

# 一种可适应的工作流的设计与实现

赵 淳 顾毓清

(中国科学院软件研究所 北京100080)

**摘要** 激烈的竞争要求组织业务能够快速灵活地变更,也使可适应的工作流研究成为热点。组织业务变更时灵活适应的工作流管理系统需要支持工作流的重用和模块化封装,我们利用消息传递将独立的工作流组织起来,建立了较好地适应现代组织管理的工作流的关系结构,并且通过对工作流部件的封装、继承和抽象的研究,提高了工作流设计的灵活性和高效性,实现了一种可适应的工作流。

**关键词** 可适应的工作流,企业应用集成,分布式系统,计算机支持的协同工作

## An Adaptive Workflow System Framework

ZHAO Chun GU Yu-Qing

(Institute of Software, Chinese Academy of Sciences, Beijing 100080)

**Abstract** Currently business information systems are expected to be able to cope with changes in the market both promptly and flexibly, this makes adaptive workflow a hot spot among research fields of workflow technology. The WfMS supporting workflow encapsulation and component reuse can respond more quickly and efficiently to changing market conditions. This paper describes an adaptive workflow framework that uses messages transmission for workflows collaboration, employs object-oriented technology to study inheritance, encapsulation and abstract of workflow components and improves the flexibility and usability of the workflow design.

**Keywords** Adaptive workflow, Enterprise application integration, Distributed system, CSCW

## 1 引言

网络和通信的发展将使分散的信息源和计算模块连接起来成为可能。这一方面促使了以集成分散的信息系统为目的的分布式信息系统的出现,另一方面也将使软件模块连接起来完成复杂计算任务的想法更具吸引力。例如工作流管理系统(WfMS),可以提供商业和工业过程(这些过程是由人和机器的活动组成的)自动控制。使用了这样系统的组织能够提高产量、减少耗费,同时还可以监视流程的执行。工作流是由计算机和人的协同活动组成的<sup>[1]</sup>。工作流模型是对控制一系列工作任务的工作过程的形式化的表示<sup>[1]</sup>。组织流程总是随着时间变化,难于预料的异常情况经常发生。若想适应变化并处理异常,工作流管理系统(WfMS)需要能够灵活地修改工作流和它的实例。开发动态的可适应的工作流是WfMS未来发展方向和难点之一<sup>[1]</sup>。这样的系统应该能够对由协作确定的上下文相关信息的更新和其他外部事件引发的工作流执行过程的变化做出明确唯一的反应<sup>[2]</sup>。

现在商业界发现自己处于一个快速连续变化的环境中,他们希望运行在这种环境下的商业信息系统能够敏捷灵活地处理市场的变化<sup>[3]</sup>。现有的工作流管理系统只支持简单的操作过程,不适合当今商业和工业的全球化的竞争环境带来的组织管理和结构的快速改变。如今追求对客户需求的变化的快速响应,激烈的竞争和技术基础设施的快速变化使这一矛盾更加突出。为了更快速和有效地响应市场的变化,组织的以往垂直孤立结构正转变为地理上分布的以流程为核心的联合结构。与此同时,因为涉及到多个并发参与的分布的功能单位,商业过程变得更复杂。这些变化对工作流管理系统提出了

新的要求:1. 能够适应变化的环境。2. 对水平的跨组织的管理提供支持。

许多商业 WfMS 已经有同其他的 WfMS 和企业应用交互的机制。但是,对于大多数的 WfMS,他们的用户不得不事先定义好整个工作流过程的每个细节才可以使它们相互操作。这样,不论何时新的 WfMS 或者企业应用加入或者改变自己的服务接口,整个工作流过程都需要被重新定义以便从其他 WfMS 和企业应用获得服务。如果我们能够使这些变化对用户或服务请求实体透明,那么 WfMS 就可以和其他的 WfMS 和企业应用更高效协作,提高了灵活性、可伸缩性和互操作性。

针对以上问题,我们将独立的工作流组织起来,借助它们之间的消息传递来组织协同工作。我们建立的工作流的关系结构对现代组织管理的结构有一定的表示能力,并且利用接口技术对工作流的封装、继承和抽象的设计进行了探索。本文简单介绍了工作流协作模型的设计和运行,阐述了封装、继承和抽象的思想;详细介绍了它们的主要实现;接着介绍了有代表性的应用场景和实例;最后给出总结,并提出下一步的工作。

## 2 协作工作流模型

### 2.1 工作流的框架模型

工作流管理联盟(WfMC)的工作流参考模型(图1)展示了 WfMS 体系的主要部件和接口,认为工作流是一个计算机辅助的自动化的商业过程或它的一部分<sup>[4]</sup>。用户可以使用过程定义工具定义或者细化工作流的操作序列。工作流引擎负责整个工作流的执行,在它的不同层次,不同的程序(用户应

用或者被调用应用)会被调用。我们参照了这个参考模型,在 J2EE 平台上设计实现了我们的系统。

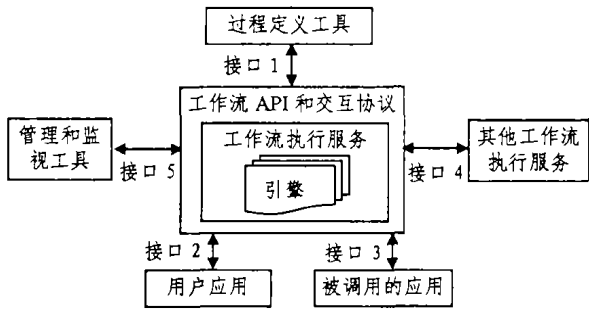


图1 工作流参考模型

## 2.2 工作流协作模型

我们认为组织的活动是由许多并发执行的子活动组成的,这些子活动之间存在着相互传递的消息。这些子活动可以属于某个部门,也可以是几个部门的协作活动。每个活动都提供自己的主题,也可以订阅其他活动的主题结果作为自身的输入。这样制定好主题的活动可以随时加入到组织中,对组织中的其他活动没有影响。组织中的子活动是一个单独的工作流,它可以随时修改自己的流程图,只要保持主题不变,就不会将修改传播到其他子活动中去。这种方式可以把一个大的工作流划分成许多独立的工作流,使工作流的制定更容易,适应组织变化的能力更强。

**2.2.1 工作流的设计视图** 我们首先将工作流划分成组,每个组拥有一些活动,每个活动就是一个独立运行的工作流。一个组可以运行在一个或多个服务器上。组内的活动是紧密耦合,组间是松散耦合的关系。主题代表工作流的事件,组内的活动通过发布和订阅主题实现协作。一个组可以是一个组织,也可以是组织中的一个关系紧密的工作流群体。例如图 2 中共有三个组。组可以建立专门负责组与组之间协作的活动,它向伙伴组提供和订阅数据,向本组的活动订阅和提供主题,我们把它称作组活动。

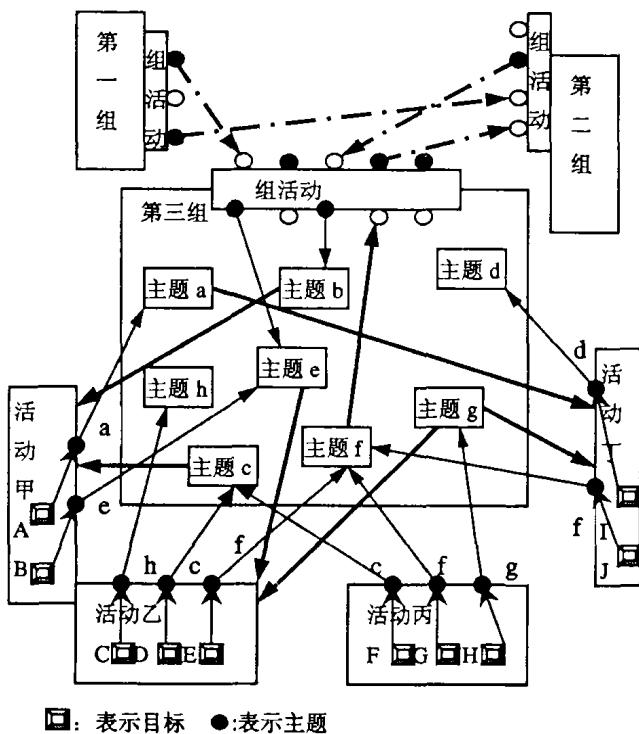


图2 设计视图

接着确定组内的活动和主题。活动的工作流的中间结点拥有的输出数据是活动的目标,它代表一个事件的发生。主题是对活动目标的概括,是组内活动相互协作的接口。活动可以向组内的主题提供数据,也可是向这些主题订阅数据。例如图 2 中,活动甲订阅了主题 b、c,向主题 a、e 提供数据。活动甲的目标 A 的数据通过主题 a 传给了活动丁。由于活动乙的目标 D 和活动丙的目标 F 相似,活动甲通过主题 c 订阅了两者。活动丙提供的主题 g 被活动乙和活动丁同时订阅。主题 b 的数据是由第二组的组活动提供的,第二组同时通过组活动订阅了主题 f。

我们认为工作流是由流程、数据和操作组成的,组织业务的变化可以通过这三者的修改来适应。通过增加主题、修改主题的发布订阅关系,我们可以快捷地修改活动间的流程。单个活动的工作流的流程的适应性修改已经有一些文献论述,例如文[3,6],这里就不再讨论了。采用面向对象的类的概念,我们将数据和操作绑定在一起,通过继承机制,可以实现对它们进行独立于流程的修改。开始设计活动的工作流时要考虑到未来可能的数据类的变化,利用基类对这些变化抽象概括。设计好的工作流调度的是基类,被执行的则是从基类继承来的子类对象。随着业务的变化,可以向工作流中添加新子类,它的对象和同一个基类的其他子类对象一起在同一个流程下被调度。活动私有的数据类只能在活动内被访问,公有数据类可以被组内的所有活动和主题访问。主题属于数据,由公有数据类表示,它也可以是由基类和子类组成的继承结构。如果活动发布的是基类,它提供的可能是任何一个子类对象;如果发布的是子类,它提供的只能是这个子类的对象。如果活动订阅的是基类,它可能收到任何一种子类的对象,如果订阅的是子类,它只能收到这个子类的对象。这样,只需要添加或修改一个数据类,无需修改活动的流程,就可以适应业务的变化。

建立在我们的工作流协作模型上的工作流设计具有以下特点。

**封装:**我们用数据类对工作流中的数据和操作进行了封装,当某个业务对象改变时,只需要修改对应数据类即可,结果会传播到整个活动或者整个组。活动是我们对工作流流程的封装,单个活动流程的改变不会影响其他活动。组活动是对整个组的封装,组内的改变不会传播到组外。

**抽象:**我们通过基类对子类的抽象实现了对数据类未来变化的逻辑编码。调度基类的活动具有较高的抽象性,可以接收并调度新添加的子类。

**继承:**我们提供了数据类的继承机制,可以添加新数据类而不用改变工作流的流程来适应业务变化。

这样我们在适应业务变化的工程中,保留了业务中不变的部分,仅修改变化的部分,使系统可以快捷地适应业务变更。

**2.2.2 工作流的运行视图** 活动就是在独立的引擎中执行的工作流,是由 EJB 来实现的。其中工作流的流程图和运行状态是要被持久保存的,可以通过消息驱动的 bean 调用工作流引擎,推动工作流的运行。工作流的目标和订阅关系由实体 bean 来管理,它运行到目标结点时,会根据订阅关系把结果消息送给其他工作流引擎。

工作流运行在独立的线程上,依靠异步消息来同步正在运行的实例。如果消息丢失或活动阻塞,订阅者的运行就不能继续,这会引发整条订阅链上的实例阻塞。我们解决这个问题的方法是检测被阻塞的活动和阻塞原因,尝试自动恢复。活动



标和主题:客户订货的数据类和汇款的数据类,它通过组活动订阅了厂家的发货信息。公司和生产厂家 A 属于同一个组。这个组共有5个主题。公司当前只涉及甲货物的业务,但未来可能还有其他货物的业务。同时,公司活动和具体的货物关系不大。所以,定义新的数据类货物,它的基类是一般货物,子类是甲货物。公司发布发货方案(主题 d)和发货要求(主题 e)。生产厂家发布运货(主题 a)、库存(主题 b)和运货计划(主题 c),其中库存没有被订阅。这五个主题都包含数据类货物,其

中公司发布和订阅的是基类,生产厂家发布订阅的是子类甲货物。整个流程是这样的:客户向公司订货,确定付款和交货方式后,公司安排厂家向客户送货。公司收到客户的订货信息后,确定付款和交货的细节,将发货方案发送给厂家,以便厂家安排生产和发货。在收到客户的一笔汇款后,公司根据合同验证厂家的运货计划,向厂家发出运货要求。厂家发出一批货物后,向客户发发货消息。经过若干次汇款、运货后,客户收到预订的所有货物,客户的一次采购活动就结束了。

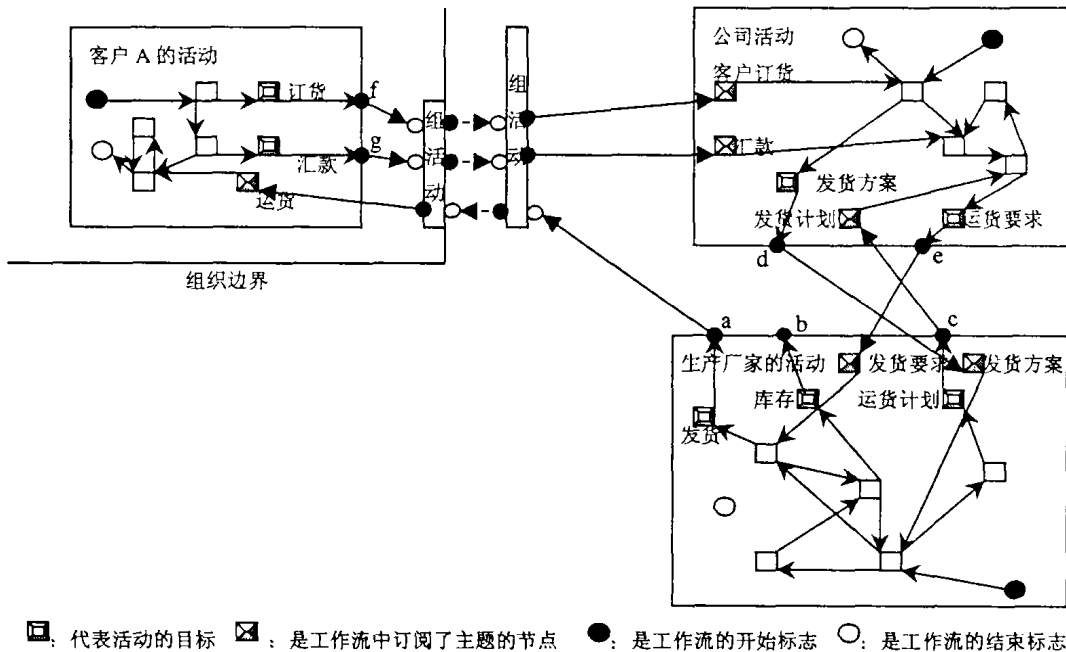


图6 应用实例

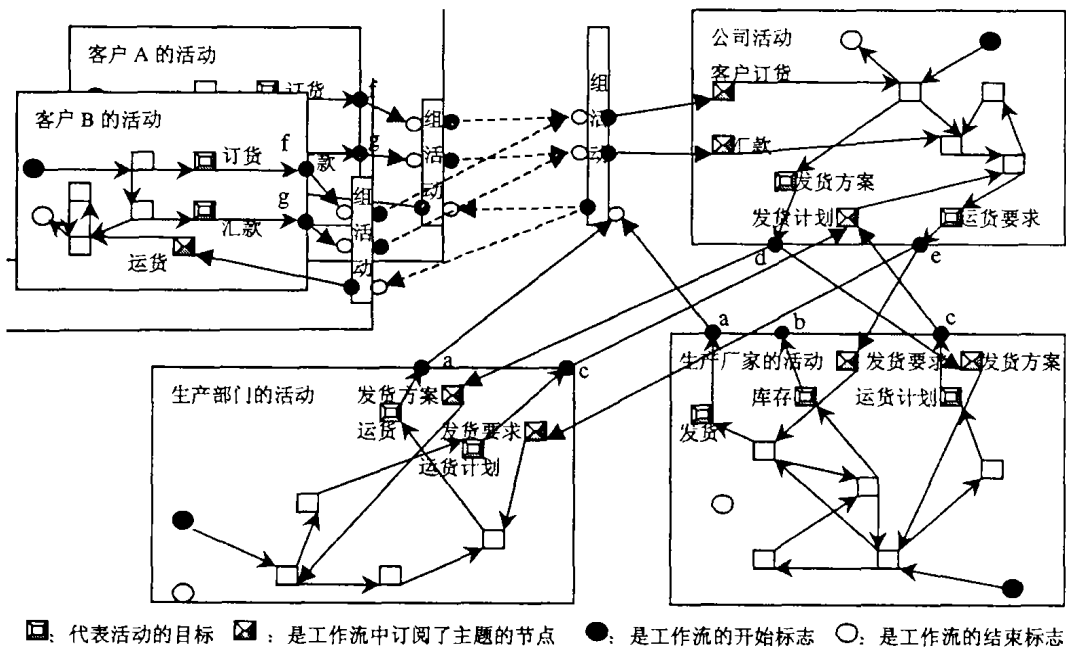


图7 业务变更后的工作流

由于市场的需要,公司决定增加经营乙货物,原甲货物的生产和销售流程不变。如图7所示,我们的系统可以较容易地适应这种变化,只需要添加乙货物的数据类和活动定义即可。从数据类货物的基类一般货物上继承出子类乙货物,为新增加的业务定义活动采购部门活动,这个活动仅处理乙货物。采购部门活动发布主题 c 和主题 a,订阅主题 d 和主题 e,只涉及数据类货物的子类乙货物。新增加的客户 B 通过以前提供的组活动接口可以完成乙货物的采购。

对于这样的业务变化,我们成功地隔离新旧业务,重用了旧业务的逻辑编码。

总结 将独立的工作流组织起来,利用它们之间的消息传递达到协同工作的目的。我们建立的工作流的关系结构对现代组织管理的结构有一定的表示能力,并且通过工作流的封装、继承和抽象的设计对工作流的可适应性进行了探索。我们的系统可以实现组织内的紧密耦合的协作,也可以实现组

(下转第134页)

作流智能的一部分<sup>[5]</sup>。

#### 4 利用 WDW 建立主动机制

主动规则的思想已经在多种环境中得到了成功的应用,包括主动数据库、工作流管理、网络管理及 XML<sup>[6]</sup>等。主动规则一般包括三个部分:事件(Event)、条件(Condition)和动作(Action),因此又称为 ECA-规则<sup>[6]</sup>。其基本执行模式为:

ON event IF condition DO action

它采用事件触发(Event-triggered)的思想,即当定义的事件发生,如果条件得到满足,即执行定义的活动。在我们的方案中,我们的 ECA 规则针对我们的 WDW 重新做了解释。

**事件** 很多事件都可以作为主动规则的触发事件,如时间事件、数据操作、异常、抽象事件等。事件既可以定义在 WDW 中,也可以定义在 WfMS 中。当预先制定的事件发生时,检查主动条件,判断是否触发主动动作 Action。

最常用的事件是时间事件,可以定义每隔一段时间检查一下条件。还可以将事件定义在 WfMS 中,例如是一个过程中从一个状态转移到下个状态时。还可以定义在 WDW 中,例如某个用户感兴趣的行为发生时或是预先定义的异常发生时。

**条件** 即用 OLAP 操作定义的布尔表达式,当某个主动规则中的事件被触发时,主动机制会判断是否满足定义的条件来决定是否执行操作。如果条件缺省,则 ECA-规则退化为 EA-规则,即只要事件发生就执行指定的动作。我们的条件都定义在 WDW 中,形如:

$Condition1 \wedge Condition2 \wedge \dots \wedge ConditionN$

其中  $Condition1 = OLAPResult1 \operatorname{operator} OLAPResult2$ ,  $\operatorname{operator}$  是比较运算符,  $OLAPResult$  是 OLAP 操作的结果,  $OLAPResult$  也可以是常数。

这样,主动规则的条件是一个以 WDW 中的数据操作的比较为元素的布尔表达式。将  $N$  个数据操作的结果进行比较后再进行一系列的布尔运算,得到的结果就是条件,当条件满足时执行主动规则定义的动作。即判断一个数据仓库中查询是否符合给定的条件。我们的数据仓库采用经典的雪花模型,所以我们的条件是关系中的布尔表达式。数据仓库中的操作结果是否符合条件最终表示为关系代数的运算。

**活动** 主动的活动多种多样。活动就是指一定的时间发生后,满足了预先制定的条件后所发生的动作。活动可能是触发某种工作流异常处理,可能是产生某种报表,也可能是将一

定的数据存入 WDW,还可能是仅仅发出一条消息触发外界的某种操作,甚至可以修改我们定义的 ECA 规则。可以制作复杂的或者是第三方的软件来响应这个消息,从而触发复杂的活动,所以活动的定义非常灵活。

如图4所示,用户事先定义好一些主动规则,存放在主动规则库中。首先按照主动规则库初始化主动处理模块。当 WfMS 或 WDW 运行过程中主动规则中定义的事件被触发时,主动处理模块会检查相应的条件是否满足。通过在 WDW 作某些数据操作,检查操作结果是否满足条件。如果满足条件,执行主动规则中定义的动作。

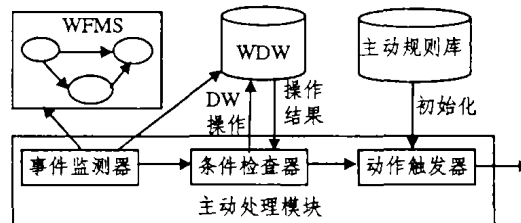


图4

**结论** 首先提出了 WDW 的概念,然后针对如何建立 WDW 提出了一个初步的模型。在我们的模型中提出了抽象维的概念,抽象维的建立对用户 OLAP 有非常重要的意义。本文还对如何加载异构的复杂的 WfMS 日志信息进行了描述。最后对 WDW 中 OLAP 应用举出了一个具体的例子,利用 OLAP 操作结果作为 WDW 主动机制的条件,建立 WDW 的 ECA 规则。

在今后的工作中,将对如何使用 OLAP 结果进行复杂的操作做进一步的研究。

#### 参考文献

- 1 Workflow Management Coalition. Workflow Reference Model
- 2 Bonifati A, Casati F, Dayal U, et al. Warehousing Workflow Data-Challenges and Opportunities. In: Proc. of VLDB'01 2001
- 3 Inmon W H. Building the Data Warehouse. John Wiley & Sons, Inc. 1996
- 4 Grigori D, Casati F, Dayal U, et al. Improving Business Process Quality through Exception. In: Proc. of VLDB'01 2001
- 5 Berry M, Linoff G. Mastering Data Mining. Wiley, 2000
- 6 Paton N W, Daz O. Active Database Systems. ACM Computing
- 7 Manolescu D A, Johnson R E. Dynamic Object Model and Adaptive Workflow. University of Illinois at Urbana-Champaign. http://micro-workflow.com/PDF/domaw.pdf, Oct. 1999
- 8 Purvis M, Purvis M, Lemalu S. An Adaptive Distributed Workflow System Framework, University of Otago. http://www.otago.ac.nz/informationscience/pubs/publications.htm
- 9 Hollingsworth D. Workflow Management Coalition. The Workflow Reference Model, Document No. TC00-1003, Issue 1. 1, 1995. http://www.wfmc.org/standards/docs/tc003v11.pdf
- 10 Kwak M, Han D, Shim J. A Framework Supporting Dynamic Workflow Interoperation and Enterprise Application Integration. In: Proc. of the 35th Hawaii Intl. Conf. on System Sciences, Hawaii, 2002
- 11 Bachmendo B, Unland R. Aspect-Based Workflow Evolution. Department of Mathematics and Computer Science. http://www.comp.lancs.ac.uk/computing/users/marash/aopws2001/papers/bachmendo.pdf, 20 July 2001
- 12 Manolescu D A, Johnson R E. Dynamic Object Model and Adaptive Workflow. University of Illinois at Urbana-Champaign. http://micro-workflow.com/PDF/domaw.pdf, Oct. 1999

(上接第131页)

织间松散耦合的协作。封装、继承和抽象的特征使工作流能够灵活、敏捷地适应业务的变化。

进一步的工作包括对异构工作流的适应性研究,可适应工作流的长事务的研究,以及可适应工作流的异常处理的研究。

#### 参考文献

- 1 Adams M, Edmond D, ter Hofstede A. The Application of Activity Theory to Dynamic Workflow Adaptation. In: 7th Pacific Asia Conf. on Information Systems, Adelaide, South Australia, July 2003
- 2 Luo Z W, Sheth A. Knowledge Sharing, Coordinated Exception Handling, and Intelligent Problem Solving for Cross-Organizational Business Processes. [PhD Dissertation]. University of Geor-