

一个 workflow 元模型的分析及其时态扩展

余 阳, 汤 庸, 梁 路, 潘 炎, 汤 娜

(中山大学 计算机科学系, 广东 广州 510275)

E-mail: yuy@mail.sysu.edu.cn; issty@zsu.edu.cn

摘 要: 时态 workflow 将时间作为一个维度引入 workflow 描述. 通过对一个典型 workflow 元模型的分析, 得出时态属性是 workflow 各要素及其关系的普遍属性; 定义了时态信息的规范表示和一组时态数据演算, 并对时态 workflow 中主要元素进行了形式化描述; 用一个例子说明如何运用上述成果进行过程建模, 并处理 workflow 中的时态数据. 结论表明时态 workflow 能综合处理过程的效率问题和角色、数据、应用程序的时间有效性问题.

关键词: 时态 workflow; workflow; 元模型; 时态信息; 时态数据演算

中图分类号: TP311

文献标识码: A

文章编号: 1000-1220(2006)08-1526-05

Analysis and Temporal Extension of a Workflow Meta-Model

YU Yang, TANG Yong, LIANG Lu, PAN Yan, TANG Na

(Department of Computer Science, Sun Yat-sen University, Guangzhou 510275, China)

Abstract: Temporal Workflow introduced time into workflow as a dimension. Firstly, analyzing a typical workflow metamodel showed that temporal attributes were the popular attributes of elements and relationships in workflow. Then, the denotation of temporal elements and a group of temporal data calculations was defined, and some primary elements in Temporal Workflow were formalized. Finally, an example showed how to build a process model by above results, and how to handle temporal data in workflow. The results show that Temporal Workflow can handle the time validity of roles, data, and invoked applications, together with the efficiency of processes.

Key words: temporal workflow; workflow; metamodel; temporal information; temporal data calculation

1 引言

时间是事物的一种基本属性, 许多事实的存在、关系的存在都是与时间相关的. 工作流“在适当的时间将适当的信息传递给适当的人用适当的工具处理”, 构成工作流的诸要素、要素间的关系都与时间相关. 目前对工作流中时间问题的研究仅仅基于过程/活动的时态属性, 本文引入了时态工作流 (Temporal Workflow) 的概念.

时态工作流就是将时间作为一个维度引入工作流系统, 全面、系统地研究工作流系统中各元素及元素间关系的时态特性及其规律.

2 相关研究

工作流应用处于一个四维时空中, 由于网络技术的发展, 空间距离因素对工作流系统转化为对时间要素的影响, 如文献[1][2]提出的工作流模型扩充了对时区 (time zone) 的表示. 时间因素对工作流应用的影响是广泛和重要的, 目前的主要方向有:

2.1 工作流的时间建模和验证

Eder 建立了包含活动时间属性的赋时活动图模型^[3]; SeaLing 等扩展了基于 Petri 网的工作流模型 WF-net, 提出了时间工作流网^[4]; Hai Zhuge^[1]、杜拴柱^[2]、李炜^[5]等在此基础上进行了进一步的研究和扩展.

2.2 工作流的性能

基于赋时过程模型, J. Leon Zhao 等提出了一种过程时间预测算法和任务时间的分配策略^[6]; 李建强等结合开环 Jackson 排队网络理论和马尔科夫链详细讨论了静态、动态性能分析的方法^[7]; 更进一步的, Jorge Cardoso 将提出了工作流的 QoS 模型^[8].

2.3 工作流的时序约束

Kafeza 将工作流活动间的时序关系归纳为七种^[9]; 刘向前等讨论了多工作流过程合并的协调机制^[10]. 综上所述, 目前与时间相关的工作流问题的研究主要集中在工作流的过程建模上, 并基于这些过程模型分析活动或过程的时间约束和时序约束, 其研究的目的是为了提提高工作流实例的执行性能或效率. 但工作流中参与活动或过程的人、工具和数据同

收稿日期: 2005-05-13 收修改稿日期: 2005-11-25 基金项目: 国家自然科学基金项目 (60573160, 60373081) 资助; 广东省自然科学基金项目 (04009746) 资助; 广东省科技计划重点引导项目 (2005B10101007) 资助; 广州市科技攻关引导项目 (2004Z3-D0261) 资助. 作者简介: 余阳, 男, 1966 年生, 副教授, 博士研究生, 主要研究方向为工作流, 时态数据库; 汤庸, 男, 1964 年生, 博士生导师, 博士, 主要研究方向为时态数据库.

样受到时间因素的影响, 时态工作流的研究综合考虑了工作流中所有元素、元素间关系的时态属性, 并研究这些时态属性对工作流相关性性质和问题的影响

3 一个工作流元模型的分析

工作流的元模型是用于描述工作流内部的各个元素、元素之间关系及其属性的。时态工作流研究的第一步就是要将时间特性赋予这些元素、关系

WMC 提出了一个基础的工作流过程定义元模型^[11]; 文献[12]从四个角度来看元模型; 文献[13]描述了一个三维工作流模型; 在此基础上, 文献[14]引入了连接符, 文献[15]引入了活动实例、过程实例等运行期元素形成了动态工作流元模型

基于这些研究成果, 我们提出的工作流元模型包含定义期元模型和运行期元模型, 其中定义期元模型包括: 组织模型、信息模型、应用程序元模型和过程定义元模型, 运行期元模型中的元素基本来自定义期元模型中元素的实例化, 限于篇幅, 不再详述

3.1 组织元模型

如图1所示, 组织元模型描述工作流与资源相关的概念及其关系。所谓资源就是活动的执行者, WFS 在为活动分配资源时, 常常通过组织单元 (Organizational Unit) 和角色 (Role) 的交集来定义, 为此引入单元角色 (UnitRole)。一个用户 (User) 可以是一个具体的人或软硬件代理 (Agent)^[16]。

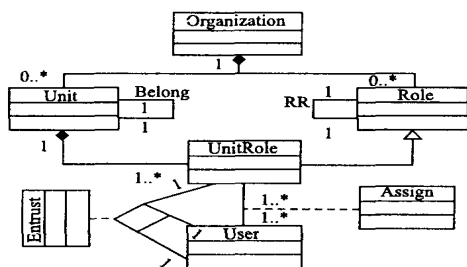


图1 组织元模型

用户和角色存在多对多的指派关系。一个用户可以将自己担任的某个单元角色委托 (Entrust) 给另一个用户, 角色间的关系 (RR) 可以有: 隶属、对等^[15]。

引入时间维: 一个组织有创建时间和有效时间, 如: 一个公司在某个日期注册成立, 营业执照的有效期是十年; 一个用户也有入职时间和有效时间 (劳动合同期); 一个用户担任某个角色有开始时间和有效时间; 一个用户从第二天起出差3天, 将自己担任的某个角色职责委托给另一个用户, 这个委托关系也有创建时间和有效时间 (第二天起3天), 等等

3.2 信息元模型

如图2所示, WFS 中影响流程的主要是相关数据, 信息元模型描述了相关数据的有关概念及其关系。相关数据包括

数据类型 Primitive、数据库字段型 DB_Field 和文件路径型 FILE_Path。表单类型 (Table_Def) 是一组相关数据的结构化表示

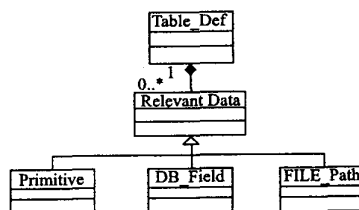


图2 信息元模型

信息模型中相关数据也具有时态属性, 如客户投诉服务系统中, 接到客户投诉即创建投诉单, 投诉必须在3天内得到处理, 于是该投诉单在正常处理流程中的有效时间是3天

3.3 应用程序元模型

如图3所示, 应用程序元模型描述了工作流中被调用程序的相关概念及其关系。一个被调用程序可以是一个常用的

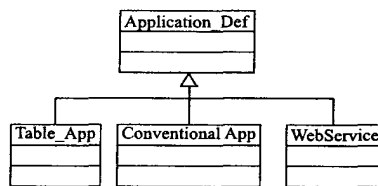


图3 应用程序元模型

表单处理程序、传统的应用程序, 也可以是一个 Web Service 的调用链接。在一个实施了 ISO 9000 的组织里, 作为工具, 应用程序从被批准使用的那一天起, 必须定期检验其有效性、准确性等, 所以, 应用程序也是有有效期的

3.4 过程定义元模型

如图4所示, 过程定义元模型定义了相关概念以描述在

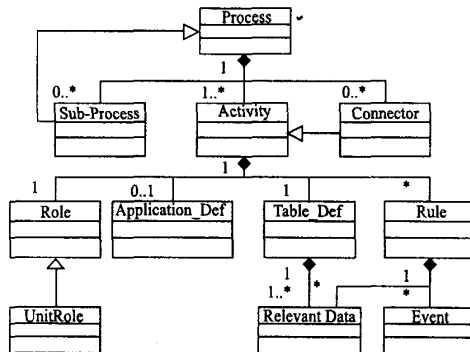


图4 过程定义元模型

过程中要执行哪些活动以及活动执行的顺序。为了支持结构化的过程定义^[17], 该模型引入了子过程的概念。活动是由一个角色使用一种工具处理相应数据的行为, 在活动执行过程

中会有各种状态(如:等待、执行、挂起、完成等).活动状态的变化和活动间的转移都是由事件(Event)触发的.状态规则(State-Rule)决定了活动状态的变化,转移规则(Transition-Rule)决定了活动转移,它们都是由事件和相关数据构成的,可以用ECA的形式描述,统称为规则(Rule).

把汇聚(join)、分支(split)结构和相关约束(AND,OR,XOR)从活动中提取出来,抽象为连接符(Connector).事实上,连接符可以看作是一种具有特定功能的活动.我们可以在活动中仅描述状态规则,而在连接符中仅描述转移规则,这样使结构的变化不影响活动的描述,增强了活动的可复用性,有利于构造柔性的 workflow

在过程模型中,每个版本的过程模版(Process)都有一个创建时间,也会有一个有效时间,它对应着现实中的管理、生产、业务过程,因而是易变的.每个活动的定义也有不同的版本,每个版本的定义也有其创建时间和有效时间.此外,为了管理需要,往往还要定义活动/过程的最早完成时间、最迟完成时间等时间属性^[3].

综合以上对元模型的分析,我们得到了 workflow 的主要元素及元素间的关系,并且它们普遍存在事务(创建)时间、有效时间等时态属性

4 时态 workflow 形式化元模型

4.1 时态信息的表示

为了规范 workflow 中时态信息的描述,我们引入两个时态信息元素(temporal element):

时间点(Time Point):把时间看成一个个孤立的时间点,当时间点的粒度(Granularity)足够小时,这一个个的时间点就可以准确的描述现实世界事件发生及变化的时间. workflow 系统中的事务时间可以用时间点描述

时间区间(time interval):通过描述时间段的起始和终止点来描述时间.也可以用时间区间表示时间点(起始和终止时间点相等).可用 $[T_s, T_e]$ 标记时间区间. workflow 系统中的有效时间可以用时间区间进行描述

为了便于时态信息的处理,我们引入如下基本的时态数据演算:

(1) 基于时间点的时态数据的运算: $<, <=, =, >, >=, !=, \min(t_1, t_2), \max(t_1, t_2)$

(2) 基于时间区间的时态数据的运算: 设 t_1, t_2 是两个时间区间

$\text{begin}(t_1)$: 时间区间 t_1 的开始时间

$\text{end}(t_1)$: 时间区间 t_1 的结束时间

$\text{start}(t_1, t_2)$: $\text{begin}(t_2) = \text{begin}(t_1)$ and $\text{end}(t_1) < \text{end}(t_2)$

$\text{finish}(t_1, t_2)$: $\text{begin}(t_1) > \text{begin}(t_2)$ and $\text{end}(t_1) = \text{end}(t_2)$

$\text{before}(t_1, t_2)$: $\text{end}(t_1) < \text{begin}(t_2)$

$\text{after}(t_1, t_2)$: $\text{end}(t_2) < \text{begin}(t_1)$

$\text{overlap}(t_1, t_2)$: $\text{begin}(t_1) < \text{begin}(t_2) < \text{end}(t_1) < \text{end}(t_2)$

$\text{meet}(t_1, t_2)$: $\text{end}(t_1) = \text{begin}(t_2)$

$\text{equal}(t_1, t_2)$: $\text{begin}(t_1) = \text{begin}(t_2)$ and $\text{end}(t_1) = \text{end}(t_2)$

$\text{during}(t_1, t_2)$: $\text{begin}(t_1) > \text{begin}(t_2)$ and $\text{end}(t_1) < \text{end}(t_2)$

(3) 其它运算: 设 T 表示时间长度, T_v 表示时间区间
 $+: T + T_v \quad [\text{begin}(T_v) + T, \text{end}(T_v)], T_v + T \quad [\text{begin}(T_v), \text{end}(T_v) + T]$
 $-: T - T_v \quad [\text{begin}(T_v) - T, \text{end}(T_v)], T_v - T \quad [\text{begin}(T_v), \text{end}(T_v) - T]$

时态元素及其运算可以封装为类,作为特殊数据类型供时态 workflow 系统使用

4.2 时态 workflow 形式化元模型

基于上述分析,我们将时间维引入 workflow 元模型,即将时态属性全面地赋予元模型中的元素和关系,并以形式化方式描述,形成时态 workflow 元模型.其中对时态属性的描述使用了 4.1 节定义的时态信息元素.限于篇幅,这里仅列出该元模型中的主要元素及关系:

定义 1 一个 workflow 的过程 $WP = \langle \text{Pid}, \text{Pn}, \text{Ver}, \text{P}, \text{AS}, \text{F}, \text{tp}, \text{Tp} \rangle$, 其中: Pid 是该过程模版的标识符; Pn 是过程名; Ver 是过程模版的版本号; P 是一个输入输出节点的集合; AS 是一个活动、连接符和子过程的集合; F 是 AS 到 P 的对应关系集合,活动和子过程只有一个输入和一个输出节点,连接符不限; tp 是 WP 的创建时间, Tp 是 WP 的有效时间

定义 2 一个活动 $WA = \langle \text{Aid}, \text{An}, \text{Ver}, \text{Pid}, \text{Rid}, \text{Tid}, \text{APid}, \text{Lin}, \text{ta}, \text{Ta}, \text{State}, \text{Rule} \rangle$, 其中: Aid 该活动 WA 的标识符; An 是活动名称; Ver 该活动(模版)的版本号; Pid 是该活动所属过程的标识符; Rid 是角色的标识符; Tid 是表单类型的标识符; APid 是应用程序类型的标识符; Lin 是该活动对相关数据的存取权限(这里采用的是基于活动的授权策略); ta 是 WA 的创建时间; Ta 是 WA 的有效时间; State 活动状态的集合; Rule 是一个状态转换规则或活动转移规则的集合

定义 3 活动基本状态集 $\text{State} = \{S_0, S_1, S_2, S_3, S_4, S_5\}$, S_0 表示活动执行所需条件尚未满足,处于等待(Waiting)状态;

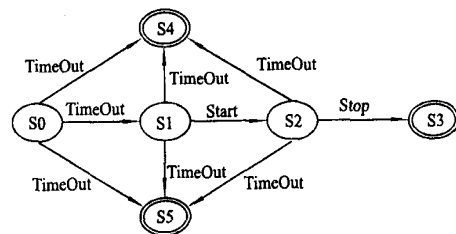


图5 活动实例状态转换图

态; S_1 表示活动处于就绪状态(Ready); S_2 表示活动处于运行期(Running); S_3 表示活动已完成(Completed); S_4 表示活动超时异常(Act_Timeout); S_5 表示数据超时异常(Data_

Timeout). S_0 是初始状态, S_3, S_4, S_5 是终止状态

状态转换如图5(见上页)所示, 活动的状态转换由事件和条件触发, 图5 定义了三种事件:

Timeout, Start, Stop. Timeout 是系统定时超时事件, 如: 可定义该事件每 12 小时发生一次 当活动状态到达终止态时, 触发活动完成事件 Done 状态转换规则见下文实例

定义4 一个过程实例 $p = \langle p_{id}, P_{id}, P, AS, F, PS, tp, Tp \rangle$, 其中: p_{id} 是该过程实例的标识符; P_{id} 是该过程实例所属过程的标识符; P 是一个输入节点的集合; AS 是一个活动、连接符和子过程实例的集合; F 是 AS 到 P 的对应关系集合; PS 是该过程实例当前的状态 (若以 Petri 网建模, 可用该过程实例所有托肯位置的集合来表示状态); tp 是 p 的事务 (创建) 时间; Tp 是 p 的有效时间

定义5 一个活动实例 $la = \langle la_{id}, p_{id}, A_{id}, U_{id}, T_{id}, AP_{id}, ta, Ta, SRule, s \rangle$, 其中: la_{id} 是活动实例的标识符; p_{id} 是 la 所属过程实例的标识符; A_{id} 是 la 所属活动 (模版) 的标识符; U_{id} 是参加该活动实例的用户标识; T_{id} 是要处理的表单标识; AP_{id} 是应用程序标识; ta 是 la 的创建时间; Ta 是 la 的有效时间; $Rule$ 是规则集; s State 表示 la 的当前工作状态

定义6 一个角色 $WR = \langle R_{id}, R_n, C, A, tr, Tr \rangle$, 其中: R_{id} 是角色的标识符; R_n 是角色名; C 是角色能力的集合; A 是该角色存取相关数据的权限 (实际用户对相关数据的存取权限是角色权限 A 和活动权限 L_{in} 的交集); tr 是 WR 的事务时间; Tr 是 WR 的有效时间

定义7 一个表单实例 $It = \langle T_{id}, T_{tid}, DataList, Frame, td, Td \rangle$, 其中: T_{id} 是表单实例 It 的唯一标识; T_{tid} 是 It 所属表单类型的标识符; $DataList$ 一组相关数据的列表; $Frame$ 是数据的结构框架; td 是 It 的事务时间; Td 是 It 的有效时间

5 一个应用实例

为了说明如何在工作流过程建模中使用时态 workflow 元模型定义的元素和关系, 也为了说明时态 workflow 在实践中的作

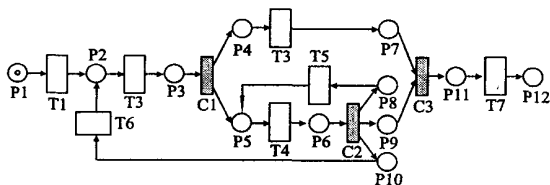


图6 工厂报建审批流程图

用, 这里给出了一个电子政务中处理工厂建设审批的实例: 申请人填写申报资料后, 首先要进行政策法规相关的预审, 若预审通过, 申报材料必须在 30 日内完成环保审批, 否则预审结论失效, 必须重新预审

Petri 网在过程建模中已得到广泛应用, 基于形式化元模型定义的基本元素及其关系, 该例的 Petri 网过程模型描述如

图6, 其中, 活动 T_1 : 填写并提交申请资料; T_2 : 预审批; T_3 : 规划审批; T_4 : 环保审批; T_5 : 活动超时处理; T_6 : 数据超时处理; T_7 : 综合审批; P_1 为开时结点, P_{12} 为结束结点; C_1, C_2, C_3 为连接符

这里仅以活动 T_4 (环保审批) 的处理来说明时态信息的作用 为使讨论清晰起见, 我们仅考虑相关数据和活动的时态性, 暂忽略组织模型中相关元素 (角色、用户等)、应用程序的时态性 定义 T_c 为系统时间, T_a, T_d 见定义 5、7 的描述, 则活动 T_4 的状态转换规则以 ECA 形式描述如下:

规则 1:

```
WHEN Timeout
  IF S = S0 and Tc > begin(Td) THEN
    SetStatus(S1)
  END IF
ENDWHEN
```

规则 2:

```
WHEN Start
  IF S = S1 THEN
    SetStatus(S2)
  END IF
ENDWHEN
```

规则 3:

```
WHEN Stop
  IF S = S2 THEN
    SetStatus(S3)
  END IF
ENDWHEN
```

规则 4:

```
WHEN Timeout
  IF (S = S0 or S = S1 or S = S2) and (end(Ta) < Tc)
  THEN
    SetStatus(S4)
  END IF
ENDWHEN
```

规则 5:

```
WHEN Timeout
  IF (S = S0 or S = S1 or S = S2) and (end(Ta) < begin
    (Td) or overlap(Ta, Td) or end(Td) < Tc) THEN
    SetStatus(S5)
  END IF
ENDWHEN
```

连接符 C_2 对应的活动转换规则:

规则 1:

```
WHEN None
  IF S = S3 THEN
    PassTokenTo(P9)
  END IF
ENDWHEN
```

规则 2:

```

WHEN None
  IF S = S4 THEN
    PassTokenTo (P8)
  END IF
ENDWHEN

```

规则3:

```

WHEN None
  IF S = S5 THEN
    PassTokenTo (P10)
  END IF
ENDWHEN

```

通过该例可以看到,通过引入时态信息和时态数据演算,使得 workflow 系统可以方便地支持 workflow 效率处理和信息有效性处理。综合处理人、信息、工具、过程等要素的时态特性,将使 workflow 系统更加符合现实工作的要求,极大地拓展 workflow 系统的应用范围。

6 结论

本文提出了时态 workflow 的概念,并在对元模型分析、改进的基础上,将时态信息全面、系统的引入到 workflow 描述,为时态 workflow 进一步的研究和开发奠定了基础。时态属性的引入对 workflow 系统的影响是广泛的,今后,时态 workflow 的研究将会涉及如下几个方面:

- 1) 基于元模型和 Petri 网技术的时态 workflow 过程模型研究;
- 2) 时态 workflow 效率和有效性研究。基于信息有效性的效率调整、活动时间的动态分配等问题进行研究将是很有现实意义的课题。
- 3) 时态 workflow 柔性研究。workflow 的动态变化和影响与时间因素密切相关。

References

- [1] Hai Zhu-ge, To-yat Cheung, Hung-keng Pung. A timed workflow process model [J]. The Journal of Systems and Software, 2001, 55: 231-243.
- [2] Du Shuan-zhu, Tan Jian-rong, Lu Guo-dong. An extended time workflow model based on TW F-net and its application [J]. Journal of Computer Research and Development, 2003, 40 (4): 524-530.
- [3] Eder J, Panagos E, Pozewalunig H, et al. Time management in workflow systems [A]. In: Abramowicz, W., Orłowska, M. E., eds [C]. Proceedings of the 3rd International Conference on Business Information Systems Heidelberg, London, Berlin: Springer-Verlag, 1999, 265-280.
- [4] Ling S, Schmidt H. Time petri nets for workflow modeling and analysis [C]. In: Proceedings of the IEEE International Conference on System, Man and Cybernetics 2000, 4: 3039-3044.
- [5] Li Wei, Zeng Guang-zhou, Wang Xiao-lin. A workflow model based on timed petri net [J]. Journal of Software, 2002, 13 (8): 1666-1671.
- [6] J. Leon Zhao, Edward A. Stohr. Temporal workflow management in a claim handling system [J]. Software Engineering Notes, March 1999, 24 (2): 187-195.
- [7] Li Jian-qiang, Fan Yu-shun. A method of workflow model performance analysis [J]. Chinese Journal of Computers, 2003, 26 (5): 513-523.
- [8] Jorge Cardoso, Amit Sheth, Krys Kochut. Implementing QoS management for workflow systems [R]. Technical Report, LSD IS Lab, Computer Science, University of Georgia, July, 2002.
- [9] Kafeza E, Kamalakara K. Temporally constrained workflows [C]. In: Internet Applications Lecture Notes in Computer Sciences 1749, Springer-Verlag, 1999, 246-265.
- [10] Liu Xiang-qian, Wang Xiao-lin, Zeng Guang-zhou. Multi-workflow combining method based on coordination mechanisms [J]. Computer Engineering, 2003, 29 (2): 118-119, 194.
- [11] Hollingsworth D. The workflow reference model [J]. Workflow Management Coalition, 1995.
- [12] Rob van Kaathoven, Manfred A. Jeusfeld, et al. Organizational memory supported workflow management [M]. Electronic Business Engineering Heidelberg: Physica-Verlag, 1999, 543-563.
- [13] W. M. P. van der Aalst. Workflow verification: finding control-flow errors using petri-net-based techniques [Z]. Business Process Management, Lecture Note in Computer Science 1806 USA: 2000: 161-183.
- [14] Zhao Wen, Hu Wen-hui, Zhang Shi-kun, et al. Study and application of a workflow meta-model [J]. Journal of Software, 2003, 14 (6): 1052-1059.
- [15] Sun Rui-zhi, Shi Mei-lin. A meta-model supporting dynamic changing workflow [J]. Acta Electronica Sinica, 2002, 30 (12A): 2052-2056.
- [16] Dickson K. W. CHIU, Qing Li, Kamalakara Karlapalem. A meta modeling approach to workflow management systems supporting exception handling [J]. Information Systems, 1999, 24 (2): 159-184.
- [17] Johann Eder, Wolfgang Gruber. A meta model for structured workflows supporting workflow transformations [C]. LNCS 2435, Springer-Verlag Berlin Heidelberg, 2002: 326-339.

附中文参考文献:

- [2] 杜柱柱, 谭建荣, 陆国栋. 一个基于 TW F-net 的扩展时间 workflow 模型及其应用 [J]. 计算机研究与发展, 2003, 40 (4): 524-530.
- [5] 李 伟, 曾广周, 王晓琳. 一种基于时间 Petri 网的工作流模型 [J]. 软件学报, 2002, 13 (8): 1666-1671.
- [7] 李建强, 范玉顺. 一种工作流模型的性能分析方法 [J]. 计算机学报, 2003, 26 (5): 513-523.
- [10] 刘向前, 王晓琳, 曾广周. 基于协调机制的多工作流过程合并方法 [J]. 计算机工程, 2003, 29 (2): 118-119, 194.
- [14] 赵 文, 胡文惠, 等. 工作流元模型的研究与应用 [J]. 软件学报, 2003, 14 (6): 1052-1059.
- [15] 孙瑞志, 史美林. 一个支持动态变化的工作流元模型 [J]. 电子学报, 2002, 30 (12A): 2052-2056.