

基于 SOA 信息系统的研发^①

隆益民 (广东女子职业技术学院 艺术设计与信息技术系 广东 广州 510300)

摘要: 分析企业信息系统存在的主要问题, 介绍面向服务架构(SOA)的基础原理、分层模型和实现技术, 提出一种基于 SOA 信息系统框架, 重点讨论面向服务的信息系统的信息模型、系统结构、实现方法, 并给出性能分析。实践表明, 该架构可以较好地实现信息系统的应用并具有良好的性能。

关键词: 信息交换平台; 信息模型; 面向服务架构; 数据采集; XML

Information System Based on SOA

LONG Yi-Min

(Department of Computer, Guangdong Women's Professional Technical College, Guangzhou 511450, China)

Abstract: This paper analyzes the major problems of many enterprises' information systems and introduces the basic principle, the hierarchical model and implementation technology based on SOA. It proposes a higher information system based on SOA which focuses on the system information model, architecture, implementations based on service-oriented architecture. In the end, it gives an example of information system application and performance. Practice shows that the architecture has good performance in college application information system.

Keywords: information exchange platform; information model; service-oriented architecture; data collection; XML

1 引言

计算机技术的持续发展提升了现代企业的管理水平, 已经渗透到了企业管理的各个角落, 客户关系管理系统、会议管理系统、进销存管理系统等等信息管理系统都同时或者陆续地在企业得到实施。一般情况下, 各个信息系统往往是由不同的厂家开发, 而且大部分系统使用的是不同的开发工具和开发平台, 对于企业的 CIO(Chief Information Officer)们来说, 如何很好地解决各个系统之间的兼容性并促进各系统的数据共享, 已经成为摆在他们面前的一个重要难题。企业的发展速度很快, 企业规模的扩大、业务流程的更改等都将导致各个信息管理系统的相应更改, 二次开发对企业来说可以说是一件无法避免的事情, 对于企业的 CIO 来说, 如何使二次开发最大限度地减少对企业的影响也是企业在部署各类系统的过程中需要重点考虑的问题。基于 SOA (Service-Oriented Architec-

ture)的架构, 站在服务的角度来设计系统, 可以很好地解决企业在发展过程中所面临的这些问题。

SOA 作为一种系统架构的方法, 最早由 Gartner 公司在 1996 年的时候为了实现每个 IT 系统各自的自由发展空间且又能按需共享而提出的。Gartner 将 SOA 描述为: “客户端/服务器的软件设计方法, 一项应用由软件服务和软件服务使用者组成……SOA 与大多数通用的客户端/服务器模型的不同之处, 在于它着重强调软件组件的松散耦合, 并使用独立的标准接口。”^[1]

基于 SOA 构建系统主要遵循以下的一些原则:

①SOA 要求一致性; ②SOA 要求简化开发; ③服务具有标准的、经过正式定义的可由计算机处理的接口; ④服务应设计为可重用。系统开发使用 UDDI (Universal Description Discovery and Integration 统一描述、发现和集成)机制作为创建、发布、发现和调

① 基金项目: 广东女子职业技术学院科学研究基金(2008A08010)

收稿时间: 2009-09-05; 收到修改稿时间: 2009-10-24

用服务的机制，确保服务的一致性。各服务层之间通过接口进行通信和数据传送，各接口采用通用的 WSDL(Web Service Description Language)语言进行定义，通过 WSDL 描述的接口将服务公开给核心引擎不仅增加了服务的通用性，同时使得各应用系统核心引擎之间的集成和通信带来极大的方便。使用 UDDI 创建服务，使用 WSDL 定义服务，提高了服务的共享性和可重用性。各服务内部之间的松散耦合使得服务内部的更改不会影响到服务的调用。

2 信息模型的设计与实现

2.1 信息模型的设计

2.1.1 信息交换系统的基本结构模式

如图 1 所示，这个信息交换体系把三层架构模式的第二层分成一个信息交换业务管理组件和多个分布式远程信息交换代理对象(在内容提供者数据服务器或远程客户服务器执行)。这个信息交换业务组件是基于可重配置的信息交换业务组件，配有一个叫 SMART PIPE(Or Policy Applicator)评估机制来优化网络的利用。信息交换业务组件为每一个客户提供一个持久的本地用户自定义策略的知识库。分布式数据信息交换体系架构模式是独立于内容提供者和网络协议的。

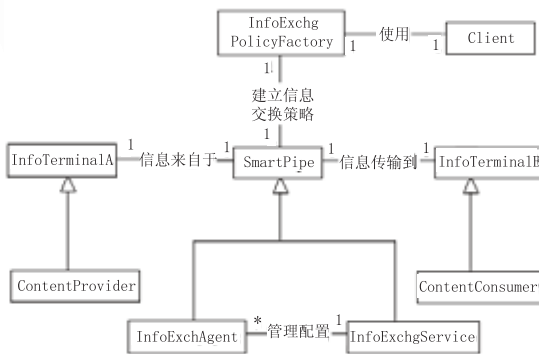


图 1 信息交换系统的基本模式

信息的提供者，如信息终端 A 把信息提供给精明管道，客户会根据需要通过信息交换策略工厂制定出有关信息的一个描述(信息的定义)精明管道又把信息定义发布到信息目录服务器，相当于把知识添加到知识库；另一端，信息的消费者，如信息终端 B，将从信息目录服务器上获得信息的定义，然后向信息终端 A 提交获取信息的请求，信息终端 A 会直接把信息传

送给信息终端 B，不经过任何中间代理。信息交换策略组件提供功能转换和生成输出，管道同步活动的用户和传输信息。SMART PIPE 合并管道和信息交换策略两重功能，同步活动的用户，进行信息交换并提供不同的功能监视或格式转换以产生信息流。客户接收到的数据流，只包含按预定义的信息交换策略或基于客户标准或规定格式的纯数据^[2]。信息交换策略组件(InfoExchgPolicy Component)能包含特定类型的数据格式或转换算法。信息交换业务组件(InfoExchg Service Component)为分布式客户的电子商务和可重编程性提供集中的访问控制和持久的数据库。

2.1.2 信息交换系统的体系结构模式

信息交换系统的体系结构由两部分组成：一个单件和多个代理组件，如图 2 所示，对应于信息交换业务组件(InfoExchg Service Component)和信息交换代理组件(InfoExchg Agent Component)。信息交换业务组件是整个系统的核心模块，主要的功能包含各种类型信息的定义、数据的采集、信息的加工、信息的存储和信息的浏览。信息交换代理模块的主要功能有信息的发布、信息的接收、信息的删除、信息的监管。

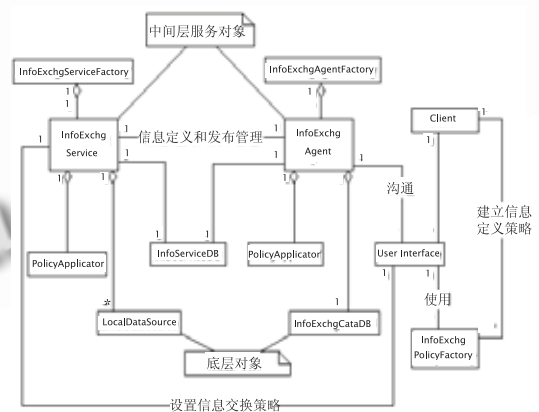


图 2 信息交换系统的体系结构模式

2.1.3 信息交换系统的信息模型

信息模型主要由信息描述(InfoScript)、信息元数据(infoMeta)、信息发布(Publish)和信息链(Info-Chain) 四大元素组成。信息描述元素描述了信息的相关属性，有信息 ID、信息名称、信息摘要、信息源主、信息发生时间、信息的最后期限。信息发布元素指出信息是向谁发布的，发布的权限、发布日期、以及记

录接收者浏览信息的时间。信息链记录了信息所经过的路线，谁浏览和何时浏览过此信息，再由谁对信息进行重发布，是信息浏览者的一个重要参考。信息元数据元素是对信息的直接描述，信息元数据模型组如图 4 所示。

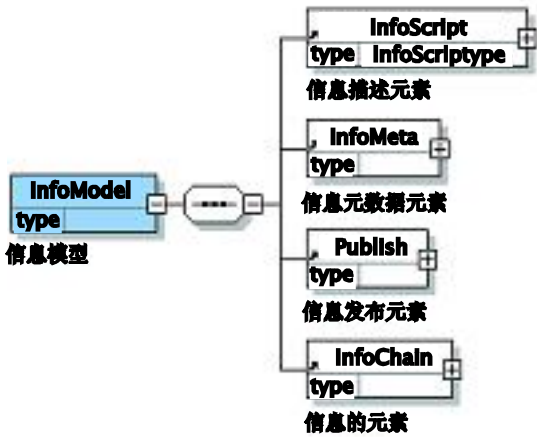


图 3 信息模型主干组件

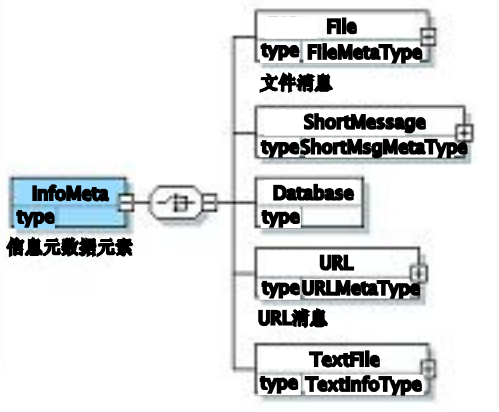


图 4 信息元数据模型组

2.2 信息模型的实现

2.2.1 PowerGate 系统的部署

PowerGate 系统的部署图如图 5 所示，主要有三个部分：信息交换目录服务器(InfoExchgCatalogServer)，用户服务器(UserServer)和信息终端(Info-terminal)。信息交换目录服务器运行着信息目录服务程序和一个信息交换目录数据库。用于存放信息终端发布过来的信息定义，并提供给其它的信息终端浏览和读取信息定义。用户服务器运行用户管理程序和一个用户信息管理的数据库。提供给用户登录时进行身份验证，新用户注册。信息终端就是参与信息发布和

消费的用户，数量不受限制，只要有合法的已注册的用户 ID 就自动成为 PowerGate 系统成员。

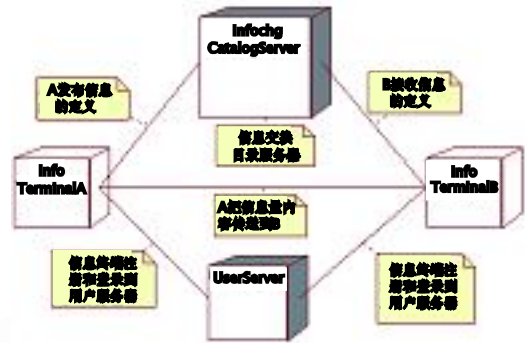


图 5 PowerGate 系统部署图

PowerGate 系统是基于 TCP/IP 协议的 P2P 的对等网系统。信息交换的基本过程是这样的：信息终端用户 A 首先在自己机器上运行的 PowerGate 客户端程序上通过用户界面定义一条信息，这条信息的定义会先存放在本地的一个数据库里，需要发布时，先从本地数据库提取信息的定义，再发布到信息交换目录服务器，信息终端用户 B 从信息目录服务器接收到发布给 B 的信息，也先保存在 B 的本地数据库里，需要执行信息的浏览时就向 A 提交请求，如验证通过，A 就把信息内容通过 TCP/IP 协议以点到点的方式直接传送给 B。B 拿到数据后用 PowerGate 自带的浏览器浏览信息。

2.2.2 PowerGate 的层次结构

从软件的层次结构这一角度来看，现有的 Power-Gate 分成如下的五个层次：网络层，数据访问层、数据处理层，数据的描述层，数据显示及操作层。如图 6 所示。

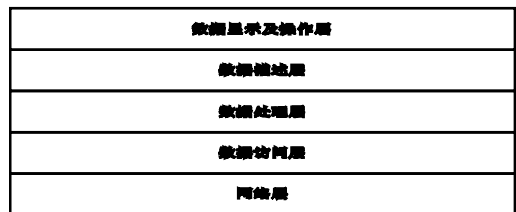


图 6 PowerGate 的层次结构

2.2.3 SQL 语句的定义、发布、接收和浏览过程

在 PowerGate 系统里，最为关键和复杂的处理就

是数据表数据的发布和浏览过程。如图 7 所示，图解了 SQL 语句的定义，存储、提取、复合定义、发布、接收和数据浏览全部过程。其中信息的浏览过程是一个较复杂的过程，由信息浏览模块完成。首先从 infotree.mdb 数据库里读取信息的定义，再判断信息的类型，如果是 SQL 语句信息，先对 SQL 语句分解，得到信息发布者的相关信息，如：发布者 ID，机器 IP 等，然后向信息发布者发送请求，得到回应后把 SQL 语句传送回去。信息发布者接到 SQL 语句后也对 SQL 语句分解，找回信息 ID，然后从 cache.mdb 数据库里读出数据库的连接字符串，把连接串和 SQL 语句一起提交给数据库，执行查询操作，并提取数据打包发送给信息接收者。信息接收者收到来自信息发布者的数据后(可能同时来自多个发布源)在本地的 cache.mdb 里生成临时数据表，执行 Global SQL 语句读取本地的临时表记录，产生新的数据表，这就是最终可浏览的数据。

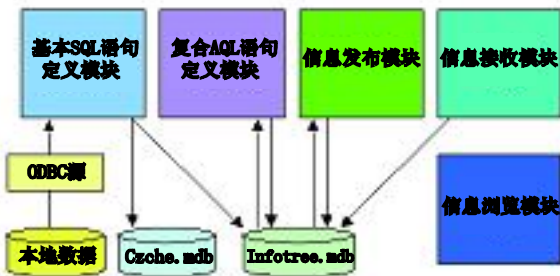


图 7 SQL 语句的生成、发布、接收和数据浏览

2.2.4 数据采集模块

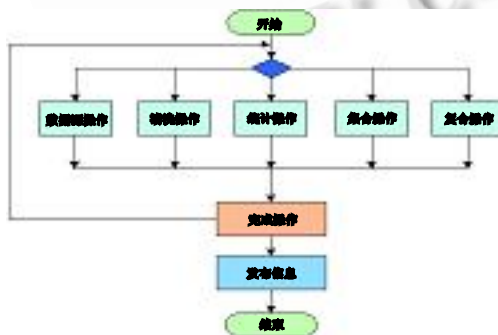


图 8 数据采集模块

PowerGate 系统里可定义的信息类型是多种多样的，如程序文件、文档文件、短消息、数据包、数

据库数据等。其中最复杂的最难以实现的是数据库数据信息的定义，因它牵涉到数据库的接口问题、SQL 查询语言、数据库访问者的身份验证等。所以，重点是与数据库相关的数据的采集。在项目工程里，独立建立数据采集模块。如图 8 所示，数据采集模块又细分为五个小模块：数据源操作、数据清洗操作、数据统计操作、数据集合操作、数据的复合操作。

2.2.5 XML 文档的处理模块

整个信息交换系统产生的信息定义都用 XML 格式来描述的。当用户通过图形界面操作定义一个信息时，操作程序就会调用 XML 文档生成模块生成 XML 文档，XML 文档以文件形式保存在当前用户的目录下或以数据形式添加到 InfoTree 数据库里^[3]。

设计的 XML 文档处理模型，较好的实现了 XML 文档的处理。XML 处理模型是以一个独立的项目存在，然后编译成动态链接库(DLL)文件，由用户通过 PowerGate 界面的操作来调用这个动态链接库。如图 9 所示：

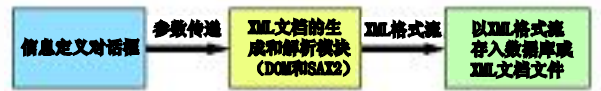


图 9 XML 处理流程

具体过程：PowerGate 用户操作消息定义界面，对信息作出定义，然后把各个定义以参数形式传递到 XML 处理模块，XML 处理模块根据传递来的参数调用 DOM 或 SAX2 接口导入这些参数进行处理，处理完毕将 XML 格式的数据流存进数据库或以 XML 文件格式保存^[4]。PowerGate 是基于 VC++ 的编码，所以这里也采用了 MSXML 开发包，这个开发包同时支持 DOM 和 SAX2，调用 DOM 接口时 DOM 又调用底层的 COM 生成 DOM 对象。处理模块可以很方便调用 MSXML 库里的函数操作 DOM 对象。整个处理效率是比较高效的。但如果 XML 文档很大时，这种处理就失去优势。这个时候最好调用 SAX2 接口，以基于事件的方式处理 XML 文档。SAX2 会根据程序预定义好的方式响应或忽略某些事件，这样，处理 XML 的效率会更高。

3 结语

本文重点研究了一种面向服务的应用系统的设计

(下转第 23 页)

(上接第 27 页)

与实现方案, 这种方案具备良好的封装性、松散耦合性、可扩展性、可靠性和见效及时等特点, 使得应用系统具有更好的互操作性、开放性, 实现了分布式的数据查询与交换功能, 较有效解决了信息孤岛问题, 使用信息模型后, 将有更广阔的使用范围。

参考文献

1 凌晓东. SOA 综述. 计算机应用与软件, 2007,24(10):

122 - 124,199.

2 O'Neil P, O'Neil E. Database Principles, Programming, and Performance 2nd Ed, 1998. 145 - 155.

3 MSxml4.0 SDK Document Microsoft XML Code Services

4 Aitken PG. 微软 XML 技术指南. 北京: 中国电力出版社, 2000. 105 - 155.