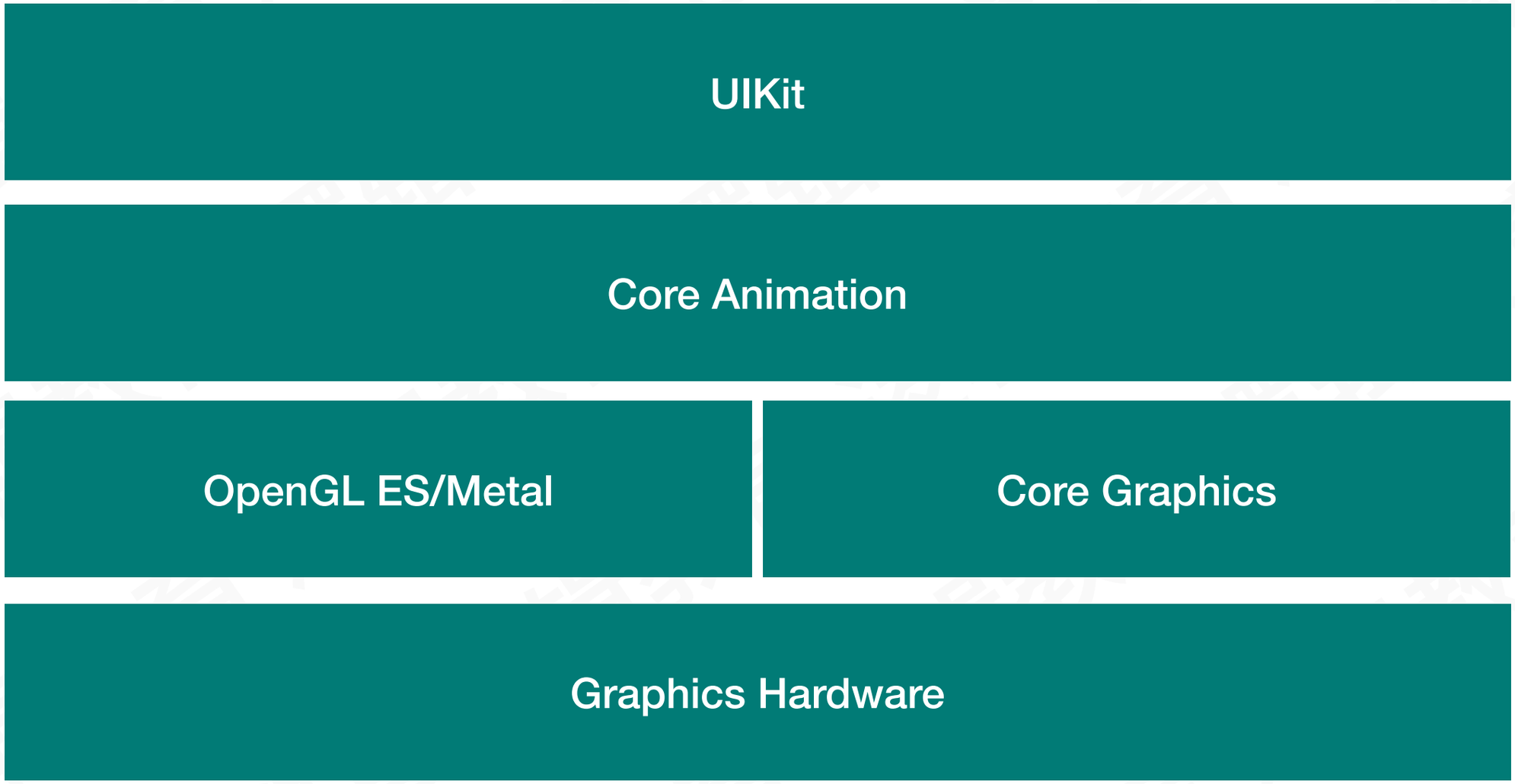
逻辑教育
Logic education

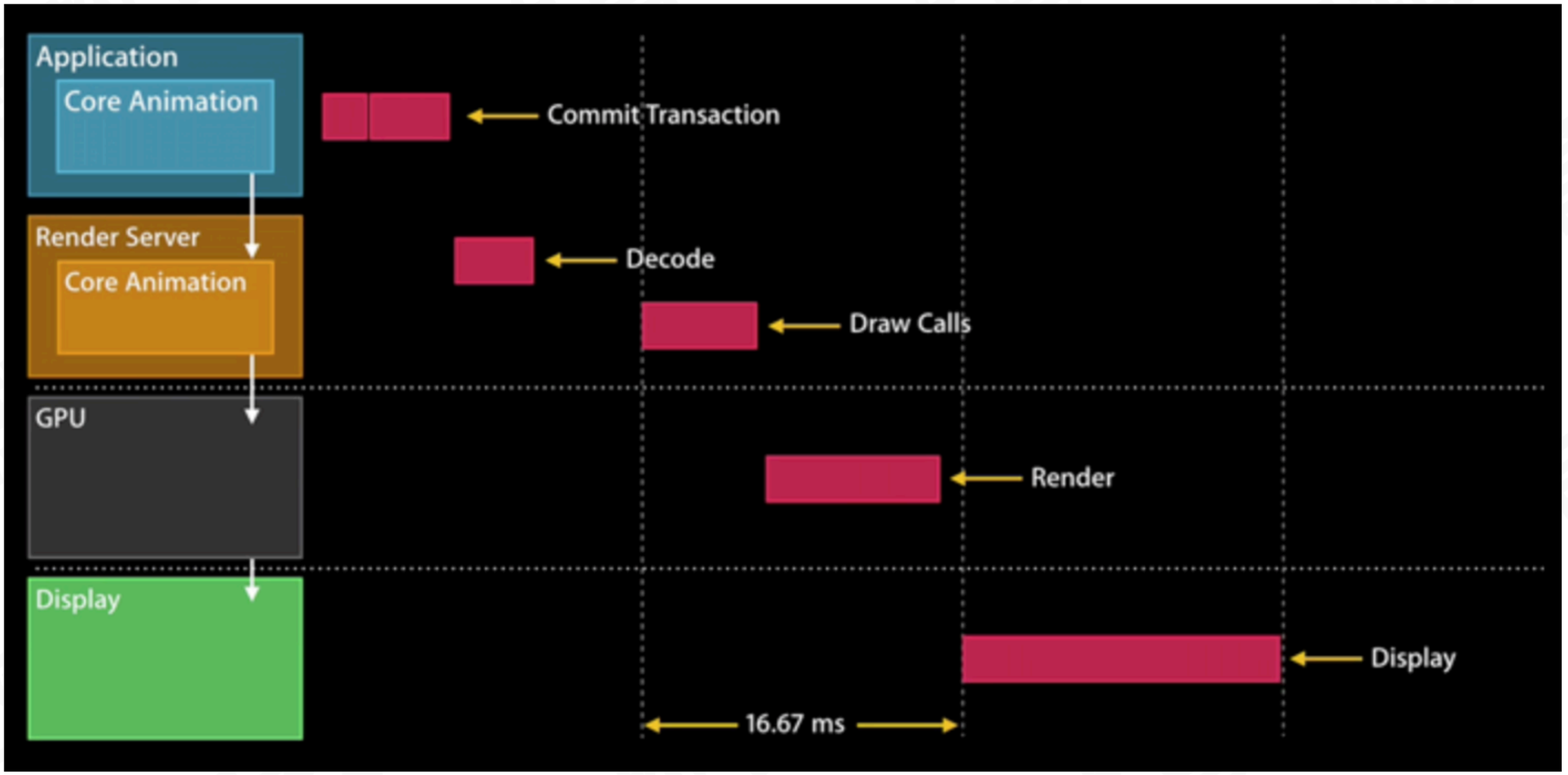
界面优化

大师班第 31 讲

LG_Kody

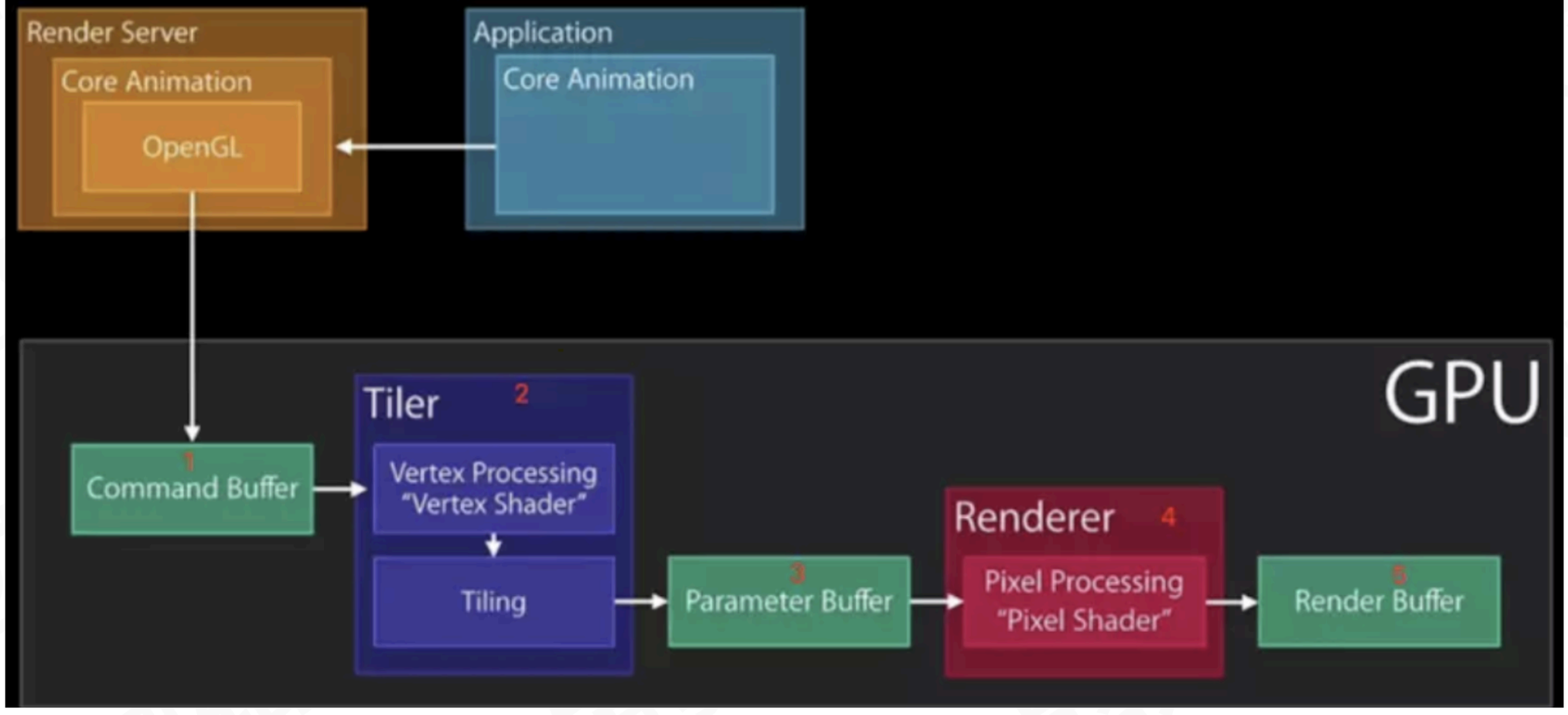
@Kody





Tile Based Rendering

Rendering pass



- Application 中布局 UIKit 视图控件间接的关联 Core Animation 图层
- Core Animation 图层相关的数据提交到 iOS Render Server，即 OpenGL ES & Core Graphics
- Render Server 将与 GPU 通信把数据经过处理之后传递给 GPU
- GPU 调用 iOS 当前设备渲染相关的图形设备 Display

逻辑教育
Logic education

Commit Transaction做了什么

- Layout，构建视图，frame,遍历的操作[UIView layerSubview],[CALayer layoutSubLayers]
- Display，绘制视图 ， display —– drawReact()， displayLyaer:(位图的绘制)
- Prepare，额外的 Core Animation 工作，比如解码
- Commit，打包图层并将它们发送到 Render Server

逻辑教育
Logic education

```
//Tips:这句代码到底是如何被系统知道，并被绘制的?
_lgView.backgroundColor = [UIColor redColor];
```

@Kody

伪代码示意图

```
CA::Transaction::commit() {
    CA::Context::commit_transaction(CA::Transaction*) () {
        CA::Layer::layout_if_needed(CA::Transaction*) () {
            CollectLayersData *layerTree;
            CA::Layer::collect_layers_(CA::Layer::CollectLayersData*);
            for in layerTree {
                [layer layoutSublayers];
            }
        }
    }
}
```

伪代码示意图

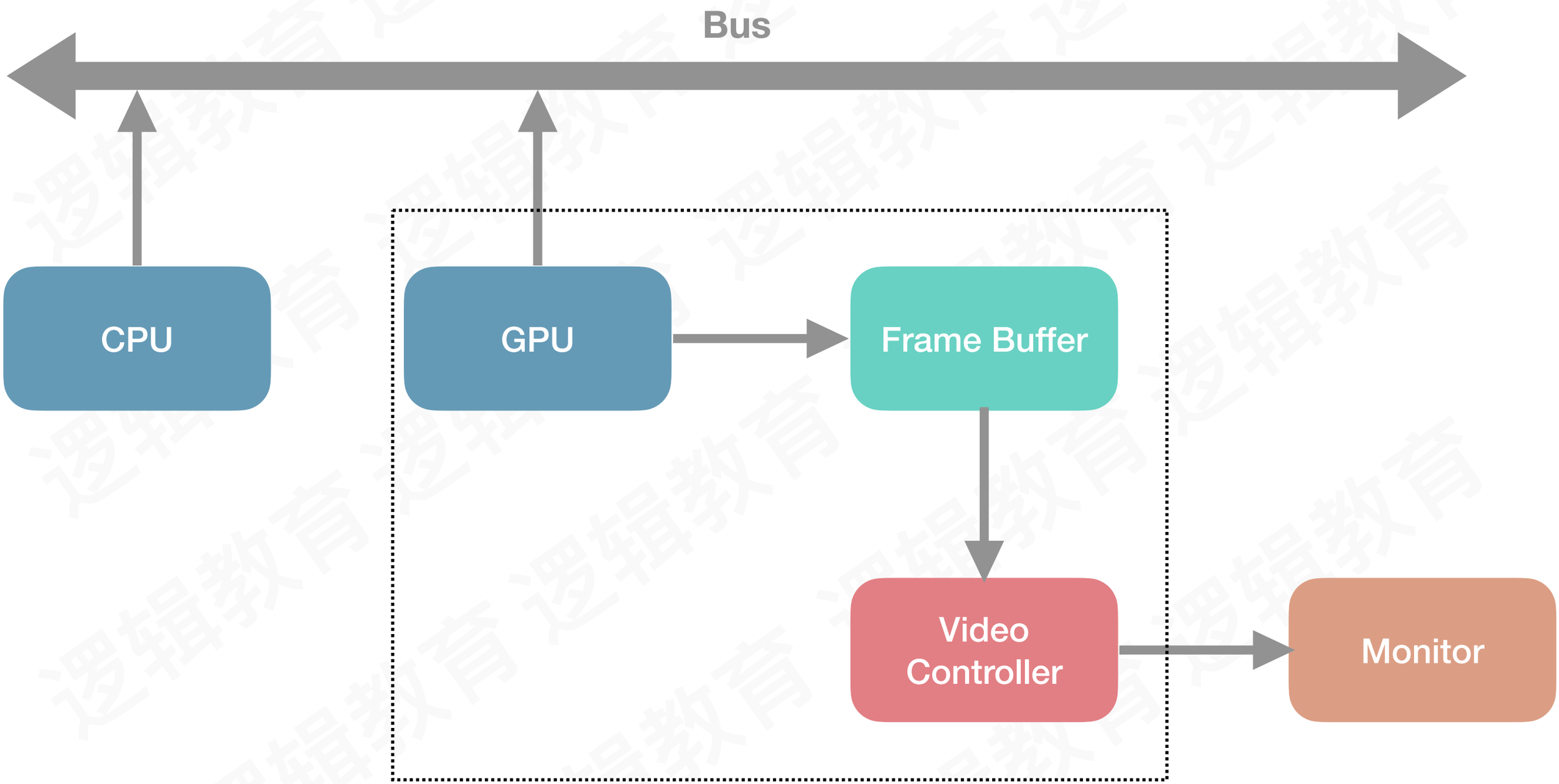
```
CoreFoundation:__CFRUNLOOP_IS_CALLING_OUT_TO_AN_OBSERVER_CALLBACK_FUNCTION__
    QuartzCore:CA::Transaction::observer_callback:
        CA::Transaction::commit();
            CA::Context::commit_transaction();
        CA::Layer::layout_and_display_if_needed();
            if(CA::Layer::layout_if_needed()){
                [CALayer layoutSublayers];
                   [UIView layoutSublayersOfLayer:];
            }else{
                [CALayer display];
                    [CALayer drawInContext:];
                    [UIView drawLayer:inContext:];
                        [UIView drawRect];
            }
```

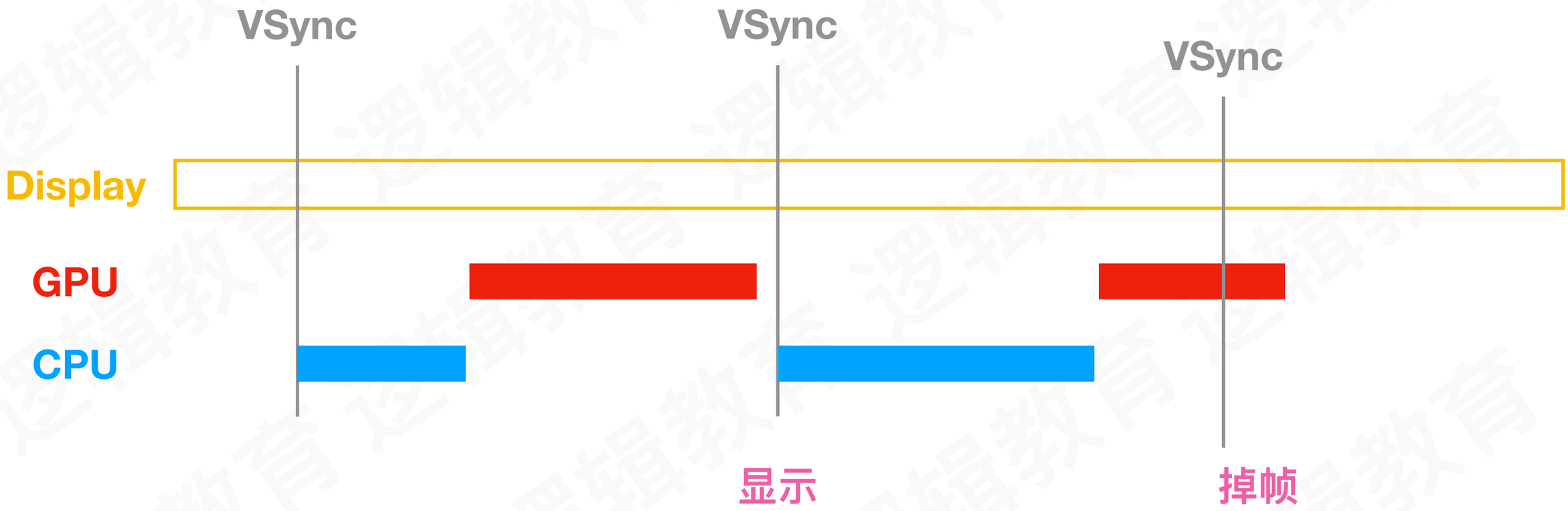
- 空对象：UIView在响应代理时默认会返回一个NSNull对象，表示属性修改后，不实现任何的动作，根据修改后的属性值直接更新视图。

- nil：手动创建并添加到视图上的CALayer或其子类在属性修改时，没有获取到具体的修改行为。此时被修改的属性会被CATransaction记录，最终在下一个runloop的回调中生成动画来响应本次属性修改。由于这个过程非开发者主动完成的，因此这种动画被称作隐式动画

- CAAction的子类：如果返回的是CAAction对象，会直接开始动画来响应图层属性的修改。一般返回的对象多为CABasicAnimation类型，对象中包装了动画时长、动画初始/结束状态、动画时间曲线等关键信息。当CAAction对象被返回时，会立刻执行动作来响应本次属性修改

逻辑教育
Logic education

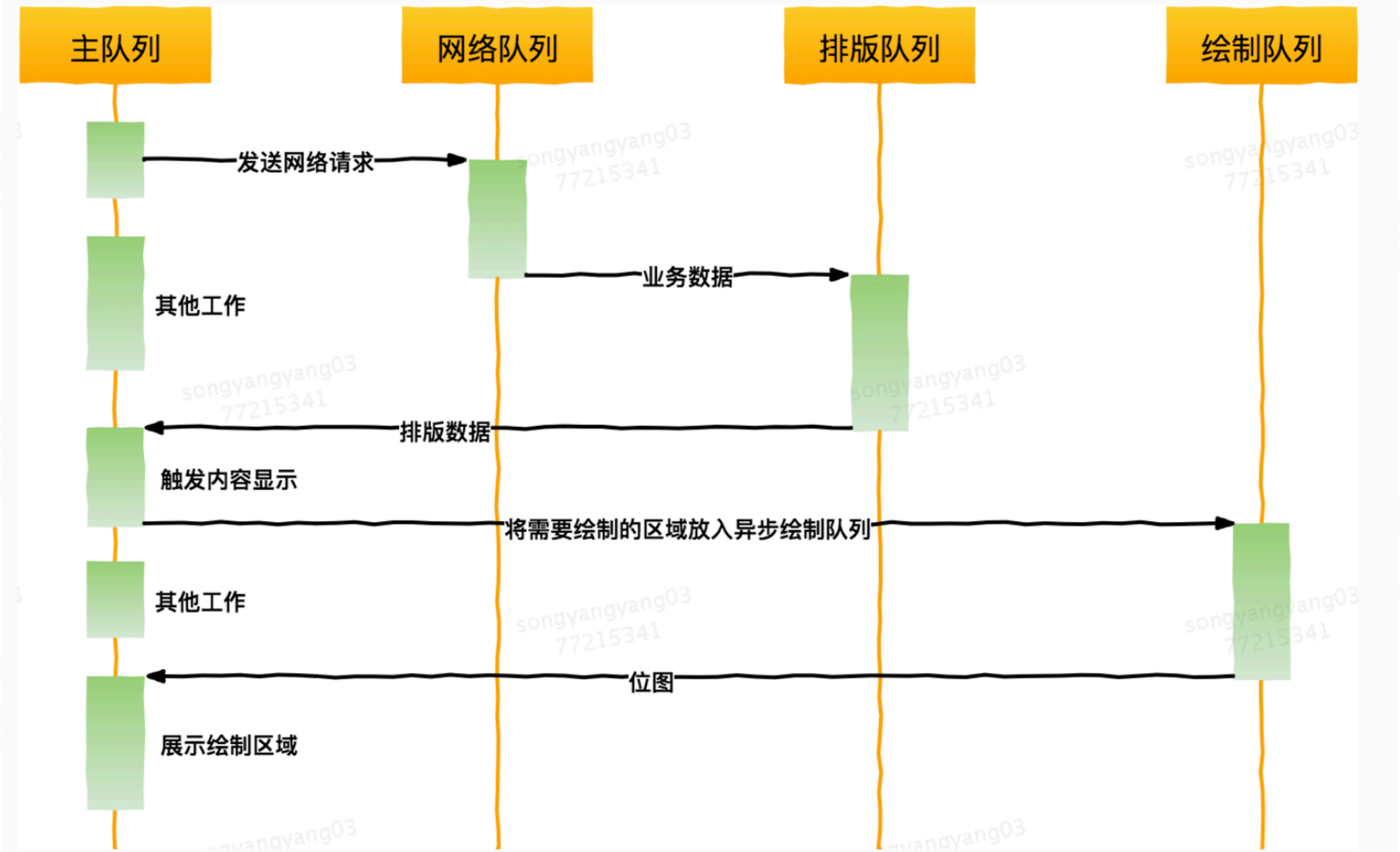




```objc
//按需加载 - 如果目标行与当前行相差超过指定行数，只在目标滚动范围的前后指定3行加载。
- (void)scrollViewWillEndDragging:(UIScrollView *)scrollView withVelocity:(CGPoint)velocity
targetContentOffset:(inout CGPoint *)targetContentOffset{

    //targetContentOffset ： 停止后的contentOffset
    NSIndexPath *ip = [self indexPathForRowAtPoint:CGPointMake(0, targetContentOffset->y)];

    //当前可见第一行row的index
    NSIndexPath *cip = [[self indexPathsForVisibleRows] firstObject];

    //设置最小跨度，当滑动的速度很快，超过这个跨度时候执行按需加载
    NSInteger skipCount = 8;

    //快速滑动(跨度超过了8个cell)
    if (labs(cip.row-ip.row)>skipCount) {

        //某个区域里的单元格的indexPath
        NSArray *temp = [self indexPathsForRowsInRect:CGRectMake(0, targetContentOffset->y, self.width,
self.height)];
        NSMutableArray *arr = [NSMutableArray arrayWithArray:temp];

        if (velocity.y<0) {

            //向上滚动
            NSIndexPath *indexPath = [temp lastObject];

            //超过倒数第3个
            if (indexPath.row+3<datas.count) {
                [arr addObject:[NSIndexPath indexPathForRow:indexPath.row+1 inSection:0]];
                [arr addObject:[NSIndexPath indexPathForRow:indexPath.row+2 inSection:0]];
                [arr addObject:[NSIndexPath indexPathForRow:indexPath.row+3 inSection:0]];
            }

        } else {

            //向下滚动
            NSIndexPath *indexPath = [temp firstObject];
            //超过正数第3个
            if (indexPath.row>3) {
                [arr addObject:[NSIndexPath indexPathForRow:indexPath.row-3 inSection:0]];
                [arr addObject:[NSIndexPath indexPathForRow:indexPath.row-2 inSection:0]];
                [arr addObject:[NSIndexPath indexPathForRow:indexPath.row-1 inSection:0]];
            }
        }
        //添加arr里的内容到needLoadArr的末尾
        [needLoadArr addObjectsFromArray:arr];
    }
}
```

逻辑教育
Logic education

Graver 渲染流程



@Kody

图片加载流程

逻辑教育
Logic education

View

UIImageView

Model

UIImage

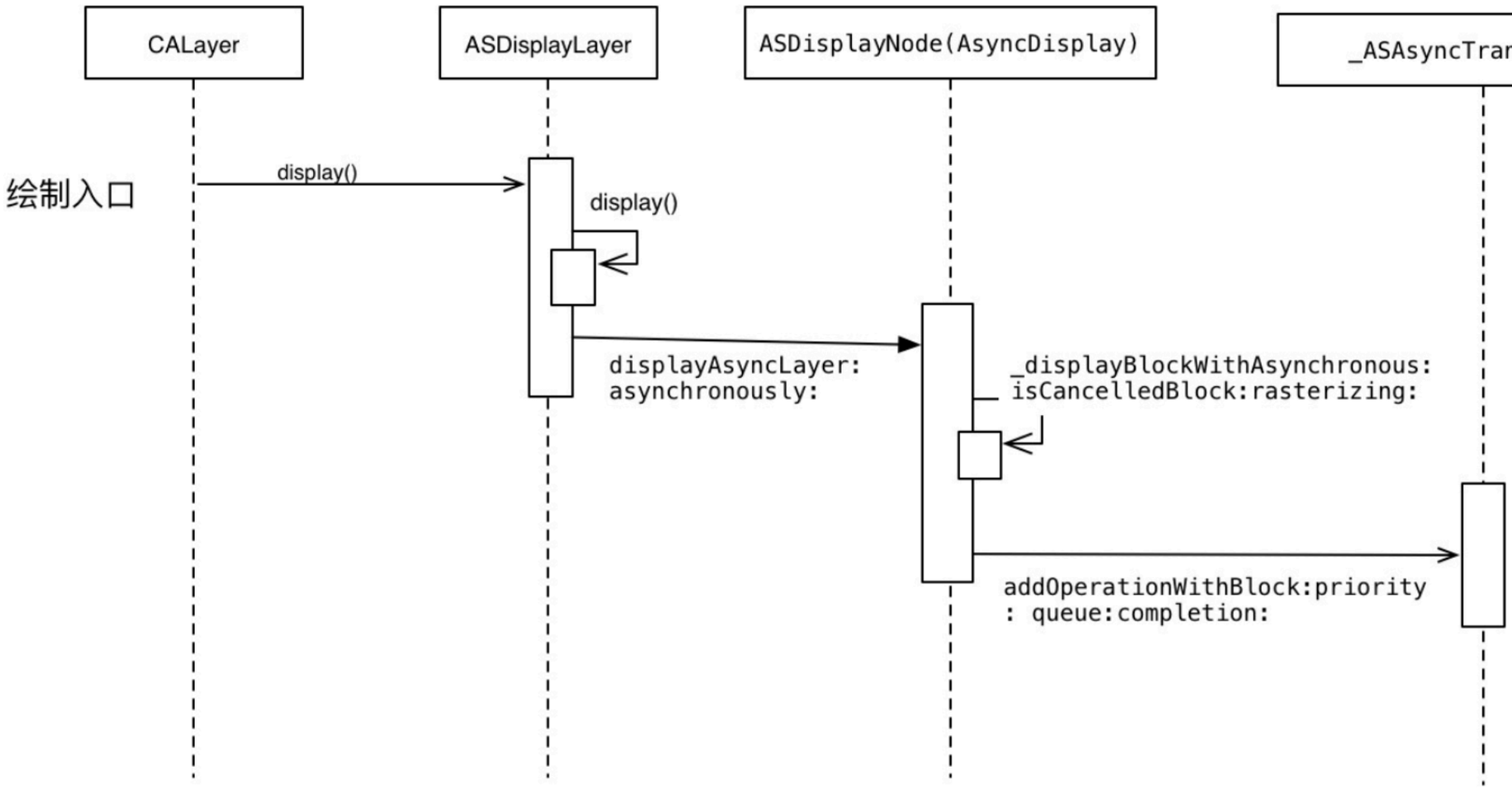
Data Buffer

image Buffer

Controller

ViewController

@Kody



逻辑教育
Logic education

作业

结合今天的知识点，从你的 App 中找出复杂页面来进行优化~
总结你的心得

@Kody

逻辑教育
Logic education

Hello Coder

当你不能坚持时,想想你身边还有很多人和你一样努力!

@Kody