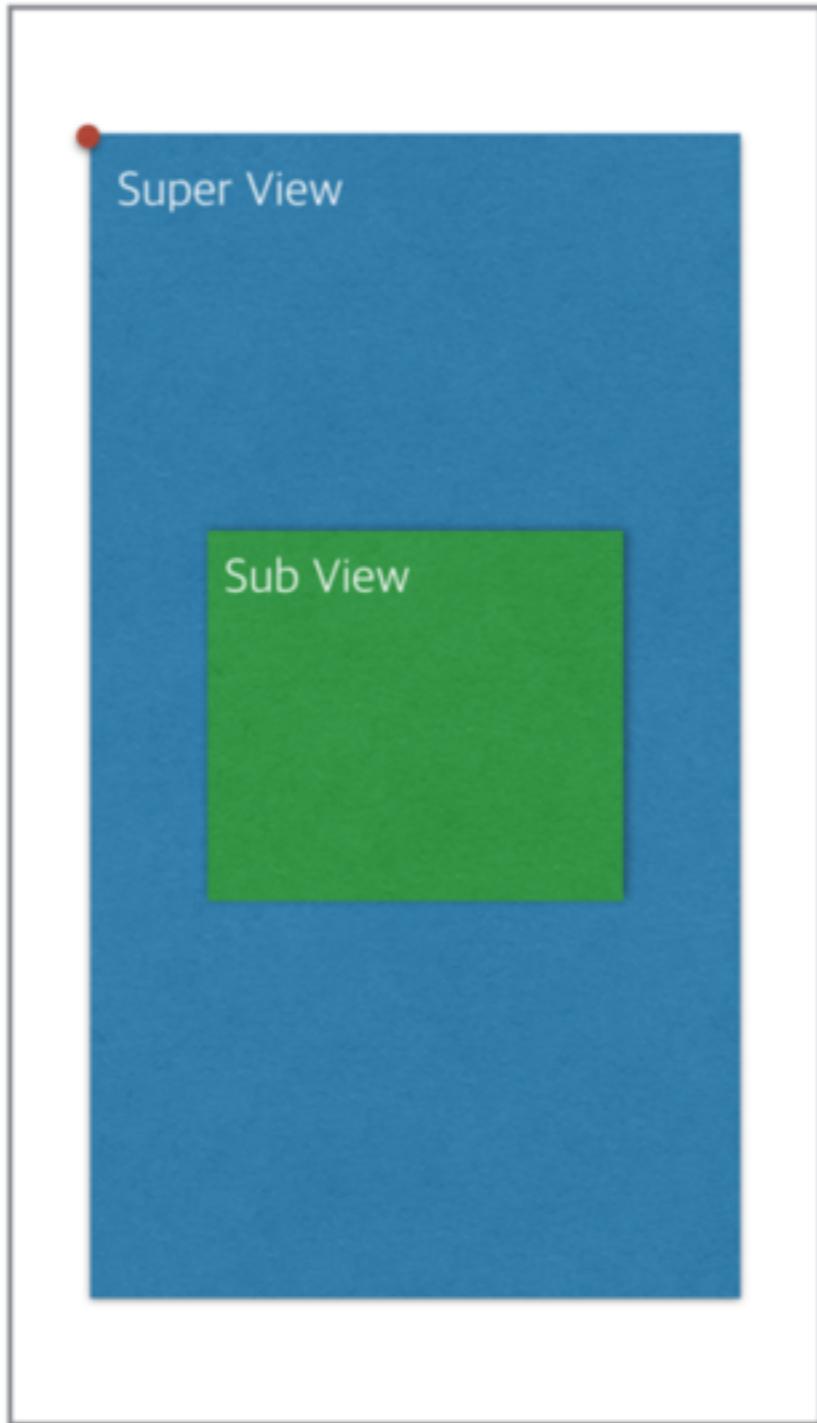
Sub View

frame (119, 217, 137, 127)
bounds (0, 0, 137, 127)

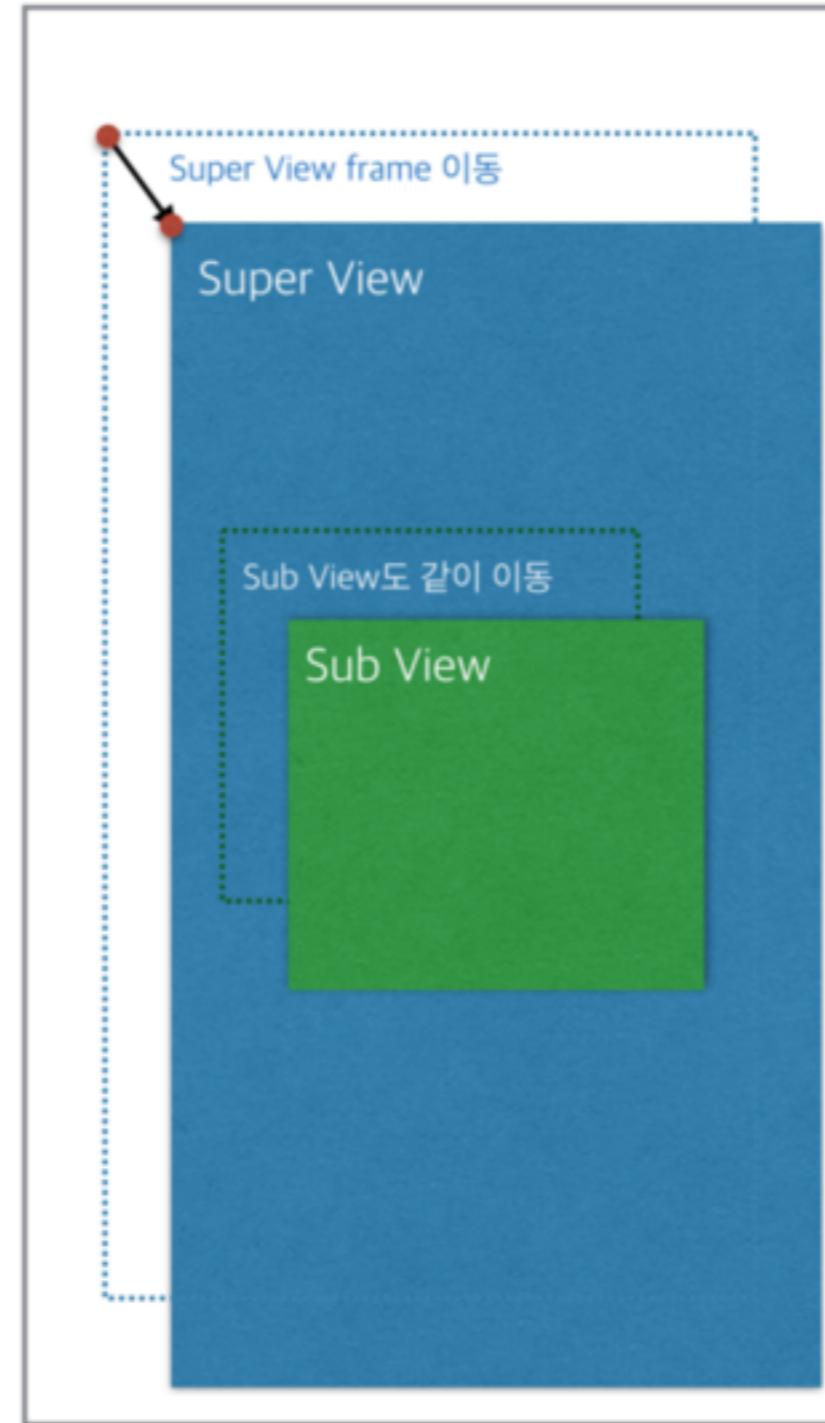
(119, 217)

Frame

Super Super View

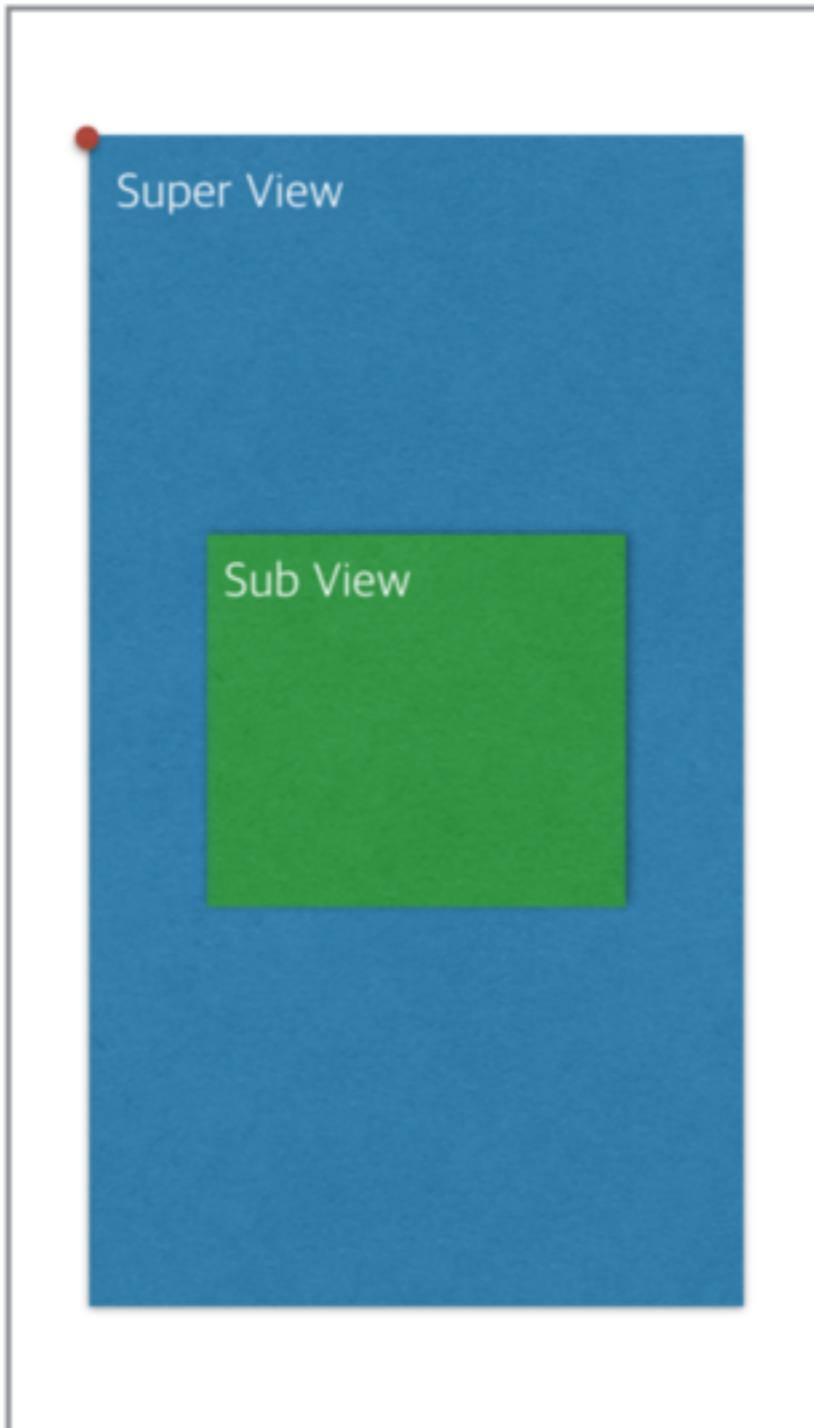


Super Super View

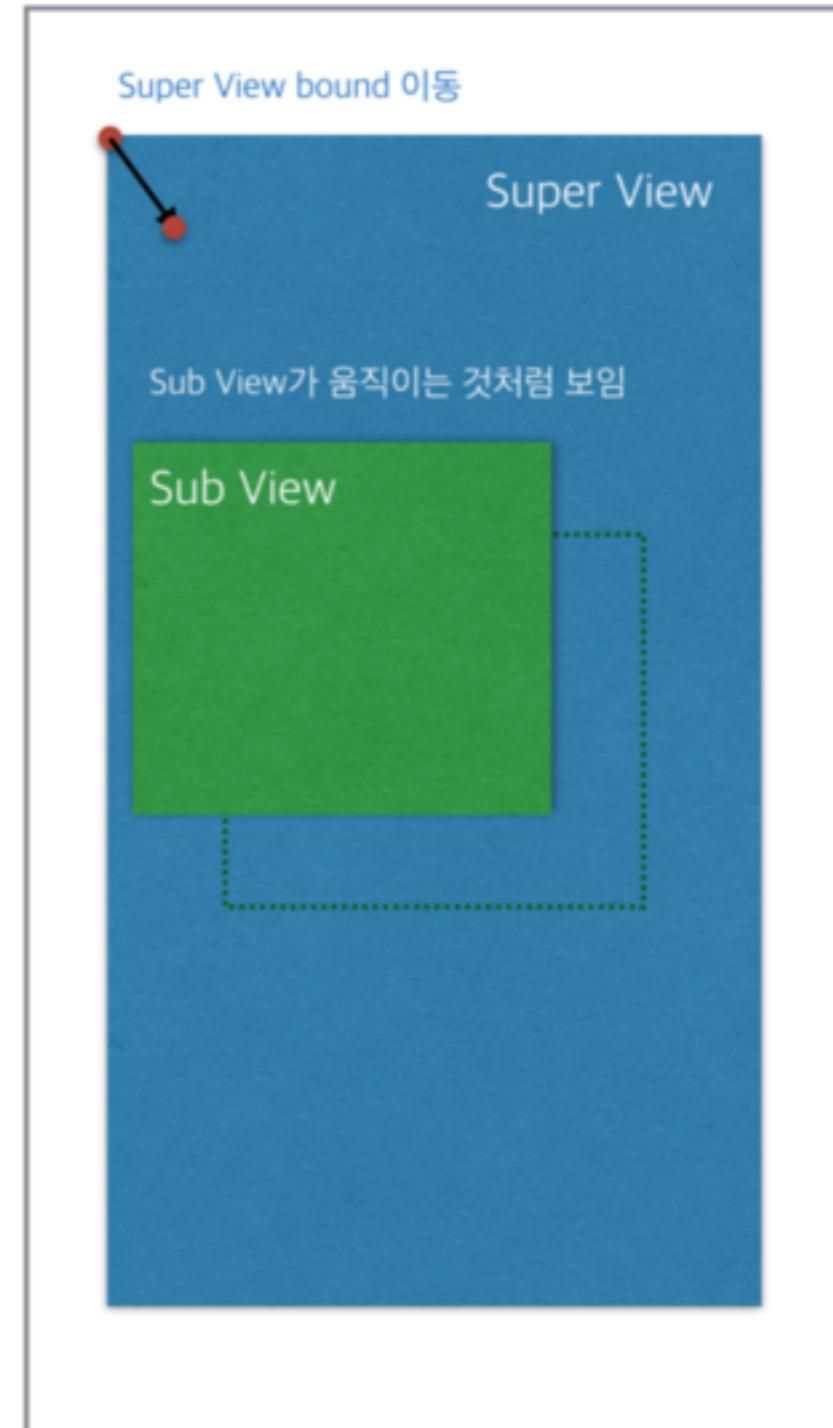


Bounds

Super Super View



Super Super View



UIViewContentMode

```
public enum UIViewContentMode : Int {  
    case scaleToFill  
    case scaleAspectFit  
    case scaleAspectFill  
  
    case redraw  
  
    case center  
    case top  
    case bottom  
    case left  
    case right  
    case topLeft  
    case topRight  
    case bottomLeft  
    case bottomRight  
}
```

———— **Scailing**

———— **Redrawing**

———— **Positioning**

Scaling

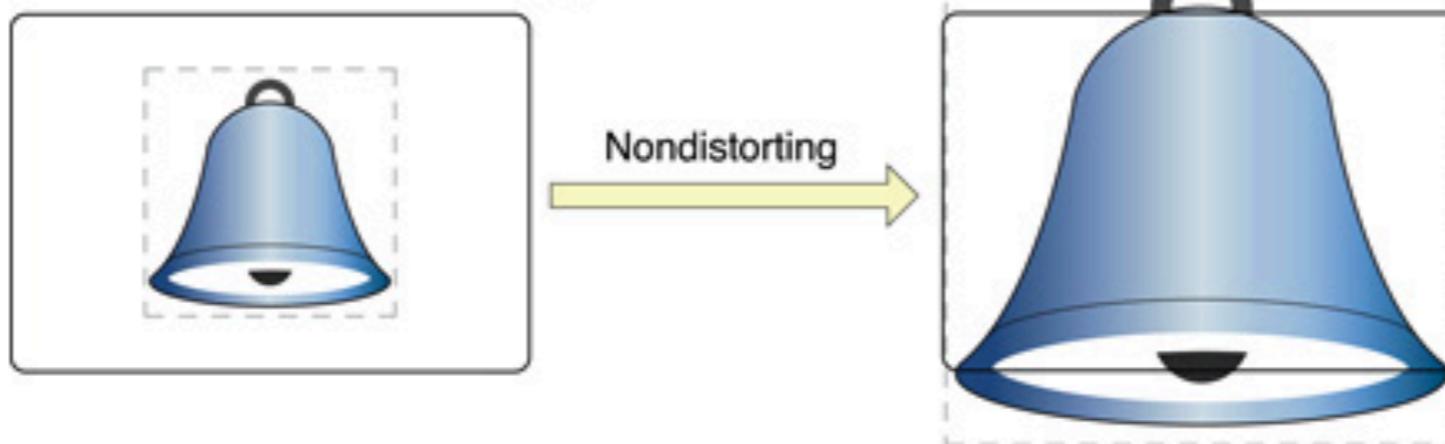
UIViewContentModeScaleToFill



UIViewContentModeScaleAspectFit



UIViewContentModeScaleAspectFill



Redrawing

- View 의 크기가 변경될 때마다 `setNeedsDisplay()` 메서드를 호출하여 연관 컨텐츠를 항상 다시 그리게 하는 것
- `draw(_:)` 메서드에 별도의 그리기 작업이 들어가고 크기에 영향을 받을 경우 설정
- 크기 변경시마다 매번 다시 그려야 하므로 성능상으로는 좋지 않은 영향

Positioning

