# 类图

一：符号

****·**** + ：表示public

****·**** - ：表示private

****·**** #：表示protected（friendly也归入这类）

****·~**** ：**省略这些修饰符表示具有package**

****·\_**** ：**下划线**表示static

****·** 斜体：表示抽象类**

二：类

分为3个部分：名称部分（Name）、属性部分（Attribute）和操作部分（Operation）

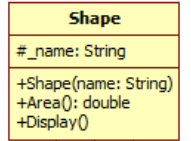
类的属性格式：

name : attribute type

类的操作格式：

name (parameter list) : type of value returned

例如：



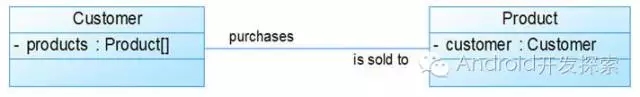
三：关系

1.单向关联关系



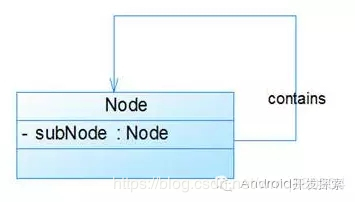
我们可以看到，在UML类图中单向关联用一个带箭头的直线表示。上图表示每个顾客都有一个地址，这通过让Customer类持有一个类型为Address的成员变量类实现。

2.双向关联关系



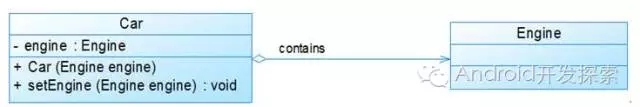
从上图中我们很容易看出，所谓的双向关联就是双方各自持有对方类型的成员变量。在UML类图中，双向关联用一个不带箭头的直线表示。上图中在Customer类中维护一个Product[]数组，表示一个顾客购买了那些产品；在Product类中维护一个Customer类型的成员变量表示这个产品被哪个顾客所购买。

3.自关联关系



自关联在UML类图中用一个带有箭头且指向自身的直线表示。上图的意思就是Node类包含类型为Node的成员变量，也就是“自己包含自己”。

4.聚合关系



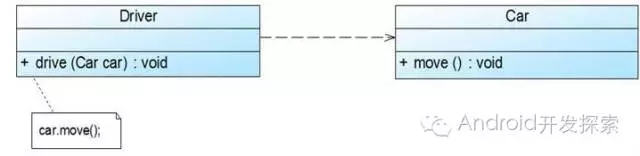
上图中的Car类与Engine类就是聚合关系（Car类中包含一个Engine类型的成员变量）。由上图我们可以看到，UML中聚合关系用带空心菱形和箭头的直线表示。聚合关系强调是“整体”包含“部分”，但是“部分”可以脱离“整体”而单独存在。比如上图中汽车包含了发动机，而发动机脱离了汽车也能单独存在。

5.组合关系



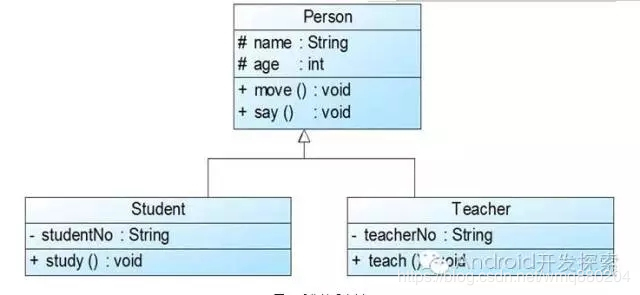
组合关系与聚合关系见得最大不同在于：这里的“部分”脱离了“整体”便不复存在，显然嘴是头的一部分且不能脱离了头而单独存在。在UML类图中，组合关系用一个带实心菱形和箭头的直线表示。

6.依赖关系



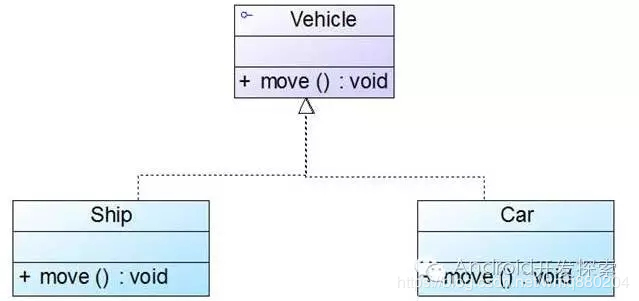
从上图我们可以看到，Driver的drive方法只有传入了一个Car对象才能发挥作用，因此我们说Driver类依赖于Car类。在UML类图中，依赖关系用一条带有箭头的虚线表示。

7.继承关系



继承关系对应的是extend关键字，在UML类图中用带空心三角形的直线表示，如下图所示中，Student类与Teacher类继承了Person类。

1. 8.实现关系



这种关系对应implement关键字，在UML类图中用带空心三角形的虚线表示。如下图中，Car类与Ship类都实现了Vehicle接口。

# 时序图

1.角色（Actor）

系统角色，可以是人、机器、其他系统、子系统；以一个小人图标表示。



2.对象（Object）



2.1对象的三种命名方式

第一种方式包括对象名和类名，例如：华为手机:手机、loginServiceObject:LoginService。

第二种方式只显示类名，即表示它是一个匿名对象，例如：:手机、:LoginSservice。

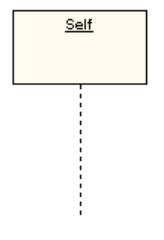
第三种方式只显示对象名不显示类名，例如：华为手机:、loginServiceObject:。

2.2对象的排列顺序

通常应遵循以下两个原则：把交互频繁的对象尽可能的靠拢；2.把初始化整个交互活动的对象放置在最左端。

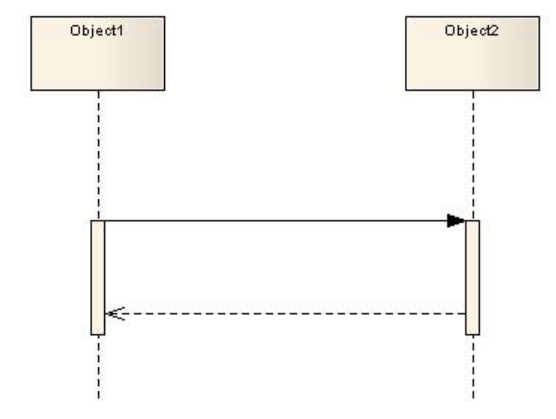
3. 生命线（Lifeline）

在时序图中表示为从对象图标向下延伸的一条虚线，表示对象存在的时间。



4. 控制焦点（Focus of Control）

又称为激活期，表示时间段的符号，在这个时间段内对象将执行相应的操作，用小矩形表示。

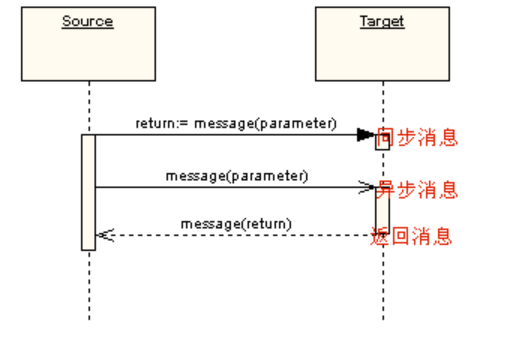


5. 消息（Message）

同步消息  实线+加粗箭头

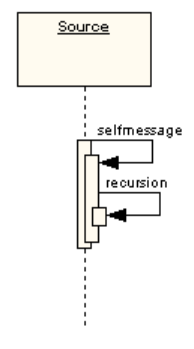
异步消息 实线+箭头

返回消息 虚线+箭头



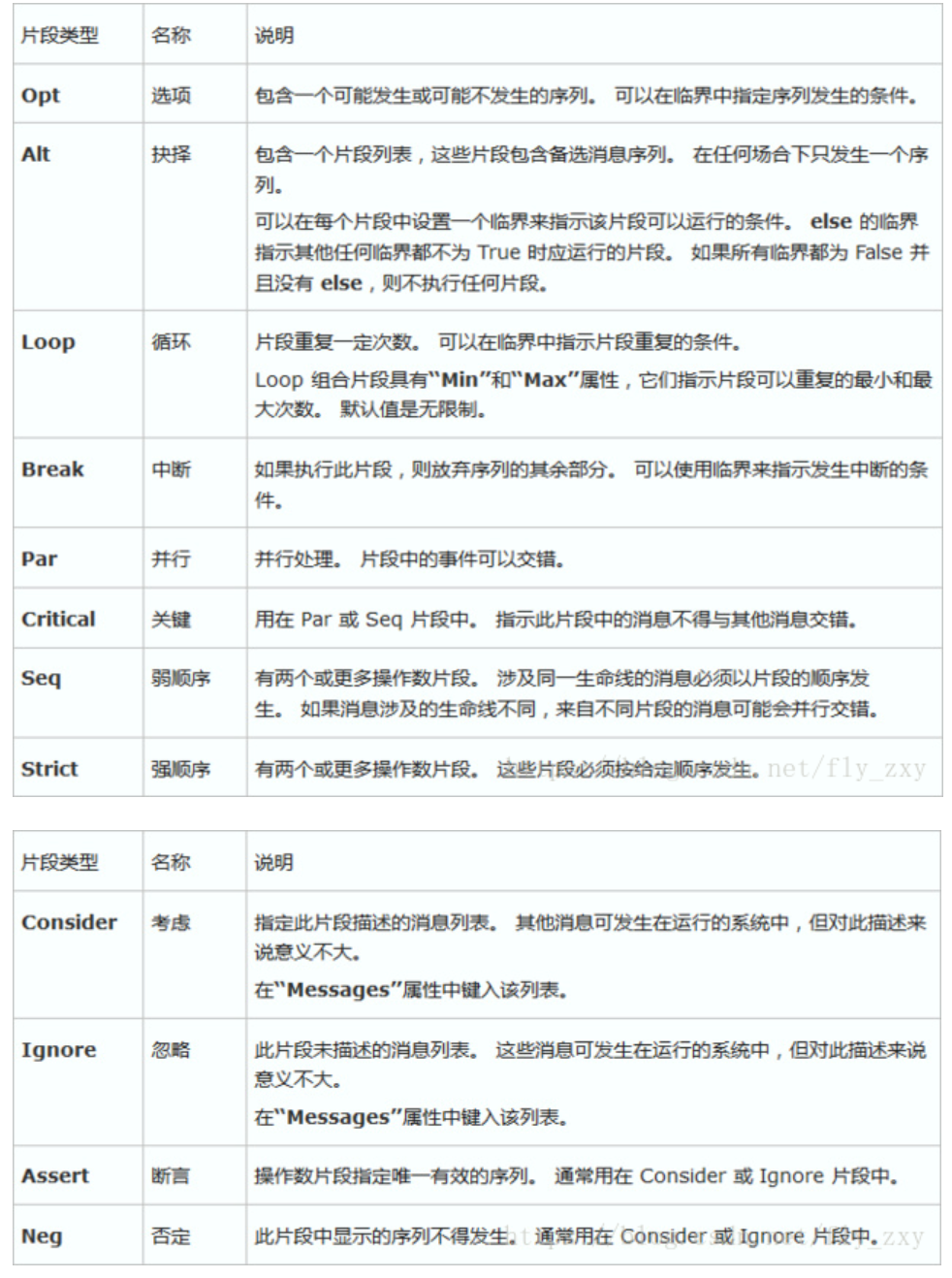
6. 自关联消息

表示方法的自身调用或者一个对象内的一个方法调用另外一个方法。以一个半闭合的长方形+下方实心剪头表示。



7. 组合片段

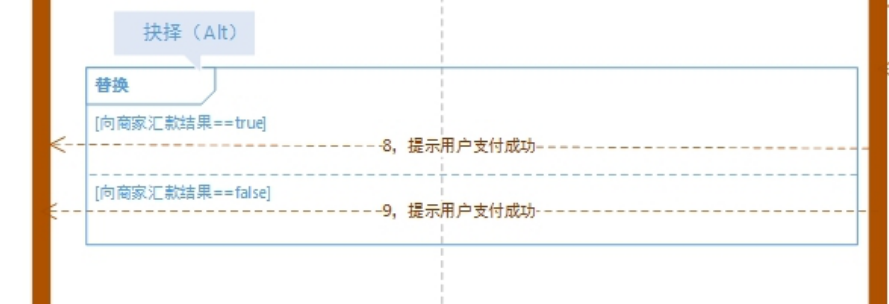
组合片段用来解决交互执行的条件和方式，它允许在序列图中直接表示逻辑组件，用于通过指定条件或子进程的应用区域，为任何生命线的任何部分定义特殊条件和子进程。组合片段共有13种，名称及含义如下：



### 7.1常用组合片段举例

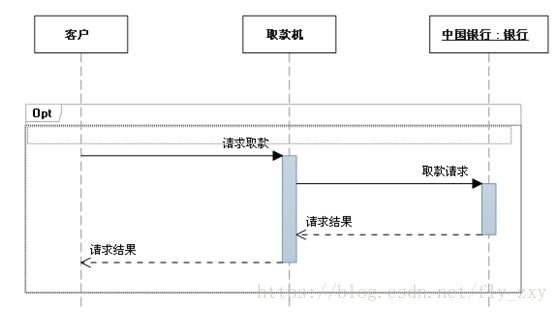
抉择（Alt）

抉择在任何场合下只发生一个序列。 可以在每个片段中设置一个临界来指示该片段可以运行的条件。else 的临界指示其他任何临界都不为 True 时应运行的片段。如果所有临界都为 False 并且没有 else，则不执行任何片段。Alt片段组合可以理解为if..else if...else条件语句。



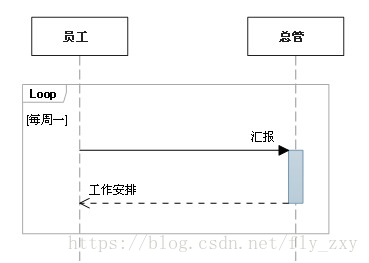
选项（Opt）

包含一个可能发生或不发生的序列。Opt相当于if..语句。



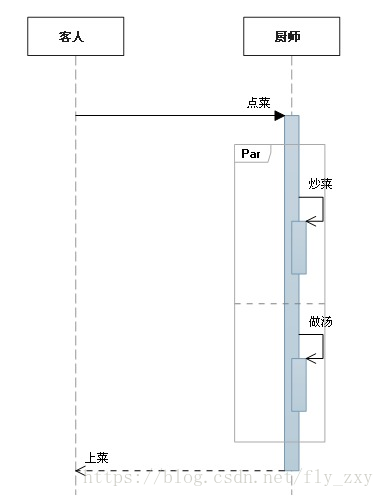
循环（Loop）

片段重复一定次数，可以在临界中指示片段重复的条件。Loop相当于for语句。



并行（Par）

并行处理，片段中的事件可以并行交错。Par相当于多线程。

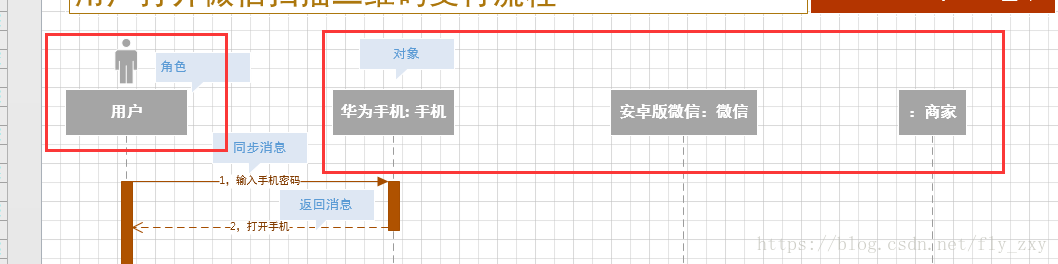


8.时序图的绘制技巧：

1，从初始消息开始画，依次画出随后消息，并给每个消息分配序号，方便理解。

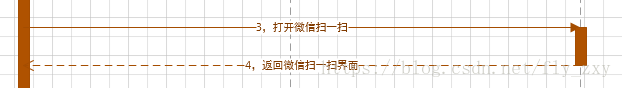
2，角色和对象用名词，消息用动词。

3，角色放在时序图的开始位置，对象重要程度或使用频率从左到右排列。这就要根据时间的流程考虑了，是一个比较主观的事情。

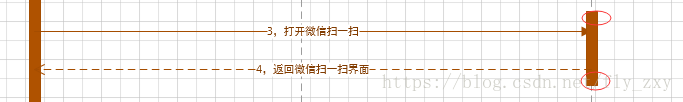


4，控制焦点两端要以消息元素封顶，控制焦点不要超过消息元素。

****正确示范****

****

****错误示范****

****