<<基于Apache CXF构建SOA>>

图书基本信息

书名：<<基于Apache CXF构建SOA应用>>

13位ISBN编号：9787121194603

10位ISBN编号：7121194600

出版时间：2013-3

出版时间：任钢 电子工业出版社 (2013-03出版)

作者：任钢

<<基于Apache CXF构建SOA>>

内容概要

《基于Apache CXF构建SOA应用》共15章，大致分为三个部分。
第一部分介绍关于SOA和WebServices的初步知识，第二部分介绍ApacheCXF框架的一些基础知识，第三部分重点介绍ApacheCXF框架的应用，包括ApacheCXF框架的前端（Frontends）应用、数据绑定（DataBindings）应用、传输协议（Transports）应用，并隆重推出了ApacheCXF框架如何实现RESTful服务、如何支持动态语言和WS—*规范等，另外，还包括ApacheCXF框架一些高级功能的用法。
最后，《基于Apache CXF构建SOA应用》还描述了ApacheCXF的工具、配置、调试、日志、部署和发布等使用的相关内容。

《基于Apache CXF构建SOA应用》最大的特点是实用性。
对于SOA和WebServices的基本概念只是初步介绍，主要内容是基于ApacheCXF框架的WebServices应用案例。
对于每一个ApacheCXF框架的功能主题，都通过一个或多个实际的案例场景来进行阐述。
对于每一个案例场景，都有源代码程序例子、架构描绘和程序实现说明。
笔者可以负责任地说每一个例子都经过调试并能够运行。
实践也是编写《基于Apache CXF构建SOA应用》的一个重要目的，最终目的就是让读者全方位地了解ApacheCXF框架能实现的功能，一方面让读者理解开发者的思路，另一方面帮助读者在实际工作中应用这些方法和编程。

作者简介

前言
软件架构的实现模式是一个发展的过程。
从以前的面向过程、面向对象，到后来的面向构件、面向整合和面向集成，接着又进化到现在的面向服务模式。
这时候，一个非常时髦的词——SOA就出现在我们的面前。

SOA是一个沉重的话题。
我很早就接触了这个概念。
那时既年轻也轻狂，觉得SOA无非就是那么几个已经耳熟能详的单词组合。
SOA的确出现得很早，但具体落地非常艰难。
仅有一些空洞的解决方案和让人发炫的理想场景。
在这样高不可及的光芒下，我们只是空喊一些口号，创造着各种新鲜、时髦和美好的词汇与概念，可没有一个明确可以下手的地方。
如何让SOA进入百姓家，这似乎成为了一个不可能完成的任务。

Web Services的出现，似乎给了我们黑的眼睛，让我们有了寻找光明的希望。
同时，Java平台对Web Services的支持，也给了我们实现SOA的利器。

基于Java规范的开源Web Services框架，我最早接触的是Apache Axis，当时还编写了基于Axis框架的一个扩展框架。
后来与一些公司的开发团队接触，才知道Apache还有一个开源Web Services框架，即Apache CXF。
这是一个合并过来的产品。
这样，国内开发人员又多了一个开源Web Services框架选择，而且这个开源框架整合了ESB和Web Services，所以我对Apache CXF未来在中国的发展前景还是充满着信心的。
对于Java支持SOA，也有很长的时间了，但国内关于SOA的方案都是一些大企业的平台，有点阳春白雪的感觉。
对于一些小企业，采用一种重量级的工具似乎有一些不堪重负或者得不偿失。
而Apache CXF框架则是一个轻量级的开源Web Services框架，在这个框架上完全可以构筑一个企业级的SOA平台。
正是在这个理念的基础上，我认真地学习了Apache CXF框架，并把在这个学习过程中的体会、经验和一些应用例子贡献给读者。

在本书的编写中，主要参考了Apache CXF官方网站的信息。
毕竟这是关于Apache CXF最权威的官方信息来源。
在参考材料中，我比较喜欢IBM的相关网站，IBM的技术网站一般都站在比较前沿的领域来讲解、讨论和分析问题，所以，书中有一些内容也是摘取或参考了IBM技术网站的思想和内容。

本书覆盖的内容较多，可以毫不夸张地说，书中的很多章节都可以独立地撰写出一本很厚的书籍。
正是出于这样的考虑，笔者不能对一些技术做详细描述，有的内容也只是蜻蜓点水地简单说明一下。
本书最大的特点是实用性。
对于SOA和Web Services的概念，以及其中的方方面面的内容，都有很多相关的书籍来进行阐述。
作为本书的基本概念，我也介绍了一些关于这方面的内容。
对于每一个例子，都进行了不止一次的编写、调试和测试。
我可以负责任地说每一个例子都是可以运行的。
对于我每一个涉及的主题，最终都是通过一个实际的案例（包括源程序代码）进行阐述的。

这本书不是一本介绍理论的书，而是充满了各种程序代码实现方式的工具书籍。

当然，阅读本书也要具备一定的基础知识，否则，有些术语和解释还是比较难以理解的。

本书从第一次编写到最后实现出版总共花费了两年半的时间。
在这个过程中我耗费了大量的时间和精力。
并且这些工作都是在业余时间内完成，白天还要照常上班，只有到了晚上或者节假日才有闲暇时光。
我一般难得有闲暇时间，如果有，也是打算去休息或放松，毕竟平时的工作还是比较劳累的。
但我还是硬着皮头坐下来写书，写书是一件非常枯燥的任务。
对于枯燥，这还不是最大的障碍。
我认为最困难的是一个接着一个的技术难题。
很多时候有些难题没有办法一下子解决，于是就做了一个例子又一个例子，编写了一个测试接着又一个测试，可还是不能达到自己理想的结果，沮丧、失败、自责、怀疑、困惑等等都涌上头来。
在这段时间中我有几次都考虑放弃，心里总是在继续写和终止写之间徘徊不定。
但总是觉得已经走了这么长的路程，不能因为一时的挫折而终止多年的辛劳，于是还是像一个孤独的苦行僧执着地编写和测试下去。
很高兴的是我还是坚持下来了，在经历了无数个寂寞和孤单的夜晚，把这本书赶写出来。

在这期间，我要感谢我的家人对我的理解和支持。
我的妻子和女儿总是抱怨我一天到晚总是待在电脑旁边。
我要感谢我的父亲和母亲，我远离家乡，没有时间照顾他们，但他们总是支持我现在做的一切。
在这期间，我的母亲永远地离开了我，我只能用这本书来纪念她。
我要感谢我的好朋友江愿兵、徐宾和卢建平，他们在很多方面给了我无尽和无私的支持与鼓励。
我把这里的一切都献给他们并祝愿他们好人一生平安。

由于笔者水平有限，书中覆盖的范围又比较广，涉及的概念也比较多，所以书中的错误和缺点在所难免，希望读者能给予批评和指正。
我的联系方式是：rengang66@sina.com。

<<基于Apache CXF构建SOA>>

书籍目录

章节摘录

版权页： 插图： REST不是一种协议，而是一种体系结构风格，这是非常重要的区别。
相比较于基于SOAP和WSDL的Web Services，REST模式提供了更为简洁的实现方案。
目前，越来越多的Web Services开始采用REST风格设计和实现，真实世界中比较著名的REST服务包括：Google Ajax搜索API、Amazon Simple Storage Service（Amazon s3）等。
基于REST的Web Services遵循以下一些基本的设计原则。
系统中的每一个对象或资源都可以通过一个唯一的URI来进行寻址，URI的结构应该简单、可预测且易于理解，比如定义目录结构式的URI。
以遵循REC—2616所定义协议的方式显式地使用HTTP方法，建立创建、检索、更新和删除（CRUD：Create，Retrieve，Update and Delete）操作与HTTP方法之间的一对一映射：若要在服务器上创建资源，应该使用POST方法；若要检索某个资源，应该使用GET方法；若要更改资源状态或对其进行更新，应该使用PUT方法；若要删除某个资源，应该使用DELETE方法。
URI所访问的每个资源都可以使用不同的形式加以表示（比如XML或者JSON），具体的表现形式取决于访问资源的客户端，客户端与服务提供者使用一种内容协商的机制（请求头与MIME类型）来选择合适的数据格式，最小化彼此之间的数据耦合。
11.1.2 RESTful原则 在Web应用程序中最重要的REST原则是，客户端和服务器之间的交互在请求之间是无状态的。
从客户端到服务器的每个请求都必须包含理解请求所必需的全部信息。
如果服务器在请求之间的任何时间点重启，客户端就不会得到通知。
此外，无状态请求可以由任何可用服务器回答，这十分适合云计算之类的环境。
客户端可以缓存数据以改进性能。
在服务器端，应用程序状态和功能可以分为各种资源。
资源是一个向客户端公开的概念实体。
资源包含多种概念，如应用程序对象、数据库记录、算法，等等。
每个资源都使用URI（Universal Resource Identifier）得到一个且唯一的地址。
所有资源都共享统一的界面，以便在客户端和服务器之间传输状态。
访问资源使用标准的HTTP方法，比如GET、PUT、POST和DELETE。
Hypermedia是应用程序状态的引擎，资源表示通过超链接互联。

编辑推荐

《基于Apache CXF构建SOA应用》适用于软件设计师、软件开发工程师和一些正在进行SOA开发的开发人员，既可以作为ApacheCXF框架的学习指南，也可以提供给软件开发工程师在设计方面进行参考。

<<基于Apache CXF构建SOA>>

版权说明