

workflow 管理系统互操作技术研究*

周世杰** 秦志光 刘锦德

(电子科技大学计算机科学与工程学院 成都 610054)

【摘要】介绍了 workflow 技术和 workflow 参考模型，讨论了该参考模型的重要组成部分及其功能特点，提出了 workflow 管理系统互操作的四种典型设计模式。在参考模型和典型设计模式的指导下，详细分析了 workflow 互操作性的分类与实现技术以及目前 workflow 互操作性存在的问题，从而在一定程度上解决了 workflow 管理系统的互操作性。

关键词 workflow; workflow 管理系统; 互操作; 分布式计算

中图分类号 TP393

Study on Interoperability of Workflow Management System

Zhou Shijie Qin Zhiguang Liu Jinde

(College of Computer Science and Engineering, UEST of China Chengdu 610054)

Abstract The application of the workflow management in IT fields is discussed generally. The basic concepts of the workflow are studied in detail. After put forwarding the workflow management reference model, its components and characteristics are probed into. Then, according the reference model, the three kinds of interoperability among (between) the workflow engines are referred. Through analyzing four general design modes of the workflow interoperability, their characteristics and features are discussed and compared. Then, the classification of the interoperability of the workflow is elucidated. The two ways to realize the interoperability of the different workflow system are addressed. Finally, the shortcomings and the trends in the future of the current workflow interoperability are also touched on.

Key words workflow; workflow management system; interoperability; distributed computing

workflow 管理的主要特点是使管理过程自动化，使人及各种应用工具相互协调工作，达到“在适当的时候把适当的信息传递给适当的人”的目的。这正符合现代企业强调以过程为中心的管理模式，成为企业在信息集成的基础上实现过程集成的有效途径之一。随着各种 workflow 管理系统产品的问世，如何保证各种 workflow 管理系统技术之间的互操作性，以减少用户的资金投入、扩大 workflow 产品选择范围，以及更好地实现企业(跨企业)的过程集成，逐渐成为 workflow 领域研究的热点^[1,2]。workflow 互操作(性)是指两个或者多个 workflow 引擎之间通过通信和协作的方式来协同工作的能力。从广义的角度看，也可将 workflow 各功能部件通过接口与 workflow 引擎之间交互的能力视为 workflow 互操作性。为了实现 workflow 系统之间的互操作性，必须对 workflow 模型、workflow 互操作分级、workflow 互操作性典型设计模式和 workflow 互操作实现技术进行深入研究。

2001年11月5日收稿

* 四川省科技攻关项目

** 男 30岁 博士生

1 workflow 技术

workflow是指整个或部分业务过程，在计算机支持下的全自动化或半自动化^[3]。workflow所关心的是业务过程的自动化，在这些业务过程中，文档、信息或任务按照预定的一组规则在所有参与者之间传递，实现或有助于实现整个业务过程。过程(也叫业务过程，Process)是相关联的一组流程或活动，它们相互协作，以实现一个企业的某个业务目标或政策目标。workflow的实现是由workflow管理系统WfMS(Workflow Management System)来完成。workflow管理系统是一个运行计算机软件程序的系统，它完全定义、管理和执行workflow。而系统中软件的执行顺序，则用计算机信息表达方式描述的workflow逻辑来驱动^[5]。在WfMS中，过程(业务过程)被分解成一个一个的活动(Activity或Work Activity)，该活动(工作活动)代表了过程中的一个逻辑步骤。因此，workflow管理系统实质上是通过管理工作活动的顺序，调用与各工作活动步骤相关的人工或IT资源来实现业务过程的自动化。一个完整的WfMS应具备三个方面的功能特征：建造功能、运行控制功能与运行交互。

在workflow管理系统中，既涉及到许多数据的交换，也涉及到应用程序之间的交互(包括功能部件之间的交互)。为满足实现企业经营过程重组的需要及考虑操作的灵活性和workflow产品最终用户方便集成的要求，很有必要对workflow管理系统进行标准化。国际workflow管理联盟(Workflow Management Coalition, WfMC)定义了workflow的参考模型，如图1所示。

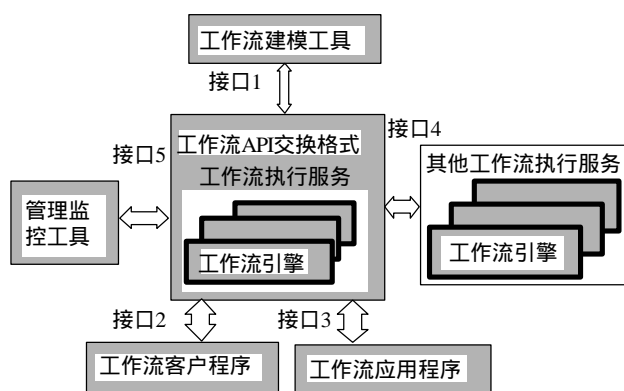


图1 workflow管理系统参考模型

由图1可知，workflow参考模型实际上由两个部分组成：1) workflow功能部件；2) 接口。workflow功能部件完成过程的自动化或半自动化，接口则完成功能部件之间以及workflow引擎之间交互的功能。

根据workflow参考模型，workflow互操作有三种模式：相同workflow执行服务的不同workflow引擎之间的互操作；不同workflow执行服务的workflow引擎之间的互操作和不同workflow系统的workflow引擎之间的互操作。其中不同workflow系统的workflow引擎之间的互操作性实现难度很大。

2 workflow 管理系统互操作模式

随着workflow产品市场的飞速发展，各种workflow系统纷纷进入市场。在规划workflow互操作性时，必须考虑对这些遗留系统的兼容。为此，设计workflow互操作性时，不可能要求workflow厂商为了互操作，放弃向用户提供功能更强大的workflow系统，而只能关注一些典型的workflow互操作场景，提出workflow互操作的典型设计模式。这样既允许厂商实现简单的workflow互操作，也使一些有技术势力的厂商能够实现复杂的workflow互操作，从而满足不同的用户需求。典型的workflow设计模式有链式、网状同步子过程式、网状轮询子过程式和事件同步式。

2.1 链式设计模式

链式workflow互操作设计模式允许某个工作流过程实例在过程内部的一个节点触发其他workflow实例，从而实现workflow之间的互操作，其设计模式示意图如图2所示。

由图2可知，workflow引擎A的过程实例A在节点A₃触发workflow引擎B中的过程实例，从而将过程实例A的一个实体(工作项、过程实例、或者是一个活动)传递到过程实例B的workflow环境中，并在该环境中独立执行。一旦该过程实例开始执行，过程实例A可能终止，也可能继续执行。实体在workflow环境中传递的方式可通过workflow应用网关的数据映射，也可通过标准的API进行。需注意的是，

连接点的选择应该依据 workflow 原对象模型, 即传递的实体必须满足原对象的条件。

2.2 网状同步子过程设计模式

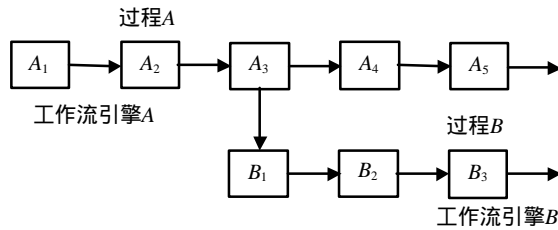


图2 链式互操作设计模式

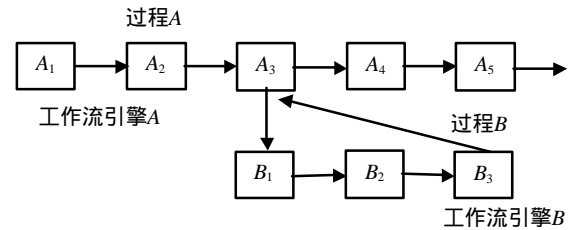


图3 网状同步子过程互操作设计模式

在网状同步子过程 workflow 互操作设计模式中, 允许特定 workflow 环境中执行的一个过程, 被封装为一个独立的任务, 并在其他 workflow 环境中去完成, 其设计模式的示意图如图3所示。

由图3可知, 过程实例A的一个活动A₃, 在另外一个 workflow 引擎B中执行。当在过程实例B中执行时, 过程实例A仍然保持活动状态。当过程实例B完成后, 将控制返回到 workflow 引擎A中, A继续执行其他活动。两者的同步是通过过程实例参数改变信号或者状态改变信号来实现。在这种设计模式下, 传递的是一个独立的子过程, 这种传递方式可以是网关映射的方式, 也可以是API的方式。此外, 在这种子过程的方式中, 可能出现子过程的进一步划分(子过程还包含子过程), 从而形成复杂的网状结构。

2.3 网状轮询子过程设计模式

在网状轮询子过程 workflow 互操作设计模式中, 允许一个 workflow 引擎在某个过程节点创建新的过程实例, 随后, 在指定的集合点查询该新过程实例的状态, 如图4所示。

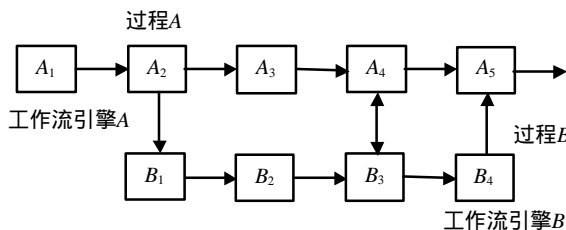


图4 网状轮询子过程互操作设计模式

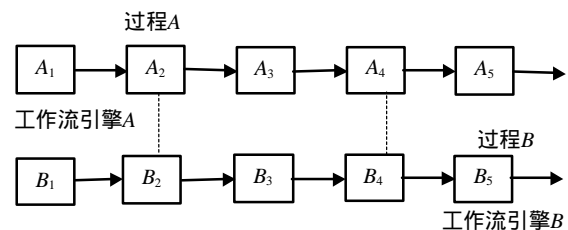


图5 事件同步互操作设计模式

由图4可知, 工作流引擎A在过程实例A的节点A₂创建新的过程实例B, 当B开始执行时, 过程实例A继续执行。当A到达和过程实例B指定的集合点(比如A₄节点)时, A轮询实例B的当前状态。此时, 过程实例B的状态有完成并终止或还未完成两种。若是前者, 工作流引擎B将自己完成的状态信息追加到应答消息队列中, 直到该消息被实例A取走。若是后者, 在实例A将轮询请求追加到请求消息队列中, 直到该请求得到响应。若是后者, 实例A是继续执行(异步方式)还是等待轮询应答(同步方式), 完全依赖应用程序的需要(图4为异步方式的示意图, 同步方式与此相似)。

2.4 事件同步设计模式

事件同步 workflow 互操作设计模式允许在工作流引擎运行过程中, 针对其他工作流引擎(或者自己)发生的事件, 创建新的 workflow 过程实例, 如图5所示。

由图5可知, 工作流引擎A和B在节点A₂、B₂和节点A₄、B₄根据事件进行同步操作。同步事件可能是一个工作流引擎的终止, 也可能是逻辑上事件检查点产生的同步事件。而在同步点之后, 工作流引擎A和B分别独立执行, 当一方或双方都到达同步点时, 继续等待新的同步事件产生。

在工作流的四种设计模式中, 其相同点是: 在两个或者多个工作流引擎中, 均有不同的 workflow 过程实例运行或者创建。但在链式设计模式中, 新的 workflow 运行实例结束后, 既不需要返回信息到

相应的其他工作流引擎，也不需要同步操作，而是两者都继续并发执行。对其他三种设计模式，或要求要返回信息，或必须进行运行同步：网状同步子过程设计模式要求原工作流引擎，必须等待相应过程实例运行完成并返回结果后，才能继续执行；网状轮询子过程设计模型是在不同的同步点进行论询来保证同步，而事件同步设计模式是根据新过程实例产生的事件来保证同步。

3 工作流互操作分类

按照国际工作流管理联盟的定义，工作流的互操作性分为八个级别。

3.1 无互操作性

属于这个级别的工作流系统，没有提供与其他工作流系统通信的手段，因而不具有互操作的能力。

3.2 共存

属于这个级别的工作流系统，不具备和其他工作流系统互操作的功能，但可以和其他工作流系统同时运行在同一个运行环境中。属于这个级别的工作流系统，其目的不是提供系统之间的互操作，而是为了增加可运行在同一平台上的工作流系统的数量，使一个完整的工作流，划分为几个部分，在不同的工作流系统中完成，相应地也扩大了用户选择适合自己的工作流系统的范围。在这种级别下，工作流系统之间的接口由人工提供(如果共存的工作流系统都遵照WFMC标准开发，也可通过标准接口通信)。

3.3 网关

工作流网关是允许在不同工作流系统之间传递实体(过程、活动或者是工作项等)的一种体制。在需要数据格式转换时，工作流网关也提供工作流系统之间的数据映射功能，实现工作流系统之间的通信。在存在多个工作流实例的环境中，网关还具有“数据路由”的功能。具体实现时，工作流网关可以是工作流系统的一部分，也可以是单独的应用程序。WFMC定义了两种工作流网关：单独网关和公共网关API。

3.4 有限公共API子集

有限公共API子集是工作流系统共同定义一组核心API。工作流系统通过这组核心API进行交互。这种级别的互操作不仅要定义一组公共API，还必须定义工作流系统之间数据交换的标准格式。对应用来说，API可以是标准的接口，也可以是数据封装的形式，标准的数据交换格式则一般通过定义“过度数据交换格式”的形式实现。

3.5 完整工作流API

完整工作流API是定义工作流系统所有操作的API规范，工作流系统之间的交互通过API来进行。与有限公共API接口相比，完整工作流API级别的互操作包含的工作流互操作范围要广泛的多，而且这些API是在更高的层次上抽象出来的。当然，完整工作流API的实现与有限公共API一样，并可以认为它是有限公共API的扩展。

3.6 共享定义格式

共享定义格式是工作流系统之间的所有过程定义(包括过程属性、信息路由描述、访问权限等)采用统一的标准。如WFMC制定的工作流过程定义语言(Workflow Process Definition Language, WPDL)就是这种级别的互操作。通过这种方式，过程定义在不同工作流系统之间共享，满足系统互操作性的要求。在实现形式上，这种级别必须支持导入(Import)和导出(Export)两个接口。

3.7 协议兼容

协议兼容指对工作流系统客户和服务端之间的通信(包括过程定义的传递，工作流的事务处理和错误恢复等)都进行标准化。为了得到这个级别的互操作，工作流系统必须提供多种通信机制(常见的有X.400、远程过程调用、其他协议)，因此实现难度较大。

3.8 等同式

等同式指 workflow 系统所有用户接口都完全相同(或至少感觉上相同)。由于商业和实际的原因, 这种级别的互操作很难实现。

上述八个级别的互操作性逐级提高, workflow 系统实现难度也是依次不断增加。为了满足不同的需求, workflow 系统实现时可以选择不同的互操作级别。

4 workflow 互操作的实现技术

从实现方式来看, workflow 互操作可有直接通信、消息传递、桥和共享数据存储四种方式。这四种方式各有特点, 使用范围也不尽相同。根据 workflow 参考模型, 可采用 workflow 引擎之间直接通信和桥的方式

4.1 workflow 引擎直接通信

这种方式的工作流互操作实现形式实质上是在 workflow 引擎之间定义一组标准的工作流应用程序编程接口和交换格式(Workflow APIs and Interchange Formats, WAPI)。通过 workflow API 和数据交换格式, workflow 引擎可以直接通信, 从而实现 workflow 互操作性。从 workflow 参考模型来看(图1), 接口4正是为实现 workflow 互操作而定义的, 如图6所示。

4.2 桥

采用“桥”来实现 workflow 的互操作, 实质就是由 workflow 网关来传递数据并对数据进行映射, 即到达互操作第三级别的互操作性要求, 这种实现形式如图7所示。

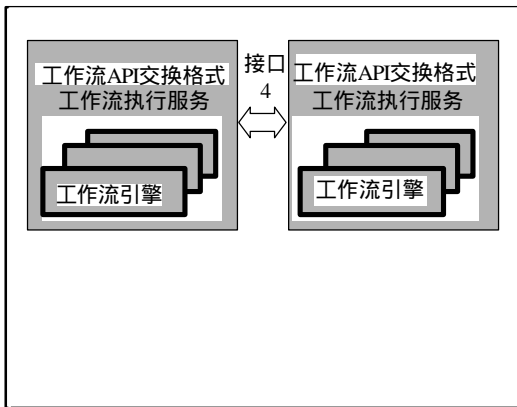


图6 workflow 接口实现互操作

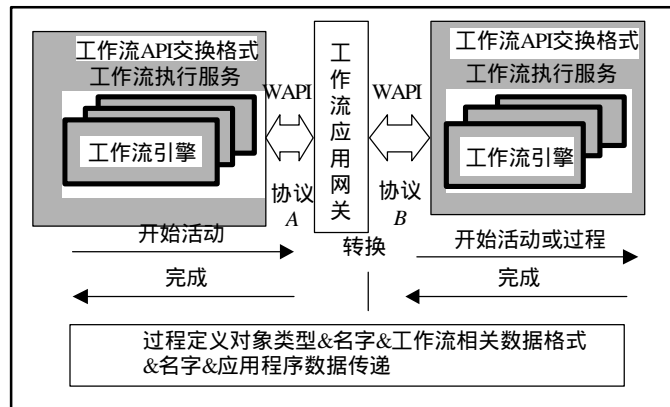


图7 workflow 网关实现互操作

5 workflow 管理系统互操作性存在的问题

随着 workflow 技术迅速发展, workflow 管理系统互操作技术仍进展缓慢, 采用上述四种典型互操作设计模式只能部分解决 workflow 互操作问题。目前, workflow 互操作技术仍存在以下三个不足的方面^[7, 8] :

- 1) workflow 模型缺乏描述互操作性的能力 虽然 WfMC 提出了一套 workflow 管理系统的标准, 也定义了 workflow 系统的参考模型, 但是该模型仅能通过 WAPI 来实现 workflow 互操作, 这显然不能完全描述和表达 workflow 系统的互操作。此外, workflow 参考模型及相关标准没有对 workflow 通信机制进行定义, 而在 workflow 系统实现时, 对选择何种通信协议存在很大的分歧, 很难实现真正意义上的互操作。
- 2) workflow 互操作过程中缺乏健壮性 workflow 实例在运行过程中可能出现资源访问冲突, 使得活动在数据操作上出现相互重叠。这种情况在 workflow 互操作过程中表现尤为明显。目前, 对互操作过程中 workflow 系统健壮性还很难保证。

3) 缺乏互操作情况下事务处理的能力 目前大部分 workflow 系统不支持事务的概念,不能保证在 workflow 系统之间互操作下一个活动执行的 ACID(Atomicity, Consistency, Isolation, Durability)属性,而且当 workflow 非正常中断时,也无法保证正确的恢复数据。

综上所述,目前 workflow 管理系统以及 workflow 互操作的发展还有许多问题亟待解决,而对这些问题的解决也正是今后 workflow 管理系统和 workflow 互操作技术的发展方向。

6 结束语

采用 workflow 参考模型可以很好地描述 workflow 管理系统的结构,但在设计 workflow 系统互操作性时,其描述能力略显不足。通过丰富 workflow 模型描述能力,充分利用典型的四种互操作设计模型,并结合 CORBA 技术、MQ 技术、WEB 技术、分布式数据库技术,可以在一定程度上解决 workflow 的互操作问题。目前 workflow 技术和 workflow 互操作技术的发展趋势,对分布式 workflow 技术的研究、workflow 与 CORBA 技术有机结合,以及 workflow 在电子商务中的应用研究都具有广泛而重要的意义。因此,建立适用于电子商务的 workflow 模型将是今后研究工作的重点。

参 考 文 献

- 1 范玉顺, 五 澄. workflow 管理技术研究及产品现状及发展趋势. 计算机集成制造系统, 2000, 1.6(1): 1-7
- 2 汪 涛, 黄力芹, 吴耿锋. workflow 管理的发展历程与趋势. 计算机工程与科学, 2001, 23(1): 97-100
- 3 Workflow Management Coalition. Workflow Reference Model, [WfMC1003]. WFMC-TC-1003, 1995: 97-100
- 4 Workflow Management Coalition. Terminology and Glossary, [WfMC1011]. WFMC-TC-1011, 1999
- 5 Workflow Management Coalition. Process Definition Meta-Modal and WPD, [WfMC1016]. WFMC-TC-1016, 1999
- 6 Alonso G. Exotica/FMQM: A Persistent Message-based Architecture for Distributed Workflow Management. Available at <http://www.almaden.ibm.com/cs/exotica>
- 7 张德壮, 李俊海, 耿继秀. workflow 系统综述. 计算机应用, 2000, 20(5): 34-39
- 8 Asuman Dogac. A workflow-based Electronic Marketplace on the Web. Software Research And Development Center. middle east Technical University. Technical report, 2001
- 9 秦志光, 刘锦德. ODP 环境中全局安全系统内的安全域之间的连接. 电子科技大学学报, 1995, 24(5): 520-523
- 10 苏 森, 唐雪飞, 刘锦德. 面向对象互操作技术. 电子科技大学学报, 1998, 27(1): 90-94