

虚拟企业中动态的过程知识管理的研究

The Dynamic Process Knowledge Managements in Virtual Enterprises

王 茜

(东南大学计算机科学与工程系 南京210096)

Abstract In this paper, the importance and phases on the management of dynamic process knowledge are mentioned in view of dynamic changes of business processes in virtual enterprises. Recently, much attention from both Workflow area and CSCW area has been paid to the integration of the definition and the execution on flexible, dynamic (or emergent) enterprises' processes. Two typical systems that come from two areas are introduced. The system architectures, common properties and their contributions on the process knowledge management are discussed.

Keywords Process knowledge, Virtual enterprise, Emergent process, Coordination

1 概述

市场的全球化和竞争的日益剧烈,迫使若干个企业常常会为了生产某个或某几个产品在短时间内组成联盟——虚拟企业,当这个或这些产品不再赢利时,这种联盟就解散。在这种动态联盟中每个企业是业务伙伴。这些业务伙伴在地理位置上是分散的,可以是不同地域,不同国家。各业务伙伴各自独立运行在异构、分布的平台上。由于新产品的生命周期越来越短,企业为了要以新产品抢占市场,必须要不断地开发新产品。为了适应新产品的需要,合作伙伴会不断变更,业务过程也会作相应的调整以降低成本,缩短时间,提高产品的性能和质量。

2 动态的过程知识

2.1 过程知识

有两种形式的过程知识,第一种是潜在的/内在的

知识(tacit /internal knowledge)。个人的经验、个人对于领域的一些心照不宣的理解,例如,个人对于如何完成项目、如何工作的认识。团队对于某一事务的共同的,这些理解隐含在他们日常工作的习惯中,或者在他们的潜意识中。这些知识存在于意识,不可捉摸^[4]。每个人都有自己的对事物的认识,不同的人有不同的观点。在虚拟企业中任何一个人要想成为所有领域的专家是不可能的。只有通过团队的协同交互,个人的潜在知识才能得到提取、分析并综合成为整个企业的资源。

第二种是明确表示的知识^[4]。对潜在的知识进行记录,使它们形式化,表示为业务过程,这种外部化了的的知识中含有团队成功执行任务的详细内容,而且它还和“为什么”(why),“如何”(how)有关——而不仅仅表示“什么”(what)^[5]。这样所获得的业务过程就是外部化了的过程知识。经过评估之后,这个业务过程可以作为模板存储在过程库中。

2.2 紧急出现的过程

一个插件用于多媒体教学或演示系统中,COM 为其它组件正确调用它的功能提供了保障机制。

参考文献

- 1 Williams S, Kindel C. the Component Object Model A Technical Overview [M] <http://www.microsoft.com>, 1994. 10
- 2 Rafail A, Shohoud Y 著. COM 与 COM+从入门到精通 [M]. 邱中潘等译. 北京:电子工业出版社,2000. 4
- 3 Armstrong T, Patton R 著. ATL 开发指南[M]. 董梁, 丁杰, 李长业等译. 北京:电子工业出版社,2000. 11

丰富的集成服务、各种易用工具 and 应用程序,它还提供了很好的可复用性,在具有了二进制和网络标准后,COM(包括 DCOM)打开了编程思维上的革新之门。

组件对象模型的核心是对于组件和它们的客户之间如何交互的具体描述。作为一个具体描述,它定义了诸多软件组件间交互的标准。此外,COM 还是一个包含于组件对象库中的具体实现。

我们实现的共享电子白板系统利用了 COM/DCOM 对 RPC 和多线程的独特支持。它可以作为一个独立的系统,COM 为将来升级这个系统,实现更复杂的功能(如支持声音等)提供了方便;它也可以作为

在过程定义阶段,会出现如下情况:

- 由于整个过程的复杂性,不可能定义所有可能的工作步骤和过程控制^[6];
- 不可能用编程语言描述所有可能出现的条件选择和可能的路径;
- 很可能错误地理解了现实世界的业务过程,于是过程本身就没有能够很好地反映真实情况,而在执行过程中发现了这些错误,必需进行修改;
- 系统所要求的资源在现实世界得不到满足,发生冲突;
- 所预料的某些任务的执行的结果与实际产生的结果不相吻合;
- 所需求的业务环境与过程执行中的业务环境发生变化。

所有这一切都导致了在项目的执行过程中会出现某些紧急事件,而这种紧急情况不是通常在编程语言中用一些例外处理的条件选择所能解决的。例如,顾客临时改变了需求,或者某个特殊的零配件不可能获得,或者某些所需要的资源短缺,于是需要重新设计、重新组织合作伙伴。为了适应新的情况,过程的结构就要在执行过程中动态地修改,而且修改的结果必须进行记录、保存,以便于可追溯(traceability),作为业务过程的历史资料、经验知识供以后借鉴。所有的合作伙伴都要重新学习修改过了的过程,不断地调整自己,然后继续执行工作流程所规定的任务。

紧急情况下对过程模型的在线修改使得过程的在线定义、学习和执行搅合在一起了,正是这种搅合扩展了传统工作流的定义^[7]。WfMC 定义工作流^[8]:"当任务或信息按照过程规则从一个参与者传到另一个参与者执行时,全部或部分业务过程自动化",通过把过程定义、执行和柔性地协同紧急情况集成在一起,把工作流的定义扩展到了柔性的协同,而不完全依赖于结构化的预定义。也就是说在工作流定义阶段并不追求对业务流程的完美定义,允许存在不完全的、不一致的过程结构^[9],留待执行的过程中再不断地完善。

3 过程知识管理的四个阶段

对过程知识的管理贯穿于整个产品的生命周期,可以分为四个阶段:

- 第1阶段,通过协同交互获得与具体项目相关的知识,认识项目的性质、目标、结果、延续时间、使用哪些资源(包括人力资源和设备)、各协作伙伴所扮演的角色以及这些角色之间如何协同工作。这些都是企业实现目标、完成任务的重要资源。
- 第2阶段,外部化合作伙伴的共识,使之成为记录下来的清晰的知识。这就是通常所说的企业建模阶段。

企业的过程知识表达成可读的、可视的图形化的过程结构。建模的过程可以以过去的某个过程模型为参考(referencing)进行某些修改,也可能对过去的某个过程模型进行部分替换或者用另外的模型中的部分进行重新组装(combining),也可能从头开始建立(creating/representing)一个全新的模型。过程知识包括产品(项目的结果)、组织和人员(人力资源)、工作流程和任务逻辑以及系统的结构、平台、人机界面和工具^[5]。这个模型很清楚地表明:各任务执行的顺序,谁执行哪个任务,每个任务中使用哪些资源,每个任务究竟如何执行以及为什么要这么执行。这样一个计划已经不仅仅是常规地完成一个任务的工作计划了,它是一个知识的集成器,它集成了各合作伙伴的经验/知识,企业的规则/文化,日常的工作习惯/模式,甚至于专业知识。

• 第3阶段,企业的业务过程的知识向所有参与者传播。要让每一个合作伙伴以及企业内部的每个参与者都学习企业的知识。

• 第4阶段,正式执行——知识起作用。正式执行和模拟执行的区别仅在于放入了工作对象和具体的属性值。工作对象是任务执行过程中要处理的应用数据。当过程实例运行时可能会遇到紧急情况,要对模型进行修改以适应新的情况。所有的参与者必需通过学习调整他们自己以适应新的过程。这时过程的定义、学习和过程的执行就搅合在一起。当产品交付顾客使用以后,顾客可能会发现一些问题并给出反馈意见,对这些反馈意见进行分析,找出原因。如果原因出自工作流程,那么就要对过程模型进行修改,或者作必要的说明。

当前对这种企业知识管理系统的支持主要来自两个方面的努力,它们是:工作流(Workflow)领域、计算机支持协同工作(CSCW)领域^[10]。工作流领域主要研究的是对过程活动的协调、调度,对人们工作过程的追溯。最近工作流领域开始从工作流管理的角度出发把目标集中在对柔性、可调整(adaptive)的过程的支持上,允许模型在过程的执行中作改变^[11]。例如 Mobile^[12], Baan^[13], WIBO^[14], ADOME^[15], AIS Workware^[7,9]等系统都或多或少地提供了对动态、柔性处理的支持;CSCW 领域主要研究的是为团队提供通信和协同的支持。最近他们也把注意力集中到了对一些更加不固定的、没有很好定义的紧急过程的支持上,他们从协同的角度出发,使得过程的定义和执行更加柔性并获得集成,例如, COORDINATOR^[16], COW-FISH^[17], chips^[18], XCHIPS^[2,3]等系统都能提供共享的信息空间支持业务过程的动态、协同的定义和执行。下面简单剖析欧共体 IST (Information Societies Technology Programme) 资助的 EXTERNAL^[20] (Extended

Enterprise Resources, Network Architectures and Learning) 项目中两个来自不同领域的典型的系统 AIS Workware 和 XCHIPS, 比较它们在过程知识的学习和管理中的特点。

4 两个典型的系统

4.1 AIS Workware

AIS Workware 是挪威 SINTEF AIS (Advanced Intranet Collaboration) 的一个在研项目^[7]。它的主要目标是要支持动态和随机的 workflow, 这种动态性和随机(ad hoc)性在当今知识密集的企业中显得越来越重要^[7,8]。系统从 workflow 的角度出发集中解决 workflow 定义和执行的集成, 也就是说, 系统通过支持可变的过程建模和不完整的模型的执行从参考模型自底向上支持对过程知识的管理。

4.1.1 过程建模 其参考模型定义了过程知识表达的四个层次(如图1所示), 它们是: 过程逻辑描述层、活动详细说明层、工作管理层以及工作执行层。工作执行层就是具体地按照定义的工作流来执行任务, 但是执行中有很大的自由度, 当一个团组或个人执行任务时, 究竟下一步如何执行完全由他(们)自己决定。

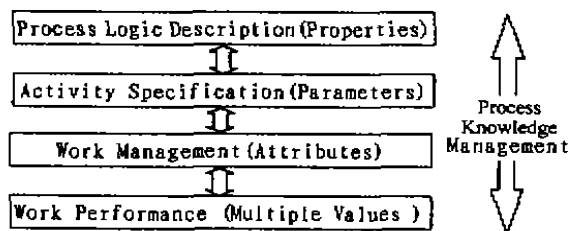


图1 过程建模的参考模型

知识管理是指对过程的收集, 这些过程将会对提高协同的整体效果、过程知识的创新、传播和使用起着显著作用。过程知识的管理在每一层上都起作用, 尤其是参考模型中最下的一层, 以往的典型过程实例将会对参考模型中上层的过程模板(template)产生影响, 它可能会修改上层的过程模板。而上层的过程模板存入模板库以后又可以作为一种资源——这种情况下的过程知识——供给将来使用。所以在参考模型的每一层都有集成的环境支持过程知识的管理, 也就是说在建立模型时(无论是预定义阶段还是在紧急处理阶段), 用户在创建知识, 这种知识可以被重用, 收获并和已有的知识相结合。

过程建模中采用图形化的、层次的、可分解的结构化的建模语言, 过程模型中的最基本的概念是工作项

(work item), 它既可以表示一个任务又可以表示一个过程。所有的工作项都在工作表(work list)中注册。每一个工作项都有一个工作台(work top), 通过工作台可以访问工作项的三种服务, 即三种功能: 计划服务, 构建过程模型; 执行服务, 提供对执行时所必要的工具和信息对象的访问; 协调服务, 提供工作项状态改变的信息, 以便观察工作的进展, 同时它还提供对外部通信的工具用以支持随机的协同工作。

4.1.2 过程的交互执行 AIS Workware 系统强调过程执行时的人机交互, 它把执行引擎称为人机交互机, 软件系统和用户之间协同工作, 共同把业务过程按照当时的具体情况向前推进, 而不像传统 workflow 系统在例外情况处理中 workflow 引擎在许多条件语句中进行选择。AIS Workware 系统的执行引擎是构建在工作项状态变迁模型、外部界面以及对协调事件的解释之上的, 用户在任何时候都可以改变状态变迁模型, 改变对协调事件的解释, 从而达到紧急处理的目的。

用户通过触发工作项的协调服务(即改变工作项的状态)来激活一个过程, 也可以通过内部执行组件对事件进行翻译激活一个过程。每一个流都是一个过程实例, 都反映了一定的具体执行情况, 实例之间互不影响, 过程执行中, 引擎判断由用户触发的工作项状态的变化, 然后自动地激活工作的流动或者使得新的工作项处于准备状态。假如一个活动细化分解的子模型中仅仅是一些工作项而不包含这些工作项之间的关系, 那么用户就可以自由地根据具体情况来定义这些工作项之间的关系。用户之间可以通过共享的工作表相互通信, 另外感知监视器可以把所发生的行动及时地通知每一个参与者。过程模型中的决策连接器(decision connector) AND, OR, XOR 决定了过程流动的路径。同样地, 这些连接器也是既可以预先定义也可以在过程的执行中根据具体情况来定义。

4.2 XCHIPS

XCHIPS 是德国 GMD-IPSI/Concert^[2,3] 的一个在研项目。它的主要目标是通过提供一个基于超媒体的协同环境来支持柔性的过程建模、过程学习和过程执行的集成。系统从协同工作的角度出发, 集中解决柔性的、协同地协调工作。这里所说的柔性指过程定义的柔性、执行的柔性、学习的柔性以及这些活动之间转换的柔性。

4.2.1 过程建模 它定义了过程知识表达的三个层次(如图2所示), 它们是: 元模型层、过程模型层以及过程实例层。元模型层提供了基于超媒体的知识共享的建模对象, 节点(node), 链接(link), 支持同步或异步通信的群件工具以及作用于过程模型语义网络的

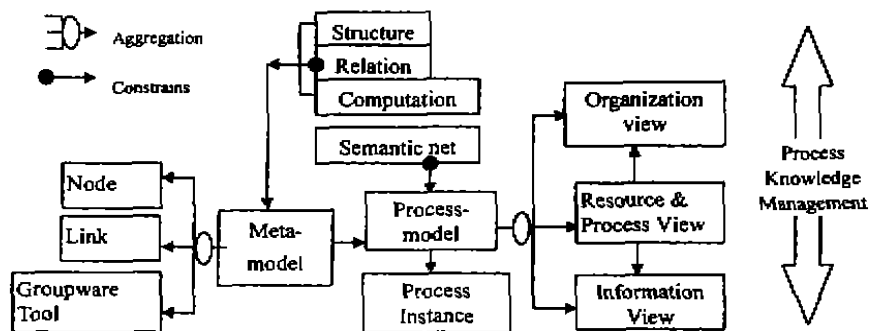


图2 过程模型的三个层次

结构约束、关系约束和计算约束。这些超媒体的对象以及对象之间的约束关系作为建模的框架(平台)为过程建模给出了一个知识共享的空间。

过程模型层向用户提供了四个视图来具体描述过程的知识^[21,22],它们是过程视图、资源视图、信息视图和组织视图。在过程视图中主要描述任务以及它们之间的逻辑,包括任务到子任务的分解关系;在资源视图中主要描述完成任务所需要的人力或设备、工具;在信息视图中主要描述虚拟企业的目标、战略、产品以及背景信息;在组织视图中主要描述完成任务所需要的组织、人员或者虚拟企业的组织概况。所有的这些描述都可以通过超媒体的连接形成层次结构或者通过系统提供的组件工具与 Word 文件、Html 文件连接并采用组件工具文件夹(folder)对文件分类保管;对过程模型复制可以生成过程实例,为了执行过程实例必须加入该过程所对应的应用数据(工作对象)并给出具体的运行属性。如果过程实例很有价值,它也可以作为过程模板保存起来,这样就完成了从模型下层向上层的抽象和提升。

从图中可以看出对过程的动态知识管理贯穿在模型的各个层次:在元模型层定义了所有共享对象的基本知识、属性、方法以及约束;过程模型层从4个视图形式化、图形化、结构/非结构化地描述过程的知识;过程实例层蕴涵了各种执行的具体情况以及非预料执行环境中的过程处理。过程知识主要包括两个方面,一个是关于所进行的项目的知识,它包括项目的类型、目标、结果、时间、需使用的资源、各合作伙伴所起的作用以及它们之间如何合作;另一个是关于项目如何执行的知识,它包括任务执行的次序、任务的执行者、使用的资源以及为什么和如何完成任务。

4.2.2 过程的交互执行 过程实例在执行中是采用人机交互的方式进行的。用户通过触发“任务”节点,使其状态(“准备”、“活动”、“完成”、“停止”)发生改变,当处于“活动”状态时,过程的执行引擎自动触发执

行软件(假如有的话),或者通知该任务相应的操作者执行任务。任务完成之后,该节点的状态变为“完成”,这个状态的改变自动触发由“连接”所引导的下一个任务改变状态为“准备”。如果一个父任务具有下一层的分解,当父任务状态变为“活动”时,它的下层分解中的第一个任务节点(起始节点)就会自动改变状态为“准备”。当所有的下层节点的任务都“完成”以后,上层的父节点就自动改变状态为“完成”。用户可以在任何时候,对任何还未执行的任务或任务逻辑进行修改,调整使之适合实际的、具体的运行环境和临时发生的任何情况,从而达到最大的柔性。如果有必要,修改后的过程实例将可以作为过程模板保存在过程库中供以后使用。由于用户是在同一个环境中在模型(实例)上进行过程知识的定义、学习以及执行的,仅仅是各个阶段环境的状态不一样:定义阶段,图形编辑器处于工作/活动状态;学习阶段,任务的具体执行被关闭;执行阶段,加入工作对象和具体参数,从而实现过程定义、学习和运行环境的统一、不同阶段的自由切换。

4.2.3 多用户的合作、协同 XCHIPS 系统是一个超媒体的协同工作的系统,系统提供了异地、异种平台协同感知的功能,使得工作在同一空间的合作伙伴互相感知对方的存在。系统提供了信息广播的功能,使得任何用户对共享对象的任何操作都能及时地反映到所有的合作者,保证了 WYSIWIS。群件工具,例如共享空间、共享白板、共享文件编辑器,使得合作伙伴可以共同访问、创建和修改非结构化的信息,实现协同工作。超媒体和工作流特性的混合使得用户不仅可以共同访问非结构化的信息,还可以协同定义和执行结构化的过程,两者之间平滑过渡。

讨论 两个典型的系统从两个不同的研究领域出发实现了共同的目标:柔性地支持动态过程知识的创建和过程的执行。它们最大的共同特点是,人机交互执行、过程活动/任务之间的松耦合。以往的工作流系统都强调过程的自动化,要求工作流引擎能够自动地根

据工作流的描述判断下一步的执行任务。于是,为了解决各种例外,以往许多系统都采用一系列的条件语句供系统进行路径的选择,但是这些都只能对期望中的例外情况进行处理,而对真正的例外情况,即非预期的例外情况,收效甚微,反而增加了系统的复杂性和减少了程序的可读性^[6]。Fabio Casati^[22]提出三种解决的方法:一是使系统变傻(fool the system),就是说负责执行的工作流处在工作流的控制之外,这样工作流停顿以后再执行,业务过程仍然能够继续进行;二是修改运行的工作流实例,就是说工作流实例按照例外情况进行修改,但是它不影响其它的实例执行;三是修改工作流定义。上述的两个典型系统采用人机交互执行,由人触发任务状态,实际上就是把操纵任务执行的控制从工作流引擎转移到了人机交互,使活动/任务之间的关系松散(通过加入人的交互),实现了有限过程自动化。

过程知识的管理是贯穿在产品生命全周期的。从两个系统的模型层次上可以看出知识管理从顶层的元模型层直到底层的实例层。从两个系统所强调的动态、柔性来看,它们更重视知识的动态性/活动性,动态的创建、修改、学习、使用、保存以及再激活、再使用(用作参考、用于结合或者直接使用)。过程知识是虚拟企业知识的集成器,它综合了企业的经验、规则/规程、文化、目标以及成功完成任务的细节。对过程知识的管理的每一个阶段都伴随着学习,只有通过学习才能产生对知识的理解,才能更好地管理知识。

致谢 感谢 GMD-IPSI/Concert 研究室对作者短期访问期间在学术上、研究环境上提供的支持和帮助。

参考文献

- 1 Kouloupoulos T. Knowledge Management Defined. <http://www.delphigroup.com/articles/1998/KM-defined.htm>
- 2 Haake J. Cooperative Learning of process knowledge: Making an argument for Cooperative Hypermedia-based Learning Environments. Submitted
- 3 Haake J. Facilitating Orientation in Shared Hypermedia Workspaces. In: Stephen C. Hayne, ed. Group'99. Proc. of the Intl. ACM SIGGROUP Conf. on Supporting Group Work. New York ACM Press, 365~374
- 4 Murray P. C. Information, Knowledge and Document Management Technology. <http://www.ktic.com/top-ic6/12-INFKM.HTM>
- 5 Lillehagen F, Karlson D. Visual Extended Enterprise Engineering And Operation Embedded Knowledge Management and Work Execution. IEMC'99-IFIP Intl. Enterprise Modeling Conf. Vardal, Norway, June 1999
- 6 Bernstein A, Dellarocas C, Klein M. Towards Adaptive Systems CSCW-98 Workshop Report. SIGMOD record, 1999, 28(3)
- 7 Carlsen S, Jorgensen H D. Emergent Workflow: The AIS

- Workware Demonstrator. <http://ccs.mit.edu/klein/cscw98/paper08/>
- 8 WfMC WfMC -Workflow Handbook 1997, edited by Peter Lawrence. Workflow Management Coalition, John Wiley & Sons Ltd, 1997
- 9 Jorgensen H D, Carlsen S. Emergent Workflow: Planning and Performance of Process Instances. Workflow Management '99, University of Muenster, Germany, 1999
- 10 METIS 2x White Paper <http://www.motis.no/>
- 11 Kavakli V, Loucopoulos P. Goal-driven Business Process Analysis Application in Electricity Deregulation. Information System, 1999, 24(3):187~208
- 12 Heint P, Horn S, Jablonski S, et al. A Comprehensive Approach to Flexibility in Workflow Management Systems. Software Engineering Notes, 1999, 24(2)
- 13 Meijer T D, Kessels H, Vunjt C, Comte R I. Realizing Run-time Adaptable Workflow by Means of Reflection in the Baan Workflow Engine. <http://ccs.mit.edu/klein/cscw98/paper20>
- 14 Fakas G, Karakostas B. A Workflow Management System Based on Intelligent Collaborative Objects. Information and Technology, 1999, 41, 907~915
- 15 Dickson K, Chiu W, Li Qing, Karlapalem K. A Meta Modeling Approach to Workflow Management Systems Supporting Exception Handling. Information System, 1999, 24(2):159~184
- 16 Medina R. Tutorial: State of the Art in workflow management research and products. A tutorial presented at ACM SIGMOD Intl. Conf. on Management of Data, June 1996
- 17 Wang W, Haake J. Supporting User-defined Activity Spaces. In: Proc. of ACM Hypertext'97, April 1997. 112~123
- 18 Haake J M, Wang W. Flexible Support for Business Processes: Extending Cooperative Hypermedia with Process Support. Information and Technology, 1999, 41, 355~366
- 19 Krogsue J, Carlsen S. An Integrated Modelling Approach for Process Support. In: Hawaii Intl. Conf. on System Sciences (HICSS-30), Maui, Hawaii, 1997
- 20 Information Societies Technology Programme: Annex 1 - Description of work. Proposal Number: IST-1999-10091
- 21 Wang Weigang, Haake J M. Flexible Coordination with Cooperative Hypermedia. In: Proc. of ACM Hypertext'98 (HT98), June, 1998. 245~255
- 22 Ellis C, Nutt G. Multi-dimensional workflow. In: Proc. of the IDPT'96, 1996
- 23 Casati F. A Discussion on Approaches to Handling Exceptions in Workflows. SIGGROUP Bulletin, 1999, 20(3)
- 24 Faustmann G. Enforcement vs. Freedom of Action An Integrated Approach to Flexible Workflow Enactment. SIGGROUP Bulletin, 1999, 20(3)
- 25 Dickson K, Chiu W, Li Qing, Karlapalem K. Exception Handling with Workflow Evolution in ADOME-WFMS: a Taxonomy and Resolution Techniques. SIGGROUP Bulletin, 1999, 20(3)