

系统分析辅导： workflow 模型设计 PDF 转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/141/2021\\_2022\\_\\_E7\\_BB\\_BB\\_E7\\_BB\\_9F\\_E5\\_88\\_86\\_E6\\_c29\\_141207.htm](https://www.100test.com/kao_ti2020/141/2021_2022__E7_BB_BB_E7_BB_9F_E5_88_86_E6_c29_141207.htm)

摘要： workflow 模型是组织机构经营过程的模型表示，但由于 workflow 建模理论研究的相对滞后，现行 workflow 管理系统中普遍存在语义不够丰富、柔性不好、表示不直观等问题。为此作者在开发 UniFlow 工组流管理系统时，设计了一套基于有向图理论的 workflow 模型，该模型有效的解决了这些问题。本文主要从 workflow 模型理论出发向读者介绍了本模型的设计思想。文章首先详细介绍了 workflow 模型的定义和组成关系，然后详细介绍了基于有向图理论的 workflow 模型的主要功能和设计思想。关键词： workflow workflow 模型 有向图 过程1 引言 workflow 模型是对组织机构业务过程的抽象表示。 workflow 模型是整个 workflow 管理系统的基础，它的确定性保证了系统内各组成部分之间交互的一致性，也决定了一个 workflow 管理系统从设计开发到运行实施的诸多环节。不同的 workflow 模型也就形成了不同的 workflow 管理系统。相对 workflow 产品的实现技术和发展而言， workflow 建模理论的研究相对滞后，在建模方法上还没有形成比较系统化的理论体系。目前已有的建模方法主要有：基于活动网络的建模方法、基于 Petri 网的建模方法、基于语言行为理论的建模方法、基于活动与状态图的建模方法和基于扩展事务模型的建模方法等。但这些 workflow 模型普遍都存在以下问题：（1）模型语义不够丰富，表示复杂问题的能力不足；（2）模型的柔性不好，不容易处理在执行过程中出现的不确定性问题；（3）表示方法不够直观等。在信息管理自动化的环境下，

workflow模型必须采用简单、直观、又具有较强描述能力的面向组织用户的模型。因此，作者设计UniFlow workflow管理系统时，在参照 workflow管理联盟（Workflow Manangement Coalition 简称：WfMC） workflow理论的基础上，博采众长，建立了基于有向图理论的UniFlow workflow管理系统 workflow模型，该模型有效的解决了 workflow模型中普通存在的问题。下文主要向读者详细介绍这一模型的相关理论与设计思想。

## 2 workflow模型概要

企业、政府机关等组织机构是一个复杂的系统，用过程的观点来看待这些组织机构，组织机构的组成元素就是过程，过程的组成元素就是活动； workflow模型就是对这些业务过程的抽象表示。过程是指为了完成组织机构的某个经营目标或任务，而按某种顺序联结在一起的一系列活动。过程的主要属性有：触发事件、过程结果、执行规律集。过程的执行规律集用来描述过程内部各活动、子过程之间的控制流程。执行规律集由顺序、并发、与、或、循环等五种连接关系组成，通过这五种连接可以完整的描述活动与活动之间的逻辑关系。活动是指一种变换或操作，是组织机构过程的最基本单元。一个活动的属性由功能输入、输出，资源输入、输出、控制输入、输出和活动的描述等组成。功能输入、输出是指活动中要运作、产生的组织机构对象，它们描述了组织机构过程的物流，所以通常与上级过程的执行规律集有紧密联系。资源输入、输出是指活动执行所占用、释放的资源，包括执行活动所必须的设备、物料、人员等，它们不仅可以在一定程度上描述不同活动之间的竞争关系，还可以用于实现对过程的分析 and 优化；控制输入是指对活动进行控制和限制的信息单位或对象；控制输出是活动结束状态的信

息单位集合；活动描述是与活动执行有关的具体信息，如活动的编号、名称、优先级、成本核算单位、开始时间、完成期限、执行者、负责人、所需资源等。功能输入、资源输入、控制输入统称为活动的前条件，前条件对活动的发生有制约作用，只有所有的前条件都满足了，活动实例才能进入“活动”状态；功能输出、资源输出、控制输出统称为活动的后条件，后条件影响着后继活动实例的发生。一个完整的工作流模型主要有工作流表示方法、工作流元模型、工作流模型接口和建模工具组成。

### 3 工作流模型的有向图表示

组织机构的一个完整的业务流程是由一系列最基本的生产活动按照一定的逻辑顺序和规则组成的，这些生产活动与他们之间的逻辑关系可以直观的映射成为一个由节点和连接弧组成的有向图。有向图中的节点即表示一个可执行的活动单元，连接于两个节点之间的有向弧即表示活动间的先后顺序关系。

#### 3.1 有向图理论基础

对基于有向图理论的工作流模型，我们作如下定义：

- 1) 有向图  $G = \{N, L\}$  为一个二元组， $N = \{n_1, n_2, n_3, \dots, n_s\}$  为节点集合， $L = \{l_1, l_2, l_3, \dots, l_s\}$  为连接弧集合，其中  $l_i = (n_j, n_k)$  为从  $n_j$  指向  $n_k$  的连接弧， $n_j, n_k \in N$ 。
- 2) 对于任意  $n_i, n_j \in N$ 。若有连接弧  $l = (n_i, n_j) \in L$ ，则  $n_i$  为  $n_j$  的一个前驱节点， $n_j$  为  $n_i$  的一个后继节点， $l$  称为  $n_j$  的一个输入连接弧，或者  $n_i$  的一条输出连接弧。
- 3) 若  $N'$  包含于  $N$ ，且  $N' = \{n' \mid (n', n) \in L\}$ ，则  $N'$  为节点  $n$  的前驱节点集，记为  $\text{Pre}(n)$ 。
- 4) 若  $N'$  包含于  $N$ ，且  $N' = \{n' \mid (n, n') \in L\}$ ，则  $N'$  为节点  $n$  的后继节点集，记为  $\text{Post}(n)$ 。
- 5) 节点状态：对于任意  $n \in N$ ，有状态函数  $\text{State}(n) = \{0, 1\}$ ，当节点  $n$  处于非执行状态时， $\text{State}(n) = 0$ 。当节点  $n$  处于执行状态时  $\text{State}(n) = 1$ 。初始时刻，任意  $\text{State}(n) = 0$ 。
- 6

) 转移函数：对于任意 $l \in L$ , 有转移函数 $Trans(l) = \{0, 1\}$ . 如果 $Trans(l) = 1$ , 则连接弧 $l$ 容许发生转移 (是否发生转移, 则要取决于后面的演进规则) : 如果 $Trans(l) = 0$ , 则连接弧不能发生转移。

7) 演进规则：有向图的演进规则是由节点状态的改变与连接弧发生转移这两个动态因素相互作用而完成的, 因此, 规则包含如下两个方面：(1) 对于任意 $n \in N$  当 $State(n) = 0$ 时, 若存在 $l = (n', n)$ 发生转移, 则 $State(n) = 1$ . 当 $State(n) = 1$ 时, 若节点 $n$ 执行完毕, 则 $State(n) = 0$ . (2) 对于任意 $l = (n, n') \in L$  当 $State(n)$ 从1变为0时, 若 $Trans(l) = 1$ , 则连接弧 $l$ 发生转移; 若 $Trans(l) = 0$ , 则连接弧 $l$ 不发生转移, 直至下一次 $State(n)$ 从1变为0的时候再使用此规则。

### 3.2 节点类别

在基于有向图的工作流模型中有向图的节点代表了具有如下特征的多种实体：1) 与业务中实际存在的事件或活动有着直接的对应关系；2) 本身有着具体的或人为定义的含义；3) 能与其他节点形成一定的逻辑关系；因此, 区分不同类别的节点、对节点进行具体的类别定义不仅可以明确节点的含义, 同时也增强了模型的语义。在这里我们赋予节点有如下几种类型的定义：活动、子过程、开始、结束、同步节点。

#### 3.2.1 活动

活动是指在一个不间断的时间间隔内为实现某一目标由人工或系统自动完成的一个原子任务, 它是组成业务流程的基本单元。一个 workflow 管理系统的所有活动的集合覆盖了组织中各类业务流程的全部细节。活动按自动化程度可分为两类：人工型活动和自动型活动。人工型活动是通过工作表 (任务表) 的生成来通知相关人员, 依靠人员手工或启动应用的方式来完成。自动型活动则是在工作流的驱动下直接驱动应用或利用自动化设备来完成的活动；这种自动型活动

充分体现了 workflow 管理系统所实现的组织内部不同应用间的过程集成。

### 3.2.2 子过程

作为组成业务流程的最基本单元，活动是指不能被进一步分解的原子单元。但在流程较复杂时活动结点数也必然增加，且关系复杂，因此，我们引入了子过程的概念。子过程是一类能够分解的节点类型，它的内部可以包含所有元素类型，实质上是一个子业务流程。子过程的引入大大增强了模型的表达能力，使模型具有了层次的概念，并支持自顶向下的建模过程。同时，我们规定子过程可以出现在任何层次。

### 3.2.3 开始节点与结束节点

开始节点和结束节点是两个标志性的节点，用来标识流程的开始和结束。一个 workflow 具有唯一开始点和结束点。

### 3.2.4 同步节点

在将一个实际的业务流程映射成为 workflow 模型时，很重要的一点，就是要保证活动间的逻辑关系不变。“与”和“或”是两类最基本的逻辑关系，它是表达各种复杂关系的基础，workflow 模型必须具备表达“与”和“或”关系的能力。我们在前面定义的演进规则，对于任意一个处于非执行状态的节点  $n$ ，只要有一条输入连接弧发生了转移，那么该节点即可被执行，这实际上就表达了“或”的关系。即  $\{n' \in \text{Pre}(n), l = (n', n) \in L, \text{Trans}(l) = 1 \text{ 且 } l \text{ 发生转移}\}$ 。

对于“与”的关系增加同步节点  $S$  来表示，它对活动起协调、同步的作用。我们规定，同步节点  $S$  的动态行为完全遵循演进规则，所不同的是，当  $S$  处于执行状态时，即  $\text{State}(S) = 1$  时，将判断它的所有输入连接弧是否发生转移；若是，则  $S$  的状态就由 1 变为 0，即  $S$  执行完毕；否则  $S$  仍处于等待状态，并继续判断，直至满足上面的条件后才执行完毕、发生转移， $\text{State}(S) = 0$ 。这意味着同步节点将使它的所有前驱节点都执行完毕后才继续推进流程，这实际上就

表达了“与”的关系  $\{n' \mid \text{Pre}(n), l=(n', n) \in L, \text{Trans}(l)=1\}$   
且 $l$ 发生转移。 100Test 下载频道开通，各类考试题目直接下载。  
详细请访问 [www.100test.com](http://www.100test.com)