

国外电子信息类系列教材

*Service oriented computing and
Web data management*

面向服务的计算和 Web数据管理

陈以农 蔡维德 编著
郭小群 等译

Electronic Information



西安电子科技大学出版社
<http://www.xduph.com>

TP393.4/360

2013

国外电子信息类系列教材

面向服务的计算和 Web 数据管理

陈以农 蔡维德 编著
郭小群 等译

北方工业大学图书馆



C00338703

RFID

西安电子科技大学出版社

内 容 简 介

《面向服务的计算和 Web 数据管理——从原理到开发》是基于作者讲授的课程 CSE445/598(分布式软件开发)的教学材料。从 2006 年秋季开始,这个课程在亚利桑那州立大学计算机科学和工程学每个学期都要讲授。本书讲授 SOA、SOC 和 SOD 这三个主题,而且在一定的宽度和深度上覆盖了每个主题的大部分内容。这些内容不仅涉及 SOA/SOC 的概念、原理和方法,还包括可运行代码的演示。通过可运行代码,学生们会对书中的概念、原理和方法有更好的理解。本书还介绍了当前最尖端的技术和工具,可以用它来开发具有适当规模和功能的可运行软件。

本书并不是一部讲授程序设计的导引性教程。它的主要读者是计算机科学与工程专业的高年级大学生、研究生,或具有程序设计背景的软件工程师。

Copyright © 2010 by Kendall/Hunt Publishing Company

ISBN 978-0-7575-7747-5

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of the copyright owner.

图书在版编目(CIP)数据

面向服务的计算和 Web 数据管理/陈以农,蔡维德编著,郭小群等译. —西安:

西安电子科技大学出版社, 2013. 1

国外电子信息类系列教材

ISBN 978-7-5606-2934-6

I. ①面… II. ①陈… ②蔡… ③郭… III. ①互连网络—数据库管理系统—教材

IV. ①TP393.4

中国版本图书馆 CIP 数据核字(2013)第 004821 号

策划编辑 曹媛媛

责任编辑 曹媛媛 李 晶

出版发行 西安电子科技大学出版社(西安市太白南路 2 号)

电 话 (029)88242885 88201467 邮 编 710071

网 址 www.xduph.com 电子邮箱 xdupfxb001@163.com

经 销 新华书店

印刷单位 西安文化彩印厂

版 次 2013 年 2 月第 1 版 2013 年 2 月第 1 次印刷

开 本 787 毫米×1092 毫米 1/16 印张 35.5

字 数 838 千字

印 数 1~3000 册

定 价 64.00 元

ISBN 978-7-5606-2934-6/TP

XDUP 3226001-1

*** 如有印装问题可调换 ***

本社图书封面为激光防伪覆膜,谨防盗版。

前 言

本书译自《SERVICE-ORIENTED COMPUTING AND WEB DATA MANAGEMENT FROM PRINCIPLES TO DEVELOPMENT》2nd Edition, 美国 Kendall/Hunt 出版社出版, 陈以农、蔡维德编著。

软件开发已经演化了好几代, 经历了命令式、过程式、面向对象, 到分布式面向对象等范型。当面向服务的计算出现后, 分布式软件开发正在从分布式面向对象的开发, 其中以 OMG(对象管理组)开发的 CORBA(公共对象请求代理体系结构)和微软开发的 DCOM(分布构件对象模型)为代表, 向分布式面向服务的开发转移。

本书的第一版介绍了 SOSD(面向服务软件开发)的概念、原理、方法和最新技术。在本书的第二版中, 增加了近两年来大量发展的概念、原理和技术等新知识, 其他各章也都作了很大的修改和扩充。第二版新增的五章是:

第 5 章 Web 应用和数据管理, 讨论了具有状态的(stateful)Web 应用软件的开发, 采用各种不同的状态管理技术, 包括视图状态、会话状态、应用状态、文件管理和 Web 快存(caching)等。

第 7 章 面向服务的体系结构和 REST 体系结构, 介绍了 Web 服务开发、服务托管以及基于 Windows 通信基础(Windows Communication Foundation)的 REST 服务开发。

第 9 章 面向服务和事件驱动的机器人应用, 研究视觉编程语言(VPL), 并用其开发机器人服务和应用软件。还提出了分布的机器人应用和机器人即服务(RaaS)的概念。

第 10 章 面向服务的软件与数据库的连接, 介绍了面向服务的软件和关系数据库、XML 数据库、LINQ(集成查询语言 Integrated Language Query)和使用 LINQ 访问对象、关系数据库和 XML 数据库之间的接口。

第 14 章 云计算和软件即服务, 介绍了 SOC 发展的最新趋向, 包括多租户(multi-tenancy)结构, Web 数据库及文件系统、调度, 容错技术和实时计算等。还介绍了谷歌、微软和 Salesforce 公司的云计算平台和开发环境, 以软件即服务作为重点。

附录部分包含了一些教学材料, 提供了如何从零开始编写可以运行的应用程序的步骤。

感谢本书的作者陈以农、蔡维德两位老师给我们提供书稿, 允许我们翻译这本书。

感谢西北大学郝克刚教授, 他组织了本书的翻译工作并亲自翻译了序言部分。

还有西北大学的很多老师和学生参与了本书的初稿翻译工作, 其中韩冰同学翻译了第 1 章, 杨睿同学翻译了第 2 章, 韩辉同学翻译了第 3 章以及第二部分的图表, 包永强同学翻译了第 4 章, 丁剑洁同学翻译了第 5 章, 郭小群老师翻译了第 6 章, 黄晓娜同学翻译了第 7

章和第 14 章的 14.1、14.2、14.3 节,赵海川同学翻译了第 8 章和第 14 章的 14.4、14.5 节,董红妮同学翻译了第 9 章和第 14 章的 14.12、14.13 节,崔莉老师翻译了第 10 章,吴江老师翻译了第 11 章和第 14 章的 14.6、14.7、14.8 节,王鹏同学翻译了第 12 章和第 14 章的 14.9、14.14、14.15 节,崔宁宁同学翻译了第 13 章和第 14 章的 14.10、14.11 节;秦宝生老师翻译了附录 A 及附录的练习题,姬祥老师翻译了附录 B,吴莎同学翻译了附录 C。

感谢参与本书初稿翻译工作的这些老师和学生。在初译的基础上,郭小群、吴江和侯红老师完成了终译和统稿,但由于译者水平有限,不妥之处还希望读者指正。

最后,感谢所有为本书制作、出版、发行做出贡献的人们。

译 者

2012 年 12 月

序 言(第二版)

基于面向服务体系结构(SOA)、面向服务的软件开发(SOSD)和面向服务的计算(SOC)代表了现代软件的工程实践和技术。本书的第一版介绍了 SOSD 的概念、原理、方法和最新技术。因为 SOSD 仍然是一个迅速发展的年轻领域,两年前本书的第一版出版后,又涌现了许多新概念和新技术。在本书的第二版中,我们增加了大量近两年来新发展的概念、原理和技术等知识。除新增加的 5 章外,其他各章也都作了很大的修改和扩充。新增的各章是:

第 5 章 Web 应用和数据管理,讨论了具有状态的(stateful)Web 应用软件的开发,采用各种不同的状态管理技术,包括视图状态、会话状态、应用状态、文件管理和 Web 快存(caching)等。

第 7 章 面向服务的体系结构和 REST 体系结构,介绍了 Web 服务开发、服务托管以及基于 Windows 通信基础(Windows Communication Foundation)的 REST 服务开发。

第 9 章 面向服务和事件驱动的机器人应用,研究可视化编程语言(VPL),并用其开发机器人服务和应用软件。还给出了分布式机器人应用和机器人即服务(RaaS)的概念。

第 10 章 面向服务的软件与数据库的连接,介绍了面向服务的软件和关系数据库、XML 数据库、集成查询语言(LINQ, Language Integrated Query)以及使用 LINQ 访问对象、关系数据库和 XML 数据库之间的接口。

第 14 章 云计算和软件即服务,介绍了 SOC 发展的最新趋向,包括多租户(multi-tenancy)结构,Web 数据库及文件系统,调度、容错技术和实时计算等。本章还介绍了谷歌、微软和 Salesforce 公司的云计算平台和开发环境,以软件即服务作为重点。

我们把全书分为三部分,每一部分都可以用于一个不同的课程。第一部分包括前六章,可以供大学高年级学生或研究生的面向服务的计算和分布式软件开发课程使用。这部分强调计算范型、数据表示、数据管理和以编程语言为基础的软件开发。它介绍了基本的概念和原则,除技术和工具之外,还介绍了在传统软件工程课程中没有的内容,如多线程、事件驱动编程、面向服务的开发、基于 Web 的编程、Web 数据管理和可靠性及安全性机制开发等。

第二部分包括接下来的八章。这些章节建立在第一部分讨论的基本概念和原理之上,但是它们并不依赖前六章的细节。这部分强调高级别数据管理的组合,以及使用服务和大的构件来构成应用软件。它包括 Web 服务和在 Windows 通信基础以及企业软件开发使用的业务流程执行语言(BPEL)上的 REST 服务开发,用于事件驱动的软件开发和机器人应用的视觉程序设计语言(VPL),面向服务的软件与数据库的接口,本体语言和应用,面向服务的应用软件体系结构,案例研究和云计算等。这些材料可以用于高年级大学生和研究生的课程。第二部分的大部分内容都基于成熟的知识和技术,也讨论了许多有助于研究生选择其研究方向和课题的问题。

第三部分为附录，提供了从零开始编写可以运行的应用程序的逐步指导。这部分材料对那些编程背景不足的学生学习第一和第二部分至关重要。这些讲解和练习可以帮助学生将一些概念连接起来，把它们变成能工作的程序。这部分可以用于新生的课程，通过游戏编程、机器人编程和 Web 编程介绍计算的概念。

在亚利桑那州立大学，我们使用本书作为两门课程的教科书。第一个课程是 CSE445/598(分布式软件开发)，其中 CSE445 是高年级大学生课程，CSE598 是研究生课程。本课程主要教授本书第一部分的内容。在每章末尾给出两周的编程作业和课题。我们还使用本书作为一个新设立的课程 CSE494/598(面向服务的计算及信息管理)的教材，教授本书第二部分的内容。这两个课程可以独立开设，而不必使得其中之一是另一个课程的先决条件。不过，第一部分的基本概念和原理，包括第 1 章、第 3 章、第 4 章的 4.1 节，应当作为开始第二部分学习以前，准备所需的概念而要复习或布置的阅读材料。

感谢参与本项目、特别是参与本书第二版编写工作的诸位赞助者、支持者和同事。其中有清华大学的白晓颖教授，SAP 的 Shuyuan Chen 博士，IBM 的 J. Y. Chung 博士，清华大学的都志辉教授，Intel 的 Mr. Marcos Garcia-Acosta，北京大学的梅宏教授，IBM 的 K. W. Hwang 博士，北京大学的金芝教授，亚利桑那州立大学的 Y. H. Lee 教授，复旦大学的李银胜教授，Intel 的 Mr. John Oliver，DoD OSD NII 的 Raymond Paul 博士，亚利桑那州立大学的 S. S. Yau 教授，北京大学的郁莲教授。他们对我们深入理解这些资料做出了很多贡献。我们也感谢 Intel 公司、美国教育部、美国国防部和国家科学基金会的大力支持。没有他们的支持，这本书不可能出版。我们也感谢亚利桑州立大学的助教和助研们，他们是 Jay Elston, Wu Li, Guanqiu Qi, Qihong Edward Raleigh, Shao Xin Sun, Le Xu 和 Peide Zhong。感谢他们审定了书中所用的许多例子和作业。最后，我们还要感谢我们的家人，感谢她们对我们上大学繁忙饱满的研究和教学工作的同时承担这个项目所给予的支持和理解。

写给导师们

本书所有的习题和作业都经过了亚利桑州立大学的课堂教学测试。此外，本书中列出的所有程序代码也都经过了开发和测试。如果你有兴趣得到更多有关本书的资料请同作者联系。本书有相应的 Web 网站 <http://venus.eas.asu.edu/WSRepository/book/socwsi.html> 和 <http://www.public.asu.edu/~ychen10/book/socwsi.html>，可以从网站上下载本书有关的资源。想得到只有教师可看的资料，请直接同作者 {yinong, wtsai}@asu.edu 联系。

亚利桑州立大学 陈以农

亚利桑州立大学 清华大学 蔡维德

序 (第一版)

软件开发已经演化了好几代,经过了命令式、过程式、面向对象,到分布式面向对象等范型。当面向服务的计算出现后,分布式软件开发正在从分布式面向对象的开发,以OMG(对象管理组)开发的CORBA(共同对象请求代理体系结构)和微软开发的DCOM(分布式构件对象模型)为代表,向分布式面向服务的开发转移。面向服务的计算和面向服务的软件开发已经被所有的主要计算机公司,包括BEA、Google、HP、IBM、Intel、Microsoft、Oracle、SAP和Sun Microsystems等所采用和支持,而且他们的技术已经被OASIS、W3C和ISO制定为标准。

在我们介绍本书的内容以前,首先要澄清三个基本概念:面向服务的体系结构、面向服务的计算和面向服务的软件开发。

面向服务的体系结构(SOA)是一种分布式的软件体系结构,它由一群松耦合的服务组成,这些服务通过标准的接口和协议相互通信。这些服务是平台独立的。服务可以在公开的或私有的目录或服务库上发布,供软件开发者们组装他们的应用软件。作为一个软件的体系结构,SOA是一个概念模型,它关心的是组织结构和软件构件(服务)之间的接口,并不关心可运行软件的开发。

面向服务的计算(SOC)涉及的是计算范型,它建立在SOA的概念模型之上。然而SOC要更前进一步,不只关心概念和原理,还涉及方法、算法、编码和评估,它包括了软件开发过程的大部分内容。

面向服务的开发(SOD)涉及基于SOA概念和SOC范型的整个软件开发生命周期,包括需求、规约、结构设计、编写、服务发现、服务实现、测试、评估、部署和维护等。SOD还包括使用最新的技术和工具有效地生成可运行软件。

我们使用“分布式面向服务的软件开发”作为本书的标题是想与广泛使用的“分布式面向对象的软件开发方法”作对比,并且强调这样的事实:面向服务的软件开发自然是分布式的。不仅软件在开发中是分布的,使用不同的计算机在不同的地点开发,而且开发过程也是分布式的,也就是说应用软件的构建者、服务中介以及服务供应商都是独立地工作在不同地点的开发商,只是他们遵循着相同的接口和标准。此外,我们在第2章专门讨论了一般的分布计算以及如何使SOA、SOC和SOD适合于一般分布式计算的框架。

最近有许多关于SOA、SOC和SOD的书出版,反映了这些领域不断增长的需求。这些书分属下面三个范畴:

- (1) SOA的高级概念和原理;
- (2) SOC的某一个方面,如BPEL、Ontology或XML;
- (3) 使用特定平台的SOD,如Visual Studio .Net、Oracle SOA Suite、Java EE或WebSphere等。

这些书大部分是由开发商组织编写的,大都针对具体的语言、平台和工具。与现有的

这些书不同,本书采取了一种折中的方法,在一个教程里讲授 SOA、SOC 和 SOD 这三个主题,而且有一定深度地覆盖了每个主题的大部分内容。本书的重点是讲授 SOA/SOC 的概念、原理和方法。

在本书中,这些概念、原理和方法不仅通过正文和图形来解释,还通过可运行代码进行解释。我相信如果学生们看过了实现概念、原理和方法的可运行代码,就会对它们有更好的理解。

我们还介绍了当前最尖端的技术和工具,可以用它来开发具有适当规模和功能的可运行软件,例如一个可运行的网上在线书店、交易网站或者一个具有人工智能的能操纵真正机器人遍历迷宫的机器人程序。如果没有最新开发的工具或没有使用由专业服务供应商提供的可用的服务和构件,这样的软件在教程中作为作业开发是不可能完成的。在每章的结尾都布置了许多练习和至少一个大的“项目”作业,让学生去实践 SOA 和 SOC 的概念并开发可运行的软件。

本书在一定的宽度和深度上涵盖了 SOA、SOC 和 SOD 三个主题。

本书的编写是基于作者讲授的课程 CSE445/598(分布式软件开发)的教学材料,从 2006 年秋季开始,这个课程在亚利桑那州立大学计算机科学和工程学系每个学期都要讲授。

CSE445 课程是为高年级大学生开设的课程,CSE598 课程是为研究生开设的课程。CSE598 还有一个网上在线课程,教授给那些在职的工程硕士研究生。这些学生中的很多人在做软件的项目管理。本书的部分高级材料也在 CSE565 课程(软件验证、确认和测试)中使用。具有以下教学目标和结果的课程可使用本教材:

1. 增进对使用并发和同步程序的软件工程的理解,有如下结果:

- * 学生能够指出并发、线程和同步的应用、优点和缺点;
- * 学生能够使用并发和同步的设计原则;
- * 学生能够设计和编写程序来演示并发、线程和同步的使用。

2. 增进对分布式软件开发的理解,有如下结果:

- * 学生能够认识到各种分布式计算范型和技术;
- * 学生能够指出分布式软件开发中软件生命周期中的各个阶段和可发布的内容;
- * 学生能够在软件生命周期的各个阶段,创建分布式软件的开发中需要发布的文档;
- * 学生理解分布式应用的安全和可靠属性。

3. 设计和发布面向服务应用的构件,有如下结果:

- * 学生理解服务发布和服务目录的作用;
- * 学生能够在服务登记中指出可用的服务;
- * 学生能够用程序设计语言设计服务,并发布服务让公众使用。

4. 学习使用当前开发分布式系统和应用技术各种技巧,有如下结果:

- * 学生能够使用当前的技术和标准开发分布式程序;
- * 学生能够使用当前的开发程序的框架,利用图形用户界面(GUI)、远程服务和工作流开发程序和 Web 应用。

本书并不是一部讲授程序设计的导引性教程。它的主要读者是计算机科学与工程专业的高年级大学生、研究生,或具有程序设计背景的软件工程师。本书读者应熟悉一种面向

对象的程序语言，如 C++、C# 和 Java 等。此外还期望读者对基本的软件工程原理有所了解。

本书由 9 章和 1 个附录组成。每章都是一个单元，根据教师教授的细致程度，每章约需要 6~9 个学时。

本书不是一部研究专著，而是教授大学生和研究生们学习 SOA、SOC 和 SOD 的教科书。然而，作研究的学生和在职的专业人员仍然会发现本书非常有用，因为它对当今的前沿内容、最尖端的技术和专业的开发工具等作了全面和深入的讨论。本书不仅基于作者在这些领域的教学经验，而且也基于作者在这些领域多年的研究工作中积累的经验 and 专业知识。

由于 SOA、SOC 和 SOD 是新的领域，而且是动态发展的，有关技术和工具在迅速地演变。也许某些材料在书印刷出版以后很快就会更新。我们意图包括最新的概念和技术，但是又不得不在某个点上中断这个过程。我们重点强调 SOA 和 SOC 概念、原则和方法，与 SOD 技术和工具比较，它们是相对稳定的。本书的材料是在 2006 年秋天开始讲授的。大部分开发的例子最初是基于 .Net 2005 的。现在 .Net 2008 已经发布。在书出版之前，我们做很少的修改或不修改，就能在 .Net 2008 中测试所有例子。我们期望这些例子在将来的新版工具下也能正常运行。

本书附录中的教程是本书的重要补充。它提供了在第 3 章讨论的 Web 应用开发和第 5 章讨论的机器人软件开发实现的全部细节。另一方面，该教程可以独立讲授，作为那些没有编程经验学生的主要教材。实际上，该教程的内容在为某些高中生开设的面向服务的计算的课程中已经教授过了。

我们非常感谢这个项目的赞助者、支持者和同事们，他们有清华大学的白晓颖教授，亚利桑那州立大学的 Gary Bitter 教授，得克萨斯大学达拉斯分校的 Farokh Bastani 教授，考文垂大学的 Kuo-Ming Chao 教授，SAP 的 Shuyuan Chen 博士，IBM 的 J. Y. Chung 博士，清华大学的都志辉教授，IBM 的 K. W. Hwang 博士，加州大学欧文分校的 Kane Kim 教授，亚利桑那州立大学的 Y. H. Lee 教授，复旦大学的李银胜教授，加州大学欧文分校的 K. J. Lin 教授，DoD OSD NII 的 Raymond Paul 博士，亚利桑那州立大学的 Mary White 博士，亚利桑那州立大学的 S. S. Yau 教授，得克萨斯大学达拉斯分校的 I-Ling Yen 教授。他们对我们理解这些材料做出了许多贡献。我们也感谢美国教育部和美国国防部的慷慨支持。没有他们的支持，这本书不可能出版。我们也感谢亚利桑那州立大学的助教和助研们。他们是 Zhibin Cao, Calvin Cheng, Sandy Chow, Jay Elston, Qian Huang, Sheng Liu, Zheng Liu, Wu Li, Xin Sun, Jingjing Xu, Xinyu Zhou, Peide Zhong。感谢他们审定了本书的很多例子和作业。最后我们要感谢我们的家人，感谢她们对我们大学繁忙饱满的研究和教学工作同时承担这个项目给予的支持和理解。

陈以农

蔡维德

2008 年 5 月 1 日

目 录

第一部分 分布式面向服务的软件开发和数据管理

第 1 章 分布式面向服务计算导引

.....	2
1.1 计算机体系结构和计算范型.....	2
1.1.1 计算机体系结构.....	2
1.1.2 软件体系结构.....	3
1.1.3 计算范型.....	3
1.2 分布式计算和分布式软件体系结构.....	6
1.2.1 分布式计算.....	6
1.2.2 N 层体系结构.....	6
1.2.3 分布式对象体系结构.....	8
1.3 面向服务的体系结构和计算.....	10
1.3.1 基本概念和术语.....	10
1.3.2 面向服务的计算.....	13
1.3.3 面向对象计算与面向服务计算.....	14
1.3.4 面向服务的企业.....	16
1.3.5 面向服务的系统工程.....	17
1.4 面向服务的开发与应用.....	18
1.4.1 传统的软件开发过程.....	18
1.4.2 面向服务的软件开发.....	19
1.4.3 面向服务计算的应用.....	21
1.4.4 Web 应用组合.....	22
1.5 讨论.....	23
1.6 练习和作业.....	26

第 2 章 多线程分布式计算

2.1 C#和 .Net 介绍.....	32
2.1.1 C#与 .Net 入门.....	33
2.1.2 C#和 C++ 的比较.....	36
2.1.3 名字空间和 using 指令.....	38
2.1.4 C#中的队列例子.....	39
2.1.5 C#中的类和对象.....	41
2.1.6 参数:用 ref 和 out 传递引用.....	44
2.1.7 基类和基类构造函数调用.....	45

2.1.8 构造函数、析构函数和垃圾回收.....	45
2.1.9 C#中的指针.....	46
2.1.10 C#的统一类型系统.....	47
2.2 内存管理和垃圾回收.....	48
2.2.1 静态变量和静态方法.....	49
2.2.2 局部变量的运行时栈.....	49
2.2.3 动态存储分配的堆.....	51
2.2.4 作用域和垃圾回收.....	52
2.3 多任务和多线程的一般问题.....	52
2.3.1 基本需求.....	53
2.3.2 临界操作和同步.....	54
2.3.3 死锁和死锁的解决.....	55
2.3.4 执行顺序.....	56
2.3.5 操作系统对多任务和多线程的支持.....	57
2.4 Java 中的多线程.....	59
2.4.1 创建和启动线程.....	59
2.4.2 线程同步.....	63
2.4.3 同步方法.....	63
2.4.4 同步语句.....	68
2.5 C# 中的多线程.....	69
2.5.1 线程类和属性.....	69
2.5.2 监视器.....	70
2.5.3 读锁和写锁.....	81
2.5.4 互斥.....	85
2.5.5 信号量.....	86
2.5.6 协调事件.....	86
2.6 异常处理和事件驱动编程.....	89
2.6.1 异常处理.....	90
2.6.2 事件驱动编程.....	94
2.6.3 案例研究:一个使用事件驱动方法开发的电子商务应用.....	98
2.7 讨论.....	103

2.8 练习和作业	103	4.2.4 Java 中的 XML 处理	172
第 3 章 面向服务软件开发概要	114	4.3 XPath	174
3.1 面向服务的软件开发环境综述	114	4.4 XML 类型定义语言	176
3.2 服务提供商:创建和托管服务	115	4.4.1 XML 文档类型定义(DTD)	177
3.2.1 使用 ASP. Net 创建 Web 服务	116	4.4.2 XML 模式	179
3.2.2 使用 WCF 创建 Web 服务	119	4.4.3 名字空间	182
3.2.3 测试 Web 服务	122	4.4.4 XML 确认	184
3.2.4 作为服务提供商托管 Web 服务	123	4.5 可扩展样式表语言	186
3.3 服务中介:服务的发布和发现	125	4.6 其他 Web 数据格式	191
3.3.1 具有全部所需特征的服务中介	125	4.6.1 XHTML	192
3.3.2 UDDI 服务注册	127	4.6.2 RSS	193
3.3.3 ebXML 服务注册和存放	135	4.6.3 Atom	196
3.3.4 临时存放表	137	4.6.4 Google 协议缓冲区和 BigTable	198
3.4 SOAP	137	4.7 讨论	200
3.4.1 SOAP 格式	138	4.8 练习和作业	200
3.4.2 基于 HTTP 的 SOAP	139	第 5 章 Web 应用和数据管理	205
3.5 WSDL: Web 服务描述语言	140	5.1 Web 应用的结构	205
3.5.1 WSDL 文档的元素	141	5.1.1 Web 应用的构件	205
3.5.2 WSDL 文档的例子	142	5.1.2 服务控件	208
3.6 服务请求者:用服务构建应用	143	5.1.3 用户控件	212
3.6.1 连接端点和代理	143	5.1.4 Web 应用配置	214
3.6.2 用 ASP. Net 创建 Web 应用项目	144	5.1.5 全局应用类	217
3.6.3 创建 GUI 并组合一个基于远程 Web 服务的应用	146	5.1.6 动态链接库	218
3.7 基于 Java 的 Web 服务开发	151	5.2 Web 应用模型	219
3.7.1 用 AJAX 构建 Web 应用	152	5.2.1 纯基于 HTML 的 Web 应用	219
3.7.2 基于 Java 的 Web 服务开发和托管	154	5.2.2 客户端脚本编写	220
3.8 讨论	156	5.2.3 服务器端脚本编写和后台代码计算	222
3.9 练习和作业	156	5.3 状态管理	223
第 4 章 XML 数据表示和处理	160	5.3.1 状态管理技术综述	223
4.1 XML 基础	161	5.3.2 视图状态	224
4.1.1 XML 和 HTML	161	5.3.3 会话状态	226
4.1.2 XML 语法	162	5.3.4 会话状态和 Cookies	229
4.1.3 XML 名字空间	165	5.3.5 应用状态和服务执行模型	230
4.2 XML 数据处理	166	5.4 将用户数据存到服务器文件系统	231
4.2.1 DOM:文档对象模型	167	5.4.1 文件系统与标准文件操作	231
4.2.2 SAX:XML 的简单 API	169	5.4.2 XML 文件的读写	232
4.2.3 XML 文本编写器	171	5.5 高速缓存	236
		5.5.1 输出高速缓存	237

5.5.2 分段高速缓存	239	6.2.1 IIS 和基于 Windows 的 安全机制	258
5.5.3 数据高速缓存	239	6.2.2 基于表单的安全	259
5.6 图形数据的生成和表示	245	6.2.3 使用安全套接层	265
5.7 讨论	249	6.3 Windows 通信基础的可信计算	265
5.8 练习和作业	250	6.3.1 WS 安全性	265
第 6 章 面向服务软件的可信性	254	6.3.2 WS 可靠性	267
6.1 基本概念	254	6.3.3 事务	269
6.1.1 可信性	254	6.4 讨论	270
6.1.2 服务的可信属性和质量	255	6.5 练习和作业	271
6.1.3 SOA 软件的安全问题	256		
6.2 Web 应用的安全设计	258		

第二部分 高级面向服务计算和系统组成

第 7 章 面向服务的体系结构和 REST 体系结构	274	8.3 无状态与有状态 Web 服务	317
7.1 Windows 通信基础综述	274	8.3.1 BizTalk 的 singleton 对象方法	318
7.2 使用模板开发 WCF 服务	276	8.3.2 BPEL 的相关性方法	318
7.3 开发自托管的 WCF 服务	280	8.4 支持 BPEL 组合的框架	321
7.4 开发使用自托管服务的客户	283	8.4.1 Oracle SOA 套件	321
7.5 开发使用 WCF 服务的 ASP. Net GUI 客户	288	8.4.2 ActiveBPEL	322
7.6 REST 概念和 REST 型服务	290	8.4.3 BizTalk	323
7.6.1 REST 概念和 REST 体系结构	291	8.5 Mashup 用于 Web 应用的组合	324
7.6.2 REST 型服务	291	8.6 其他组合语言	327
7.6.3 把 SOAP 服务转换为 REST 型服务	295	8.6.1 OWL-S	327
7.6.4 使用 REST 型服务	296	8.6.2 SCA/SDO	328
7.7 讨论	297	8.6.3 workflow 基础和 Silverlight	329
7.8 练习和作业	297	8.6.4 WSFL: Web 服务流语言	330
第 8 章 通过组合进行面向服务软件开发	300	8.7 讨论	331
8.1 组合语言概述	300	8.8 练习和作业	331
8.2 BPEL	302	第 9 章 面向服务和事件驱动的机器人应用	335
8.2.1 BPEL 活动和结构	303	9.1 面向服务的机器人计算	335
8.2.2 BPEL 流程	303	9.2 事件驱动的机器人应用	336
8.2.3 BPEL 流程的 WSDL 接口定义	305	9.3 云计算中机器人即服务	339
8.2.4 BPEL 流程	307	9.4 Robotic Studio 和可视化编程语言	341
8.2.5 一个调用实际的 Web 服务的例子	310	9.5 用 VPL 仿真 ALU	345
		9.5.1 算术和逻辑单元的逻辑设计	345
		9.5.2 创建 VPL 活动	346
		9.5.3 VPL 图和测试	347

9.6 有限自动机和 VPL 图	349	11.5.5 OWL Lite、DL 和 Full 简介	409
9.7 开发面向服务的机器人应用	353	11.6 本体开发环境	411
9.7.1 Robotics Studio 的服务库	353	11.7 讨论	412
9.7.2 传感器服务	354	11.8 练习和作业	413
9.7.3 迷宫导航算法	354	第 12 章 面向服务应用的体系结构 ..	416
9.7.4 用 VPL 实现迷宫导航算法	355	12.1 引言	416
9.8 VPL 到其他平台的映射	361	12.2 应用的体系结构	416
9.9 讨论	363	12.2.1 动态体系结构和动态组合	419
9.10 练习和作业	364	12.2.2 动态再组合	420
第 10 章 面向服务软件和数据库的		12.2.3 嵌入在运行基础设施中的	
接口	367	生命周期管理	421
10.1 面向服务软件中的数据库	367	12.3 面向服务应用的体系结构的例子 ..	422
10.2 面向服务软件中的关系数据库	368	12.3.1 IBM WebSphere 的体系结构 ..	423
10.2.1 数据库和软件之间的接口	368	12.3.2 企业服务总线	424
10.2.2 ADO.Net 中的 SQL 数据库 ..	370	12.3.3 SAP 的 NetWeaver	425
10.2.3 ADO.Net 中的数据适配器和数据集	376	12.3.4 用户为中心的面向服务的体系结构	426
10.3 基于 XML 的数据库和查询语言 XQuery	378	12.4 讨论	427
10.3.1 查询的表达	379	12.5 练习和作业	428
10.3.2 XML 文档的转换	381	第 13 章 面向服务软件开发的模拟练习	430
10.3.3 XQuery 讨论	383	13.1 引言	430
10.4 综合查询语言 LINQ	383	13.2 领域模型举例	433
10.4.1 LINQ 的目标	383	13.2.1 本体系统	433
10.4.2 λ 表达式	383	13.2.2 发布的服务	437
10.4.3 LINQ 与对象	386	13.2.3 发布的工作流	439
10.4.4 LINQ 与 SQL	387	13.2.4 物流领域协作模板	440
10.4.5 LINQ 与 XML	390	13.3 项目的特定需求	441
10.5 讨论	392	13.4 一个可运行的例子	442
10.6 练习和作业	393	13.5 讨论	449
第 11 章 本体和语义 Web	396	13.6 练习和作业	450
11.1 语义 Web 和本体概述	396	第 14 章 云计算和软件即服务	452
11.2 本体语言 RDF	397	14.1 引言	452
11.3 RDF 模式	399	14.2 SaaS 的成熟度模型	455
11.4 本体的推理和验证	405	14.3 多租户 SaaS 的数据库设计	458
11.5 Web 本体语言 OWL	406	14.3.1 资源隔离模式	459
11.5.1 从 RDF 到 OWL	406	14.3.2 安全性	463
11.5.2 OWL 的类和属性	407	14.3.3 可伸缩性	464
11.5.3 类的布尔组合	408	14.4 Google App 引擎	465
11.5.4 属性限制	408	14.4.1 服务	466

目 录

14.4.2 数据存储	466	14.8.1 Hadoop 生态系统	480
14.4.3 开发可伸缩应用的提示	467	14.8.2 Hadoop HDFS 和 MapReduce	481
14.4.4 开发工具	467	14.8.3 例子	481
14.4.5 其他约束	468	14.9 微软的 Azure	483
14.5 Google 文件系统	468	14.9.1 Azure 的体系结构	483
14.5.1 GFS 系统结构和操作	469	14.9.2 Azure 的元素	484
14.5.2 GFS 开发中的经验	470	14.9.3 Azure 的云应用	486
14.5.3 其他类似项目	470	14.10 Salesforce.com	486
14.6 BigTable	471	14.10.1 顶层体系结构	486
14.6.1 主要构件	471	14.10.2 元数据驱动的体系结构	487
14.6.2 BigTable 概述	472	14.10.3 多租户的数据库设计	488
14.7 MapReduce	477	14.10.4 对象和字段元数据表	488
14.7.1 MapReduce 编程模型	477	14.11 优先级和调度	492
14.7.2 例子	478	14.12 云计算算法	497
14.7.3 应用	478	14.13 数据区分器的应用	508
14.7.4 执行概述	479	14.14 讨论	509
14.7.5 任务粒度和容错计算	479	14.15 练习和作业	509
14.8 Hadoop	480		

附录 基于构件的面向服务的软件开发指南

附录 A 基于构件的动画和游戏编程	512	附录 C 面向服务的机器人应用	533
附录 B Web 应用开发	517		

第一部分

分布式面向服务的 软件开发和数据管理

第 1 章

分布式面向服务计算导引

本章介绍计算机体系结构、不同的计算范型特别是分布式计算范型和面向服务的计算(SOC)范型。

1.1 计算机体系结构和计算范型

软件体系结构和分布式软件开发与执行软件的计算机系统的体系结构是相关的。本节介绍计算机系统结构以及各种计算范型。

1.1.1 计算机体系结构

对于单处理器计算机,计算机体系结构通常指的是处理器的体系结构,它是软件和硬件之间的接口或者处理器的指令体系结构[Patterson 2004]。对于多处理器计算机,体系结构通常指的是指令和数据流。*Flynn's Taxonomy* [Flynn 1972]将计算机体系结构分为四种类型:

- 单指令单数据流(SISD),即简单的处理器系统;
- 单指令多数据流(SIMD),例如向量或阵列式计算机;
- 多指令单数据流(MISD),例如对同一数据流执行冗余计算并对结果进行表决的容错计算机系统;
- 多指令多数据流(MIMD),由拥有自己的内存和控制器、算术逻辑运算部件以及输入输出部件的独立计算机系统组成的计算机系统。

MIMD 被看做是分布式系统,不同的分布式系统有不同的关注点,如图 1.1 所示。分布式计算关注用并行或(和)分布方式表达计算时涉及的原理、方法以及技术。分布式软件体系结构关注的是软构件间的组织和接口。网络体系结构研究的是网络节点的拓扑结构以及连接。网络通信研究协议层,协议层允许节点之间相互通信并理解彼此之间的数据格式。有些研究者用操作系统区分分布式系统和网络。当一组网络节点有独立的操作系统时,分布式系统是内聚性的操作系统。