

面试题--CALayer 与 UIView 的关系

CC老师

在 **iOS** 中，几乎所有的 `view` 都是由一个底层的 `layer` 来驱动的。`view` 和它的 `layer` 之间有着紧密的联系，`view` 其实直接从 `layer` 对象中获取了绝大多数它所需要的数据。

在 **iOS** 中也有一些单独的 `layer`，有很多特殊的图层,比如

- `CAShapeLayer`--用来实现贝塞尔曲线绘制
- `CATextLayer` --以图层的形式绘制文字，除 `UILabel` 所有的绘制特性还额外提供一些新特性。
- `CATransformLayer`--它能够用于构造一个层级的3D结构，比如立方体等
- `CAGradientLayer`--生成多种颜色的平滑渐变，绘制使用了硬件加速GPU
- `CAReplicatorLayer`--高效生成许多相似的图层
- `CAScrollViewLayer`--可定义滚动的区域范围，但不提供键盘或鼠标事件处理，也不提供可见的滚动条。
- `CATiledLayer`--将大图分解成小片然后单独按需载入，可以解决载入大图时的性能问题
- `CAEmitterLayer`--高性能的例子引擎，可以创建实时的例子动画
- `CAEAGLLayer`--显示任意的OpenGL图形，OpenGL ES渲染环境。
- `AVCaptureVideoPreviewLayer`--显示摄像机捕获的内容.预览图层

`view` 和 `layer` 之间是如何协同工作和被精心设计的。无论何时一个可动画的 `layer` 属性改变时，`layer` 都会寻找并运行合适的 'action' 来实行这个改变。在 `Core Animation` 的专业术语中就把这样的动画统称为动作 (action，或者 `CAAction`)。

CAAction：技术上来说，这是一个接口，并可以用来做各种事情。但是实际中，某种程度上你可以只把它理解为用来处理动画

`layer` 将像文档中所写的的那样去寻找动作，整个过程分为五个步骤。第一步中的在 `view` 和 `layer` 中交互的部分是最有意思的：

`layer` 通过向它的 `delegate` 发送 `actionForLayer:forKey:` 消息来询问提供一

个对应属性变化的 `action`。`delegate` 可以通过返回以下三者之一来进行响应：

1. 它可以返回一个动作对象，这种情况下 `layer` 将使用这个动作。
2. 它可以返回一个 `nil`，这样 `layer` 就会到其他地方继续寻找。
3. 它可以返回一个 `NSNull` 对象，告诉 `layer` 这里不需要执行一个动作，搜索也会就此停止。

当 `layer` 在背后支持一个 `view` 的时候，`view` 就是它的 `delegate`。在 iOS 中，如果 `layer` 与一个 `UIView` 对象关联时，这个属性必须被设置为持有这个 `layer` 的那个 `view`。

著作权归作者所有，任何形式的转载都请联系作者获得授权并注明出处。
如未能核实来源或转发内容图片有权利瑕疵,请及时联系[小雁子]助理老师
QQ:1900009930.