

[返回目录:全网各大厂 iOS 面试题-题集大全](#)

## 什么是离屏渲染？什么情况下会触发？该如何应对？

离屏渲染就是在当前屏幕缓冲区以外，新开辟一个缓冲区进行操作。

离屏渲染出发的场景有以下：

- 圆角（`maskToBounds` 并用才会触发）
- 图层蒙版
- 阴影
- 光栅化

### 为什么要有离屏渲染？

大家高中物理应该学过显示器是如何显示图像的：需要显示的图像经过 CRT 电子枪以极快的速度一行一行的扫描，扫描出来就呈现了一帧画面，随后电子枪又会回到初始位置循环扫描，形成了我们看到的图片或视频。

为了让显示器的显示跟视频控制器同步，当电子枪新扫描一行的时候，准备扫描的时候发送一个水平同步信号(HSync 信号)，显示器的刷新频率就是 HSync 信号产生的频率。然后 CPU 计算好 frame 等属性，将计算好的内容交给 GPU 去渲染，GPU 渲染好之后就会放入帧缓冲区。然后视频控制器会按照 HSync 信号逐行读取帧缓冲区的数据，经过可能的数模转换传递给显示器，就显示出来了。具体的大家自行查找资料或询问专业人士，这里只参考网上资料做一个简单的描述。

离屏渲染的代价很高，想要进行离屏渲染，首选要创建一个新的缓冲区，屏幕渲染会有一个上下文环境的一个概念，离屏渲染的整个过程需要切换上下文环境，先从当前屏幕切换到离屏，等结束后，又要将上下文环境切换回来。这也是为什么会消耗性能的原因了。

由于垂直同步的机制，如果在一个 HSync 时间内，CPU 或者 GPU 没有完成内容提交，则那一帧就会被丢弃，等待下一次机会再显示，而这时显示屏会保留之前的内容不变。这就是界面卡顿的原因。

### 为什么要避免离屏渲染？

CPU GPU 在绘制渲染视图时做了大量的工作。离屏渲染发生在 GPU 层面上，会创建新的渲染缓冲区，会触发 OpenGL 的多通道渲染管线，图形上下文的切换会造成额外的开销，增加 GPU 工作量。如果 CPU GPU 累计耗时 16.67 毫秒还没有完成，就会造成卡顿掉帧。

圆角属性、蒙层遮罩 都会触发离屏渲染。指定了以上属性，标记了它在新的图形上下文中，在未愈合之前，不可以用于显示的时候就出发了离屏渲染。

- 在 OpenGL 中，GPU 有 2 种渲染方式
  - On-Screen Rendering: 当前屏幕渲染，在当前用于显示的屏幕缓冲区进行渲染操作
  - Off-Screen Rendering: 离屏渲染，在当前屏幕缓冲区以外新开辟一个缓冲区进行渲染操作
- 离屏渲染消耗性能的原因
  - 需要创建新的缓冲区
  - 离屏渲染的整个过程，需要多次切换上下文环境，先是从当前屏幕（On-Screen）切换到离屏（Off-Screen）；等到离屏渲染结束以后，将离屏缓冲区的渲染结果显示到屏幕上，又需要将上下文环境从离屏切换到当前屏幕
- 哪些操作会触发离屏渲染？
  - 光栅化，layer.shouldRasterize = YES
  - 遮罩，layer.mask
  - 圆角，同时设置 layer.masksToBounds = YES、layer.cornerRadius 大于 0
  - 考虑通过 CoreGraphics 绘制裁剪圆角，或者叫美工提供圆角图片
  - 阴影，layer.shadowXXX，如果设置了 layer.shadowPath 就不会产生离屏渲染

[返回目录:全网各大厂 iOS 面试题-题集大全](#)

---

## 更多精选大厂 · iOS 面试题答案 PDF 文集



获取加小编的 iOS 技术交流圈: [937 194 184](#), 直接获取