

# 11 活动图

## 11.1 活动图

### 11.1.1 语义

活动图是状态机的变体，变体中的状态表示动作或子活动的执行，转换由动作或子活动的完成触发。活动图表示过程自身的状态机。

### 11.1.2 表示法

活动图是状态图的特例，其中所有（至少大多数）的状态是动作或子活动，并且其中所有（至少大多数）的转换由在源状态中的动作或子活动的完成触发。整个活动图附属（通过模型）到类（如用况）、包或操作的实现。活动图注重于由内部进程驱动的流（与外部事件相比）。在所有或大多数事件都表示实现内部产生的动作（即，控制的过程流）的情形下，使用活动图。在发生异步事件的情境下，使用普通的状态图。

### 11.1.3 例子

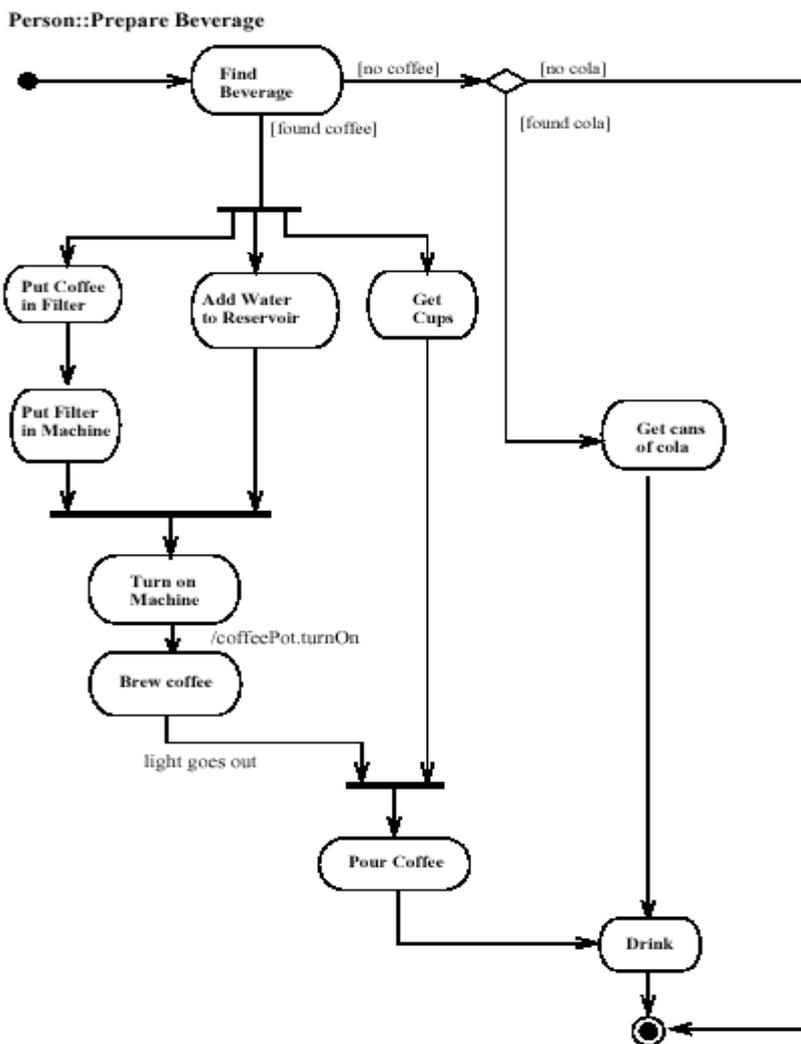


图 11-1 活动图

#### 11.1.4 映射

活动图型映射为活动图

## 11.2 动作状态

### 11.2.1 语义

动作状态是这样的状态，它具有一个入口动作和至少一个外出转换，外出转换中包含完成入口动作的隐含事件（如果有监护条件，可能有几个这样的转换）。动作状态不能有内部转换或者基于显式事件的外出转换，对这种情形，使用一般的状态。通常用动作状态对算法（过程）或工作流的执行步骤进行建模。

### 11.2.2 表示法

可以用一个上下是直线、左右是外凸的圆弧的图形表示动作状态。动作表达式放在图形之内。动作表达式在图中可以是不唯一的。

从动作状态出发的转换不应包括事件特征标记。状态中动作的完成隐式地触发这样的转换。转换可以包括监护条件和动作。

### 11.2.3 表示选项

可以用自然语言、伪代码或者程序语言代码描述动作。描述动作时仅可以使用拥有这个动作的对象的属性和链。

注意，可以在普通的状态图中使用动作状态表示法；然而，它们在活动图中更加常用，活动图是状态图的特殊情况。

### 11.2.4 例子



图 11-2 动作状态

### 11.2.5 映射

动作状态符号映射为动作状态，这样的动作状态带有被映射为状态的入口动作的动作表达式。动作状态既没有退出也没有内部转换。状态通常是匿名的。

## 11.3 子活动状态

### 11.3.1 语义

子活动状态调用活动图。当进入一个子活动状态时，就执行“嵌套”在其中的活动图，这样的活动图就象其他的活动图一样。除非达到嵌套图的最后一个状态，或者在由于子活动状态出来的转换上发生了触发事件，否则不退出子活动状态。因为在活动图中通常状态没有触发事件，所以子活动状态经常在嵌套图完成后退出。单个的活动图可以被多个子活动状态唤醒。

### 11.3.2 表示法

子活动状态的表示法和动作状态的表示法类似，不同的是在右下角多了一个用来描述嵌套活动图的图标。在符号中放置子活动的名字。子活动在图中不必是唯一的。

### 11.3.3 例子



图 11-3 子活动状态

### 11.3.4 映射

子活动状态符号映射为子活动状态。子活动的名称映射为子活动状态和其名为该名的状态机之间的子机链。子活动状态通常是匿名的。

## 11.4 决策

### 11.4.1 语义

当用监护条件指明依赖拥有对象的布尔条件的可能的不同转换时，状态图（源自活动图）能用以表达决策。本部分提供了一个缩记，用来显示决策，并把不同路径归并起来。

### 11.4.2 表示法

可以通过标记带有不同监护条件的动作的多个输出转换来表示决策。

为决策提供的图标是传统的钻石状图形，该图形带有一个入箭头和两个或多个出箭头，每一个箭头用不带事件触发的不同的监护条件标明。所有可能的结果都应该出现在外出转换上。可以为至多一个外出转换定义具有“else”的预定义监护。如果所有标在其它转换上的监护都为假，就激活这个转换。

可以用同一个图标来合并决策的多个路径，把这样的情况称为“合并”。合并有两个或多个进入箭头和一个外出箭头。

注意决策链可以是复杂转换的一个部分，但只有在这个链中的第一段可以包含事件触发标志。所有的段都可以有监护表达式，但源自合并的转换不能有触发标志或监护表达式。

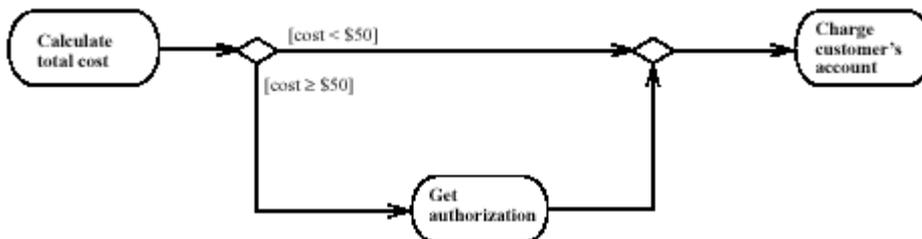


图 11-4 决策和合并

### 11.4.3 映射

决策符号映射为种类为连接的伪状态。朝外箭头上的各标号映射为从伪状态引出的相应转换上的监护。合并符号同样映射为种类为连接的伪状态。

## 11.5 泳道

### 11.5.1 语义

可以把动作和子活动组织成泳道。按照类，用泳道为动作和子活动组织责任。它们经常对应于业务模型中的组织单元。

### 11.5.2 表示法

可以把活动图分成可见的“泳道”，用垂直线把每个泳道和相邻的泳道分开。每个泳道代表整个活动的部分职责，最终可以用一个或多个对象实现它。泳道的相对顺序在语义上并不重要，但是可以表达某种亲和力。为每个活动分配一个泳道，转换可以跨越泳道。转换路

径的具体路由并没有什么意义。

### 11.5.3 例子

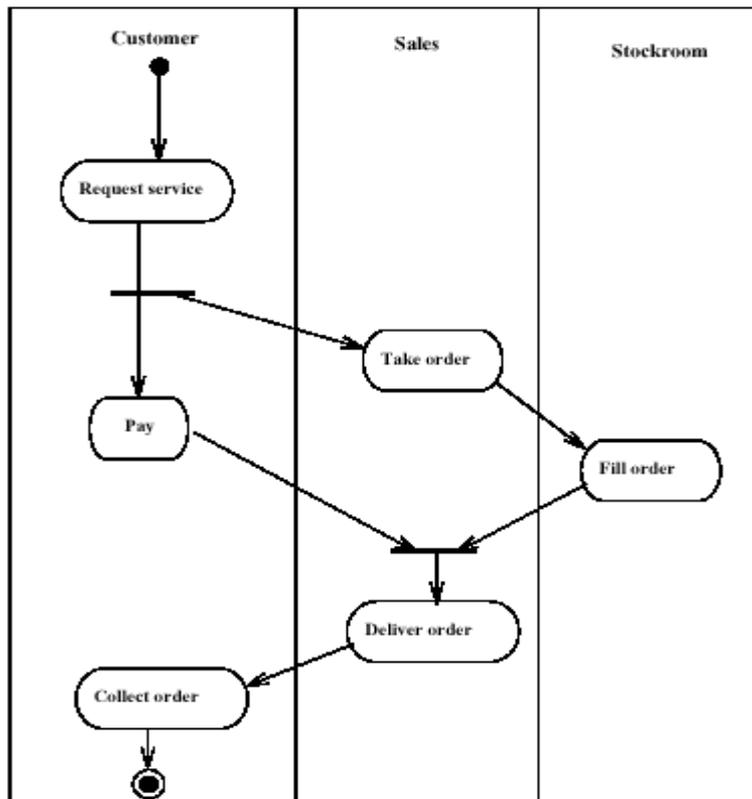


图 11-5 活动图中的泳道

### 11.5.4 映射

泳道映射为活动图中的状态分隔区。泳道中的状态符号使得对应的状态归属于相应的分隔区。

## 11.6 动作—对象流关系

### 11.6.1 语义

动作通过对象并在对象上起作用。这些对象要么对初始化动作有主要的责任，要么由动作使用或控制。通常动作指定拥有活动图的那个对象（初始化动作）和作为动作的目标的对象之间的调用发送。

### 11.6.2 表示法

#### 11.6.2.1 负责动作的对象

在顺序图中，通过画一条生命线并且把动作放在生命线上来表示对象负责执行动作，见 8.2 “顺序图”。活动图不显示生命线，但是每个动作指定那一个对象执行它的操作。这些对象也可以按一定方式与泳道相关联。同一个对象或者多个对象完全能操纵一个泳道内的动作。

#### 11.6.2.2 对象流

可以用对象符号表示输入到动作或从动作输出的对象。从动作状态到输出对象画一条虚线箭头，从输入对象到动作状态画一个虚线箭头。同一个对象可能（而且往往）是一个动作的输出，也可能是一个或者多个后续的动作的输入。

当对象流（虚线）箭头提供冗余约束时，必须删除控制流（实线）箭头。换句话说，当

状态产生了下一个状态的输入的输出时，对象流关系暗示控制约束。

### 11.6.2.3 对象的状态

同一个对象经常被一些连续的动作或子活动操作。显示一个带有到或来自所有相关动作和子活动的箭头的对象是可能的，但是为了更清晰起见，或许在一个图上多次画一个对象。对象的每次出现表示对象生存期中的一个不同点。为了区分同一个对象的不同出现，在每一点上，把对象的状态放在方括号中，并跟在对象名字的后面。（比如，PurchaseOrder[approved]）。也可以在协作图和顺序图中使用这种表示法。

### 11.6.3 例子

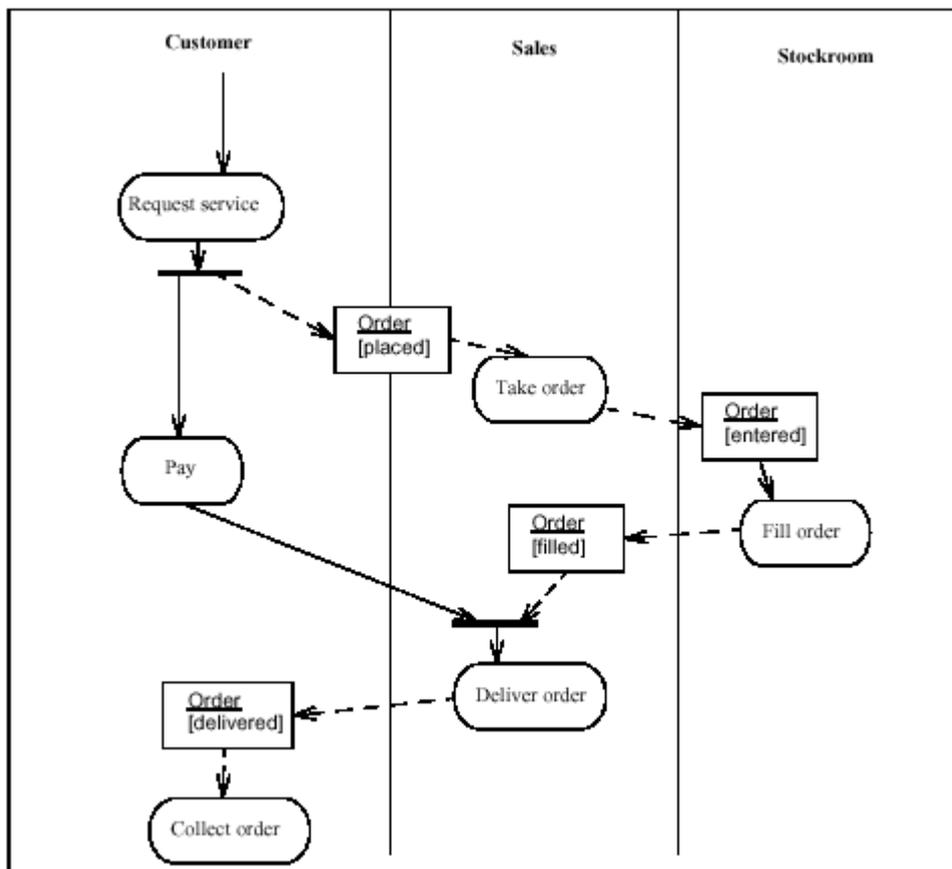


图 11-6 动作和对象流

### 11.6.4 映射

对象流符号映射为对象流状态，其进入与外出转换对应进入与外出箭头。该转换没有附加物。对象流符号的类名和 (可选的)状态名映射为对应该名称的类或在状态中的类目。实线和虚线都映射为转换。

## 11.7 控制图标

下列图标为确定种类的能在转换上描述的信息提供清晰的符号。这些图标不是构造活动图所必需的，但是许多用户喜欢它们提供的添加效果。

### 11.7.1 表示法

#### 11.7.1.1 信号接收

可以用一个凹的五边形表示信号接收，这个五边形看起来象其中的一个边上带有一个三角形凹口的矩形。信号的特征标记表示在符号的里面。从上个状态到五边形画一个没有标记

的转换箭头，并且从五边形到下一个动作状态画另一条没有标记的转换箭头。可以从对象标志到五边形的凹口画一条虚线箭头，用来表示信号的发送者，这是可选的。

#### 11.7.1.2 信号发送

可以用一个凸的五边形来表示信号发送，这个五边形看起来象一条边上带有一个三角形的矩形。信号的特征标记表示在符号的里面。从上一个状态到五边形画一个没有标签的转换箭头，并且从五边形到下一个动作状态画另一条没有标记的转换箭头。可以从五边形的尖端到对象标志画一条虚线箭头，用来表示信号的接收者，这是可选的。

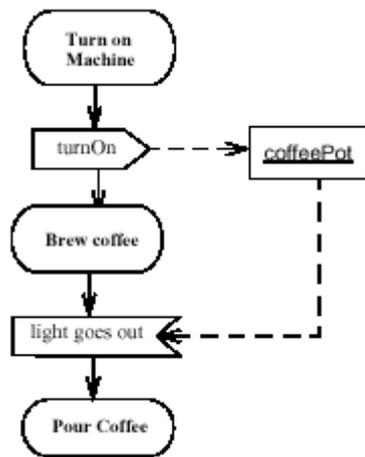


图 11-7 用于信号接收和发送的符号

#### 11.7.1.3 延缓事件

当一些其他的动作或子活动正在进行时，发生的事件必须要被延迟而等待使用，这是一种很常见的情况。通常没有及时处理的事件会丢失。对于这种情况，可以认为就象有一个内部转换处理，它处理这个事件，并且把它放在一个内部队列中，直到这个事件被需要或被丢弃。每一个状态指定一个被延缓的动作集（如果它们在这个状态期间发生，并不用于触发状态转换）。如果一个事件不在一个状态的延缓事件集中，而且它不触发转换，那么即使这个事件发生了，也会把它从队列中舍弃。如果转换依赖事件，那么若事件已经在内部队列中就立即点火转换。如果可能有几个转换，那么队列中第一个事件优先。

把可延缓事件列在状态里，其后面跟有一个斜线和一个特殊操作 `defer`。如果这样的事件发生，它被保存下来并且在对象转换到另一个状态的时候重现，在新状态中或许该事件再次被延缓。当对象到达该事件不被延缓的那一个状态时，就必须接收或者丢失这个事件。可以把表示放在复合状态或者它的等价体上以及子机和子动作状态上，在这些情况下，它保持在整个复合状态中是可延缓的。包含的转换仍然可以由可延缓事件触发，这样就从队列中移走它。

在动作状态中延缓事件并不是必要的，因为这些状态对事件处理来说是不可中断的。在这种情况下，在这个状态期间发生的延缓事件和非延缓的事件同时被延缓，直到该状态完成。这意味着定时转换将同样地忽略事件和状态完成的相对次序，并且忽略是否延缓事件。



图 11-8 延缓事件

11.7.2 映射

信号接收符号映射为没有动作和内部转换的状态。其具体事件映射为它和下一个状态之间的向外转换的触发器事件。

信号发送符号映射为在它和前面状态之间的进入转换上的发送活动。

附属到状态的延迟事件映射为从状态到事件的延迟事件关联。

### 11.8 同步状态

当同步状态有一个引入和一个外出转换并且没有约束限制时，在活动图中可以省略同步状态表示法。如果包含了同步状态圈（就象为状态机定义的那样），则其语义与映射是相同的。

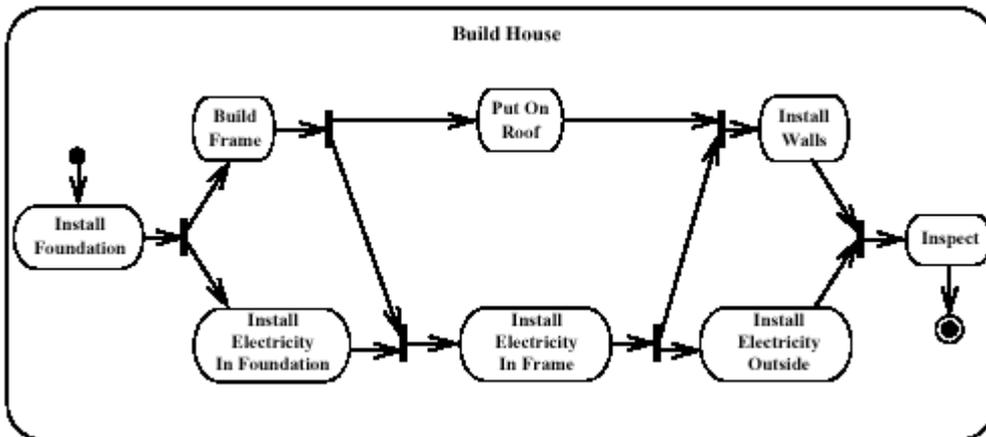


图 11-9 同步并行活动

## 11.9 动态调用

### 11.9.1 语义

动作状态的动作或子活动状态的活动图可以多次并发地执行。在运行时用并发表达式决定并发调用的数量，并发表达式对一组参数表求值，每一个调用有一个参数表。

### 11.9.2 表示法

如果动作或者子活动状态的动态并发不总是只有一个，它的多重性可以显示在状态的右上角。否则，就什么也不显示。

### 11.9.3 映射

活动或者子活动状态右上角中的多重性字符串映射为状态的动态多重性属性中的值，这两个值是相同。多重性字符串也映射为状态的“是动态的”属性的真值。如果不存在多重性，“是动态的”属性值为假。

## 11.10 条件分叉

在活动图中，从分叉外出的转换可以有监护。这意味着由分叉转换初始化的区域也许不启动，因而不要求在相应的结合处结束。可以在从分叉外出的转换上使用常用的监护的表示法和映射。