工作流过程定义中的分层结构与正则 Petri 网*)

王斌君[1] 郝克刚[2]

(中国人民公安大学科技系 北京100038)[1] (西北大学软件工程研究所 西安710069)[2]

The Hierarchical Structure of the Workflow Process Definition and Normal Petri Net

WANG Bin-Jun[1] HAO Ke-Gang[2]

(Department of Science & Technology, Chinese People's Public Security University, Beijing 100038)[1]

(Software Engineering Institute, Northwest University, Xi'an 710069)[2]

Abstract The abstract level problem and the classification of the abstract level problem in the workflow process model are discussed. The formalized problem of the abstract level problem in the process model is researched. The concept and definition of the petri net with end-transition and normal petri net according to the control abstract level problem are put forward and discussed. The equivalence of general Petri Net and general Petri Net is proved by the petri net with priori relationship.

Keywords Petri net, Petri net with priori relationship, Petri net with end-transition, Normal petri net, Block transition of petri net, Petri net with block transition, Workflow, Process definition, Complex sytem

1 引言

工作流管理系统,也称为过程管理系统[1],是一种特殊的计算机支持的协同工作软件,它所解决的问题是分布式环境下各种异步活动的协同工作问题。过程管理系统被誉为是继数据库管理系统和人机界面管理系统之后第三次从应用中分离出来,并成为一个应用系统不可或缺的子系统。已有大批的研究机构[1]、专家学者[2~4]和大型软件公司对其研究。

工作流管理系统中一个核心问题就是对工作流模型的研究。按工作流模型的抽象程度应该将其分为三个层次[5,6]:WPDL 语言层、信牌驱动模型层和 Petri 控制模型层。WPDL 语言层是与其它工作流模型交流过程定义的基础;信牌驱动模型是对 WPDL 语言中控制结构和数据结构的一种抽象;Petri 网模型是对信牌驱动模型中控制结构的进一步抽象。文[6,7]定义并完善了一种工作流过程定义的扩展信牌驱动模型,并在此基础上研究了信牌模型中最常用的基本概念和控制结构所对应的非确定 Petri 网控制模型[8]。但由于篇幅的限制,文[8]对信牌驱动模型中的分层结构和子工作流问题的形式化没有详细讨论,本文将完善这部分内容,它可看作文[8]的续篇。

用扩展的信牌驱动模型对一些大而复杂的实际问题进行过程定义时,并非所有的活动都处在同一层次上,过程定义应该是一些处在不同抽象级别上的活动构成的有层次的协同工作网络。处在不同层次上的活动,它们出现的抽象粒度是不一样的,把它们放置在同一个层次上,不仅产生了混乱,而且使它们之间的关系复杂化了。另外,出于工程的需要,对一个中型问题进行过程定义也不可能将所有的活动、信牌箱以及它们之间的关系都创建在同一张图上。对这类复杂问题的一个行之有效的解决方案就是使用分层、信息隐蔽和封装[9]。

本文将讨论扩展的信牌驱动模型中各种不同粒度和抽象级别的活动。与这种分层问题的需求相对应,在扩展的信牌驱动模型中引入了内置块和子过程等概念以适应不同应用场合对过程定义的需求。为了将这种模型进一步进行形式地描述,我们定义了带结束变迁的 Petri 网和正则 Petri 网。最后,论述了带结束变迁的 Petri 网与正则 Petri 网之间的等价关系。

2 工作流过程定义中的分层结构

下面是扩展的信牌驱动模型静态结构的定义。

定义1(扩展的信牌驱动模型) 多元式 $XP=(H,X;F,H_0,H_f,W,H_{SPLIT,JOIN}H,D)$称为扩展信牌驱动模型的静态结构,其中:

①D 表示扩展的信牌驱动模型所涉及的所有数据,其值域用$\overset{\vee}{D}$表示;

②H 表示活动集合,$\forall h\in H, h=(GH,HH)$,GH 和 HH 分别称为功能函数和后继函数。GH 被定义为$2^D\to 2^D$,HH 根据出函数定义,参见④;

③X 表示信牌箱集合;

④$F\subseteq X\times H\cup H\times X$,称为 XP 的流关系,其中 $X\times H$ 和 $H\times X$ 分别称为入关系和出关系。对出关系定义一个出函数 $FO:H\times X\times D\to TF$(TF 是一个真假值的集合)。$\forall h\in H$,FO|h 表示与 h 相关的出函数,被称为 h 的后继函数;

⑤$H_0\in H$ 是唯一的活动,称为开始活动,$^*H_0=\phi$;

⑥$H_f\subseteq H$ 是一个活动的集合,称为结束活动,$H_f{}^*=\phi$;

⑦$W:F\to N$ 称为转移的权重;

⑧H_{SPLIT}是 H 的一种划分$\{H_{ALL},H_{AND},H_{XOR}\}$,即 $H_{ALL}\cap H_{AND}\cap H_{XOR}=\phi$,$H_{ALL}\cup H_{AND}\cup H_{XOR}=H$,

$_{JOIN}H$ 是 H 的另一种划分$\{_{ALL}H,_{AND}H,_{XOR}H\}$,即$_{ALL}H\cap _{AND}H\cap _{XOR}H=\phi$,$_{ALL}H\cup _{AND}H\cup _{XOR}H=H$。

利用上述定义,对扩展的信牌驱动模型中过程、子过程和块等概念描述如下:

*)本文受国家"十五"重点科技攻关项目资助(项目编号:2001BA107C)。王斌君 博士,副教授,目前研究方向为软件工程、工作流模型和软件理论等。郝克刚 教授,博士生导师,目前研究方向为构件技术、分布式计算和软件理论等。

过程:在扩展的信牌驱动模型中,活动和信牌箱按规则构成的描述工作流的网结构必由一个唯一开始活动和若干个结束活动组成,其中的 AND-SPLIT 与 AND-JOIN 必须扩展地正则配对(详见文[6]),而 ALL-SPLIT 与 ALL-JOIN、XOR-SPLIT 与 XOR-JOIN 可以扩展地正则配对,也可以不扩展地正则配对。

块:在扩展的信牌驱动模型中,由一个标记为块开始的活动和一个块结束的活动以及其它若干个活动和信牌箱构成某个过程中的子网称为块。块必须满足扩展的正则的配对结构。块的作用有如下几条:块可以在一个过程中被共享,而不需要重复地定义;可以把块看作一个整体,在扩展的信牌驱动模型中可以将块看作是一个更高抽象级别的"活动",以便简化对具有复杂控制结构的过程定义的描述和理解。借用块的概念,我们可以将扩展的信牌驱动模型组织成一个分层的过程定义,下层的一个块(正则配对的子网)对应上层的一个抽象"活动"。这种分层的过程定义表达了对复杂问题的抽象和工程设计的需要。

子过程:当一个过程被另外某个过程的活动调用或者激活时,这个过程被称为一个子过程。因此,一个子过程的基本要素和构成规则与一个过程相同。它的作用像块一样可以简化复杂业务流程的描述;可以在多个过程定义间被共享。为了使子过程能够被多个过程定义共享,子过程的定义需要设置参数,以适应不同环境下的调用。

由此可见,无论是块还是子过程都是为了解决共享和重用性问题、表达不同粒度的抽象以便更有效地组织过程定义而设计的机制。研究发现这种带结束活动的控制机制对应于一种被称为带结束变迁的 Petri 网。进一步研究表明,通过带优先关系的 Petri 网,可以将带结束变迁的 Petri 网等价地转换为一个通常意义下的 Petir 网。下面将分别描述这些概念,并证明带结束标记的 Petri 网与一般 Petri 网之间的等价关系。

3 带优先关系的 Petri 网

定义2(带优先关系的 Petri 网) ppt=(Σ,ρ)被称为一个带优先关系的 Petri 网,其中:

①$\Sigma=(S,T;F,K,W,M_0)$是一个 Petri 网系统;

②优先关系 ρ 是 T 上的偏序关系,(t1,t2)∈ρ 表示 t1的优先级高于 t2的优先级。

当在一个标识下没有比变迁 t 更高优先级的可激活变迁时,该变迁 t 才能发生。下面是具有优先关系的 Petri 网发生权的形式化定义。

定义3(ρ-发生权) 设 ppt=(Σ,ρ)是一个带优先关系的 Petri 网,对 M∈[M_0>,t 称为在 M 下有 ρ-发生权,当它满足:M[t>且不存在 t',使得(t',t)∈ρ,记作 M[t>ρ。

(Σ,ρ)的发生结果同一般的 Petri 网系统,这里不再赘述。

定义4(广义补位置) 设$\Sigma=(S,T;F,K,W,M_0)$是一个 Petri 网系统,P⊆S 是一个非空的集合。P 的广义补位子 P' 被定义为满足下列条件的位子:∀ t∈T

$W(t,P')=\max\{0,\Sigma_{s\in P}(W(s,t)-W(t,s))\}\wedge$

$W(P',t)=\max\{0,\Sigma_{s\in P}(W(t,s)-W(s,t))\}\wedge$

$K(P')=\Sigma_{s\in P}M_0(s)+M_0(P')$

定理1 设$\Sigma=(S,T;F,K,W,M_0)$是一个 Petri 网系统,P⊆S 是一个非空的集合,Σ▶P 是扩展 P 的广义补位子而得到的新的 Petri 网系统,则 Σ 和 Σ▶P 等价。

证明:略。(参见文[6]或[10])

定义5 设 ppt=(Σ,ρ)是一个带优先关系的 Petri 网,其中:

①$\Sigma=(S,T;F,K,W,M_0)$是一个 Petri 网系统;

②优先关系 $\rho=\{(t_{11},t_{12}),(t_{21},t_{22}),\cdots\cdots,(t_{n1},t_{n2})\}$是一个 T 上的偏序关系,$(t_{i1},t_{i2})\in\rho$ 表示 t_{i1}的优先级高于 t_{i2}的优先级。

令 $P_i={}^{\bullet}t_{i1},i=1,2,\cdots,n$。$\Sigma^1=(S',T,F'';K',W'',M_0')$是一个由下列变换得到的 Petri 网系统:$\Sigma\blacktriangleright P_1\blacktriangleright P_2\cdots\blacktriangleright P_n$。那么,$\Sigma_\rho=(S',T;F',K',W',M_0')$,其中:∀ $P_i'\in S'$

①假如 $W''(P_i',t_{i2})=0$,则将(P_i',t_{i2})加入到 F' 中,并且使 $W'(P_i',t_{i2})=1$;否则 $W'(P_i',t_{i2})=W''(P_i',t_{i2})$。

②$W'(t_{i2},P')=W''(t_{i2},P')-W''(P',t_{i2})+W'(P_i',t_{i2})$

定理2 设$\Sigma=(S,T;F,K,W,M_0)$是一个 Petri 网系统,(Σ,ρ=(t,t'))是一个优先系统,P=${}^{\bullet}t$,(Σ▶P=(S',T;F'',K',W'',M_0')。那么,由 Σ▶P 按定义5所构造的 Petri 网系统 $\Sigma_\rho=(S',T;F',K',W',M_0')$与(Σ,ρ=(t,t'))等价。

证明:略。(参见文[6]或[10])

4 带结束标记 Petri 网与正则 Petri 网

下面先给出带结束标记 Petri 网和正则 Petri 网的定义,然后,利用带优先关系 Petri 网证明一个带结束标记 Petri 网与等价于一个正则 Petri 网。

在下面的各定义和定理中,没有说明的定义和概念参见文[11,12]。

定义6(带结束标记的 Petri 网系统) 一个带结束变迁的 Petri 网系统被定义为一个七元组:FP = {S,T;F,K,W,M_0,T_F}。其中:

①$\Sigma=\{S,T;F,K,W,M_0\}$是 Petri 网系统;

②$T_F\subseteq T$ 是一个结束变迁的集合,当 $t\in T_F$ 发生后,系统的所有变迁均不能再发生。

在不引起误会的前提下,可将带结束变迁的 Petri 网系统简称为带结束变迁的 Petri 网。一个带结束变迁的 Petri 网是对信牌驱动模型控制部分的抽象(文[8]已证明了信牌驱动模型中 END-SPLIT 和 XOR-SPLIT 控制结构对应的非确定 Petri 网与一般 Petri 网之间的等价关系),它是在不考虑组织模型、角色以及角色分配、各种资源和各种数据等因素的前提下,对信牌驱动模型中控制部分的形式化描述。

定义7(正则 Petri 网系统) 一个正则 Petri 网被定义为一个多元组:NP={S,T;F,K,W,S_0,S_f,T_0,T_f}。其中:

①$\Sigma=\{S,T;F,K,W,S_0\}$是 Petri 网系统,S_0是一个被称为初始位子的特殊位子,它也表示该 Petri 网的初始标识中只有 S_0含有 token;

②S_f 是一个被称为结束位子的特殊位子,它也表示当该 Petri 网的所有变迁均无发生权的标识(也称为结束标识)中只有 S_f 含有 token;

③$T_0\in T$,被称为开始变迁,${}^{\bullet}T_0=S_0$,且 $S_0^{\bullet}=T_0$;

④$T_f\in T$,被称为结束变迁,$T_f^{\bullet}=S_f$,$S_f^{\bullet}=T_f$,且当 T_f 发生后,除 S_f 有 token 外,∀ $s\in(S-S_f)$均不含 token。

由上面的定义可见,正则 Petri 网有唯一的初始标识、开始变迁、结束标记和结束变迁。初始状态下只有 T_0能发生;当 T_f 发生后,该正则 Petri 网的标识变为 S_f,系统将不再允许任何变迁发生。其中,S_0(S_f)即表示一个位子,也表示初始标识

(结束标识),可根据上下文确定其含义。

定理3(带结束变迁的 Petri 网的等价性定理) 对任何给定的带结束变迁的 Petri 网系统 $\Pi=\{S,T;F,K,W,M_0,T_F\}$ 都可以构造一个正则的 Petri 网与其等价。

证明:首先,对给定的带结束变迁的 Petri 网系统 $\Sigma=\{S,T;F,K,W,M_0,T_F\}$,按如下的方法构造一个带优先关系的正则的 Petri 网 (NP,ρ),其中,$NP=\{S',T';F',K',W',S_0,S_f,T_0,T_f\}$ 按如下的方式构造:

令:$T_{si}=\{t_s|\forall\ s\in S$,构造一个对应的 $t_s\}$,由于 Σ 是有限 Petri 网,设 T_{si} 中的元素为:$t_{s1},t_{s2},\cdots,t_{sn}$,b 是一个特殊的位子,则:$S'=S\cup\{S_0\}\cup\{S_f\}\cup\{b\}$。

$T'=T\cup T_0\cup T_f\cup T_{si}$。

$F'=F\cup\{(S_0,T_0)\}\cup\{(T_0,s)|s\in M_0\}\cup\{(s,t_s),(b,t_s),(t_s,b)|t_s\in T_{si}\}\cup\{(t_F,b)|\ t_F\in T_F\}\cup\{(b,T_f)\}\cup\{(T_f,S_f)\}$

$K'=K\cup(K'(S_0)=1)\cup(K'(S_f)=1)\cup(K'(b)=1)$

$W'=W\cup(\{F'-F\}\rightarrow 1)$

按上述方法构造的 Petri 网 $NP=\{S',T';F',K',W',S_0,S_f,T_0,T_f\}$ 是一个正则的 Petri 网。图1是上述形式构造的直观表示。

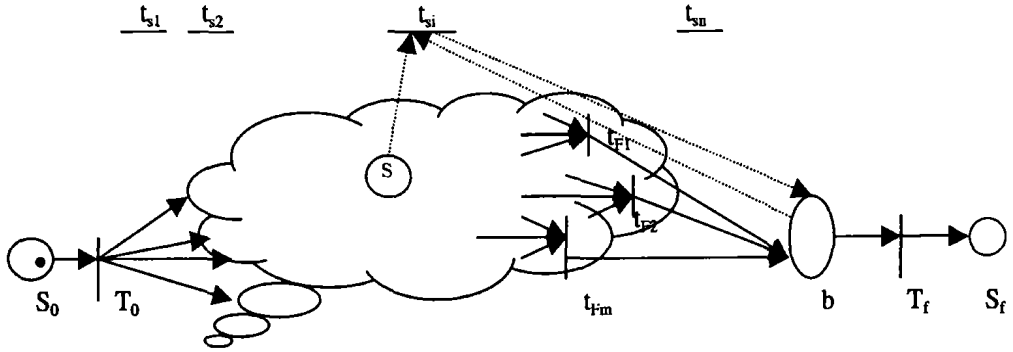

图1 带结束变迁的 Petri 网与正则 Petri 网

其中,  表示原 Petri 网系统,而 T_0、t_{si} 和 T_f 是为了构造与其等价的正则 Petri 网而新增加的变迁,S_0、S_f、b 是新增加的位子,T_0 表示开始变迁,$t_{si}(i=1,2,\cdots,n)$ 表示每个位子对应的新变迁,T_f 表示最终结束变迁(以与原 Petri 网中结束变迁 t_{Fi} 区别),它们都是在原 Petri 网上增加的变迁。图中的其它流关系非常明确,这里不再解释。

优先关系 ρ 的构造方法如下:为保证与原带结束变迁的 Petri 网系统 Σ 等价,当 $\forall\ t_{Fi}\in T_F$ 发生后,原 Petri 网中的任何变迁均不可发生,这要利用带优先关系的 Petri 网系统。设 ρ 是我们要建立的优先关系,为了真正到达 S_f,而不让 Σ 中有残留的 token,需要在激活 T_f 之前清空残留的 token,这一任务由 $t_{si}(i=1,2,\cdots,n)$ 完成,即 t_{si} 的优先级高于 T_f,所以,$\{(t_{si},T_f)|\ t_{si}\in T_{si}\}\subset\rho$。进一步,为了使 $t_{si}(i=1,2,\cdots,n)$ 顺利清空残留的 token,而不使 $\forall\ t\in T$ 发生,需要使 t_{si} 比 t 有更高的优先权,所以,$\{(t_{si},t)|\ t_{si}\in T_{si}\wedge t\in T\}\subset\rho$。因此,$\rho=\{(t_{si},T_f)|t_{si}\in T_{si}\}\cup\{(t_{si},t)|t_{si}\in T_{si}\wedge t\in T\}$。

下面对其意义进行直观的解释:当系统 (NP,ρ) 开始执行后,第一个有发生权的变迁只有 T_0。T_0 一旦发生就变成了带结束变迁的 Petri 网 Σ 的初始状态,由于位子 b 的设计,能保证系统按 Σ 的发生规律进行。一旦某个 $t_{Fi}(i=1,2,\cdots,m)$ 发生,按照 Σ 的语义,Σ 系统中不应该有任何的变迁再发生。此时,b 中有 token,由于 (NP,ρ) 中 $t_{si}(i=1,2,\cdots,n)$ 的优先级高于其它的变迁,它保证了可将 Σ 中所有位子的残留 token 全部清空。最后,由 T_f 发生,(NP,ρ) 系统结束。从上面的执行过程可以看出,(NP,ρ) 在 Σ 上的投影与 Σ 的动态语义相同,即它们等价。

然后,再按2定义的优先 Petri 网的概念和等价性定理2,将 (NP,ρ) 转化为一个与之等价的不带优先关系的 Petri 网系统 NP_ρ。

由于 Σ 与 (NP,ρ) 等价,(NP,ρ) 又与 NP_ρ 等价,因此 Σ 与 NP_ρ 等价。 □

下面利用正则 Petir 网定义 Petri 网的块和具有块变迁的 Petri 网,以描述扩展的信牌驱动模型中不同的抽象层次。

定义8(Petri 网的块) 设 $PN=\{S,T;F,K,W,M_0\}$ 是一个 Petri 网,$PNB=\{S',T';F',K',W',S_a,S_b,T_a,T_b\}$ 称为是 PN 的一个块,其中:

①$S'\subseteq S,T'\subseteq T$;②$F'=F|_{T'\times S'\cup S'\times T'}$;

③$K'=K|_{T'\times S'\cup S'\times T'}$;④$W'=W|_{T'\times S'\cup S'\times T'}$,

⑤$S_a\in S',S_b\in S'$;⑥$T_a\in T',T_b\in T'$;

⑦$PNB=\{S',T',F';S_a,S_b,T_a,T_b\}$ 是一个正则的 Petri 网;

⑧只有当 T_b 发生后,T_a 才可再次发生。

记为 Γ。

也就是说,Γ 是 $PN=\{S,T,F;K,W\}$ 中满足上述条件的元素构成的一个整体。PN 中所有 Γ 的集合记为 $\Gamma(N)$。由于 Petri 网的块是一个正则的 Petri 网,它有唯一的初始标识、开始变迁、结束标记和结束变迁。只有当结束变迁发生后,开始变迁才可以再次发生,并且,块中的 token 全部清除,所以,一个 Petri 网的块在上层可以抽象地表示为 $(\{S_a,S_b\},tt,\{(S_a,tt),(tt,S_b)\})$,其中的 tt 被称为块变迁。它可用图2表示。

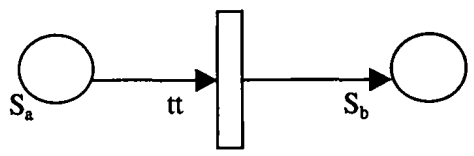

图2 Petri 网的块

有了 Petri 网块的概念,将化简 Petri 网的构造。这种机制有利于描述大而复杂的过程定义问题。此时,对 Petri 网的定义要做如下修正:

定义9(具有块变迁的 Petri 网) $\Pi=\{S,T;F,K,W,M_0,TT\}$ 被称为具有块变迁的 Petri 网,其中:

①$\Sigma=\{S,T;F,K,W,M_0\}$ 是 Petri 网系统;

②TT 是块变迁的集合,$\forall\ tt\in TT,|{}^\bullet tt|=|tt^\bullet|=1$。

从 C/E 网到 P/T 网再到 Pr/T 或着色网是一种提高抽象层次的方法,本文提出的 Petri 网块的概念从另一个角度给

(下转第163页)

与前提条件$(r_1,r_2)\in ATE$相矛盾。命题得证。

②当$(r_1,r_2)\in RTE$

由运行时互斥的定义可直接证明，在此略去。

定理4是用角色互斥实现职责分离的必要条件。

4.3 与角色继承有关的角色互斥性质

定理5 具有继承关系的角色不可能互斥，即任何互斥的角色之间均没有继承关系：

$(\forall r_1,r_2)(r_1,r_2)\in E\Rightarrow\neg(r_1\geqslant r_2\vee r_2\geqslant r_1)$

证明：不失一般性，设$\exists$ s 使得 $r_1\in A(s)$且 $U(s)=u$。

用反证法证明，不失一般性假设 $r_1\geqslant r_2$。

由性质4得，$(r_1\in A(s)\wedge r_1\geqslant r_2)\Rightarrow r_2\in A(s)$

故得如下结论：$r_1\in A(s)\wedge r_2\in A(s)$，这与前提条件$(r_1,r_2)\in E$相矛盾。命题得证。

定理6 任何角色不能多重继承互斥的角色，即若$(r_1,r_2)\in E$，则不存在 r：

$r\geqslant r_1\wedge r\geqslant r_2$。

证明：不失一般性，设$\forall r_1,r_2,r$ 且$(r_1,r_2)\in E$ 且 $r\geqslant r_1$；$\exists$ s 使得 $r_1\in A(s)$。现需证明 $r\geqslant r_2$不成立。

用反证法证明，假设 $r\geqslant r_2$成立。

由性质4得，$(r\in A(s)\wedge r\geqslant r_1)\Rightarrow r_1\in A(s)$

$(r\in A(s)\wedge r\geqslant r_2)\Rightarrow r_2\in A(s)$

故可得如下结论：$r_1\in A(s)\wedge r_2\in A(s)$，这与前提条件$(r_1,r_2)\in E$相矛盾。命题得证。

定理7 一角色若继承了互斥角色中的某一方，则该角色也与另一方互斥，即若

$(r_1,r_2)\in E$ 且$\exists$ $r\geqslant r_1$，则$(r,r_2)\in E$。

证明：由互斥定义得，

$(r_1,r_2)\in E\Rightarrow(\exists\ p_1\in PA(r_1),\exists\ p_2\in PA(r_2))(p_1,p_2)\in Ep$

由角色继承定义得，$r\geqslant r_1\wedge p_1\in PA(r_1)\Rightarrow p_1\in PA(r)$

故$\exists\ p_1\in P_A(r),\exists\ p_2\in PA(r_2)$且$(p_1,p_2)\in Ep$，由角色互斥定义得，$(r,r_2)\in E$。命题得证。

在公文审批中，“起草者”与“审批者”是互斥角色，由定理5可以防止这两个互斥角色中的任一方通过继承另一方而破坏职责分离规则；由定理6和定理7可以防止其他角色通过多重继承而同时进行公文起草与审批。

结束语 本文将角色继承划分为泛化继承和监管继承，分析了由于角色监管继承而可能破坏数据的完整性和一致性、角色缺席可能对系统正常运行造成破坏的问题，在此基础上提出了解决这两个问题的方案。在 RBAC 模型中利用角色互斥可以实现现实领域中职责分离的安全需求，认真研究角色互斥及其性质，可以更好地划分用户职责，防止滥用权利对系统安全造成的危害。本文所研究的内容不仅具有理论意义，而且具有较大的实用价值。本文所研究的内容在我们开发的“法院综合信息管理系统”中得到了很好的应用。

参 考 文 献

1 Yan Han, Liu Feng-Yu, Zhang Hong. An object-oriented model of access control based on role. ACM SIGSOFT Software Engineering Notes, 2000, 25(2): 64～68

2 Sandhu R S, Coyne E J, Feinstein H L, et al. Role-based access control models [J]. IEEE Computer, 1996, 29(2): 38～47

3 Sandhu R, Ferraiolo D, Kuhn R. The NIST Model for Role-based Access Control: Towards A Unified Standard [A]. In: Proc. of 5th ACM2000. Workshop on Role-Based Access Control [C]. ACM, Berlin, Germany, July 2000

4 Simon R T, Zurko M E. Separation of Duty in Role-Based Environments. In: Proc. of Computer Security Foundations Workshop X, Rockport, Massachusetts, 1997

5 Moffett J D. Control Principles and Role Hierarchiees. In: 3rd ACM Workshop on Role-Based Access Control(RBAC). 1998

6 Moffett J D, Lupu E C. The Uses of Role Hierarchies in Access Control. In: 4th ACM Workshop on Role-Based Access Control (RBAC). 1999

（上接第159页）

出了 Petri 网的抽象方法，它可使人们用不同的粒度，对不同抽象级别的概念进行有效的建模，化简了问题的复杂度，使人们在可控制的前提下充分地利用 Petri 网这种直观的形式化工具对问题进行正确的建模和分析。

结束语 本文描述了信牌驱动模型的分层结构（块和子过程）所对应的正则 Petri 网。至此，通过文[6～8]和本文完整地论述了一个功能强大、使用方便的工作流模型——扩展的信牌驱动模型，解决了过程定义的建模问题。同时，也详细地讨论了一个与之对应的形式化模型——正则 Petri 和非确定 Petri 模型，以及将信牌驱动这种工作流过程定义模型转化为了一个 Petri 网的方法。即我们不仅可以用灵活、方便、表达能力强的信牌驱动模型对工作流的过程进行适当、准确和全面的建模，而且，还能将这种模型能转化为一个 Petri 网。这就为进一步对信牌驱动模型进行分析、验证、甚至仿真、执行和其他性能分析等等提供了坚实的理论基础和保障。

参 考 文 献

1 WfMC. The Workflow Reference Model. http://www.aiim.org/wfmc

2 史美林，向勇，杨光信. 计算机支持的协同工作理论与应用. 北京：电子工业出版社，2000

3 范玉顺. 工作流管理技术基础. 北京：清华大学出版社，2001

4 van der Aalst W M P. The Application of Petri Nets to Workflow Management. The Journal of Circuits, Systems and Computers, 1998. 1～53

5 Wang Binjun, Hao Kegang. Three Levels of Workflow Model, in Proc. ISFST2001

6 王斌君. 工作流过程模型的层次研究及其分析：[西北大学博士论文]. 2002. 4

7 岳晓丽，杨 斌，郝克刚. 信牌驱动式工作流计算模型. 计算机研究与发展，2000，37(12)：1513～1519

8 郝克刚，王斌君. 非确定 petri 网. 小型微型计算机系统(已录)

9 Booch G. Object_Oriented Design With Applications. The Benjamin/lemmings, 1991

10 Best E, Koutny M. Petri net semantics of priority system. Theoretical Computer Science, 1992. 175～215

11 Peterson J L. Petri Net Theory and The Modeling of Systems. Prentice-hall, 1981

12 袁崇义. Petri 网原理. 北京：电子工业出版社，1998