

UML

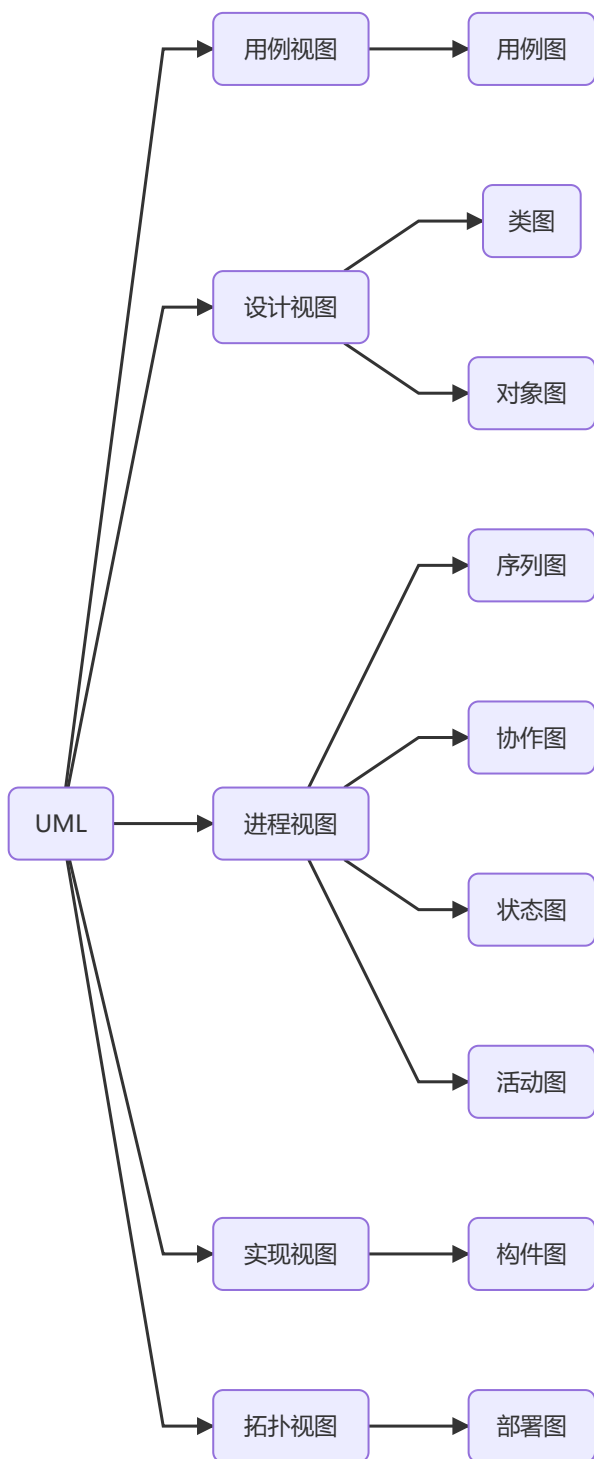
UML (Unified Modeling Language) 是一种统一建模语言，为面向对象开发系统的产品、说明、可视化和编制文档的一种标准语言。

一、基本概念

UML图分为用例视图、设计视图、进程视图、实现视图和拓扑视图，又可以动静划分为静态视图和动态视图。

静态视图分为：用例图、类图、对象图、包图、构件图、部署图。

动态图分为：状态图、活动图、协作图，序列图。



1、用例图

用例图主要回答了两个问题：1、是谁用软件。2、软件的功能。从用户的角度描述了系统的功能，并指出各个功能的执行者，强调软件的使用者，系统为执行者完成哪些功能。

2、类图

用户根据用例图抽象成类，描述类的内部结构和类和类之间的关系，是一种静态结构。在UML类图种，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）。

各种关系的强弱关系：泛化 = 实现 > 组合 > 聚合 > 关联 > 依赖。

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，既有老虎的特性也有动物的共性。

【箭头及指向】：带三角形的实线，三角形指向父类。

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。

【箭头及指向】带三角形的虚线，三角形指向接口。

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：老师和学生，关联关系可以使双向的，也可以使单向的。

【箭头及指向】：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】：是整体和部分的关系，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。

【箭头及指向】：带实线的空心菱形，菱形指向整体。

2.5 组合关系

【组合关系】：是整体和部分的关系，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，没有公司就没有部门。

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种代表整体的对象负责代表部分的对象的生命周期。

【箭头及指向】：带实心菱形的实线，菱形指向整体。

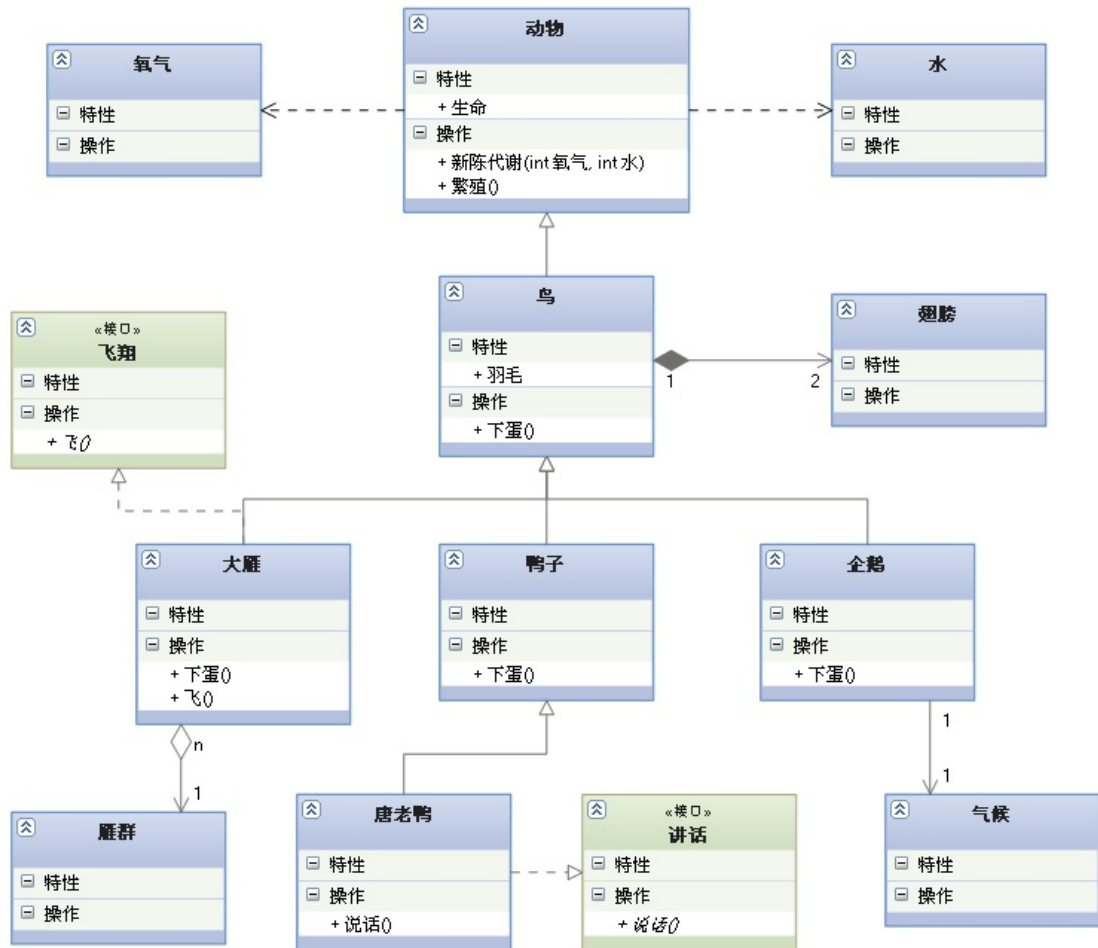
2.6 依赖

【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。

【代码实现】：局部变量，方法的参数或对静态方法的调用。

【箭头及指向】：带箭头的虚线，指向被使用者。

2.7 各种类图的关系



3、对象图

描述的是参与交互的各个对象在交互过程中某一时刻的状态。对象图可以被看做是类图在某一时刻的实例。

4、状态图

是一种由状态、变迁、事件和活动组成的状态机。用来描述类的对象所有可能的状态以及时间发生时状态的转移条件。

5、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。

活动图是一种表述过程基理、业务过程以及工作流的技术。它可以用来对业务过程、工作流建模，也可以对用例实现甚至是程序实现来建模。

5.1 带甬道的活动图

泳道表明每个活动是由哪些人或哪些部门负责完成。

5.2 带对象流的活动图

用活动图描述某个对象时，可以把设计到的对象放置在活动图中，并用一个依赖将其连接到进行创建、修改和撤销的动作状态或活动状态上，对象的这种使用方法就构成了对象流。对象流用带有箭头的虚线表示。

6、序列图-时序图

交互图的一种，描述了对对象之间消息发送的先后顺序，强调时间顺序。

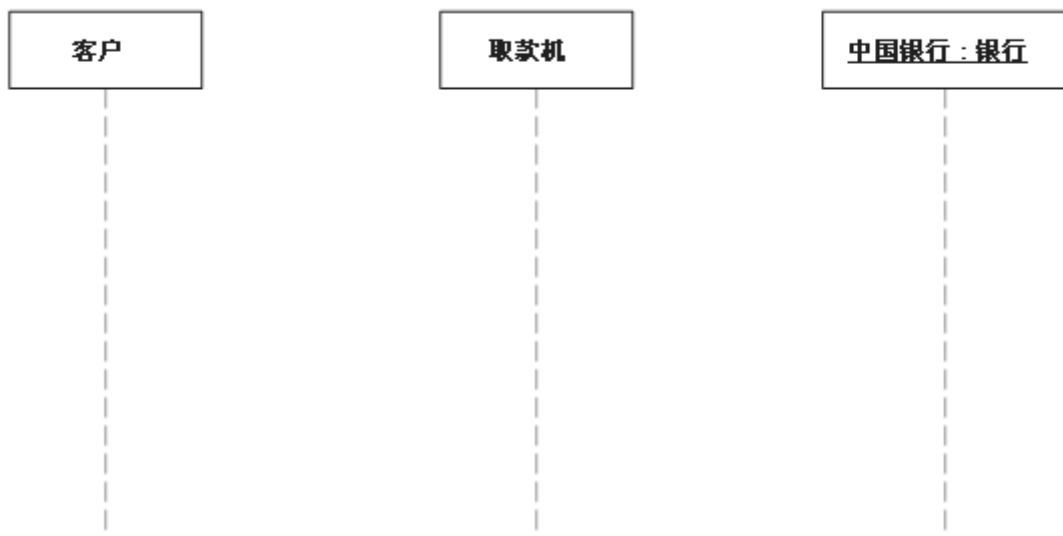
序列图的主要用途是把用例表达的需求，转化为进一步，更加正式层次的精细表达。用例常常被细化为一个或更多的序列图。同时序列图更有效地描述如何分配各个类的职责以及各个类具有相应职责的原因。

消息用从一个对象的生命线到另一个对象生命线的箭头表示。箭头以时间顺序在图中从上到下排列。

序列图中涉及的元素：

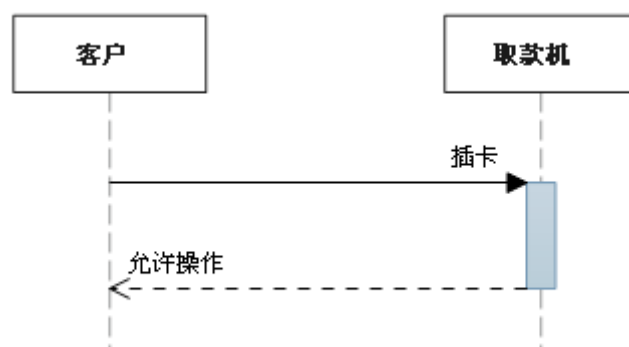
6.1 生命线

生命线名称可带下划线。当使用下划线时，意味着序列图中的生命线代表一个类特定实例。



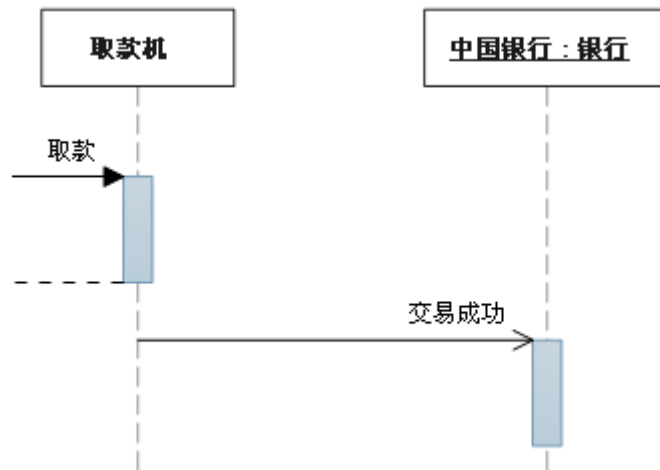
6.2 同步消息

同步等待消息。

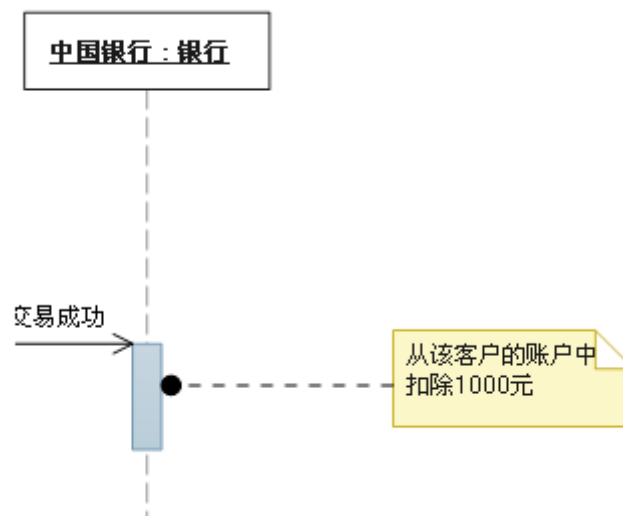


6.3 异步消息

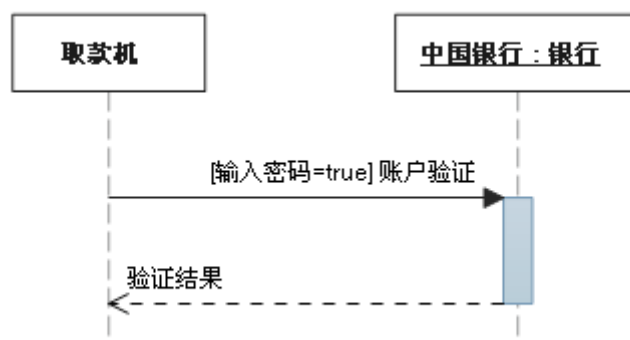
异步发送消息，不需要等待。



6.4 注释



6.5 约束



6.6 组合

组合片段用来解决交互的条件及方式。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。常用的组合片段有：抉择、选项、循环、并行。

7、协作图

交互图的一种，描述了收发消息的对象组织关系，强调对象之间的合作关系。时序图按照时间顺序布图，而协作图按照空间结构布图。

8、构件图

构件图是用来表示系统中构件与构件之间，类或接口与构件之间的关系图。其中，构件图之间的关系表现为依赖关系，定义的类或接口与类之间的关系表现为依赖关系或实现关系。

9、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。强调物理设备以及之间的连接关系。

[以上内容来自知乎](#)

[以下内容来自](#)

UML图表可大致分为结构性图表和行为性图表两种。

结构性图表显示了系统在不同抽象层次和实现层次上的静态结构以及它们之间的相互关系。结构性图表中的元素表示系统中具有意义的概念，可以包括抽象的、现实的和实作的概念。

结构性图表有七种类型

1. 类图 (Class Diagram)
2. 组件图 (Component Diagram)
3. 部署图 (Deployment Diagram)
4. 对象图 (Object Diagrams)
5. 包图 (Package Diagram)
6. 复合结构图 (Composite Structure Diagram)
7. 轮廓图 (Profile Diagram)

行为性图表显示了系统中对象的动态行为，可以表达系统随时间的变化。

行为性图表有七种类型

1. 用例图 (Use Case Diagram)
2. 活动图 (Activity Diagram)
3. 状态机图 (State Machine Diagram)
4. 序列图 (Sequence Diagram)
5. 通讯图 (Communication Diagram)
6. 交互概述图 (Interaction Overview Diagram)
7. 时序图 (Timing Diagram)