

前一节我们学到了用来表示对象交互顺序的时序图，那么有没有图形可以表示对象的状态变化呢？

答案是：有，它就是**状态机图**。

《手册》的设计规约章节有对状态图的规定：

【强制】如果某个业务对象的状态超过 3 个，使用状态图来表达并明确状态变化变化的各个触发条件。

那么我们思考下面几个问题：

- 什么是状态机图？
- 状态机图的使用场景是什么？
- 如何画状态机图？

2.1 概念

状态机图也叫状态图、状态转换图、简单状态图，它描述一个事物随着事件的状态变化。

简而言之，**状态机图主要描述对象的状态以及引起对象状态变换的事件**。

状态机图可以对对象、用例甚至整个系统的行为建模。

大多数面向对象技术都适用状态机图来描述一个对象在其生命周期中的状态变化。

2.2 使用场景

在业务建模时，可以创建状态机图对用例场景建模（和手册描述的情况类似）。

在分析和设计时，可以建模事件驱动对象；我们还可以使用多个状态机图来表现同一个状态机和行为的不同方面。

3.1 核心组件

为了更全面地描述核心组件的构成，本小节采用 visual paradigm 官网状态机图组件给大家讲解分析。

3.1.1 状态

状态表示一个对象生命周期中的一种条件，它可以满足执行某些活动的条件后者等待接收某些事件。

状态包括 5 个部分：

- 状态名：状态的名称
- 进入（）：进入状态的行为
- 行为（do）：在进入状态时执行的行为
- 退出行为（）：离开状态时执行的操作

- 延迟触发 (Deferrable Trigger) : 暂时不触发状态变更的行为

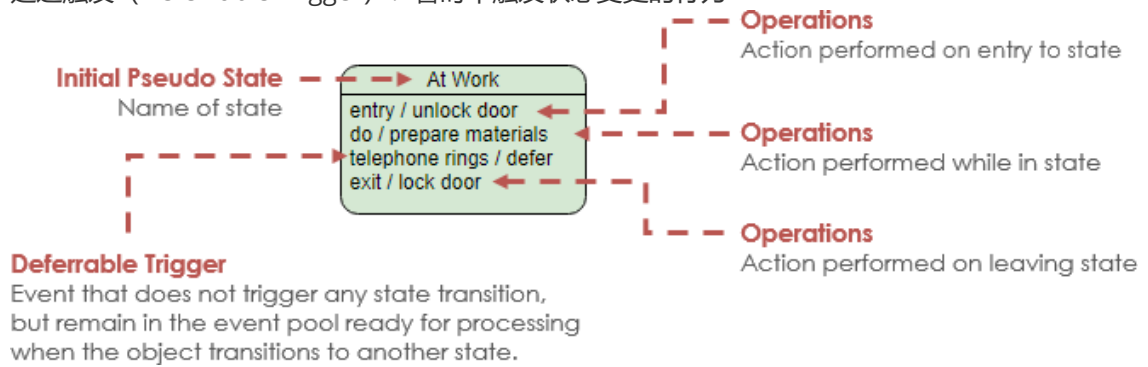


图 1 : 状态 (图片来自 visual paradigm)

3.1.2 转换

转换是指前一个状态到后一个状态经历的事件。转换前的状态称为源状态, 转变后的状态称为目标状态。

转化分为 5 个部分:

- 源状态: 受变换影响的状态
- 事件触发器: 一种可以触发源状态以满足保护条件的机制
- 保护条件: 在接收事件触发器转换时需要计算的布尔表达式
- 操作: 可执行的原子计算, 可以直接作用于拥有状态机的对象, 并间接作用于对象可见的其他对象
- 目标状态: 转换完成后所处的状态

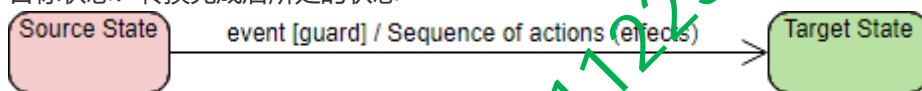


图 2 : 转换 (图片来自 visual paradigm)

3.1.3 决策节点

分节点 (Fork node) 是一种为状态, 用来表示进入的转换将分为两个或者多个目标状态。从 fork 定点传入的流转不允许有守护条件或者触发器, 至少有两个或两个以上的转出条件。

合节点 (Join node) 也是一种伪状态, 合并从分支节点转换而来的多个状态, 合节点至少有两个或两个以上的转入和一个转出。

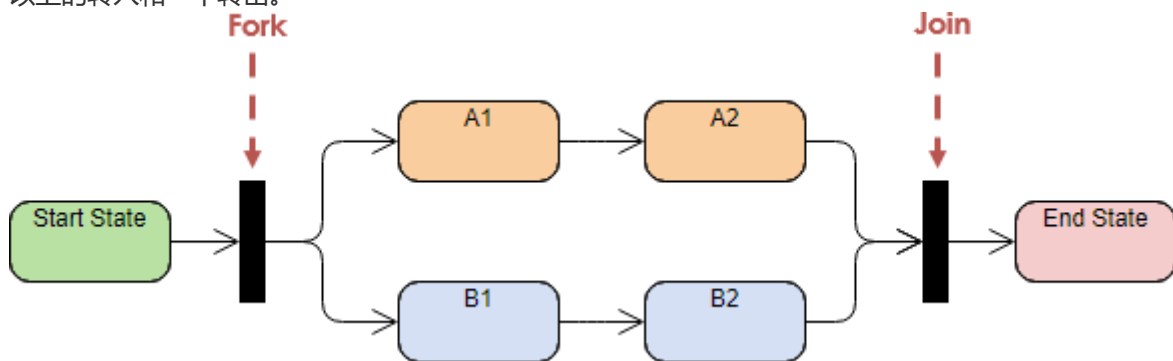


图 3 : fork and join 节点 (图片来自 visual paradigm)

选择 (Choice) 是一种伪状态，通过守护条件判断执行的路径。

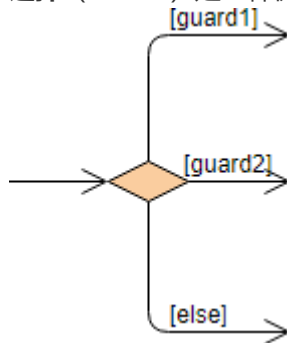


图 4: choice (图片来自 visual paradigm)

如下图所示下单时，如果库存足够则进入确认订单状态，如果库存不足或用户取消，则进入取消状态。

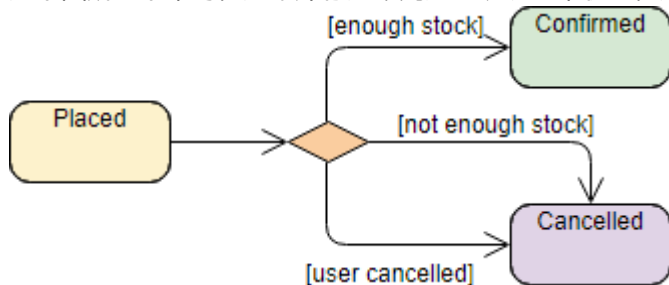


图 5: 状态机图选择节点案例 (图片来自 visual paradigm)

终止

终止也是一种虚拟状态，表示状态机的生命周期结束。终止用待叉号的箭头表示。

和最终状态不同的是，终止状态表示由于上下文对象的结果而导致状态机的结束

如下图所示，断电导致激活状态转为终止状态。

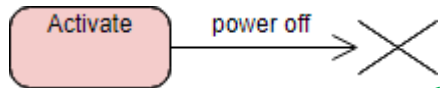


图 6: 状态机图终止 (图片来自 visual paradigm)

3.1.4 组合状态

普通的状态是不包含子结构的，而组合状态则需要画在状态内部或者画到独立的状态图中，表示将一个状态分为多个状态。包含子状态的状态称之为组合状态。

如下图所示的 Active 状态就是一个组合状态，其中的 Inspection / Choice / Transaction 只在 Active 状态下才存在。

需要注意的是，组合状态最多只有一个初始和最终状态。

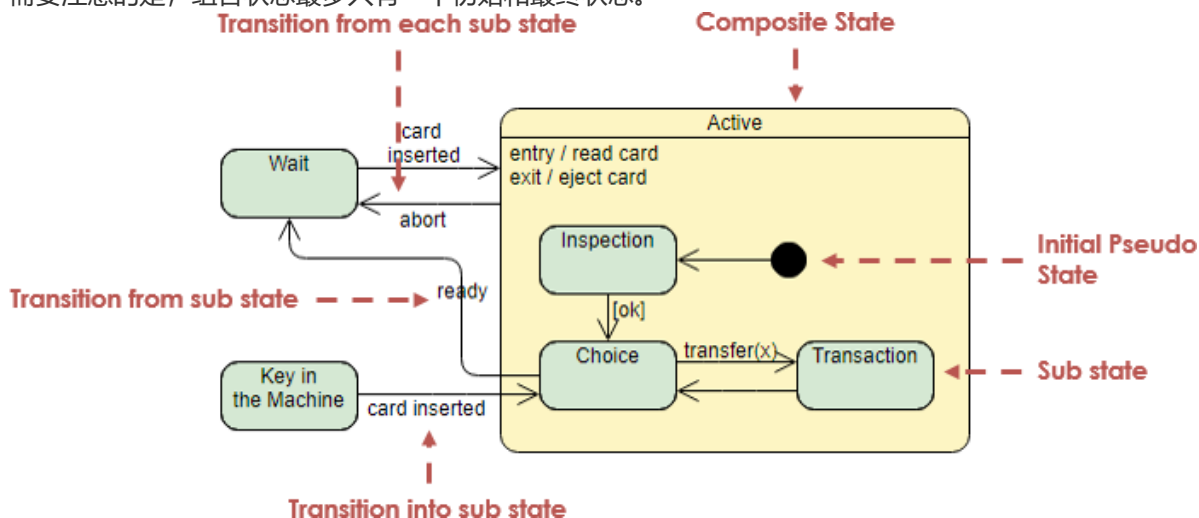


图 7：状态机图终止（图片来自 visual paradigm）

组合状态和子状态机图

下图为例，左侧表示组合状态，右侧表示子状态机图，两者语义上等价。

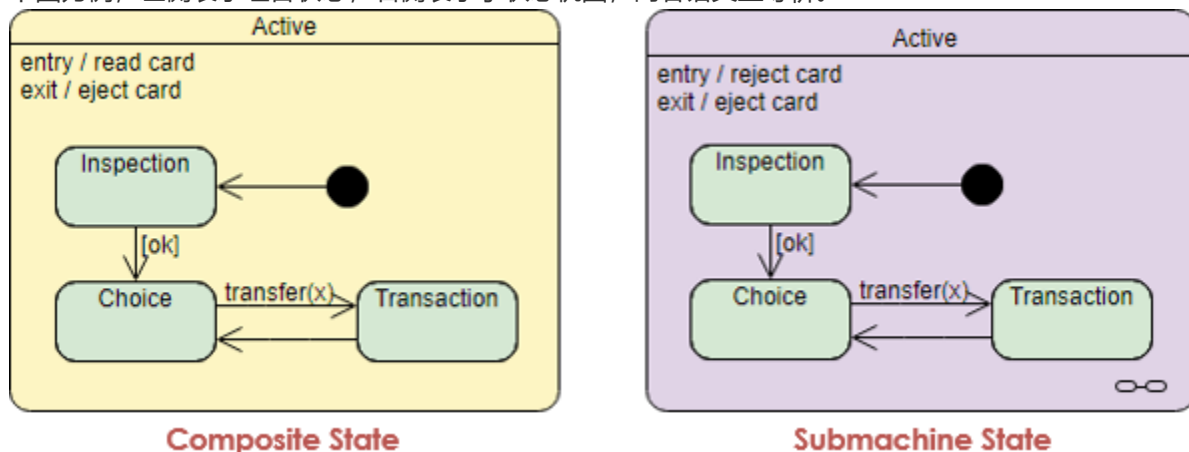


图 8：组合状态和子状态机图（图片来自 visual paradigm）

正交状态

如果一个组合状态包含两个及两个一样的子区域，则称之为正交。正交状态通过虚线分隔为多个部分。

如下图所示，S2 转换到状态 S1（转换到正交状态的边界）即代表激活了所有区域的初始状态。在正交状态中，每个区域都必须能够触达结束状态从而能够触发完成事件。S3 转换到 S1 则表示并发的情况。

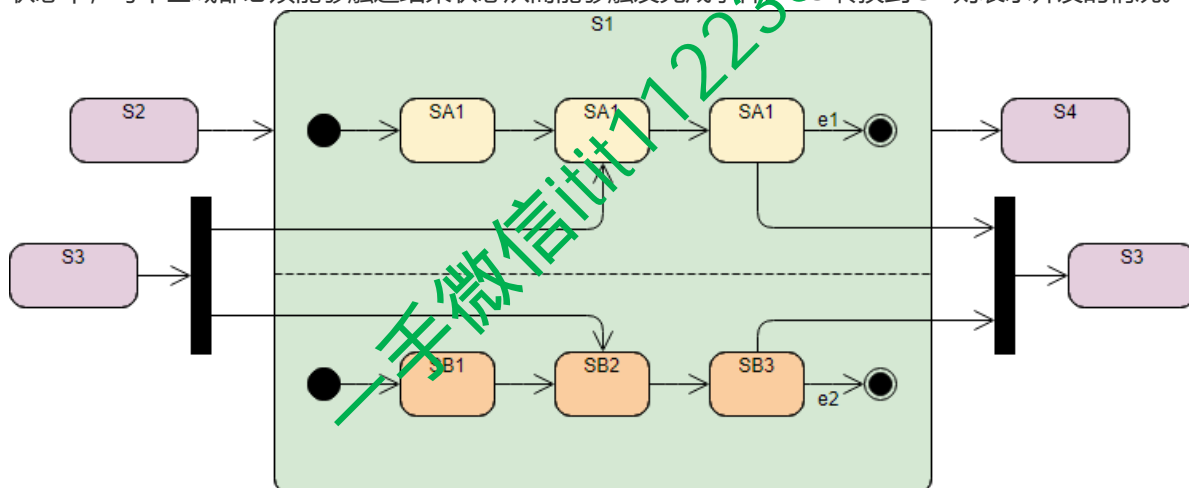


图 9：正交状态（图片来自 visual paradigm）

历史状态

历史状态允许状态机重新进入已经离开的组合状态。

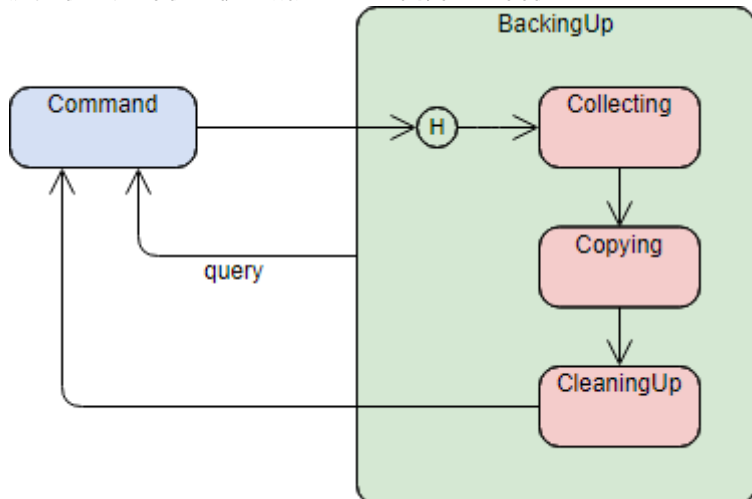


图 10：历史状态（图片来自 visual paradigm）

3.2 绘图步骤

寻找主要状态。画状态机图最重要的步骤是寻找主要状态。

确定状态间的转换。可以先将状态绘制出来，然后再分析状态之间的转换过程，绘制过程中可以调整状态的位置，以实现更好的视觉效果。

细化状态内的活动和转换。绘制完状态和状态间的转换后，可以根据需要添加内部转换，进入和退出转换，和其他的相关活动。

组合和正交状态的使用。如果某些状态可以归结为一个状态，那么可以使用组合状态绘制。如果一个状态可分为多个部分（转换流程），可以使用正交状态。

4.1 人生如梦

人的一生从婚姻角度看，包括单身、已婚、离异三个大状态；从职场角度看主要包括待业、受雇、退休三个大状态，当然从其他角度还可以画出更详细的状态图。

注：大家可以在 IDEA 中安装 PlantUML 插件，在项目视图下找到 Scratches and Consoles 选项选项卡，在此目录下作图。

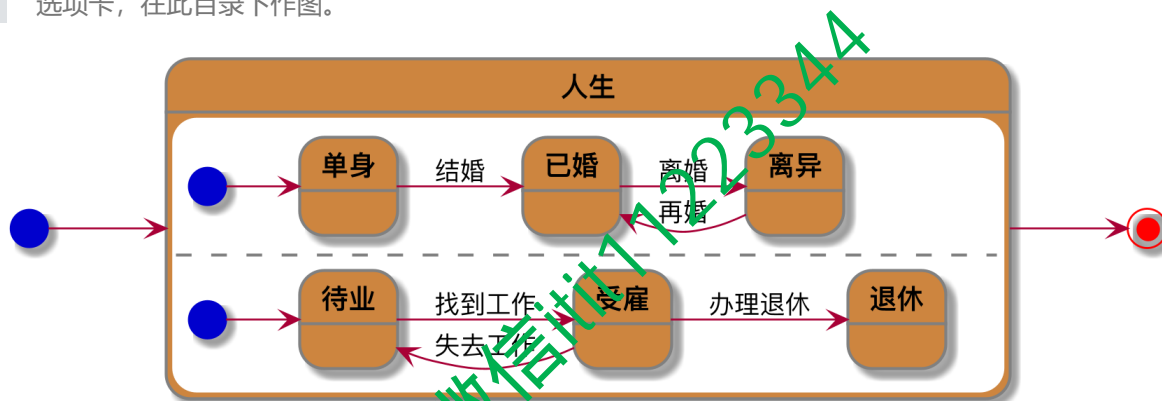


图 11：人生状态图

4.2 评论审核状态图

另外以某个博客评论审核系统为例，演示状态图的画法。

某博客系统发表评论后需要审核，审核通过后读者才可见，审核不通过则需要重新修改，修改通过才能读者可见。

我们先寻找主要状态：待审核、待修改、公开，然后根据实际情况填充状态之间的转换。

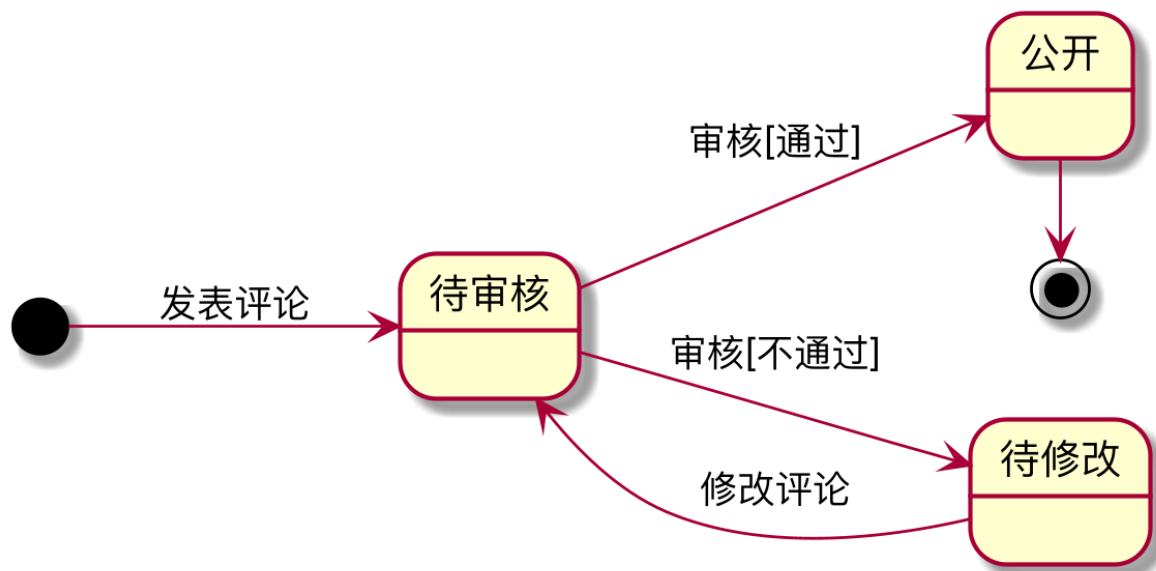


图 12：评论审核状态图

本节主要讲述了状态机图的主要概念，常见的使用场景，核心的组件，并给出了使用范例。希望大家能够熟练掌握并运用到项目梳理和技术方案的设计中。

下一节将讲述活动图的概念，活动图 and 状态图的区别，以及活动图的画法等。

请参考本节人生的状态图，自行绘制一模一样的图形。将该图受雇状态细化为包括普通员工、公司主管、CTO 三个状态采用组合状态丰富人生的状态图。然后从年龄段的角度讲婴儿、少儿、青少年、青年、中年、老年补充到人生的状态图中。

- 阿里巴巴与 Java 社区开发者.《Java 开发手册 1.5.0》华山版. 2019
- 维基百科 - 状态图
- [状态图的语法和功能](#)
- 谭云杰.《大象：Thinking in UML》.中国水利水电出版社. 2012

}