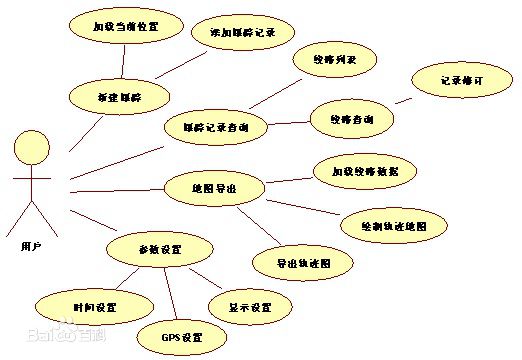
# 用例图

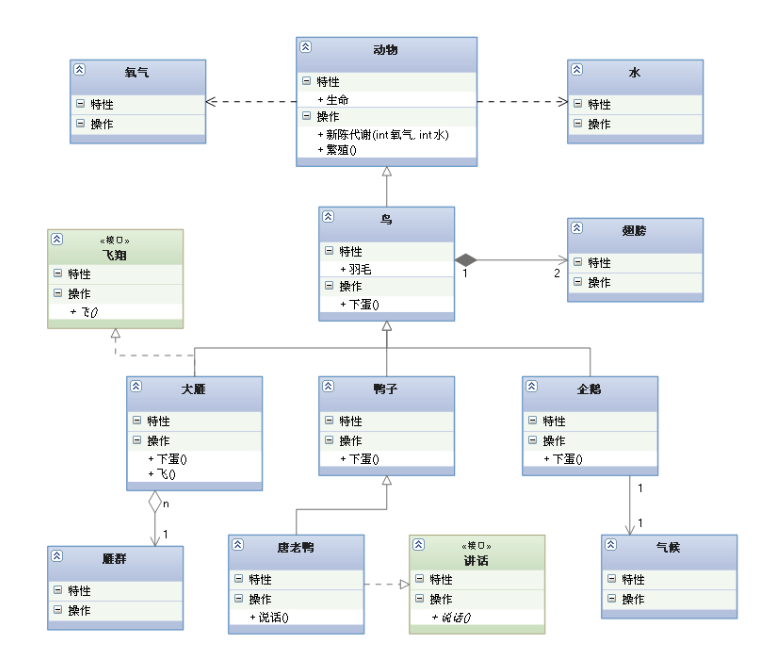
用例图是指由参与者（Actor）、用例（Use Case），边界以及它们之间的关系构成的用于描述系统功能的视图。

用例图由参与者（Actor）、用例（Use Case）、系统边界、箭头组成，用画图的方法来完成。



# 类图

　用户根据用例图抽象成类，描述类的内部结构和类与类之间的关系，是一种静态结构图。 在UML类图中，常见的有以下几种关系: 泛化（Generalization）, 实现（Realization），关联（Association)，聚合（Aggregation），组合(Composition)，依赖(Dependency)。 各种关系的强弱顺序： 泛化 = 实现 > 组合 > 聚合 > 关联 > 依赖



类图中类包含 类名和属性值

## 2.1.泛化

　　　　【泛化关系】：是一种继承关系，表示一般与特殊的关系，它指定了子类如何继承父类的所有特征和行为。例如：老虎是动物的一种，即有老虎的特性也有动物的共性。

实线箭头 表示



## 2.2.实现

　　　　【实现关系】：是一种类与接口的关系，表示类是接口所有特征和行为的实现。

虚线箭头



## 2.3关联

【关联关系】：是一种拥有的关系，它使一个类知道另一个类的属性和方法；如：老师与学生，丈夫与妻子关联可以是双向的，也可以是单向的。双向的关联可以有两个箭头或者没有箭头，单向的关联有一个箭头。



## 2.4.聚合

　【聚合关系】：是整体与部分的关系，且部分可以离开整体而单独存在。如车和轮胎是整体和部分的关系，轮胎离开车仍然可以存在。

聚合关系是关联关系的一种，是强的关联关系；关联和聚合在语法上无法区分，必须考察具体的逻辑关系。

空心菱形箭头+箭头



## 2.5.组合

　　　　【组合关系】：是整体与部分的关系，但部分不能离开整体而单独存在。如公司和部门是整体和部分的关系，没有公司就不存在部门。

　　组合关系是关联关系的一种，是比聚合关系还要强的关系，它要求普通的聚合关系中代表整体的对象负责代表部分的对象的生命周期。

实心菱形箭头



## 2.6.依赖

　　【依赖关系】：是一种使用的关系，即一个类的实现需要另一个类的协助，所以要尽量不使用双向的互相依赖.

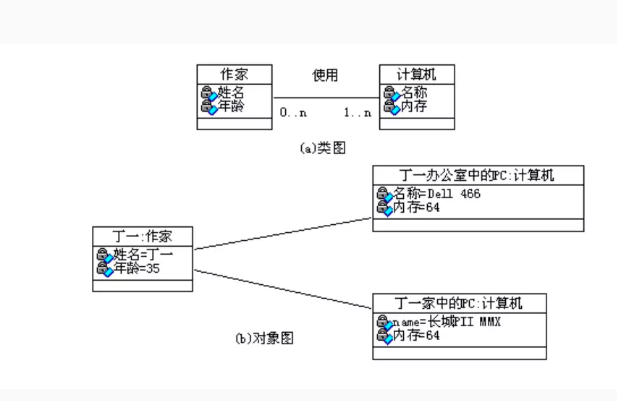
【代码表现】：局部变量、方法的参数或者对静态方法的调用



# 对象图

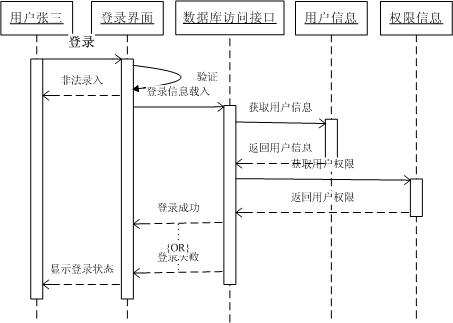
对象图(Object Diagram) 是显示了一组对象和他们之间的关系。使用对象图来说明数据结构，类图中的类或组件等的实例的静态快照。

对象图是类图的实例，几乎使用与类图完全相同的标识。他们的不同点在于对象图显示类的多个对象实例，而不是实际的类。一个对象图是类图的一个实例。由于对象存在生命周期，因此对象图只能在系统某一时间段存在。



# 序列图

交互图的一种，描述了对象之间消息发送的先后顺序，强调时间顺序。

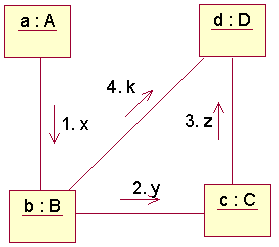


# 协作图（通信图）

而“协作”作为一个结构事物用于表达静态结构和动态行为的概念组合，表达不同事物相互协作完成一个复杂功能。故UML 2.0以后通信图不再是协作图，没有专门的”协作图“，只有”协作“。

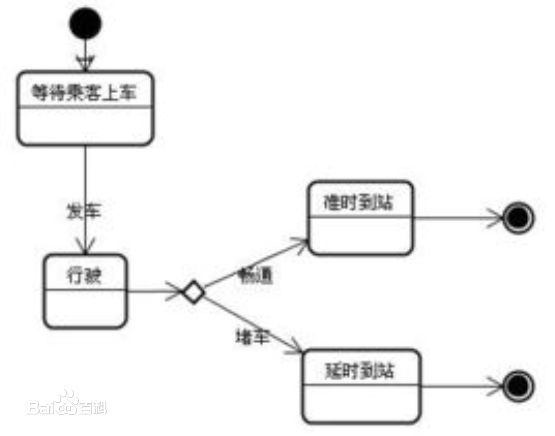
框中的名称带有“：”表示一个对象 前半部分为对象名，后半部分为类名

对象之间的箭头表示的对象之间的通信的消息



# 状态图

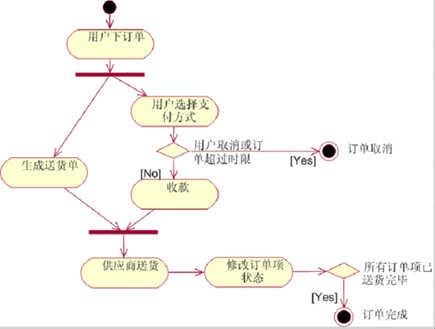
状态图(Statechart Diagram)是描述一个实体基于事件反应的动态行为，显示了该实体如何根据当前所处的状态对不同的事件做出反应。通常我们创建一个UML状态图是为了以下的研究目的：研究类、角色、子系统、或组件的复杂行为。

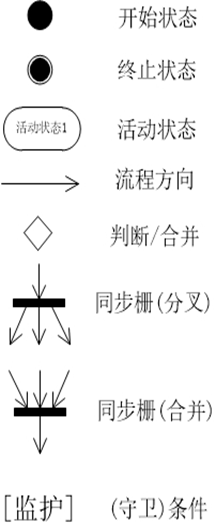


# 活动图 （动态图）

活动图 是状态图的一种特殊情况，这些状态大都处于活动状态。本质是一种流程图，它描述了活动到活动的控制流。

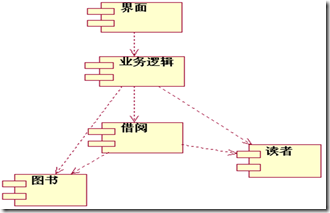
交互图强调的是对象到对象的控制流，而活动图则强调的是从活动到活动的控制流。





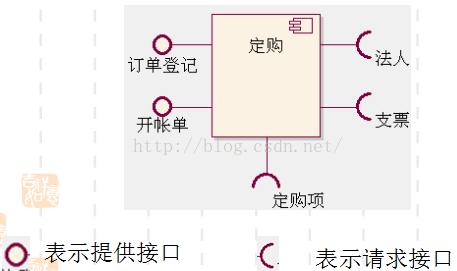
# 构件图

构件图主要用于描述各种软件构件之间的依赖关系，例如，可执行文件和源文件之间的依赖关系，所设计的系统中的构件的表示法及这些构件之间的关系构成了构件图。构件图由一边有两个小矩形的一个长方形表示



供接口用“棒棒糖”式的图形表示，即由一个封闭的圆形与一条直线组成。

需接口用“插座”式的图形表示，即由一个半圆与一条直线组成。



# 部署图

描述了系统运行时进行处理的结点以及在结点上活动的构件的配置。强调了物理设备以及之间的连接关系。

部署模型的目的：

描述一个具体应用的主要部署结构，通过对各种硬件，在硬件中的软件以及各种连接协议的显示，可以很好的描述系统是如何部署的；平衡系统运行时的计算资源分布；可以通过连接描述组织的硬件网络结构或者是嵌入式系统等具有多种硬件和软件相关的系统运行模型。

