

# 内存

## 第 2 个问题：内存泄漏

对于横向 (ViewController 之间) 数据传输, 现在通过 JDRScene (相当于横向 Pipeline) 来实现, 在跳转下一页面前, 当前页面调用 startScene: 方法开启一个 scene, 在 startScene: 方法里通过调用 JDRScene 的 jd\_addController 和 jd\_observer 方法实现当前 ViewController 对 scene 属性的监听, 每当这个时候 Instruments 的 Leaks 工具会报一个新的内存泄漏, 如下图...

# 内存

Details > Leaks > Leaks by Backtrace Instrument Detail

Leaked Object	#	Address	Size	Responsible Libr...	Responsible Frame
Malloc 48 Bytes	1	0x60800084c6...	48 Bytes	StockGod	-[JDRScene jd_observer]
Malloc 48 Bytes	1	0x608000849540	48 Bytes	StockGod	-[JDRScene jd_observer]

Stack Trace

- malloc\_zone\_malloc
- malloc
- class\_copyPropertyList
- [JDRScene jd\_observer]
- [JDRScene jd\_addController:]
- [JDRBaseViewController setScene:]
- [JDRBaseViewController startScene:]
- \_\_42-[SGWarrenListViewController addObserver:]\_block\_invoke
- [NSObject(KVO) observeValueForKeyPath:ofObject:change:con...
- NSKeyValueNotifyObserver
- NSKeyValueDidChange
- [NSObject(NSKeyValueObservingPrivate) \_changeValueForKey...
- [NSObject(NSKeyValueObservingPrivate) \_changeValueForKey:...
- \_NSSetObjectValueAndNotify
- [SGWarrenListView tableView:didSelectRowAtIndexPath:]
- [UITableView \_selectRowAtIndexPath:animated:scrollPosition:n...
- [UITableView \_userSelectRowAtIndexPath:]
- \_runAfterCACommitDeferredBlocks
- \_cleanUpAfterCAFlushAndRunDeferredBlocks
- \_afterCACommitHandler
- \_CFRunLoop\_IS\_CALLING\_OUT\_TO\_AN\_OBSERVER\_CALLBAC...
- \_CFRunLoopDoObservers