

# SOA 中 Data Service 的分析与设计

张大鹏, 邱锦伦

(上海大学计算机工程与科学学院, 上海 200072)

**摘要:** 针对当前分析与设计面向服务的架构(SOA)系统时面临的数据访问多样性问题, 提出一种新的基于 SOA 的 Data Service 分析设计策略, 该方法以数据实体为中心, 以任务为导向, 采用定位数据实体和分析数据操作相结合的方法, 提供一套清晰的 SOA 系统数据访问层设计思路, 从而实现 Data Service 的粒度和重用性之间的平衡。实验结果表明, 该方法具有一定应用价值。

**关键词:** 面向服务的架构; 实体; 操作

## Analysis and Design of Data Service in SOA

ZHANG Da-peng, QIU Jin-lun

(School of Computer Engineering and Science, Shanghai University, Shanghai 200072)

**【Abstract】** Aiming at the data access multiformity problems in analysis and design of Service-Oriented Architecture(SOA), a novel SOA-based Data Service analysis strategy is proposed, which makes data entity as core and the task as guide. It uses the method combined with location data entity and analysis data operating to provide a set of design idea for SOA system data access level, and balances the relation between granularity and reuse of Data Service. Experimental results show this method has values of application.

**【Key words】** Service-Oriented Architecture(SOA); entity; operation

数据和数据管理几乎是所有企业软件解决方案的关键因素, 其中包括面向服务的架构(Service-Oriented Architecture, SOA)。有效的数据建模和管理是成功实现 SOA 的基础。要将数据提高一个层次, 就要把数据转换成信息; 要将信息提高一个层次, 就要把信息转换成知识。本文引入 Data Service 的概念, 即  $Data\ Service = Data\ Integration + Web\ Service$ 。

### 1 Data Service 简介

图 1 为数据服务在整个 SOA 基础服务中所占的层次, 可以看出, 通过 Data Service 商业流程可以直接映射到 IT 系统的流程, 为企业的商业战略提供强有力的支持。

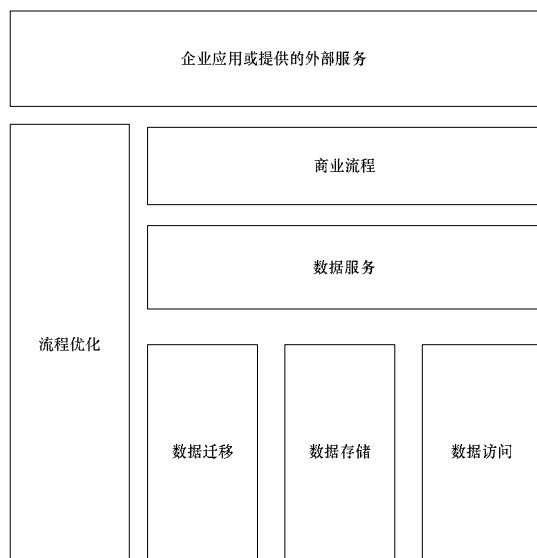


图 1 Data Service 在 SOA 架构中的位置

这里讨论的 SOA 是建立在包括 SOAP, WSDL, UDDI, WS-\* 协议族和 BEPL 等标准之上的基于 Web Service 的 SOA。

### 2 Data Service 的功能与特点

数据访问服务通常被总称为企业信息系统(Enterprise Information System, EIS), 也称数据库和文件系统。它们可以是遗留系统、记录系统, 包装好的商业应用程序、用户、伙伴、第三方应用程序和服务以及 Web Services。它们的共同之处是向其他应用程序提供数据和/或信息。本文称这些模块为 Data Service。

Data Service 具有以下 3 个特点<sup>[1]</sup>:

(1) 松耦合的服务架构: SOA 满足了应用环境的高灵活性和易维护性的特点, 各个服务之间相互独立, 靠相互发送消息进行通信。

(2) 标准化的统一接口: 在 SOAP, WSDL, UDDI 统一标准下, 各个 Service 之间相互合作完成统一的应用需求, 这也为服务在更高层次上的组合(例如: 企业间供应链)提供了统一的标准。

(3) 组合和重用的服务: SOA 可以用 BEPL 之类的流程控制语言实现动态的重用服务。服务在不同的流程中被多次重用, 提供相同的服务, 减少 IT 成本。

Data Service 的自治与可组合特点如图 2 所示。

**基金项目:** 上海市重点学科建设基金资助项目(J50103)

**作者简介:** 张大鹏(1983 - ), 男, 硕士研究生, 主研方向: 分布式软件, SOA 技术; 邱锦伦, 副教授、硕士

**收稿日期:** 2009-06-20 E-mail: dapeng@shu.edu.cn

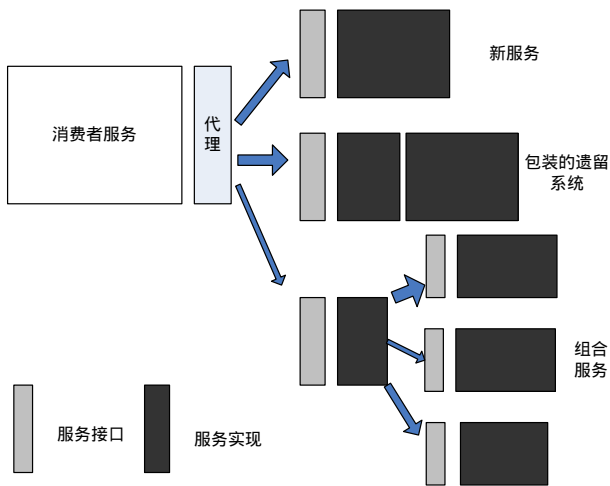


图 2 Data Service 的自治与可组合

### 3 Data Service 存在的问题

目前，Data Service 存在以下 4 个问题：

(1)Data Service 和业务流程的不一致性。具体表现是，一个 Data Service 所描述的业务操作不完整，或是在组合 Data Service 时某些服务的操作重叠。

(2)Data Service 的粒度不合适。具体表现是，很多的 Data Service 只是提供简单的 CRUD(Create, Read, Update, Delete)操作，导致组合的服务通信太多，性能急剧下降。

(3)Data Service 的接口暴露了实现过程。例如，根据接口参数判断出 Data Service 实现时所使用的数据库，这样使应用层和实现层直接强耦合，不利于服务的重用。

(4)不当的 Data Service 组合层次。由于不当的粒度导致 Data Service 在组合业务流程时候层次模糊不能清楚地反映服务所处的层次，因此服务只是局限在某个具体的流程中，不能在多个流程中进行编排。

造成这些问题的原因很多，其中最重要的问题在于 SOA 项目前期分析设计时，使用了和当前软件开发环境不一致的分析设计方法。例如，使用面向对象与基于组件的编程方法，通过使用对象、类和组件而实现关注点的分离和聚合，而在 SOA 环境中数据实体和数据操作将是 IT 所要面临的抽象对象。文献[2]描述了用面向对象的设计模式的思想来设计 Data Service，但是还是不能够从根本上解决抽象层次错位的问题。

本文提出一个新的基于数据实体和数据操作的分析与设计方法。

## 4 Data Service 的分析与设计方法

### 4.1 Data Service 的分析设计

造成问题(1)、问题(2)和问题(4)的一个主要原因是在 SOA 建模时所采取的策略不当，文献[3]提到 3 种建模策略：Top-Down 方式，Bottom-Up 方式，Meet-Middle 方式。在此，先对 3 种方式进行分析。

(1)Top-Down 方式：从业务需求出发，建立业务过程分析模型，在分析模型得到仿真与优化之后生成建立业务过程执行模型，并在过程执行模型的指导下，抽取具体的面向服务对象模型。这种方式对项目成功的风险太大，需要对业务有很深很透彻的理解才能完全实施，可能会导致项目陷入“无限设计”的状态，对大的项目来说很难做到。

(2)Bottom-Up 方式：与 Top-Down 方式相反，企业可能已经存在一些遗留应用系统。直接对遗留系统的 API 进行简单包装会导致问题(2)。同时，由于缺乏对全局流程的了解，因此容易导致问题(4)。

(3)Meet-Middle 方式：结合以上 2 种方式，企业可以同时建立业务过程模型，以及面向服务的数据实体模型，在某个阶段两者结合。某个阶段时机很难在实际的项目中把握，从而导致这种方式在使用时难度很大。

服务建模应该是个循序渐进的过程。在这过程中必须确定一个围绕的核心。如果以数据操作为核心，势必导致 Data Service 包含于流程语境中独特的数据操作相关的操作，导致 Data Service 的粒度过大，重用性不好。于是，以数据实体为核心的思路，与以数据操作为核心的服务相比，以数据实体为核心的服务明显增加了敏捷性，使得面向服务的流程能够重新建模，但是以数据实体为核心的方式明显增加了更多的预先分析，这既增加了每个服务的成本，又增加了产生服务所花费的时间。

**定义 1** 数据实体：在流程中操作的数据对象，可以是数据库的表、视图、文件或是其他提供这些数据的程序或服务。

**定义 2** 数据操作：在流程中，对数据实体进行的 CRUD 运算。

基于定义 1 和定义 2，新的分析步骤如下：

(1)确定数据实体模型，如订单、客户、发票、产品信息等。这需要分析当前系统所涉及的所有数据实体。这里的实体模型类似在建立数据库时采用的 E-R(实体-关系)模型中的 E，在这里可以是关系数据库中的表、视图、XML 文件、Excel 中的数据表格、文件系统中的单个文件等。实体的选择是决定 Data Service 粒度的重要因素，详见文献[4]。

(2)列举数据操作，例如提交订单、生成订单、打印发票、订阅产品信息，是由一系列发生在不同数据实体上的 CRUD 运算组成的。

(3)建立数据操作-数据实体关联矩阵。关联关系包括增、删、改、查，分别用 C,D,U,R 表示。每个数据操作都对相应的数据实体存在一个或多个操作作为相应矩阵元素的值。某些操作可能有限定条件或者操作规则，关联矩阵可以根据需要设定更多的值以表现更复杂的情况。

(4)以数据实体为数据源，以数据操作为函数确定服务接口和操作。如果同一个数据实体存在多个操作，可以合并增、删、改操作作为一个服务。查询由于参数的多样化，不能和其他的操作共用一个服务。

(5)根据新产生的服务来编排原来的任务，如果符合，则结束；否则，拆分大粒度服务数据操作，并回到步骤(4)继续进行。

产生合适粒度的 Data Service 并不是个简单的过程，这些步骤里面每一步都需要细致的工作和积累。什么样的粒度是最好的，实际上很难有统一的标准，不同的目标有不同的结论。

### 4.2 实例研究

下面用一个实际的例子简单地演示本文提出的分析和设计 Data Service 的步骤。笔者在设计上海人民电器厂销售系统时面临这样一个问题：公司有 2 种客户——协议用户和普通用户。普通用户的订单流程为

客户浏览产品信息  
 客户建立订单  
 检查客户账户信息和当前库存信息  
 处理订单  
 运输货物  
 确认订单  
 协议客户订单流程为  
 客户浏览产品信息  
 客户建立订单  
 处理订单  
 选择送货方式，运输货物  
 确认订单

根据之前提出的方法，对该流程进行分析与设计：

(1)该流程中的数据实体包括：产品信息，协议客户产品订单，普通客户产品订单，客户信息，产品库存。

(2)和这些数据实体发生关系的任务包括：读取产品信息，建立订单，处理订单，确认订单，检查客户信息，检查产品库存。

(3)建立数据实体-数据操作关联矩阵，如表 1 所示，其中，0 代表没有相关操作。

表 1 数据实体-数据操作关联矩阵

数据操作	数据实体			
	产品信息	客户订单	产品库存信息	客户信息
读取产品信息	0	0	0	0
建立订单	0	C	0	0
处理订单	R	C	0	0
检查客户信息	0	0	0	R
确认订单	0	U	0	0
检查产品库存	0	0	R	0

(4)根据关联矩阵产生的服务包括：产品信息订阅，订单服务，客户信息服务，检查产品库存服务。

(5)由于订单服务包含建立、修改、查询 3 种操作，因此无法组合原流程，且在多个不同用户同时进行操作时，产生效率瓶颈。

(6)回到第(4)步，把有关数据实体客户订单的数据操作拆分成 3 部分：创建订单服务，处理订单服务和确认订单服务。具体设计流程如图 3 所示。

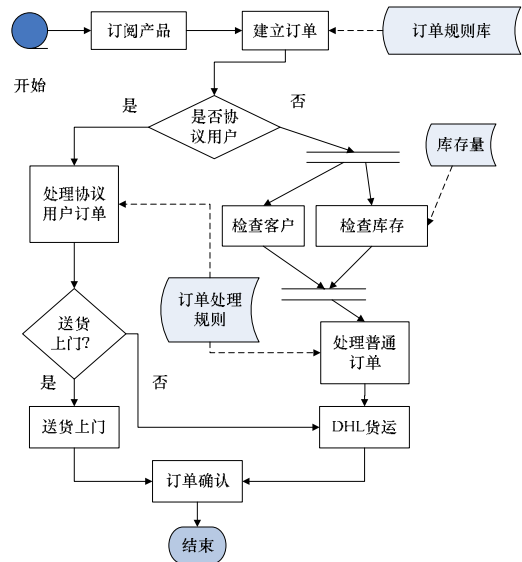


图 3 订单处理流程

该流程在实际的应用中能够较好地处理公司不同的订单，并且服务的粒度得到合理安排。在访问不同数据实体时不存在重叠和交叉，性能没有受到影响，很好地满足了客户的需求。

## 5 结束语

本文描述了在实际的工程项目中分析设计 SOA 中 Data Service 的一种方法。目前，对于服务产生结果的评判并没有形式化的描述，同时，由于软件产品需求的多样化，因此在实际的分析过程中很难有统一的标准。实验结果表明，本文提出的方法能够给分析和设计人员提供一个清晰的思路以开展工作，并能获得较好的业务敏捷性和服务重用性。

## 参考文献

- [1] Xu Yang, Xin Zhanhong. A Study on the Integration Model of EIS Based on SOA[M]. Boston, USA: [s. n.], 2007.
- [2] Byun Sang-Yong. Study on the Web Service Composition and Connection Patterns[C]/Proc. of the 5th International Conference on Software Engineering Research, Management and Applications. [S. l.]: IEEE Press, 2007.
- [3] Thomas E. Service-Oriented Architecture Concepts, Technology, and Design[M]. [S. l.]: Prentice Hall PTR, 2004.
- [4] Heuser C A. Towards a Complete Conceptual Model: Petri Nets and Entity-Relationship Diagrams[J]. Information Systems, 1993, 18(5): 275-298.

编辑 陈文

(上接第 104 页)

## 参考文献

- [1] 钱夕元, 荆建芬, 侯旭遏. 统计过程控制(SPC)及其应用研究[J]. 计算机工程, 2004, 30(19): 144-146.
- [2] Florac W A, Careton A D. Measuring the Software Process—Statistical Process Control for Software Process Improvement[M]. [S. l.]: Pearson Education, Inc., 1999.

- [3] Florac W A, Carleton A D. Measuring the Software Process Improvement[M]. [S. l.]: Addison-Wesley, 1995.
- [4] 张公绪. 现代质量管理学[M]. 北京: 中国财政经济出版社, 1999.
- [5] 张公绪. 选控图理论与实践[M]. 北京: 中国邮电出版社, 1984.

编辑 顾逸斐