论文题目：

Combining dynamic Workflow and transactional semantics using a pattern-based approach

论文作者及单位：

Imed Abbassi, Mohamed Graiety, Nejib Ben Hadj-Alouane and Walid Gaaloulz （作者）
University of Tunis El Manar（单位）

发表期刊信息：

2014 IEEE 23rd International WETICE Conference

技术问题：

如何使用一个基于模式的方法来结合动态工作流和事务性语义

现实背景：

网络服务是模块化和自述性的软件应用，这就会导致当根据用户需求添加新的网络服务的不可靠性以及不灵活性，因此有效结合动态工作流和事务性语义来确保网络服务的可靠和灵活显得十分有必要。

作者思路（idea）：

将所有网络服务抽象成一个简单的事务并找到相应实体；然后根据这些实体间的描述来规定实体间的约束；通过给这些实体规定一个入口（in）和出口（out），并将一些可并行处理的实体聚合在二者之间。

解决方案：

第一步定义事务服务，首先将网络服务抽象成一个简单的事务包括，compensatable (c), retriable (r)，pivot (p)。并建立一个函数 TP (TP∈SERVICE→P({c， p，r})),并由逻辑关系推导

成：∀s•s∈SERVICE 可得 TP(s) ∈ {{c}, {p}, {r}, {c, r}, {r, p}}。第二步定义动态网络服务构成，

即该网络服务参与了构造过程但是是在运行过程中被选择及关联，这样就可在预料之外的情况下重新配置。然后定义了网络服务的独立成分包括，sequence, compensation, abortion, cancellation, 以及它们之间的约束关系并得到了以下关系：activation dependencies（depAct）指定一个部分组件服务的顺序执行；compensation dependency（depCps）可以表达一个向后恢复机制；cancellation dependency（depCnl）允许信号服务执行失败,其他服务被取消他们的并行执行；abortion dependency（depAbt）允许在传播失败时废弃掉该服务。第三步根据控制流框架建立有限状态自动机并规定 3 个规则来产生事务流。最终使得整个系统正常运行并且满足了网络服务的可靠和灵活。

创新贡献：

论文中通过多个实例来运用该方法进行重新建模，将具体事务抽象成具有代表性的实体，用一个统一的模式来结合动态工作流和事务性语义，保证了网络服务的可靠性和灵活性，为网络服务的建模提供了很好的借鉴。

效果评价：

这篇文章首先论证了当下处理网络服务的两种方式的优势和不足，再利用抽象化思维将这两种方法结合起来，使得网络服务的架构更加合理；

优点：作者为了跟加清晰的讲述这个方法的实现过程以及之前单一方法的不足，分别用了两个实例来具体说明。再加上一些框架流程图等，让读者从对作者论述了解的更为清晰。此外，单论方法对于实际问题的解决而言，这种方法不但很好的解决了问题，而且还为类似问题的解决提供了思路。

缺点：文章内容涉及到的专业知识很高，只适合计算机专业的人员看，可事实上，这种抽象思维还适用于大部分的非计算机事务处理，因此在我看来，这篇文章应该从更低的层次入手，从日常事务入手慢慢深化到计算机领域，这样才能更有参考性。

约束条件：在复杂的网络服务中，抽象事务有一定难度以及实体间的约束较难把握。

个人观点：

这篇文章就结构而言十分清晰，包括问题陈述以及对现下解决方案不足及优势的分析还有利用实例来阐述新方法的原理和过程。辅以流程图，控制流图，状态图等，让读者对新方法有了更直观的认识。内容上，作者首次提出了将现下流行的两种方法相结合，利用一个基于模式的方法，让这个方法有了普及的可能，这是非常好的一面。当然，在阅读过程中，我也遇到了很多问题，包括一些单词的翻译，以及对函数间关系的不理解等，但通过查文章后面的参考文献以及百度搜索已经搞懂了大部分问题，但是也不知道自己的理解是否准确，对与一些概念的解释还是有些模糊，但是文章的摘要和总结写的很精练，让我对整篇文章有了清楚地认知，再加上论点清晰，所以，整体来讲，我还是对这个方法有所领悟。总的来说，阅读者篇文章受益匪浅，也认识到阅读论文和理论学习相辅相成的作用。