

深入 Enterprise JavaBeans

编程

陈松 李亚伟 编著

- Enterprise JavaBeans 是 Java 技术所遵循的“一次编写，随处运行”的跨平台特性。
- 本书针对 EJB，对 CORBA 和 J2EE 等关键理论作了讲述，并且对 RMI-IIOP、JNDI 和 JTS 程序的设计均作了较为深入详细的讲解。
- 帮助读者熟悉分布式计算的三大标准、面向对象程序的开发和设计，掌握 J2EE、CORBA 和 EJB 的结合关系、Enterprise JavaBeans 的基础理论、EJB 程序的开发、EJB 的互操作技术以及 EJB 的一些高级程序开发。
- 本书适合于有面向对象程序设计和 JAVA 程序设计经验的人阅读，也适合于作为 EJB 的教程。



中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

深入

ENTERPRISE

JAVABEANS

编程

陈松
李亚伟
编著

中国铁道出版社

2002 • 北京

(京)新登字 063 号

内 容 简 介

Enterprise JavaBeans (EJB) 是 SUN 公司提出的开发三层企业级 Java 应用的工业标准, 已被目前绝大多数应用服务器厂商作为遵循的规范, 具有“编写一次, 随处运行”的跨平台特性。本书针对 EJB; 对 CORBA 和 J2EE 等关键理论作了讲述, 并且对 RMI-IIOP、JNDI 和 JTS 程序的设计均作了较为深入详细的讲解。

通过本书的学习, 可以帮助读者熟悉分布式计算的三大标准、面向对象程序的开发和设计, 掌握 J2EE、CORBA 和 EJB 的结合关系、Enterprise JavaBeans 的基础理论、EJB 程序的开发、EJB 的互操作技术以及 EJB 的一些高级程序开发。本书适合于有面向对象程序设计和 JAVA 程序设计经验的人阅读, 也适合于作为 EJB 的教程。

图书在版编目 (CIP) 数据

深入 Enterprise JavaBeans 编程/陈松、李亚伟编著. —北京: 中国铁道出版社, 2002. 8

(开发者说)

ISBN 7-113-04815-3

I. 深… II. ①陈…②李… III. JAVA 语言-程序设计 IV. TP312

中国版本图书馆 CIP 数据核字 (2002) 第 053857 号

书 名: 深入 Enterprise JavaBeans 编程

作 者: 陈 松 李亚伟

出版发行: 中国铁道出版社 (100054, 北京市宣武区右安门西街 8 号)

策划编辑: 严晓舟 魏 春

责任编辑: 苏 茜 田志国 黄园园

封面设计: 孙天昭

印 刷: 北京市燕山印刷厂

开 本: 787×1092 1/16 印张: 19.75 字数: 462 千

版 本: 2002 年 8 月第 1 版 2002 年 8 月第 1 次印刷

印 数: 1~4000 册

书 号: ISBN 7-113-04815-3/TP·753

定 价: 33.00 元

版权所有 侵权必究

凡购买铁道版的图书, 如有缺页、倒页、脱页者, 请与本社计算机图书批销部调换。

前言

Internet 的日益发展导致了传统的客户机/服务器计算模式的革新。在两层计算模式中，客户机与服务器紧密结合，不仅性能低，而且管理复杂。业务逻辑的变动则导致必须重写应用程序，从而影响系统的可扩展性。同时，客户机直接访问服务器端的数据库及对数据库的各种操作，使系统安全性难以得到保障。因此，“表示层+业务逻辑层+数据层”的三层或多层的模式已经成为分布式 Internet 计算的主流。

Enterprise JavaBeans (EJB) 是 SUN 公司提出的开发三层企业级 Java 应用的工业标准，目前已被绝大多数应用服务器厂商作为遵循的规范。EJB 技术以 Java 2 平台为基础，继承了 Java 语言中的所有优点，如“编写一次，随处运行”的跨平台特性。J2EE/EJB 规范提供了各种公用服务，分别为表示层、业务逻辑层和数据层上的应用组件提供了功能完善的运行平台。

完全遵循 J2EE/EJB 工业标准的应用服务器，可以在异构的环境中，以完备的功能、充分的可靠性和强大的性能满足各种需求的应用系统，将覆盖电子商务网站、电信、银行、邮政、电力、计算机、办公设备、金融证券、投资理财、制药、保健、航空货运、航天、电子设备、保险等各种领域。J2EE 应用服务器将复杂的技术问题与业务问题分离，使用户能专注于业务问题的解决，从而为企业应用带来了可伸缩性，让系统随业务的增长一起增长；同时为企业提供一个整套开发、部署、调试和运行所需应用的工具，帮助用户快速开发分布式三层应用。

在国际上，世界 500 强中有超过 80% 的企业使用应用服务器。国内，目前与应用服务器相关的应用集中在邮电、金融、证券和电子商务等领域。随着电子商务技术的成熟，将会有更多的传统行业使用应用服务器作为业务迁移到 Web 的引擎，特别是中小型企业对应用服务器的需求将会日益增加。



EJB2.0 新特性

- ◆ 引入了消息分发 Bean。
- ◆ EJB 容器管理实体 Bean 的持久性。
- ◆ 实体 Bean 查询语义的定义。
- ◆ 实体 Bean 选择方法。
- ◆ 在 Home 接口中提供了独立于 Enterprise Bean 实例的方法来实现业务逻辑。
- ◆ Enterprise Bean 中增加了安全标志。
- ◆ 定义了互操作协议。



本书特点

理论知识丰富：本书不仅仅从程序开发的角度给读者讲解 EJB 技术，而且从 EJB 体系结构、分布计算和异构平台互操作的角度论述了三层分布计算的理论知识。

涉及面广：CORBA 技术是 EJB 互操作和对象访问的基石，J2EE 是 EJB 与 Web 结合的标准接口。本书针对 EJB，对 CORBA 和 J2EE 等关键理论作了讲述，最后，对 RMI-IIOP、JNDI 和 JTS 程序的设计均做了较为深入详细的讲解。

编著人员经验丰富：本书的编著人员具有 CORBA 和 J2EE/EJB 产品核心的设计和开发经验，对 EJB 技术和面向对象程序设计有较深入的了解和深刻的体会。立足于此，本书结合 EJB 的核心细节对 EJB 开发做了细致的描述。



目标

通过对本书的学习，可以帮助读者掌握以下内容：

- ◆ 了解分布式计算的三大标准。
- ◆ 熟悉面向对象程序的开发和设计。
- ◆ 了解构件化技术的发展热点。
- ◆ 掌握 J2EE、CORBA 和 EJB 的结合关系。
- ◆ 掌握 Enterprise JavaBeans 的基础理论。
- ◆ 掌握 Enterprise JavaBeans 程序的开发技巧。
- ◆ 掌握 Enterprise JavaBeans 的互操作技术。
- ◆ 掌握 Enterprise JavaBeans 的一些高级程序开发技巧。



阅读指南

本书的内容安排如下：

第 1 章，对象中间件。对象中间件是支持面向对象开发和应用集成的中间件。本章介绍了分布式计算的三大标准 COM/DCOM、J2EE/EJB 和 CORBA，比较了各自的优缺点和发展历程。

第 2 章，CORBA 的相关概念。CORBA 是基于对象管理体系结构的。对象请求代理 ORB，规定了分布对象的定义（接口）和语言映射，实现对象间的通讯和互操作；在 ORB 之上定义了很多公共服务，如名字服务、事件服务、事务（交易）服务和安全服务等；IIOP 通信协议是 EJB 和其他系统及 EJB 之间进行互操作的基础。

第 3 章，J2EE 概述。J2EE 对企业级关键应用提供了高可靠、高性能和安全的开发平台。J2EE 支持多种工业标准的协议和界面引擎，包括数据库互联标准（ODBC 和 JDBC）、组件标准（CORBA、COM/DCOM 及 Enterprise JavaBeans（EJB））和通信协议（IIOP 和 HTTP）等。

第 4 章，EJB 概述。EJB 是由 Sun 公司提出的基于 Java 的面向对象的组件标准，EJB 组件包含一定的业务规则，运行在服务器端。采用 EJB 技术能有效地进行软件复用，提高开发

人员的效率,降低软件的开发和维护成本,提高软件的质量,控制所构建系统的复杂性。会话对象、实体对象、消息分发对象和配置描述符等是构成 EJB 技术的基础。

第 5 章,会话 Bean。会话 Bean 是被创建用来执行客户的某些业务逻辑并和客户进行交互的 EJB。会话 Bean 有无状态会话对象和有状态会话对象之分,本章介绍了会话 Bean 的编程方法和函数接口。

第 6 章,实体 Bean。实体 Bean 是被创建用来对企业系统中的数据进行封装的 EJB,实体 Bean 和数据库拥有相同的生命周期。本章介绍了实体 Bean 的编程方法和函数接口。

第 7 章,实体 Bean 的持久性管理。本章主要介绍了如何实现 Bean 和容器的持久性管理,并针对具体的 DEMO 程序进行阐述。

第 8 章,EJB 对分布式计算和互操作的支持。互操作是异构平台之间进行通信的一个重要概念。本章介绍了 RMI 的互操作、事务的互操作、名字的互操作和安全的互操作等。

第 9 章,IDL 文件的编写。接口定义语言是分布对象计算的基石,本章主要介绍了接口定义语言的语义及接口定义语言的编写。

第 10 章,EJB 远程通信模型。RMI 和 RMI-IIOP 是 EJB 的两种远程通信模型。本章介绍了两种模型的特点,并介绍了如何编写基于 RMI 和 RMI-IIOP 的远程通信程序。最后阐述了 EJB 如何与 CORBA 对象之间进行交互并附有一个完整的例子。

第 11 章,EJB 与 Java 名字目录服务。名字目录服务提供了一种对象按名访问的方式。本章介绍了名字服务的基本概念和有关名字服务所提供的函数接口,最后介绍了名字服务器的启动和注册的方式。

第 12 章,EJB 的交易模型。事务服务为分布式面向对象的环境下联机事务处理应用系统的开发和运行提供了灵活且易用的平台,可以保证联机事务处理系统运行的高效性和数据的完整性。本章介绍了 Bean 和容器两种交易管理模式以及使用 EJB 实现交易处理的方法。



适用对象

本书适合于有面向对象程序设计和 JAVA 程序设计经验的人阅读,也适合于作为 EJB 的教材使用。



致谢

本书第 1 章至第 3 章由王珊编写,第 4 章至第 10 章由陈松编写,第 11 章至第 12 章由李亚伟编写,王敏毅、李琪林负责了本书的审校工作。

本书在编写过程中得到了很多人的帮助:东方通科技的刘邦涛在本书的组织过程中做了大量的指导工作;杨涛为本书提供了大量的资料;王敏毅、李琪林两位博士为本书做了审校工作,在此一并致谢!

另外,参与本书审校工作的还有:孙忠、刘小伟、邓勇、欧阳劲、张云勇、卢军、唐寅、邹思轶、张炯明、李彬、卫星、贺玉龙、陈明、李宋琛、邓海、吴文锦。



特别说明

若读者、网友发现有网站未经作者及出版社授权，而转载本书内容或提供各种形式的下载服务，请予举报。经查属实，将予以重奖。

由于本书篇幅较大，涉及技术内容广泛，加之时间仓促，书中难免存在错误和疏漏之处，希望广大读者给予批评指正。

延伸服务：如果读者愿意参加“深入 Enterprise JavaBeans 编程”的培训，或是在学习过程中发现问题或有更好的建议，欢迎致电联系。联系电话：(028) 5404228 5460593；网址：www.bojia.net；E-mail:bojiakeji@163.net。通讯地址：成都四川大学（西区）建筑学院成都博嘉科技资讯有限公司，邮编：610065。

目 录

第 1 章 分布式对象技术	1
1-1 对象中间件	2
1-1-1 中间件	2
1-1-2 分布式对象中间件	2
1-2 分布式计算的三大标准	3
1-2-1 分布式对象技术的发展	3
1-2-2 J2EE/EJB 技术	4
1-2-3 CORBA 技术	5
1-2-4 COM/DCOM	6
1-3 J2EE/EJB 的优点	7
1-3-1 J2EE 技术	7
1-3-2 J2EE/EJB 技术的领先优势	8
第 2 章 CORBA 的相关概念	13
2-1 CORBA 基础	14
2-1-1 CORBA 的体系结构	14
2-1-2 何谓 ORB	15
2-1-3 接口定义语言 (IDL)	15
2-1-4 对象适配器	19
2-1-5 CORBA 的消息通信机制 GIOP/IIOP	20
2-1-6 互操作	26
2-1-7 对象引用	27
2-1-8 客户存根和服务骨架	27
2-2 CORBA 3.0 的新特性	28
2-2-1 Java 与 Internet 的集成	28
2-2-2 服务质量控制	29
2-2-3 CORBA 构件技术	29
2-3 CORBA 服务	32
2-3-1 CORBA 服务概述	32
2-3-2 名字服务	33
2-3-3 事件服务	34
2-3-4 持久对象服务 POS 和持久状态服务 PSS	34
2-3-5 事务服务	36
2-3-6 安全服务	37

2-3-7 CORBA 消息服务.....	40
第 3 章 J2EE 概述.....	43
3-1 Java 基础	44
3-1-1 JDBC 数据库连接.....	44
3-1-2 Java 消息服务	49
3-1-3 JavaServlet 和 JSP.....	51
3-2 J2EE 基础.....	51
3-2-1 J2EE 概念	51
3-2-2 J2EE 基本结构.....	52
3-2-3 JavaIDL 编译器.....	55
3-2-4 Java 名字目录服务	58
3-2-5 Java 事务服务和事务处理 API	59
3-2-6 XML 基础	59
第 4 章 EJB 概述.....	67
4-1 Enterprise Beans 组件	68
4-1-1 EJB (Enterprise JavaBeans) 概念.....	68
4-1-2 Session Bean	69
4-1-3 Entity Bean.....	69
4-1-4 容器管理和 Bean 管理的持久性	70
4-1-5 EJB Container	70
4-1-6 EJB 的设计模式和命名约定	71
4-1-7 JavaBean 和 EJB 的区别.....	71
4-2 EJB 的角色.....	72
4-3 部署描述符	73
4-3-1 EJB 开发的角色	73
4-3-2 EJB 部署的步骤	73
4-3-3 定义部署描述符	74
4-3-4 EJB Jar 文件	74
4-4 EJB 到 CORBA 的协议映射.....	75
第 5 章 会话 Bean.....	77
5-1 会话 Bean 的基本概念	78
5-2 会话 Bean API 接口.....	80
5-2-1 javax.ejb.SessionBean 接口.....	80
5-2-2 javax.ejb.SessionContext 接口	81
5-2-3 javax.ejb.SessionSynchronization 接口.....	85
5-2-4 使用 EJB 的安全接口.....	86
5-3 无状态会话 EJB.....	89

5-3-1	"Hello World"无状态会话 Bean 实例.....	90
5-3-2	无状态会话 Bean 的生命周期管理.....	92
5-4	有状态会话 Bean.....	93
5-4-1	会话 Bean 的不可重入性.....	93
5-4-2	状态的激活与钝化.....	93
5-4-3	一个有状态会话 Bean 的例子.....	93
5-4-4	有状态会话 Bean 的生命周期.....	98
第 6 章	实体 Bean.....	99
6-1	持续化对象.....	100
6-1-1	Java 对象的序列化.....	100
6-1-2	对象/关系的映射.....	100
6-1-3	对象数据库.....	101
6-2	实体 Bean.....	101
6-2-1	实体 Bean 的组成.....	101
6-2-2	实体 Bean 的特点.....	102
6-2-3	实体 Bean 与会话 Bean 的区别.....	103
6-3	实体 Bean 中的 API 及主要概念.....	108
6-3-1	javax.ejb.EntityBean 接口.....	108
6-3-2	javax.ejb.EntityContext 接口.....	112
6-3-3	主键对象.....	113
6-3-4	实体 Bean Home.....	113
6-4	深入 Jonas 容器内部.....	117
6-4-1	实体 Bean 代表的持续化对象的生命周期.....	117
6-4-2	实体 Bean 实例的生命周期.....	118
6-4-3	实体 Bean 生命管理的具体实现.....	120
第 7 章	实体 Bean 的持久性管理.....	125
7-1	Bean 管理的持久性.....	126
7-1-1	一个 BMP 实体 Bean 实例.....	126
7-1-2	BMP 中容器的实现.....	139
7-2	容器管理的持久性.....	140
7-2-1	一个 CMP 实例.....	141
7-2-2	CMP 中容器所做的工作.....	147
7-2-3	CMP 的标准配置.....	158
7-2-4	CMP 中与容器相关的标准配置.....	158
第 8 章	EJB 对分布式计算互操作的支持.....	161
8-1	对分布式的支持.....	162
8-2	互操作概述.....	162

8-2-1	异步环境概述	162
8-2-2	互操作的目的	163
8-2-3	互操作语义	163
8-2-4	互操作需求概述	166
8-3	远程引发互操作	166
8-3-1	将 Java 远程接口映射到 IDL	167
8-3-2	将值对象映射到 IDL	167
8-3-3	系统异常的映射	167
8-3-4	获取 stub 和客户代理	168
8-4	交易互操作	169
8-4-1	交易互操作的要求	169
8-4-2	和没实现交易互操作的容器的互操作	171
8-5	名字互操作	173
8-6	安全互操作	174
8-6-1	概述	174
8-6-2	EJB 引发的保护	175
第 9 章	IDL 文件的编写	179
9-1	用 IDL 描述应用接口	180
9-1-1	编写 IDL 代码的一般规则	180
9-1-2	基本的 IDL 词法和语法规则	180
9-2	用 IDL 定义对象属性	190
9-3	IDL 定义继承	191
9-3-1	单重继承	191
9-3-2	多重继承	193
9-4	IDL 定义异常	193
9-4-1	使用标准异常	194
9-4-2	定义用户自定义异常	194
9-5	定义上下文对象	194
第 10 章	EJB 远程通信模型	197
10-1	JAVA RMI	198
10-1-1	术语定义	198
10-1-2	分布对象应用的需要	198
10-2	RMI-IIOP	199
10-2-1	何谓 RMI-IIOP	199
10-2-2	EJB 远程通信模型	199
10-3	RMI-IIOP 编译器	200
10-3-1	rmic 编译器	200
10-3-2	idlj 编译器	201

10-4	使用 RMI 的 EJB 程序	203
10-4-1	编写源文件和 HTML	203
10-4-2	编译和部署 RMI 程序及文件	210
10-4-3	运行程序	211
10-5	将 RMI 程序转换为 RMI-IIOP 程序	213
10-5-1	将 HelloWorld 程序转换为 RMI-IIOP 程序	213
10-5-2	将客户 Applet 转换为应用	216
10-6	使用 RMI-IIOP 的 EJB 程序	217
10-6-1	编写源程序	217
10-6-2	编译实例	221
10-6-3	运行程序	222
第 11 章	EJB 与 Java 名字目录服务	223
11-1	JNDI 的概念与体系结构	224
11-1-1	名字目录服务的基本概念	224
11-1-2	JNDI 的出现	224
11-1-3	JNDI API 中的几个基本概念	225
11-2	名字服务接口	226
11-3	JNDI SPI	237
11-3-1	初始化 ORB 环境	238
11-3-2	Java 远程对象与 CORBA 对象的转换	239
11-4	JNDI 的使用	239
11-4-1	创建初始名字上下文	239
11-4-2	一个具体的使用 JNDI API 的例子	240
11-5	使用 JNDI 接口访问 RMI 对象	241
11-5-1	远程接口的定义	242
11-5-2	远程对象的实现	242
11-5-3	服务器端程序	242
11-6	使用 JNDI 接口访问 CORBA 对象	244
11-6-1	CORBA 服务器对象实现	244
11-6-2	根据 IDL 接口定义的 RMI 远程接口	246
11-6-3	客户端程序	246
11-7	JNDI 名字上下文和 EJB 运行环境	248
11-7-1	客户通过 JNDI 访问 EJB	248
11-7-2	EJB 引用的部署	250
11-7-3	资源对象引用的部署	252
11-7-4	环境变量的部署	253
第 12 章	EJB 的交易模型	255
12-1	交易处理的基本概念	256

Chapter



分布式对象技术

本章要点:

本章介绍了对象中间件的基本定义及在应用系统的重要地位，并着重介绍了当前流行的分布式对象技术：EJB、CORBA 和 DCOM。通过本章的学习，读者可以了解对象中间件的概念及基本应用，同时掌握分布式计算的三大标准。

本章主要内容:

- ☐ 对象中间件
- ☐ 分布式计算
- ☐ J2EE/EJB

1-1 对象中间件

越来越多的应用系统采用面向对象技术以增加系统的灵活性和可维护性，分布式系统也是如此。设计者往往需要一种使自己的分布式应用开发简便、灵活且面向对象化的工具，对象中间件正是符合上述需要的高级软件开发工具和平台。

1-1-1 中间件

在软件体系结构中，中间件处于操作系统软件与用户应用软件之间，即在操作系统、网络系统软件和数据库之上，应用软件之下。

中间件将应用软件所要面临的共性问题进行提炼和抽象，在操作系统之上再形成一个可复用的部分，以供成千上万的应用软件重复使用。中间件可以帮助用户灵活高效地开发和集成复杂的应用软件。

中间件=平台+通信，这是软件技术发展的一种潮流，近年来有着很好的势头。

中间件正是构件化软件的一种表现形式。中间件抽象了典型的应用模式，应用软件制造者可以基于标准的中间件进行再开发，这种操作方式其实就是软件构件化的具体实现。

中间件所带来的好处已经在很多的 IT 应用中得到了体现。BEA 是世界领先的中间件软件厂商，曾有一位 BEA 的客户从 IBM 平台迁移到 Sun 的平台上，原来开发的 200 万行 C 语言代码仅仅花了 3 个小时就重新编译完成，没有遇到任何问题。如果没有中间件，这样的平台迁移的工作量将十分巨大。

1-1-2 分布式对象中间件

简单地说，对象中间件就是用面向对象技术实现的，支持面向对象开发和集成应用的中件。其功能本质上和其他一些中间件（如消息中间件和交易中间件）类似，只不过它是面向对象的。

面向对象技术的核心是对象，即可通过属性与方法提供服务的实体。不过常规的面向对象开发和系统运行是本地的，即提供服务的对象和客户在同一个地方。而对象中间件将对象提供服务的能力延伸到网络上，客户可以借助它来访问远程的对象资源。下面将以两个实例来介绍对象中间件。

如果把商店柜台（包括售货员）看成一个对象，那么它所拥有的基本功能是：展示商品、替用户查找合适的商品、收款并将商品交给用户等。通常这些功能是在本地完成的，但在 Internet 时代，情况有所变化，用户可能更愿意在家里的计算机上通过网络购物。实际上除了拿到货物还需要商家的支持外，其他的现在都可以在网上实现了。尽管用户只感觉到自己在轻松地点击鼠标，但事实上背后有复杂的程序在支持各种网上购物的行为。对象中间件所做的就是将柜台延伸到了网页上，让用户远程感受到上面提及的柜台的存在。也许还有很多其他技术同样可以做到这一点，但对象中间件的解决思路更为简单、清晰而且灵活，因此这种应用是当今流行的趋势。

对于一个网络管理员而言，最头痛的是各种复杂的设备，其设备大到一个机站，小到每

块板卡。如果把设备看做被管理的对象,那么所提供的功能无非是查询、设置和通知等,只是内容有简有繁。如果管理员总是亲临现场进行管理,就设备的地理位置而言,也十分难于管理。同样,对象中间件可以方便地将设备的管理操作轻易地映射到管理主控台上,管理员只需要选择需要被管理的设备,就可以将剩下的许多事情交给对象中间件来处理。当然,用其他技术也可做到这一点,只是用对象中间件的解决方案和本地进行的管理最接近,无论从管理员或系统开发者的角度,本地和远程都没有很大的区别。但是,其他技术对系统开发者来说就不是这样了。

对象中间件最主要的作用是将面向对象技术平滑地延伸到网络上,并在此基础上给用户提供更其他更多的服务。

由于对象中间件广泛地被企业界所接受和应用,因此在标准化方面比较理想。目前业界有三种主流的标准,分别是:OMG(对象管理组织)的 CORBA、Microsoft 公司的 DCOM 和 Sun 公司的 EJB。实际上,这三种标准在许多方面是类似的。

由于标准的存在,对象中间件有一个非常重要的特点,即不同厂商的产品只要符合同一标准就可以实现互通。

分布式对象技术成为分布式计算环境发展的主流方向。其技术特点如下:

- 主要针对异构环境下的应用互操作问题。
- 系统核心的对象管理将客户机/服务器(Client/Server)模型与面向对象技术结合在一起。
- 提供面向对象的 API。
- 已经成为建立集成框架和软件部件标准的核心技术。

1-2 分布式计算的三大标准

1-2-1 分布式对象技术的发展

分布式对象技术是伴随网络而发展起来的一种面向对象的技术。以前的计算机系统多是单机系统,多个用户通过联机访问终端,没有网络的概念。网络出现后,产生了 Client/Server 的计算服务模式,多个客户端可以共享数据库服务器和打印服务器等。随着网络的更进一步发展,许多软件需要在不同厂家的网络产品、硬件平台和网络协议的异构环境下运行,应用的规模也从局域网发展到了广域网。在这种情况下,Client/Server 模式的局限性也就暴露出来了,于是中间件便应运而生。

中间件是位于操作系统和应用软件之间的通用服务,主要作用是屏蔽网络硬件平台的差异性和操作系统与网络协议的异构性,使应用软件能够比较平滑地运行于不同平台之上。同时,中间件在负载平衡、连接管理和调度方面都起了很大的作用,使企业级应用的性能得到大幅提升,满足了关键业务的需求。但是在这个阶段,客户端是请求服务的,服务器端是提供服务的,这之间的关系是不对称的。随着面向对象技术的进一步发展,出现了分布式对象技术。可以这么说,分布式对象技术是随着网络和面向对象技术的发展而不断地完善起来的。

分布式对象技术不同于传统的面向对象技术，传统的面向对象技术有两个基本的特点：封装性和继承性。通常其强调的是代码的复用，对象往往仅存在于一个程序中，程序的外界并不可能感知和访问这些对象。而分布式对象技术主要使用了面向对象技术的封装性，组件可以分布在网络的任何位置。对外界来说，所需关心的只是组件的接口，至于内部是如何实现的则无需考虑，远程客户通过方法调用来访问。这是分布式对象技术和传统的面向对象技术的最大区别。

目前，国际上的分布式对象技术有三大流派——CORBA、COM/DCOM 和 EJB。CORBA 技术是最早出现的，1991 年 OMG 颁布了 CORBA 1.0 标准版，揭开了分布对象计算的序幕。Microsoft 公司的 COM 系列由最初的 COM 发展成现在的 DCOM，形成了 Microsoft 一套分布对象的计算平台。Sun 公司的 Java 平台，在其最早推出的时候只提供了远程的方法调用，是属于网络计算里的一种，并不能被称为分布对象计算，接着推出的 JavaBeans，也还不足以和上述两大流派抗衡。其目前的版本是 J2EE，推出了 EJB，除了语言之外还有组件标准以及组件之间协同工作通讯的框架。

1-2-2 J2EE/EJB 技术

EJB 是由 Sun 公司提出的基于 Java 的面向对象组件标准，与原来的 JavaBeans 不同，EJB 组件包含一定的业务规则，运行在服务器端。EJB 技术使复杂的多层结构应用系统开发变得容易。

EJB 技术建立在已取得极大成功的 Java 语言之上，采用 EJB 技术能有效地进行软件复用，提高开发人员的效率，降低软件的开发和维护成本，提高软件的质量，控制所构建系统的复杂性。

EJB 是用于开发和部署多层结构的、分布式的和面向对象的 Java 应用系统的、跨平台的构件体系结构。EJB 的结构是开发和配置基于组件的分布式商务应用程序的一种组件结构。用 EJB 结构开发的应用程序是可伸缩的、事务型的、多用户安全的。采用 EJB 可以使开发商业应用系统变得容易，应用系统可以在一个支持 EJB 的环境中开发，开发完之后部署在其他的环境中，随着需求的改变，应用系统可以不加修改地迁移到其他功能更强且更复杂的服务器上。

在开发能够支持关键任务的企业级信息系统的过程中，EJB 规范代表了 Java 技术的下一个发展阶段。EJB 组件将随 JavaBeans 规范引入的 Java 组件模型扩展到服务器领域，从而使业务逻辑组件的开发可以跨企业应用程序复用，并且可以跨支持 Java 的平台移植。由于 EJB 包含了基于 RMI 技术的对象分布，所以支持跨多层的可执行组件的分立，从而允许实现最大的灵活性和高度可伸缩性。如果将常规的企业应用程序运行时服务重新定义为可指定给容器抽象的对象服务，则允许 EJB 组件的开发人员将精力集中在业务逻辑上，从而减小了通常与运行时服务相关的复杂性和平台的相关性。

EJB 和 CORBA 在技术上虽然处于竞争地位，但实际上，EJB 技术很好地补充了 CORBA。CORBA 提供了一个强大的基于标准的基础结构，并可在此结构之上构建 EJB 服务器。这使 EJB 技术在 CORBA 基础结构的顶层构建应用程序变得更为容易了。

EJB 和 CORBA 规范说明的是不同的技术，目前 EJB 的实现利用 CORBA 技术的某些方

面,如 RMI/IIOP。EJB 规范要求 EJB 组件及其容器使用 Remote Method Invocation (RMI) 技术,实现分布式对象之间的方法调用。RMI 规定远程方法的语法和语义,但并不规定应使用何种传输协议提供网络连接。CORBA IIOP (Internet 对象请求代理程序间协议)基本上定义了通过 TCP/IP 传输 CORBA 消息的一种方法。开发使用 IIOP 消息形式交换 RMI 数据的 EJB 实现,说明了 EJB 应用程序怎样才能有效地使用 CORBA 技术的各部分。这种网络也支持与 CORBA 应用程序的互操作性,后者使用 IIOP 发送本地 CORBA 消息,与 RMI 无关。IBM 的 EJB 实现,即 WebSphere Application Server 优化了 IIOP 的使用,其方法是:弄清楚分布式对象何时驻留在同一台服务器上,并且只在对象确实在远程时才调用 IIOP。

为了方便既并入 EJB 技术、又并入 CORBA 技术的企业系统的开发,Sun Microsystems 在 EJB 规范和 CORBA 之间创建了一种映射。会 EJB 体系结构映射到 CORBA 会影响到 EJB 技术的几个方面,包括对象分布、命名和事务。CORBA 映射的主要目的是,保证不同厂商构建的 EJB 服务器之间的互操作性。

互操作性提供了以下好处:CORBA 客户机可以访问部署在基于 CORBA 的 EJB 服务器上的 EJB 组件;客户机程序在事务中可以将对 CORBA 对象的调用与对企业级 Bean 的调用混合在一起;事务可以跨多个 Bean,而这些 Bean 不仅可以是来自不同厂商的基于 CORBA 的多台 EJB 服务器,也可以访问另一个厂商基于 CORBA 的 EJB 服务器上的 Bean。对于要访问 EJB 组件的 CORBA 客户机来说,Bean 接口被映射到 IDL。例如,可将股票交易 Bean 中定义的 buy () 和 sell () 方法,指定为 IDL 文件中的 CORBA 操作。非 Bean 的 CORBA 客户机(如 C++客户机)可以访问这个 Bean,并用标准 CORBA 调用来调用 Bean 的方法。如果容器使用 IIOP 作为分布式对象协议,则该容器的职责是生成与企业级 Bean 及其接口对应的 IDL。

1-2-3 CORBA 技术

CORBA 是基于对象管理体系结构(Object Management Architecture,即 OMA)的,OMA 为构建分布应用定义了非常广泛的服务。CORBA 标准主要分为 3 个层次:对象请求代理、公共对象服务和公共设施。最底层是对象请求代理 ORB,规定了分布对象的定义和语言映射,实现对象间的通讯和互操作,是分布对象系统中的“软总线”;在 ORB 之上定义了很多公共服务,可以提供诸如名字服务、事件服务、事务(交易)服务和安全服务等各种各样的服务;最上层的公共设施则定义了组件框架,提供可直接为业务对象使用的服务,规定业务对象有效协作所需的协议规则。

1991 年 10 月,OMG 推出 CORBA 1.0。该版本的主要内容包括 CORBA 对象模型、接口定义语言(Interface Definition Language 了,即 IDL)、用于动态请求管理和动态调用的 API 集合及接口仓库等。CORBA 1.1 提出了对象适配器(Object Adapter)的概念,并提供了基本对象适配器(Basic Object Adapter,即 BOA)的接口。对象适配器是 ORB 与对象实现之间的联系纽带,功能主要包括:创建 CORBA 对象和对象应用;识别客户向 CORBA 对象发送的请求;将请求分发给服务端的对象实现;CORBA 对象的激活。

1996 年 8 月,OMG 推出了 CORBA 2.0。该版本在解决互操作问题方面迈上了一个新台阶。其中一个突出的贡献是引入了 GIOP/IIOP (General Inter-ORB Protocol/Internet Inter-ORB