

IBM/Lotus
技术丛书

WebSphere

环境下

Servlet与JSP高级编程

IBM红皮书

(美) Ueli Wahli 等著 瞿裕忠 张剑峰 王丛刚 等译

Servlet and JSP Programming
with IBM WebSphere Studio
and VisualAge for Java



机械工业出版社
China Machine Press

IBM

IBM/Lotus技术丛书

WebSphere环境下Servlet 与JSP高级编程

IBM 红皮书

(美) Ueli Wahli 等著

瞿裕忠 张剑峰 王丛刚 等译



机械工业出版社
China Machine Press

本书介绍IBM WebSphere Studio和VisualAge for Java环境的技术知识,以及在这个环境下的Web应用开发技术,特别是servlet与JSP编程技术。本书不仅介绍Java servlet和JSP的理论知识,还通过实例介绍了电子商务模式的实现。本书适合Web应用开发的技术人员参考。

Ueli Wahli, et al: Servlet and JSP Programming with IBM WebSphere Studio and VisualAge for Java.

本书中文版由IBM公司授权机械工业出版社独家出版。未经出版者书面许可,不得以任何方式复制或抄袭本书内容。

版权所有,侵权必究。

图书在版编目(CIP)数据

WebSphere环境下Servlet与JSP高级编程/(美)瓦利(Wahli, U.)等著;瞿裕忠等译.-北京:机械工业出版社,2001.8

(IBM/Lotus技术丛书)

书名原文:Servlet and JSP Programming with IBM WebSphere Studio and VisualAge for Java
ISBN 7-111-09042-X

I. W... II. ①瓦... ②瞿... III. JAVA语言-程序设计 IV. TP312

中国版本图书馆CIP数据核字(2001)第038308号

机械工业出版社(北京市西城区百万庄大街22号 邮政编码 100037)

责任编辑:朱裴华 吴 怡

北京昌平第二印刷厂印刷·新华书店北京发行所发行

2001年8月第1版第1次印刷

787mm×1092mm 1/16·18.75印张

印数:0 001-4 000册

定价:38.00元

凡购本书,如有倒页、脱页、缺页,由本社发行部调换

译者序

Java servlet是独立于协议和平台的服务器端软件组件。servlet是装载到Web服务器或应用服务器的Java虚拟机内运行的，这非常像applet装载到Web客户端的Java虚拟机内运行。但就功能而言，servlet更像公共网关接口（CGI）脚本或程序：servlet能响应来自HTML请求的用户事件，然后动态构建送回给用户的HTML响应。JSP (JavaServer Pages)与HTML文件类似，但是它能够在Web页面中显示动态内容。JSP技术用来把开发Web页面的动态内容从静态的HTML页面设计中分离出来。JSP有许多标记，使页面设计者能够把JavaBean对象的属性和脚本插入一个JSP文件中。JSP技术具备分隔动态部分和静态部分、独立于平台、支持组件重用、支持脚本和标记等特性。目前，基于servlet、JSP和JavaBean体系结构的Web应用受到广泛的关注。

本书给出了IBM WebSphere和VisualAge for Java环境的技术知识，并阐述了在这个环境下的Web应用开发技术，特别是servlet与JSP编程技术。全书共15章，分为两个部分。第一部分（前10章）给出有关servlet和JSP程序设计技术，并详细阐述如何在WebSphere Application Server中运行servlet和JSP，如何在VisualAge for Java中开发与测试servlet和JSP，以及如何使用WebSphere Studio进行开发和发布。第二部分（后5章）提供一个完整的应用，并展示如何在开发环境和产品环境中运行它，还讨论了电子商务模式（Patterns for e-business）在这个应用示例中的使用。书中给出应用示例的源代码和安装步骤。我们衷心期望这本书能够有益于我国Java servlet和JSP技术及其在电子商务中应用的研究和开发。

本书第1、2、3、6、9、10章由瞿裕忠翻译，第4、5章分别由陆海涛和彭晓晖翻译，第7、8章分别由王丛刚和张剑峰翻译，第11、15章由程伟华翻译，第12、13、14章由赵旭东翻译。全书由瞿裕忠统一审稿。有关Java servlet和JSP技术及其在电子商务中应用的详细资料和应用示例，请访问东南大学电子商务论坛（<http://ebi.seu.edu.cn>）。

在翻译过程中，译者参阅了大量的国内外资料。在此，谨向书中提到和参考文献列出的作者表示感谢。本书的翻译工作得到了机械工业出版社和IBM中国有限公司的支持，也得到了东南大学XObjects Group（<http://xobjects.seu.edu.cn>）全体成员的大力支持，在此，我们深表谢意！

限于译者的水平及时间，本书尚有许多不足之处，恳请读者指正。

瞿裕忠

2001年1月于东南大学

前 言

本IBM红皮书详细介绍如何有效地使用IBM WebSphere和VisualAge for Java环境，以构造、管理和部署基于servlet、JavaServer Pages (JSP) 和JavaBean体系结构的Web应用。

第一部分描述在我们的环境中用到的产品，并提供有关产品安装和配置的指导。接着，讨论servlet与JSP程序设计，传授这些技术的理论知识，使读者实际理解其组成元素，以及所述概念的示例。关于示例代码的执行，我们给出配置WebSphere Application Server以及在WebSphere中部署和运行Web应用示例的有关信息。使用这些章节所揭示的知识，本书详细介绍VisualAge for Java和WebSphere Studio所提供的开发环境，以帮助读者使用这些工具所提供的特性，如综合调试、“WebSphere测试环境”、Studio向导和Web站点资源的发布等。我们还描述如何将Rational公司的ClearCase产品集成到我们的环境以便进行软件配置管理。

第二部分描述PDK (Pattern Development Kit) 应用示例，包括安装、配置和操作，还讨论了电子商务模式 (Patterns for e-business) 在应用示例中的使用。电子商务模式为创建电子商务应用提供了设计决策信息。

本IBM红皮书适合于需要一般了解或详细了解如何在WebSphere环境中使用servlet和JSP进行软件开发的读者。本书要求读者对Java和HTML基本知识有很好的理解。

Internet上的示例代码

本红皮书中的示例代码可以从Internet上的ITSO红皮书站点下载，文件名为5755samp.zip和5755pdk.zip，URL为<ftp://www.redbooks.ibm.com/redbooks/SG245755/>，请阅读附录C“附加资料的使用”。

写作班子简介

本红皮书由来自世界各地的一个专家组创作，这些专家在“国际技术支持组织”(International Technical Support Organization, ITSO)的San Jose中心工作。

Ueli Wahli是ITSO的一位信息技术顾问。他在16年之前加入ITSO，曾在IBM Switzerland的技术支持部门工作。他主要从事IBM全球培训课程的写作与授课，涉及领域广泛，包括应用开发、对象技术、VisualAge产品、数据字典和图书馆管理。Ueli从Swiss Federal Institute of Technology (瑞士联邦技术学院)取得数学学位。他的电子邮件地址是wahli@us.ibm.com。

Mitch Fielding是一位电子商务专家，为FishTech&Partners公司工作，该公司是IBM的一个业务伙伴，位于澳大利亚的悉尼。他在软件开发和技术咨询方面有10年的经验。目前，他从事围绕WebSphere技术的Internet新产品的开发，开发环境为VisualAge for Java和DB2。他的电子邮件地址是mfieldin@fishtech.com.au。

Gareth Mackown是一位信息技术顾问，在英国Hursley电子商务服务部门工作。他为IBM工

作了将近5年，主要从事开发和咨询，偶尔也授课。他的专长主要在Java，包括VisualAge for Java、WebSphere和对象技术。Gareth从Durham大学取得数学与计算机科学联合荣誉学位。他的电子邮件地址是gareth_mackown@uk.ibm.com。

Deborah Shaddon是一位来自美国芝加哥IBM全球应用发送部门的信息技术资深专家。她在软件开发和体系结构方面有12年的经验，主要在银行和金融行业。目前，她的专长包括为IBM客户开发电子商务解决方案，使用包括WebSphere、VisualAge for Java和Lotus Domino在内的各种技术。Deborah从Illinois的Bradley大学取得商务信息系统的学位，目前正在攻读芝加哥DePaul大学软件工程专业的硕士学位。她的电子邮件地址是dmsnadd@us.ibm.com。

Gert Hekkenberg是一位来自荷兰阿姆斯特丹IBM软件部EMEA北方区的信息技术资深专家。他有超过15年的应用技术经验，特别在软件配置管理领域。目前，他是一位技术销售顾问，为大宗客户设计E2E应用开发解决方案。他多年来编写了多本有关应用开发和供应链管理方面的红皮书，还参与开发了各种ITSO 研讨课。Gert从荷兰Amsterdam的Vrije大学取得经济学学士学位，从荷兰Rotterdam的Erasmus大学取得商务信息系统的硕士学位。他的电子邮件地址是hekkenberg@nl.ibm.com。

下列人员为本书作出了很大的贡献，特表谢意！ITSO San Jose中心的Pat McCarthy。Joaquin Picon和Markus Muetschard对本书在应用开发和出版等各方面给予了自始至终的支持。IBM Toronto的Sheldon Wosnick在servlet开发技术和VisualAge for Java 的“WebSphere测试环境”配置方面给予了帮助。IBM UK的Jonathan Adams领导了电子商务模式的开发。来自IBM Hursley的一个小组开发了本书第二部分描述的PDK应用示例，他们是Anthony Griffin，Rob Veck，Joe Parman，Dave Mulley，Mark Campbell和Robert James。来自美国IBM Raleigh的Chris Gerken提供了有用的JSP。另外，IBM WebSphere Application Server，WebSphere Studio和VisualAge for Java开发组也为本书提供了帮助，在此一并致谢！

目 录

译者序

前言

第一部分 Web应用开发技术

第1章 环境概述	1
1.1 WebSphere执行环境	1
1.2 VisualAge for Java开发环境	2
1.3 WebSphere Studio开发环境	2
1.4 VisualAge for Java与WebSphere Studio 的相互作用	3
1.5 完整的环境	4
第2章 产品概述	6
2.1 产品如何一起工作	6
2.2 IBM HTTP Server	6
2.3 WebSphere Application Server	6
2.4 WebSphere Studio	7
2.5 VisualAge for Java	8
2.6 Distributed Debugger	9
2.7 DB2 UDB	9
2.8 SecureWay Directory	9
第3章 产品安装	10
3.1 起始环境	10
3.2 创建一个专门的用户ID	10
3.3 JDK	10
3.4 IBM HTTP Server	11
3.4.1 产品的安装	11
3.4.2 安装的测试	11
3.5 DB2 UDB	12
3.5.1 产品的安装	12
3.5.2 安装的测试	13
3.6 VisualAge for Java	13
3.6.1 产品的安装	14

3.6.2 安装的测试	15
3.6.3 存在的错误	15
3.7 Distributed Debugger	15
3.8 WebSphere Application Server	15
3.8.1 产品的安装	15
3.8.2 安装的测试	17
3.9 WebSphere Studio	19
3.9.1 产品的安装	20
3.9.2 安装的测试	20
3.10 SecureWay Directory	21
3.10.1 与DB2 UDB的不兼容性	21
3.10.2 产品的安装	21
3.10.3 配置SecureWay Directory	23
3.10.4 安装的测试	24
3.11 完成的安装	25
第4章 Servlet技术	26
4.1 Java servlet技术的概况	26
4.1.1 servlet的处理流程	26
4.1.2 Java Servlet API	27
4.1.3 servlet 的生命周期	27
4.2 基本的servlet示例	30
4.2.1 简单的HTTP servlet	30
4.2.2 产生HTML表单的servlet	32
4.2.3 处理HTML表单的servlet	34
4.2.4 简单计数servlet	37
4.2.5 servlet初始化参数	38
4.2.6 HTTP请求处理工具servlet	40
4.3 其他的servlet 示例	40
4.3.1 Cookie servlet	40
4.3.2 URL重写servlet	42
4.3.3 一个真正持久的servlet	43
4.3.4 用户会话	45

4.3.5 用户会话计数器servlet	45	6.2 WAS管理	83
4.3.6 JDBC servlet	46	6.3 WAS拓扑结构	84
4.3.7 用于SHTML的servlet标记	48	6.4 内部servlet	88
4.4 servlet交互技术	49	6.5 创建自己的Web应用	89
4.4.1 servlet合作: 过滤和链	50	6.6 类装入与重新装入	93
4.4.2 从servlet调用servlet	53	6.7 在WAS中使用JNI	94
4.4.3 响应重定位	53	6.8 设置连接池	95
4.4.4 请求分派	54	6.9 在WAS中使用JSP	97
4.4.5 使用资源	56	6.10 安全	98
4.4.6 在作用范围内的对象共享	58	6.10.1 WAS中的安全机制	98
4.5 Java Servlet API 2.2的新特征	62	6.10.2 配置企业应用	99
4.6 总结	63	6.10.3 WAS的安全设置	101
第5章 JSP技术	64	6.11 XML配置接口	107
5.1 概况	64	6.11.1 移出配置数据	107
5.2 JSP的工作原理	65	6.11.2 移入配置数据	108
5.3 JSP的组成元素	65	6.11.3 示例	108
5.3.1 HTML标记	66	6.12 用户简要表	109
5.3.2 JSP指令	66	6.13 故障排除	109
5.3.3 声明	67	6.13.1 WAS内的跟踪支持	109
5.3.4 Scriptlet	67	6.13.2 资源监控	110
5.3.5 注释	68	6.13.3 参考资料	110
5.3.6 表达式	68	第7章 使用VisualAge for Java进行	
5.3.7 WebSphere对JSP scripting的扩展	68	开发与测试	111
5.3.8 访问隐含对象	69	7.1 VisualAge for Java 概述	111
5.3.9 元素集成	69	7.2 使用VisualAge for Java进行	
5.4 JSP的交互作用	71	应用程序开发	112
5.4.1 从URL调用JSP	71	7.2.1 快速应用程序开发 (RAD)	112
5.4.2 从JSP调用Servlet	71	7.2.2 创建工业标准的Java应用程序	113
5.4.3 从Servlet调用JSP	72	7.2.3 维护程序的多个版本	113
5.4.4 从JSP调用JSP	74	7.2.4 VisualAge for Java组成部件	113
5.5 在JSP中创建动态内容	74	7.2.5 在VisualAge for Java中浏览	116
5.5.1 标准JSP标记	74	7.2.6 VisualAge for Java的其他概念	120
5.5.2 WebSphere特有的标记	77	7.3 Servlet开发	122
5.6 JSP应用示例	81	7.3.1 快速servlet开发	123
5.7 JSP 91版与JSP 1.0版的区别	81	7.3.2 开发过程	123
第6章 用servlet与JSP进行Web应用设计	82	7.3.3 开发我们的第一个servlet	124
6.1 WAS概貌	82	7.4 WebSphere测试环境	126

7.4.1 配置VisualAge for Java来 模拟WebSphere	127	8.3 创建一个项目	152
7.4.2 “WebSphere测试环境”配置	127	8.4 设置JSP版本	154
7.4.3 启动WebSphere测试环境	127	8.4.1 设置文件夹	154
7.5 在WebSphere测试环境下测试JSP	130	8.4.2 在项目里增加文件	155
7.5.1 配置VisualAge for Java来支持JSP	130	8.4.3 设置文件状态	157
7.5.2 配置VisualAge for Java 使用的JSP版本	130	8.5 编辑项目资源	157
7.5.3 运行我们的第一个JSP	131	8.5.1 文件的检出和检入	158
7.5.4 创建并运行一个JSP	132	8.5.2 调用Page Designer	158
7.6 调试servlet和JSP	133	8.5.3 使用表单和输入域	159
7.6.1 调试基础	133	8.5.4 调用一个servlet	160
7.6.2 调试servlet	134	8.5.5 预览表单, 显示HTML源代码	160
7.6.3 JSP执行监视器	136	8.5.6 把一个JavaBean插进JSP	162
7.6.4 调试由JSP生成的源代码	138	8.5.7 修改JavaBean和servlet	163
7.7 WebSphere测试环境——高级配置	138	8.5.8 编译源文件	164
7.7.1 资源的类型	138	8.6 发布阶段和发布目标	164
7.7.2 附加的servlet示例	138	8.6.1 设置测试阶段	165
7.7.3 资源位置	138	8.6.2 设置产品阶段	166
7.7.4 四个关键的配置文件	139	8.6.3 发布到一个Web应用	167
7.7.5 用于servlet链、过滤器和SHTML 的配置	141	8.7 项目关系和完整性	169
7.8 开发和测试附加的servlet和JSP配置	141	8.8 发布一个项目	170
7.9 WebSphere测试环境: 多Web应用程序	142	8.9 测试发布的文件	170
7.9.1 配置多Web应用程序	142	8.10 WebSphere Studio向导	170
7.9.2 配置ServletEngine类	144	8.11 SQL向导	171
7.9.3 载入ServletEngine	145	8.11.1 运行SQL向导	171
7.9.4 使用ServletEngineConfigDumper servlet	146	8.11.2 修改SQL语句	176
7.9.5 恢复SERunner	147	8.12 数据库向导	176
7.10 配置并测试servlet和JSP交互	147	8.12.1 运行数据库向导	176
7.11 对JavaBean的支持	148	8.12.2 数据库向导生成的代码	180
7.12 小组开发	148	8.12.3 运行生成的应用	181
7.13 资源管理	149	8.12.4 改进应用	182
第8章 使用WebSphere Studio进行开发	151	8.13 JavaBean向导	184
8.1 WebSphere Studio概观	151	8.13.1 运行JavaBean向导	184
8.2 WebSphere Studio 集成开发环境	152	8.13.2 测试JavaBean向导的代码	187
		8.13.3 JavaBean向导——用来干什么的?	187
		8.14 使用WebSphere Studio开发一个应用	187
		8.14.1 为一个部门的雇员创建SQL语句	188
		8.14.2 为雇员照片创建SQL语句	189

8.14.3 为部门雇员生成代码	189	9.5.4 创建项目	210
8.14.4 为雇员照片生成代码	189	9.5.5 创建视图	213
8.14.5 改变生成的数据源	190	9.5.6 在WebSphere Studio环境中 使用ClearCase	214
8.14.6 改正问题	190	9.5.7 将项目元素带入ClearCase环境	215
8.14.7 在VisualAge for Java里测试	190	9.5.8 使用WebSphere Studio	217
8.14.8 显示一张照片	191	9.5.9 关于SCM过程的思考	218
8.14.9 链接servlet	191	9.5.10 在其他scm环境下使用WebSphere Studio和ClearCase	219
8.14.10 运行应用	193	9.6 Rational Rose	219
8.15 问题	193	9.7 小结	220
8.15.1 解决解析的问题	194	第10章 用servlet与JSP进行Web 应用设计	221
8.15.2 Web应用在发布阶段时的文件夹	194	10.1 Web应用的结构	221
8.15.3 SQL向导为BLOB列生成错误 的数据类型	194	10.1.1 HTML页面	221
8.15.4 数据库向导JSP代码在METADATA 标记里编译	194	10.1.2 Servlet	221
8.15.5 JavaBean向导生成错误代码	195	10.1.3 Command bean	222
8.16 和VisualAge for Java的接口	195	10.1.4 Data bean	222
8.16.1 设置环境	195	10.1.5 View bean	222
8.16.2 WebSphere Studio	196	10.1.6 JSP	222
8.16.3 从VisualAge for Java里接收更新	196	10.2 MVC设计模式	222
8.16.4 把更新送往VisualAge for Java	196	10.3 详细信息	222
8.16.5 把VisualAge for Java作为 编辑器使用	196	第二部分 PDK: 一个应用示例	
8.17 存档	197	第11章 PDK概述	223
8.18 在小组里工作	197	11.1 背景	223
8.19 更多的信息及例子	197	11.2 应用描述	223
第9章 软件配置管理	198	11.3 应用预演	223
9.1 引言	198	11.3.1 欢迎页面	224
9.2 针对基于体系模式开发的SCM	199	11.3.2 主页	224
9.3 在WebSphere Studio环境中实现SCM	202	11.3.3 拓扑结构1——历史数据	225
9.3.1 SCM考虑的因素	202	11.3.4 拓扑结构2——访问行星	227
9.3.2 选择ClearCase作为单个控制点	204	第12章 使用电子商务模式来构建PDK	229
9.4 Rational的SCM工具集	204	12.1 使用电子商务模式的好处	229
9.5 WebSphere Studio环境中的ClearCase	206	12.2 应用电子商务模式	229
9.5.1 安装	206	12.2.1 选择一种商业模式	229
9.5.2 WebSphere Studio与ClearCase	208	12.2.2 选择一种相关的逻辑模式	230
9.5.3 建立一个ClearCase项目	208		

12.2.3 选择一种相关的物理模式	231
12.2.4 其余步骤	232
12.3 设计使用的技术	232
12.3.1 模型 - 视图 - 控制台框架	232
12.3.2 命令bean设计模式	233
12.4 PDK的设计	234
12.4.1 拓扑1	234
12.4.2 拓扑2	235
12.4.3 总结	238
第13章 在WebSphere中运行PDK	239
13.1 解压缩资源文件	239
13.2 定制安装系统	239
13.2.1 用户ID	239
13.2.2 设置环境参数	239
13.2.3 定制XML文件	240
13.3 安装并运行PDK	241
13.3.1 重启HTTP服务器	241
13.3.2 创建自签名SSL证书	241
13.3.3 建立Web站点	242
13.3.4 定制IBM HTTP Server	242
13.3.5 重启IBM HTTP Server	243
13.3.6 快速测试HTTP Server配置信息	243
13.3.7 LDAP配置	243
13.3.8 构建ITSOTOPO数据库	244
13.3.9 拷贝为应用而定制的文件	244
13.3.10 在WebSphere中导入XML 配置信息	244
13.3.11 运行PDK应用程序	245
13.4 复位改动	245

第14章	在VisualAge for Java中 运行PDK	246
14.1	自动配置	246
14.1.1	运行配置脚本	246
14.1.2	准备项目并导入Java代码	247
14.2	servlet引擎配置	247
14.3	手工配置	247
14.4	运行应用程序	249
14.5	复位改动	250
第15章	使用WebSphere Studio开发PDK	251
15.1	概述	251
15.2	构建WebSphere Studio项目	251
15.2.1	创建WebSphere Studio项目	251
15.2.2	定义发布阶段	252
15.3	与VisualAge for Java连接	254
15.4	管理Studio中的项目	256
15.5	发布文件	258
15.5.1	发布到WebSphere应用服务器	258
15.5.2	发布报告	259
15.5.3	发布到VisualAge for Java	259
15.6	编辑文件	260

第三部分 附 录

附录A	JSP标记语法	261
附录B	实用servlet和实用JSP	265
附录C	附加资料的使用	274
附录D	特别注意事项	281
附录E	有关出版物	283
附录F	术语表	286

第一部分 Web应用开发技术

在这个部分中，我们描述servlet和JSP程序设计。然后，详细阐述如何在WebSphere Application Server中运行servlet和JSP，如何在VisualAge for Java中开发并测试servlet和JSP，如何使用WebSphere Studio进行开发和发布。我们并不描述EJB（Enterprise JavaBeans）。有关开发与测试EJB的详细信息，请参考IBM红皮书“Servlet/JSP/EJB Design and Implementation Guide, SG24-5754”。

第1章 环境概述

本章给出书中涉及到的有关环境的描述与示意性概述。主要四个环境如下：

- WebSphere执行环境。
- VisualAge for Java开发环境。
- WebSphere Studio开发环境。
- VisualAge for Java与WebSphere Studio的相互作用。

本章中给出的图示是一些高层次框图，旨在阐明各种环境中的重要组件及其交互关系。

1.1 WebSphere执行环境

本书中使用的执行环境及其相关代码如图1-1所示。这个环境中的基本组件如下：

- WebSphere Application Server
- IBM HTTP Server

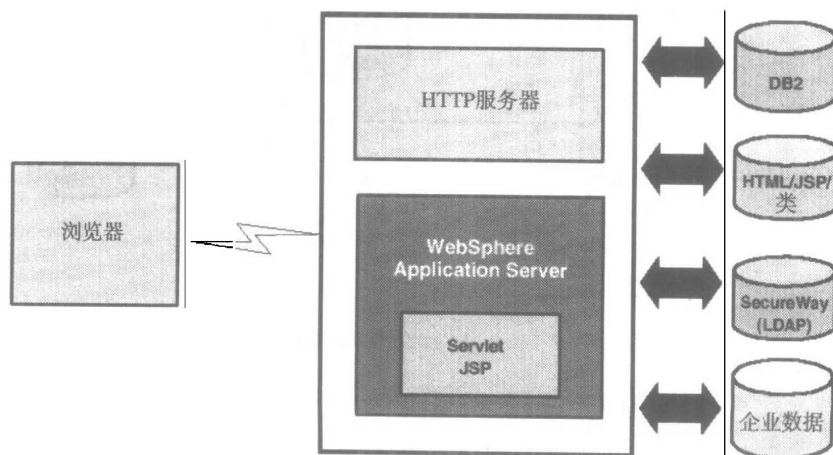


图1-1 WebSphere执行环境

- DB2

这个执行环境中显示的其他一些组件包括：

- 类文件和HTML/JSP文件。贯穿全书有很多使用在应用服务器和Web服务器中的servlet、JavaBean和HTML/JSP页面示例。
- IBM SecureWay Directory。IBM SecureWay Directory为PDK (Patterns Development Kit) 示例提供了LDAP用户鉴别。Patterns Development Kit将在本书的第二部分中给出。
- 企业数据。本书并不讨论连接企业数据的连接子，但是，我们在执行环境中勾画了这种连接子以表明对各种企业数据资源的连接支持。PDK示例包含对CICS和MQSeries的连接，但并不在本书中讨论。

1.2 VisualAge for Java开发环境

VisualAge for Java为整个开发生命周期提供丰富的功能，包括Java 代码的编辑与调试工具、JSP调试工具、WebSphere测试环境。这个开发环境如图1-2所示。

VisualAge for Java还包括了一个存储项目代码和编译代码的储存库，以及一个移入/移出设施来支持与文件系统的交互。

“WebSphere测试环境”是VisualAge for Java的重要特性之一。这一特性提供了开发机上的应用与Web服务器环境，从而支持本地测试与调试一个Web资源。这个环境提供了一个完整应用服务器的许多功能，包括对服务的访问，如LDAP与企业资源。

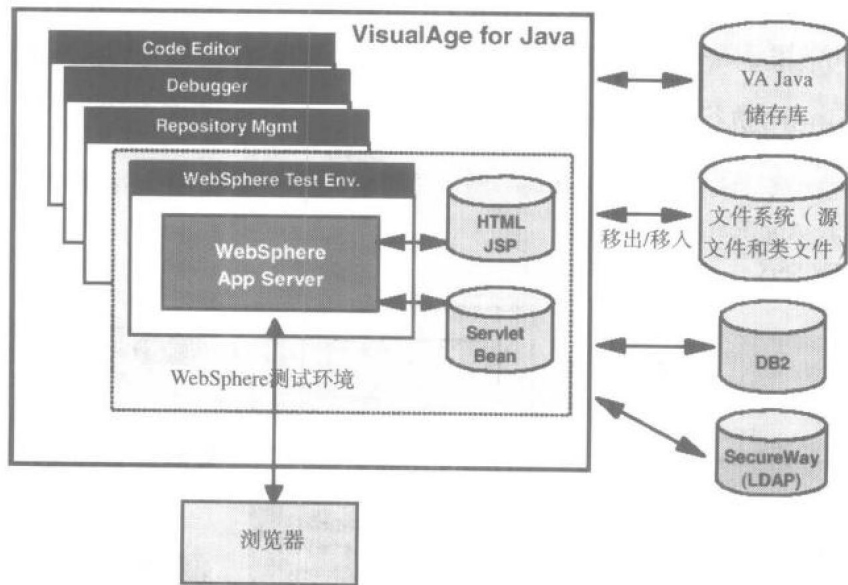


图1-2 VisualAge for Java开发环境

1.3 WebSphere Studio开发环境

WebSphere Studio用来为一个Web站点开发、管理和部署资源。图1-3显示了WebSphere

Studio的基本特性和交互作用。

WebSphere Studio维护文件系统中的项目文件，并提供小组开发和版本控制的支持。WebSphere Studio的部署特性可以用来设定项目以部署到几个不同地方，如WebSphere Application Server或者VisualAge for Java的“WebSphere测试环境”。

WebSphere Studio还包括了多个向导来指引开发者完成有关任务，如生成SQL语句、创建与数据库和JavaBean交互的Web页面。开发者能够使用WebSphere Studio中的“Page Designer”工具编辑这些生成的页面或者创建自己的HTML和JSP页面。

WebSphere Studio中的任何Java代码都能够使用该工具提供的Java编译器进行编译。

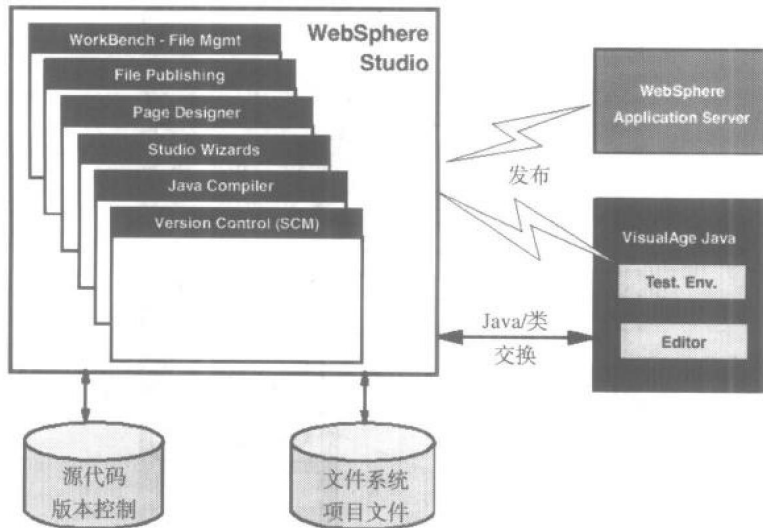


图1-3 WebSphere Studio环境

1.4 VisualAge for Java与WebSphere Studio的相互作用

WebSphere Studio的一个特别重要的特性是具备与VisualAge for Java交互的能力。该特性是

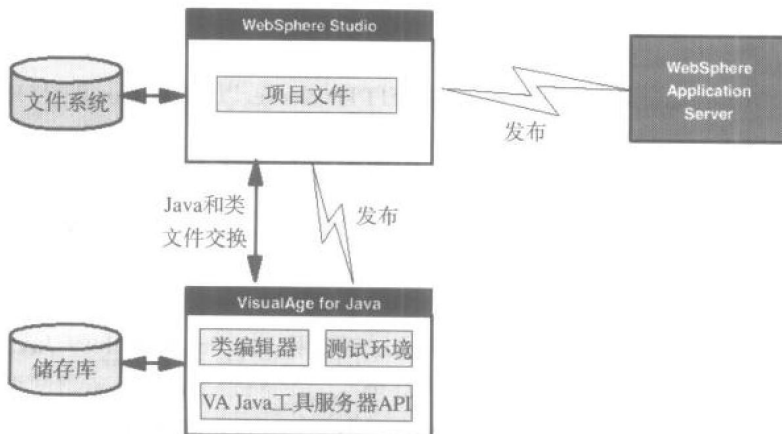


图1-4 WebSphere Studio与VisualAge for Java的相互作用

具有一个储存库的小组开发环境中，这个储存库用来集中存储多个开发者的代码。

- WebSphere Studio提供了一个HTML文件和JSP的开发环境。它还提供了有关向导来生成Web基本页面、servlet和JSP，以存取数据库和JavaBean。

工具“Page Designer”用来编辑静态HTML页面和包含动态内容的JSP页面。发布设施能够将HTML文件、JSP和servlet代码放置到合适的目录以便在应用服务器中运行或者在VisualAge for Java中测试。WebSphere Studio还提供了与VisualAge for Java的直接交互，以便将Java文件和类文件存储到VisualAge for Java储存库或者从VisualAge for Java储存库中提取Java文件和类文件。

- Web应用能够通过连接子访问企业资源，如DB2、CICS、MQSeries、IMS、SAP以及其他资源。

第2章 产品概述

本章给出书中用到的有关产品的简单概述。本章第一节介绍如何在不同方面使用各种不同的产品，然后分节对各个产品给出一个概述。关于产品的安装指导，请参阅第3章“产品安装”。

2.1 产品如何一起工作

图2-1描绘了各种产品在一个完整的开发环境中如何一起使用的概貌，这些产品将在本书中使用。有关主要产品的更详细信息将在后续章节中讲解。

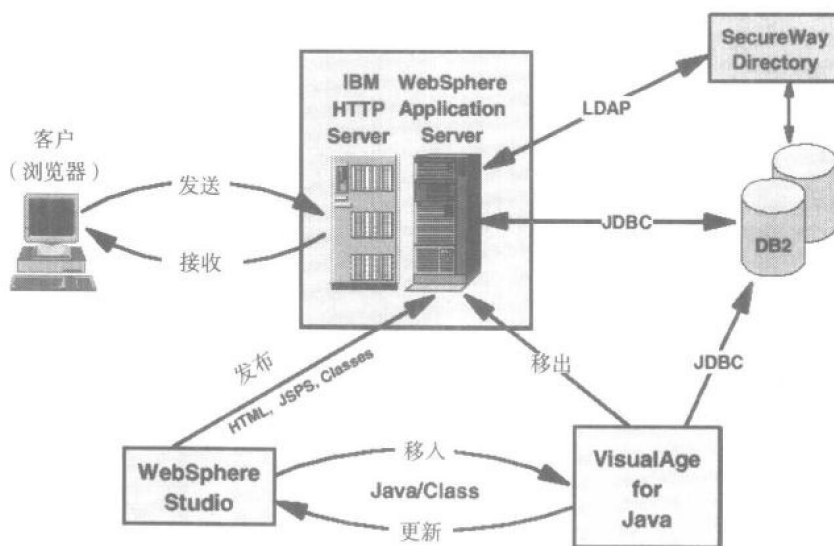


图2-1 开发环境中的产品

2.2 IBM HTTP Server

IBM HTTP Server是建立在Apache HTTP Server之上，并且可以在AIX、Solaris、Windows NT和Linux上运行。IBM增强了这个基于Apache的HTTP服务器，比如，IBM增加了SSL以支持安全的事务处理。有关IBM HTTP Server的更详细信息，请参阅产品文档或访问Web站点：

<http://www.ibm.com/software/webservers/httpservers/>

2.3 WebSphere Application Server

WebSphere Application Server (WAS)用来扩展一个标准Web服务器的功能。WAS为电子商务应用提供了Web事物处理及其与一个坚固的部署环境的交互能力。它提供了可移植的、基于Java的Web应用程序运行平台，着重于支持和执行Java Servlet、JavaBean和JSP文件。