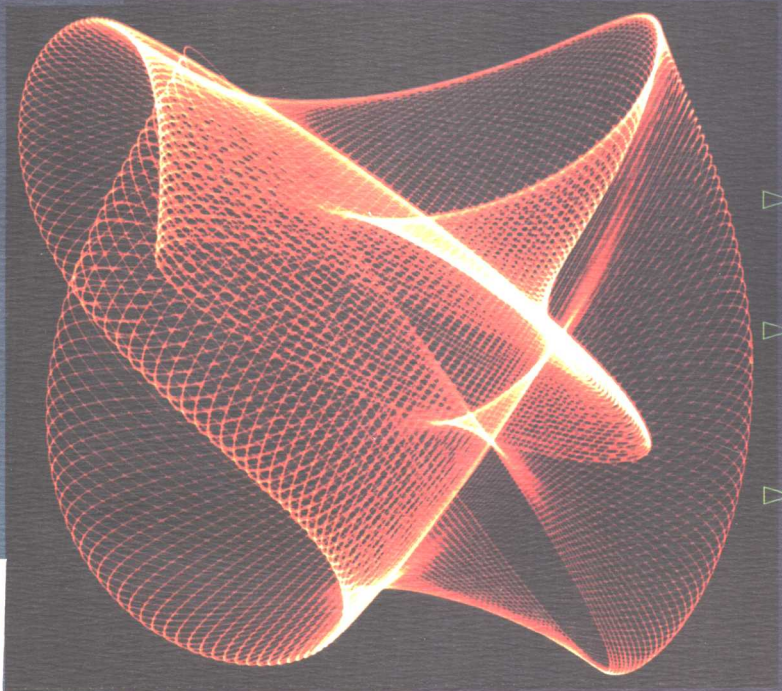
Mc Graw Hill Education

.NET & J2EE Interoperability

.NET 与 J2EE 互操作



- 成功地集成 J2EE 和.NET 技术
- 使用XML和串行化来指导高级编程
- 研究公共语言运行库和提供互操作性的其他解决方案

(美) Dwight Peltzer 著
杨 飞 黎 媛 等译

Mc Graw Hill
清华大学出版社

.NET 与 J2EE 互操作

（美）Dwight Peltzer 著

杨飞 黎媛 等译

清华大学出版社

北 京

Dwight Peltzer
.NET&J2EE Interoperability
EISBN: 0-07-223054-1

图书在版编目(CIP)数据

.NET 与 J2EE 互操作/(美)培特(Peltzer, D.)著；杨飞，黎媛等译.—北京：清华大学出版社，2004.10
书名原文：.NET & J2EE Interoperability
ISBN 7-302-09424-1
Ⅰ.N… Ⅱ.①培…②杨…③黎… Ⅲ.①计算机网络—程序设计②JAVA 语言—程序设计
Ⅳ.①TP393②TP312
中国版本图书馆 CIP 数据核字(2004)第 090622 号

出 版 者：清华大学出版社 地 址：北京清华大学学研大厦
http://www.tup.com.cn 邮 编：100084
社 总 机：010-62770175 客户服务 010 62776969

组稿编辑：曹 康
文稿编辑：王 军
封面设计：康 博
版式设计：康 博
印 装 者：清华大学印刷厂
发 行 者：新华书店总店北京发行所
开 本：185×260 印张：14.5 字数：301 千字
版 次：2004 年 10 月第 1 版 2004 年 10 月第1 次印刷
书 号：ISBN 7-302-09424-1/TP・6580
印 数：1～4000
定 价：30.00 元

前　言

在当今的数字经济时代，使用不断出现的新技术来集成传统的系统、业务应用和代码已经变得越来越重要。互操作性，即本书的主题，将主要研究 2 个平台——Microsoft .NET 和 J2EE。本书将从开发者的观点来深入介绍这两个主要平台如何相互交互。本书中的大量示例将演示.NET 与 J2EE 如何提供应用集成。例如，第 5 章介绍了公共语言运行库(CLR)，并且介绍了它如何作为类似于 IBM 的 WebSphere 的 Web 容器使用。CLR 提供了底层的服务，例如内存管理和自动垃圾回收，以及 ASP.NET、ADO.NET 和所有使用.NET Framework 的语言的安全性。

本书前 4 章主要介绍 J2EE 技术。第 5、6 和 7 章则集中介绍.NET。第 8 章引入 JNBridgePro 技术，它是提供 J2EE 与.NET 之间实际跨平台互操作性的创新业务解决方案。JNBridgePro 技术允许开发者生成相关代理，这些代理有助于两种方式的集成。Java 开发者可直接访问.NET 的函数，而.NET 的开发者则可以利用 Java 技术。JNBridgePro 业务解决方案代表了发展互操作性的重要进步。

本书指南

第 1 章“企业软件的互操作性”是本书的简单介绍，其中概述了两个主要平台之间的内部和外部集成。例如，关于 J2EE 与.NET 之间互操作性技术和它们共同支持 Web 服务的讨论就是本书的中心主题。

本章引入了一个案例分析，其目的在于重点说明 Java 或.NET 中的特殊概念。称为国际外币汇兑组织(International Finance Corporation Exchange，IFCE)的财务公司主要负责外币的买卖。该案例分析的重点在于业务进程：例如，IFCE 检查消费者的状态，以此来确定该消费者是否建立了一个账户。在最后一章中，在 Visual Studio .NET 中构建的项目将说明如何创建一个货币转换软件。例如，在对话框中输入货币数量(以英镑为单位)，这些货币可转换为欧元、马克或其他货币。

第 2 章“J2EE Servlets、Java Server Pages 和 Web 服务”中首先简要介绍了 Sun Microsystem 的 3 个开发平台，即 Java 2 Platform, Micro Edition(J2ME)、Java 2 Platform, Standard Edition(J2SE)和 Java 2 Platform Enterprise Edition(J2EE)。

本章也将讨论 3 种技术，分别是应用之间的通信、表示层和业务应用层。

本章的重点在于讨论 Java servlet、Java Server Pages 和它们在创建分布式企业

应用中的角色。本章的主题包括 servlet 的生命周期、使用 servlet 管理会话状态、Java Server Pages(JSP)、JSP 的页面指令以及如何应用这些指令将请求的结果返回给客户端。

第 3 章“Enterprise JavaBeans、接口和 JDBC 的持久性”首先简单介绍了 Enterprise JavaBeans 的开发和部署，接下来介绍了两种类型的 JavaBeans——同步和异步。然后，本章深入研究了会话 Bean 和实体 Bean。为了让读者更好地了解 Enterprise JavaBeans，讨论过 EJB 接口后，本章又详细描述了 EJB 的生命周期。

接下来，本章集中介绍了容器和它们的服务。IBM 的 WebSphere 可作为容器模型。本章解释了为何 Enterprise JavaBeans 存活在容器中以及接收容器提供的所有服务的方法。

本章从客户端的角度介绍了 Enterprise JavaBeans，并且介绍了访问由 EJB 容器提供的服务所需要的技术。

第 4 章“RMI-IIOP、JNDI 和部署描述符”首先介绍了使用远程方法调用(RMI)来调用分布在 Web Farm 或企业级网络上的对象。最初，RMI 技术用于通过导入 java.rmi 数据包来执行远程对象访问。RMI 提供分布式垃圾回收和对象激活。然而，J2EE 使用 RMI-IIOP 和 Java 命名和目录接口(JNDI)来访问远程对象。

本章的下一个主题是部署描述符在部署 J2EE 应用时所扮演的角色。实体 Bean 的开发者在部署描述符中说明容器应该如何管理 Enterprise JavaBeans。他们可以指定某个 Bean 是否是会话 Bean、会话 Bean 的类型(有状态和无状态)、或者是实体 Bean 的一种类型。开发者应该指明该 Bean 是否是容器管理的持久性(CMP)或 Bean 管理的持久性(BMP)。部署描述符还应该指明该 Bean 是否是消息驱动的 Bean。如果是，则部署描述符应该指明消息驱动 Bean 应该没有接口。本章的最后一部分描述了如何将 JAR 文件部署到容器中以进行分布。

第 5 章“.NET 语言集成组件”提供了有关公共语言运行库的详细信息。类似于 J2EE 的 Web 容器，CLR 可作为.NET Framework 的管理器并提供各种服务，包括在编译前检验数据类型的有效性、确保应用遵循公共类型规范(CTS)和通用语言规范(CLS)，以及管理安全性。本章也包含程序可执行(PE)文件的讨论，并且介绍了包含执行应用所需的所有程序二进制内容和文件引用的程序集。

第 6 章“ASP.NET 体系结构”介绍了 ASP.NET 的基础结构，并且将列举 ASP.NET 的优点和新功能作为开始。新功能包括从程序逻辑中分离 HTML 的后台编码模型、允许开发者编写事件并为其创建处理程序的事件驱动程序模型、以及通过使用视图状态来管理客户端状态的一些服务器控件。

本章介绍了 ASP.NET 的命名空间和 System.Web.UI 类，通过演示 ASP.NET 页面如何与 CLR 和 Framework 交互的示例介绍了 ASP.NET 页面的生命周期，这些示例提供了开发用户友好的 ASP.NET 应用的经验。

第 6 章也介绍了 Web 窗体，并且讨论如何将服务器端的控件以声明的方式或编

程方式添加到窗体中。随后，本章介绍了如何创建用户定义的控件以及如何将它们添加到 Web 窗体中。本章以讨论安全性和错误处理作为总结。

第 7 章“ASP.NET 与 Web 服务”详细介绍了 Web 服务与 ASP.NET 的关系。本章一开始提出以下问题：“什么是 Web 服务”，然后阐述了面向服务的体系结构。本章也探讨了 XML 以及它是如何与 Web 服务集成的。

接下来，本章介绍了客户端如何使用简单对象访问协议(SOAP)和 HTTP 来访问 Web 服务。此外，本章也解释了如何生成 Web 服务描述语言(WSDL)文件，同时详细描述了通用描述、发现和集成(UDDI)以及它在查找 Web 服务方面的作用。

第 7 章也提供有关面向对象编程的信息，并且描述了有关接口编程、支持组件重用方法的最佳实践。接下来，本章讨论了.NET 的体系结构，并且分析了如何通过一系列组件和接口来组成 Framework。本章也强调了 ASP.NET 的基础结构如何与.NET 的 CLR 进行协作，以提供灵活性并支持可扩展的 Web 服务的开发。

第 8 章“第三方供应商的互操作性解决方案”介绍了 JNBridgePro。该软件包可帮助开发者获得实际的跨平台互操作性。JNBridgePro 生成各种代理，开发者可使用这些代理从 J2EE 中访问.NET 的函数和方法，如同这些函数和方法是在 J2EE 中编写而成，反之亦然。用户友好的界面允许开发者选择所需的类和接口，然后生成有助于应用和方法互操作性的代理。

第 9 章“最佳实践、设计模式、安全性和业务解决方案”总结了本书，回顾 J2EE 和.NET 的技术，以及讨论了它们如何从内部和外部获得互操作性。本章分析了最佳实践、如何应用最佳实践以及何时应该使用最佳实践。本章也介绍了选择业务模型，如 MVC 模型，并且解释了如何在应用设计中有效地使用该模型。

此外，本章也讨论了在分布式环境中应用安全方法和管理应用状态，同时使用一些示例来演示如何实现这些任务。本章通过一个使用 Visual Basic .NET 编写的示例作为总结，以此来演示 IFCE 案例分析中的多语言互操作性。

附录 A 提供了 J2EE 的 Java 连接器体系结构(JCA)的简单概述。连接器有助于开发者访问基于 Java 的企业信息系统(EIS)和传统的数据存储。该附录指导开发者从何处获得有关连接器的信息并下载 JCA 规范。

作为总结，附录 B 提供了推荐书籍的列表，同时提供了本书中各种规范的主要来源。

本书的读者

第 1 章介绍了互操作性并且指出了每一章所涵盖的内容。对于 IT 专家、开发者和希望学习如何通过 J2EE 技术来实现内部和外部应用集成的用户，第 2 章～第 4 章将非常有用。

第 5 章～第 7 章向读者介绍了.NET Framework，并且详细解释了.NET 技术如何支持多语言的集成，这是近年来一个显著的技术进步。对于已经熟悉 J2EE 并且需要进一步了解.NET 技术的开发者，这一部分将非常有用。

第 8 章中，通过 JNBridgrPro 提供的跨平台解决方案，开发者可将 J2EE 和.NET 结合起来。对于 IT 专家、开发者和需要同时使用这两个平台的用户，学习这一章将获得很多帮助。第 9 章中重点介绍了最佳实践，并且提供了一个在 Visual Basic .NET 中创建的货币转换器解决方案，来演示如何在任意语言和任意平台上编写应用。

相关的 Web 站点和下载

本书中所有的源文件都可从 http://www.osborne.com 上下载。请按照 McGraw-Hill/Osborne 的介绍来定位有关本书的特定链接。

目　录

第Ⅰ部分　J2EE 的互操作性

第 II 部分 Microsoft .NET 的内部互操作性

第Ⅲ部分 跨平台的互操作性

第Ⅳ部分 附 录

第 I 部分

J2EE 的互操作性

- 第 1 章　企业软件的互操作性
- 第 2 章　J2EE Servlets、Java Server Pages 和 Web 服务
- 第 3 章　Enterprise JavaBeans、接口和 JDBC 持久性
- 第 4 章　RMI-IIOP、JNDI 和部署描述符

第 1 章　企业软件的互操作性

主要内容：

- 分布式应用开发的介绍
- 企业软件的互操作性
- J2EE Servlets、Java Server Pages 以及 Web 服务
- Enterprise JavaBeans、接口以及 JDBC 持久性
- RMI-IIOP、JNDI 以及部署描述符
- .NET 语言集成组件
- ASP.NET 体系结构
- ASP.NET 和 Web 服务
- 第三方供应商的互操作性解决方案
- 最佳的实践、设计模式、安全性以及业务解决方案
- Java 连接器体系结构(Java Connector Architecture，JCA)规范
- 案例分析：国际外币汇兑组织(International Finance Corporation Exchange，IFCE)

本书是实现两种平台无关的技术，即 Sun Microsystems 的 Java 2 平台企业版(J2EE 1.4)和 Microsoft 的.NET 之间集成的指南。本书的中心主题是业务伙伴间跨平台的通信和企业中关键业务数据的传递。本书说明了为什么互操作性对于当今数字化经济中获得成功是必要的，同时介绍了如何实现各种技术间的集成。Web 服务和分布式环境中的远程过程调用要求严密的安全机制，以此来保持跨越多个服务器的数据完整性。本书讨论的重点在 J2EE 和.NET 为开发人员提供的服务上。

本章将介绍 J2EE 和.NET 间通过对 Web 服务的共同支持而进行集成的技术概念，并且将总结下面章节将介绍的主要内容。很多主题都和每章的标题对应，读者可进入相关章节对某个主题作进一步的分析。例如第 2 章通过介绍 J2EE Servlets、Java Server Pages 以及 Web 服务来展开对 J2EE 的介绍。第 5 章“.NET 语言集成组件”由介绍.NET 关键组件展开。

本章还描述了在本书中使用的案例分析，该案例以现实的例子说明了支持 J2EE 和.NET 间互操作性的各种技术。

1.1　分布式应用开发的介绍

Sun 公司的 Java 2 平台企业版(J2EE 1.4)为设计多层解决方案提供了可扩展的架

构。J2EE 技术是由很多规范组成的，而不是一组可下载的技术。这些规范(或者约定)定义了 J2EE 连接器或用于 XML 的 Java API(JAX)等技术如何运行，它们适用于所有的 J2EE 技术。例如，在使用 J2EE 技术提供的某些特定服务前，业务伙伴必须遵循专门的约定或规范。

Sun 公司对构建服务器端应用程序的支持具有可靠且成熟的经验。为了征求主要机构的建议以改进所有的 Java 平台，包括 J2ME、J2SE 和 J2EE，Sun 公司率先发起了 Java 标准化组织(Java Communitey Process，JCP)，它也用于标准化关键 Java 技术。对于 Sun 公司来说，向第三方供应商提供在 Java 的未来发展方向和实现方面提出看法和建议的机会是一个明智的举动。Sun 允许人们查看相同的代码，而不是对源代码进行专利保护，从而所有的 Java 平台都可以得到持续改进。Sun 致力于为市场带来先进的技术，以及为他们的客户提供更好的服务。JCP 的出现推动了开放源代码应用程序的开发。

最近，Sun 加入了对构建基于 XML 的 Web 服务的支持。他们还在其强大的规范组中增加了 Java 连接器技术(有关 JCA 规范，请参见附录 A)。Sun 频繁地发布 Web 服务开发人员程序包(Web Services Developers Pack，WSDP)。他们认为对 Web 服务的支持是实现企业中跨平台、分布式、交互业务事务的基础。J2EE 技术提供的最重要特性之一是：只需编译应用程序一次就可以将其部署在任何平台中。

Microsoft 的.NET Framework 和 Visual Studio 的.NET 系列工具已经在 Java 和 Microsoft 阵营中产生了较大的讨论和争议。讨论的焦点围绕着平台成熟度和稳定性，以及市场占有率方面。提出最多的问题就是“Java 和 Microsoft 的.NET 是否可以共存”。Microsoft 的集成化产品系列的使用正在增加，而 Sun 加入对 Web 服务的支持也为所有开发人员提供了更有利的开发环境。所以，此时提出这个问题非常有意义。Gartner 组织最近做出预测，未来 3 年内 90%的大中型企业将在 Java 和.NET 中进行权衡。而这两种技术的综合使用，对业务和用户来说是一个双赢局面。

Microsoft 通过 Windows Server 2003、Microsoft .NET Framework、Visual Studio .NET 以及组成框架的编程语言进入企业应用领域。它们的产品系列包括 Microsoft 事务服务器(Microsoft Transaction Server，MTS)、COM+、旗舰产品 SQL Server 2000 数据库以及 Microsoft 消息队列(Microsoft Message Queue，MSMQ)。.NET 体系结构是以 Web 服务为中心的，这就为.NET 平台中使用技术的互操作性铺平了道路。Web 服务是绑定不同 Web 应用程序的粘合剂。对基于 Microsoft 的开发人员来说，主要是增加了使用针对此架构的所有编程语言构建应用程序的能力。这些语言包括 Visual Basic .NET、J# .NET、托管的 C++ .NET 以及 C# .NET。开发人员可以选择语言编写分布式应用程序，进行编译，然后部署这个应用程序。不过到目前为止，Microsoft .NET 应用程序只能运行在 Windows 平台上。希望 Microsoft 能够尽快意识到修改其服务器和系列工具的必要性，以方便用于其他的平台和操作系统。

.NET 的基础架构是基于组件的多层分布编程方式，其指定简单对象访问协议