对象方法消息转发：<https://www.bilibili.com/video/av61343372?from=search&seid=6031356826523251613>

OC：消息机制语言

消息机制例子：2 + 3: 给2发送消息（方法+），给他一个参数3

[[Person new] sendMessage:@”hello”]

背后编译成objc\_msgSend([Person new], @selector(sendMessage:),@”hello”)

对象、类本质的了解：类的本质是结构体

objc\_object结构体有一个成员变量isa

先找objc\_Class 中的Method\_array\_t里面有类的方法列表

再找superClass中的method\_array

…

找NSObject，如果还没有找到，走**消息转发流程**

<https://www.jianshu.com/p/a0dc0364d717>

A close up of a map

Description automatically generated

消息转发流程：

ResolveInstanceMethod 动态方法解析（动态添加一个方法）

动态方法解析的意思就是，征询消息接受者所属的类，看其是否能动态添加方法，以处理当前“这个未知的选择子（unknown selector）“。实例对象在接受到无法解读的消息后，首先会调用其所属类的下列类方法:

+ (BOOL)resolveInstanceMethod:(SEL)selector

类对象在接受到无法解读的消息后，那么运行期系统就会调用另外的一个方法，如下：

+ (BOOL)resolveClassMethod:(SEL)selector

如果运行期系统已经执行完了动态方法解析，那么消息接受者自己就无法再以动态新增方法的形式来响应包含该未知选择子的消息了，此时就进入了第二阶段——完整的消息转发。运行期系统会请求消息接受者以其他手段来处理与消息相关的方法调用。

ForwardingTargetForSelector快速转发（找到备用接收者处理当前的消息）

当前接受者如果不能处理这条消息，运行期系统会请求当前接受者让其他接受者处理这条消息，与之对应的方法是：

- (id)forwardingTargetForSelector:(SEL)selector

方法参数代表未知的选择子，返回值为备援接受者，若当前接受者能找到备援接受者，就直接返回，这个未知的选择子将会交由备援接受者处理。如果找不到备援接受者，就返回nil，此时就会启用“完整的消息转发机制”。

ForwardingInvocation慢速转发（把信息封装到NSInvocation，丢出去找到更多能处理当前消息的接受者）

A screenshot of a cell phone

Description automatically generated

method和SEL和IMP的关系

method\_t结构体

A close up of text on a screen

Description automatically generated

方法包含方法编号(SEL)和IMP

（方法签名：可以获得返回值和参数类型 / 方法编号：不知道返回值和参数类型）

forwardInvocation：丢漂流瓶，消息中转中心

**isa：**每个Objective-C对象都有一个隐藏的数据结构，这个数据结构是Objective-C对象的第一个成员变量，它就是isa指针。一个对象（Object）的isa指向了这个对象的类（Class），而这个对象的类（Class）的isa指向了meta class。这样我们就可以找到静态方法和变量了

类和元类：

对象（方法、属性、协议）存储在objc\_class（method\_array, property\_array）

类对象（类方法，类属性）存储在元类中