

J2EE 应用服务器的企业级应用体系结构研究与实例评测

唐贤伦^{1,2} 仇国庆² 曹长修¹ 郎风华³ 李学勤²

(重庆大学自动化学院 重庆400044)¹ (重庆邮电学院自动化学院 重庆 400065)²

(重庆邮电学院计算机学院 重庆400065)³

摘要 应用服务器向企业级用户提供了设计、开发、部署、运行和管理一个 Internet 企业应用系统的平台。本文对 J2EE 应用服务器在企业级应用中的体系结构进行了研究,并给出测试用例,对典型应用服务器的功能和性能等方面进行了现场实例综合评测,对评测结果进行了分析和说明。

关键词 应用服务器, J2EE, 体系结构, 实例评测

Architecture Research and Instance Evaluating of J2EE Application Server in Enterprise Application

TANG Xian-Lun^{1,2} QIU Guo-Qing² CAO Chang-Xiu¹ LANG Feng-Hua³ LI Xue-Qin²

(College of Automation, Chongqing University, Chongqing 400044)¹

(College of Automation, CQUPT, Chongqing 400065)²

(College of Computer, CQUPT, Chongqing 400065)³

Abstract Application Server provides a platform that helps users to design, develop, deploy, execute and manage an Internet enterprise application system. We in this paper research the architecture of J2EE application server in enterprise application. A comparative study among some leading application servers is given by evaluating their functions and performance.

Keywords Application server, J2EE, Architecture, Evaluating

1 引言

应用服务器是在当今 Internet 上企业级应用迅速发展的需求下产生的一种新技术,通过它能将一个企业的商务活动安全、有效地实施到 Internet 上,实现企业应用。应用服务器可使一个商业系统得以快速简便的开发和部署,也可以适应商业系统的用户增加而无需重构系统,这一切都是因为它处于一个相对独立的结构层。在实际应用中,一个企业系统可以由多个应用服务器、多个 Web 服务器和多个数据库服务器组成,应用程序代码可以分布在这多个应用服务器上。

应用服务器除了可以满足可扩展性的需求,还可提供很多其它方面的益处。目前基于 Web 的开发已经迅速从数据发布、少维护量的应用转向完整、复杂的商业应用。应用服务器可以简化新的商业交互程序的开发;随着应用系统规模的不断扩大,逻辑重用和团队开发变得越来越重要。此外,由于应用程序需要获得组织内更多的数据资源,对数据访问的集中管理和集中安全也越来越重要。应用服务器通过对象和组件技术为此提供了很好的解决方案。

2 应用服务器的多层企业应用体系结构

应用服务器是企业级 Web 应用的必经之路,新的企业级应用需要关键性应用能对不断增加的用户数量保持持续可用,此外企业应用还需要高的安全性和可靠性,无论访问系统的用户数量发生变化,还是系统数据资源发生变化,应用服务器都应始终保持运行。应用服务器的强大功能能够为企业级应用系统的开发和部署提供有力的保证。

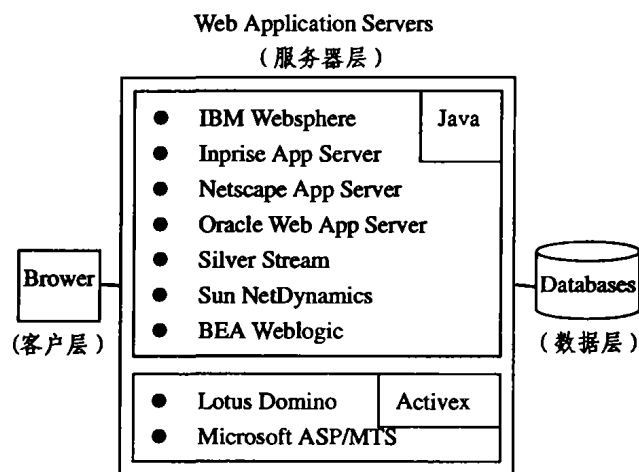


图1 三层企业应用体系结构

应用服务器最常用的场合是基于 Web 的具有三层结构的应用。第一层为客户层(client tier),该层是用户接口和用户请求的发出地,为用户提供 GUI 图形接口,典型应用是网络浏览器和胖客户(如 Java 程序);第二层为中间层(server tier),典型应用是 Web 服务器和运行业务代码的应用服务器;第三层为数据层(data tier),典型应用是关系型数据库和其他后端(back-end)数据资源,如 Oracle 和 SAP、R/3 等。在企业级应用中,应用服务器是位于企业数据(以及其他企业遗留系统)和访问企业数据的客户之间的中间件,它提供了业务代码的存放和运行的环境。它从物理上把业务逻辑(Business logic)同客户端和数据资源分离开来。业务逻辑层处于中间

层,不需要关心由哪种类型的客户来显示数据,也可以与后端系统保持相对独立性,有利于系统扩展。三层结构具有更好的移植性,可以跨不同类型的平台工作,允许用户请求在多个服务器间进行负载均衡。三层结构中安全性也更易于实现,因为应用程序已经同客户隔离。如图1所示,应用服务器运行于浏览器和数据资源之间,一个简单的实例是:客户从浏览器中输入一个资源查询和更新请求,Web 服务器将该请求发送给应

用服务器,由应用服务器执行处理逻辑,并且获取或更新后端用户数据。

3 典型应用服务器的实例综合测评

各应用服务器产品凡是通过了 J2EE 认证都需要先通过 SUN 公司的严格测试,各产品的各项重要指标比较如表1。

表1 几种典型应用服务器的重要指标比较

开发商	SILVERSTREAM	BORLAND	IBM	BEA
产品	EXtend Application Server	AppServer	WebSphere	WebLogic
运行平台	SUN/IBM/HP 等	SUN/IBM/HP 等	SUN/IBM/HP 等	SUN/IBM/HP 等
负载均衡	√	√	√	√
容错性	√	√	√	√
可移植性	在采用 IIOP 协议的平台间可行	在采用 IIOP 协议的平台间可行	在采用 IIOP 协议的平台间可行	将客户端程序中的 T3 协议头改为 RMI 或 IIOP 即可
可管理性	SERVER CONSOLE	SUN 平台上无法启动	√	√
	RICH CLIENT CONSOLE	√	√	√
	Command line	√	√	√
	Application Management	热部署	热部署	热部署
开发和支持工具	JAVA	JAVA,C,C++	JAVA,C,C++	JAVA,C,C++
XML	√	√	√	√
Web Service	√	√	√	√

对已获得 J2EE 认证的产品本文忽略对 J2EE 规范中各具体指标的测试,而重点关注其可移植性、多平台支持能力、在生产环境中使用的友好性等抽象指标。本文共选取了 SILVERSTREAM、BORLAND、IBM、BEA 进行综合测试,对各产品(特别是 IBM 公司的 Websphere)的功能和性能等方面进行了现场实例评测并比较。

3.1 Silver Stream

现场演示了 Hello World 范例,对 ORACLE 数据库中的大表进行了联合查询,同时演示了使用其开发工具 Work Bench 进行开发的全过程,最后在 WIN2000 和 SUN 平台上安装了应用服务器试用版。测试结果表明:

1. 由于 SilverStream 的 Work Bench 与 Application Server 的开发主力源于 Power Builder 的开发队伍,因此操作界面与 Power Builder 较类似,简洁、直观,开发、调试和发布应用很方便;

2. SUN 平台上安装应用服务器后,服务器端的管理控制台(CONSOLE)无法启动,只能通过客户端的 CONSOLE 进行管理;

3. Work Bench 与 Application Server 只用于基于 Java 技术的开发,无法开发 CORBA 应用,但对 CORBA 应用提供 I-IOP 接口。

3.2 Borland AppServer

Borland AppServer 5.0 包含了 EJB 容器与其服务的自动丛集式机制,让软件开发人员全心致力于建构具有商务逻辑的 EJB,由位于底层的 VisiBroker 的 CORBA 引擎所组成的基础架构提供广为业界所认可的 IIOP 协定通讯机制以及诸如 Portable Object Adapter (POA)、Objects-by-value (OBV)与 RMI-over-IIOP 等最新 CORBA 标准。

现场采用 Jbuilder 6.0 进行了 JSP、Servlet、EJB、CORBA、EJB 调用 CORBA、CORBA 调用 EJB 范例的开发和发布;演示了负载均衡和容错性能。最后在 SUN 平台上建立了试运行环境。

测试过程中运行了大量实例,未出现任何例外现象。但在后续的试开发过程中,出现了一些比较严重的问题:1. bean 较多时 server 解析会出现明显问题;2. bean 太多导致配置文件太大时,finder 方法会失效。除此之外还有一些较小的 bug。这些问题对开发进度影响较大。

3.3 IBM WebSphere

现场结合 IBM 的开发工具 WSAD 演示了范例程序和数据库操作,并且专门搭建了异构平台的 cluster 环境:用两台 PC 机和一台 SUN 主机作为集群应用服务器,另外一台 PC 机作为 Web 服务器,另外一台 SUN 主机作为数据库服务器,然后用专门的压力测试工具进行了压力测试。

3.3.1 测试环境

1. 硬件环境

如图2, WAS 节点连接到后台 ORACLE 数据库上,该 ORACLE 数据库运行在一台配置较低 SUN 服务器上。节点 3 的环境说明,机型:Ultra-60;OS:Solaris 2.8;内存:1G;芯片:2 X UltraSPARC-II 450MHz。Oracle 数据库服务器环境说明,机型:Ultra-60;OS:Solaris 2.6;内存:512M;芯片:1 X UltraSPARC-II 360MHz;DB:Oracle 8.1.7。

2. 软件环境

操作系统均采用 Windows 平台,在硬件拓扑的对应机器上分别安装 IBM WebSphere Application Server、IBM Http Server 及 IBM WebSphere 自带的 HTTP Plugin,测试软件采用通用的微软 Web Application Stress Tool 压力测试工具。

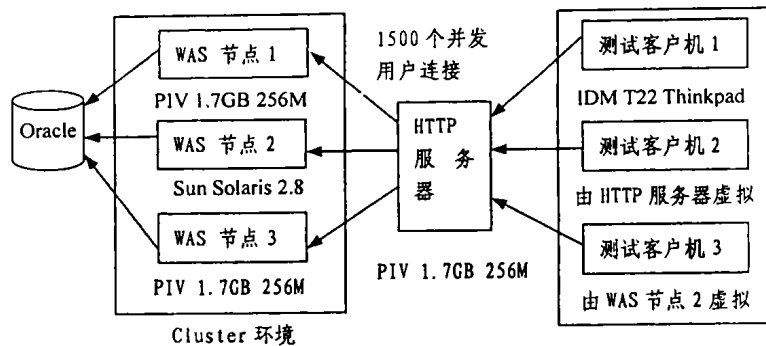


图2 测试硬件环境

3.3.2 测试用例 测试使用的例子是通过一个 Servlet 去访问 Session EJB 来完成数据库的访问。Servlet 每调用一次 EJB, 完成对 Oracle 数据库的一个表进行插入、选择和更新的操作, 可将其当作一个 Transaction。

数据库表 TestDB 的结构:

表2 数据库表 TestDB 的结构

字段名	字段类型
Name	VARCHAR
Account	NUMBER
Amout	NUMBER
ID	VARCHAR

通过测试及分析表明:

1. 可以在 Windows 2000 主域控制器上安装 WAS, 需要建立有本地存取权限的用户;
2. 可以搭建异构平台之间的 Cluster 环境, 实现负载均衡;
3. 有 HTML 的 console, 在单用户高级版 (WebSphere Advance Edtion Single Server: AEs) 中提供基于 Web Browser 的管理方式, 在 SUN 平台上可以安装, 需要预先安装必要的补丁;

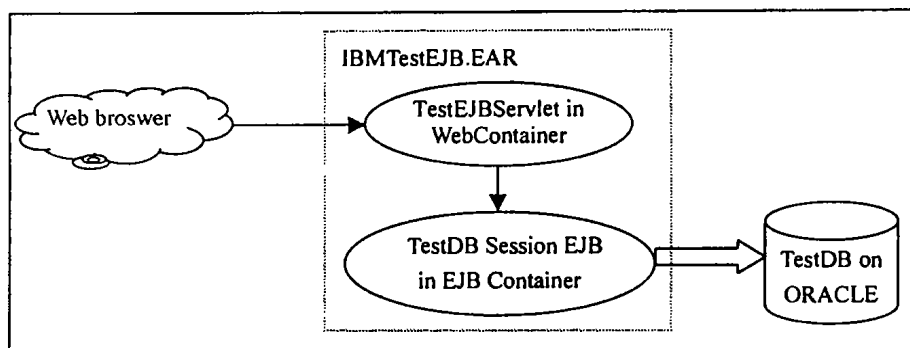


图3 测试软件架构

4. 支持 CORBA 的开发模式;
5. 管理数据库一般采用 DB2, 业务数据库可以是 ORACLE 等大型关系型数据库。但是管理数据库根据具体需要, 也完全可以不用 DB2, 而采用 ORACLE 等其它数据库。

3.4 BEA Weblogic

BEA WebLogic 是用于开发、集成、部署和管理大型分布式 Web 应用、网络应用和数据库应用的 Java 应用服务器。由于 BEA WebLogic 全面支持 Enterprise Java 标准, 因此它能有效地保护用户投资, 并使建立可移植、可伸缩的应用成为可能。同时, 这些应用能够完美地与其它应用和系统实现互操作。BEA WebLogic 应用服务器为 BEA 端到端企业中间件解决方案提供了关键的前端 Web 组件。

现场结合 UEDIT32 演示了 Hello World 级范例和简单数据库查询、负载均衡和容错。

1. 测试表明 BEA WebLogic 安装、使用方便, 占用资源少;
2. 作负载均衡时需要配置主、从服务器;
3. 其 IIOP 协议通过底层的 T3 协议实现, 运行效率高, 但在实现应用移植时需要替换客户端应用中的 T3 协议头。

结论 Application Server 是 Internet 上企业应用的基础,

采用 Application Server 技术可大大缩短开发周期、减小风险、降低成本。它在技术上并没有任何标准和规范是为某个商业性公司所独占, 全部都基于 Internet 上的开放标准, 因而具有极强的生命力。综合整个评测过程, 得出评测结果:

1. IBM 的 WebSphere WAS 产品, 在功能、性能、稳定性、集群功能、负载均衡能力、对各种硬件/操作系统平台的支持、对各种数据库的支持、开发/调测环境, 技术支持/售后服务等各个方面都有出色的表现, 并且可以和 IBM 的 MQ series, Workflow 等产品很好地实现无缝集成, 是一个很好的 J2EE 开发平台。
2. Borland AppServer 将 CORBA 和 EJB 较完美地结合在一起, 可以兼顾两个方向的发展。该产品紧跟 J2EE 标准, 综合性能名列前茅, 而作为 Borland AppServer 基础的 VisiBroker 正在充当 CORBA 技术的领导者。但测试过程也表明 AppServer 目前的版本不适合大型应用的开发, 对于规模较小的项目可以考虑。
3. SilverStream Application Server 提供对 Java 应用的支持, 对 C 应用通过 IIOP 协议提供 CORBA 接口, 适合开发和发布中、小规模基于 J2EE 标准的企业级应用, 对于小型的纯 Java 应用, 可以考虑该产品。

(下转第228页)

2.2 区域合并

经过上述的粗分割,图像中仍然会存在大量区域具有相近特征,我们根据区域色调距离和区域邻接关系对图像进行相似区域合并,以期在色调方面进一步接近人的视觉要求。设置一个色调距离函数,

$$D_{ij} = \frac{|r_i| \cdot |r_j|}{|r_i| + |r_j|} \|\mu_i - \mu_j\|$$

其中 $|r_i|, |r_j|$ 分别代表 i, j 区域包含的像素个数, μ_i 和 μ_j 表示两个区域的色调均值, $\|\cdot\|$ 表示欧氏距离。

当色调距离满足条件,并且区域相邻时进行合并,从而实现图像的较好分割。

图3是对图A分别根据R、G、B、H、S、L六个分量直方图对图像进行分割,在RGB颜色空间中R、B分量分割把路面

分成了不同的部分,产生了过分割,同理H、S分量也产生了过分割,而G、L分量相对分割效果要好一些,因为L是主要特征量,而在RGB空间中G分量包含较多的亮度信息,因此G、L两分量的分割效果较好。

3 实验与结论

图4的三组图片中,对比了人工分割的目标图,K-mean算法,Jseg算法和本文算法的分割结果。K-mean算法由于没有考虑区域邻接关系,过分割现象明显;Jseg算法则未考虑主特征量,分割的效果不稳定,有欠分割的情况。而本文算法由于自动选择主要特征量,之后融合色彩空间的色调信息和区域邻接信息进行区域合并,分割结果比较符合人的视觉特性。

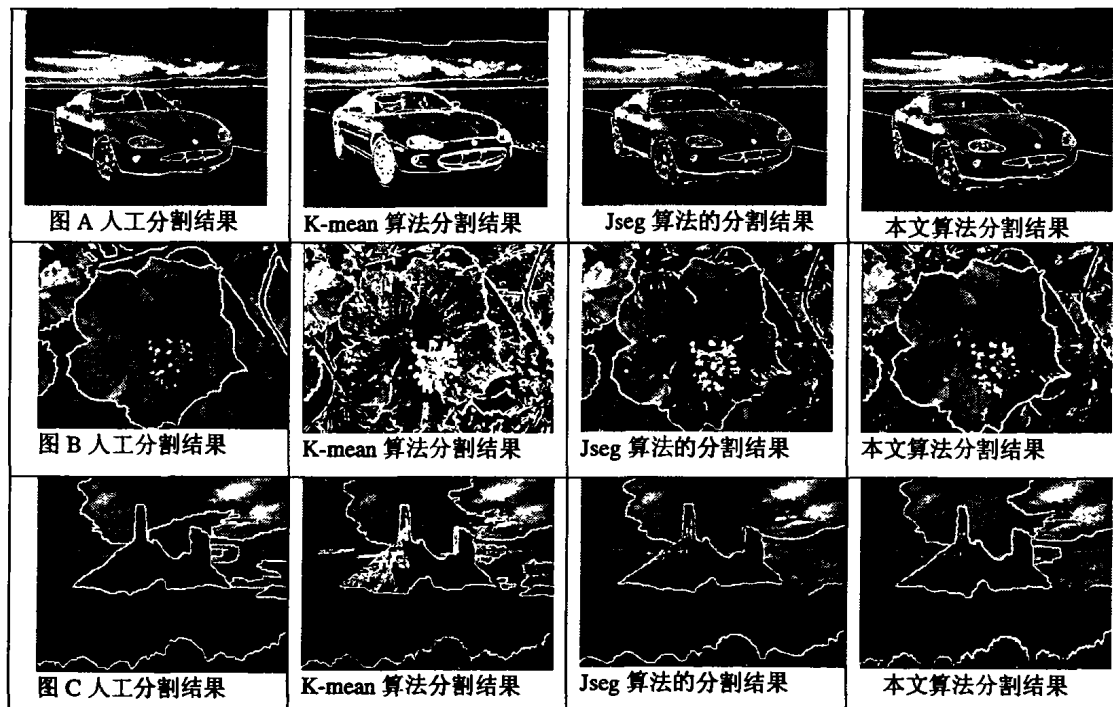


图4 不同算法分割结果对比

通过对大量图片进行分割处理比较,我们提出的算法对无显著纹理的图像具有良好的分割效果。通过RGB和HSL两个色彩空间的结合,避免了分割过程中由于颜色空间的奇异性造成的图像突变,并采用三点循环平滑直方图技术,自适应地确定主要特征量,实现对图像的粗分割,再考虑像素的颜色距离和区域邻接关系进行区域合并,分割结果和人的视觉具有良好的一致性。未来的工作中,可以将图像的纹理特征融入算法中,分割性能将进一步的提高。

参考文献

1 叶齐祥,高文,王伟强,黄铁军.一种融合颜色和空间信息的彩色图

(上接第146页)

4. BEA WebLogic 性能完备,但由 T3协议造成的移植性问题可能成为未来的隐患。

参考文献

1 Sun Microsystems, Inc. Enterprise JavaBeans TM Specification, Version 2.0 Final Release[OL]. <http://java.sun.com/j2ee>, 2001-08-22

像分割算法.软件学报,2004,15(4):522~530

2 Pham T D. Image Segmentation using Probablistic Fuzzy C-Means Clustering. In: Proc. of Intl. Conf. on Image Processing. 2001,21(3): 722~725
 3 龚声蓉,熊璋,孙文彦.一种基于HSL概率密度的彩色图像分割方法.小型微型计算机系统,2001,22(6):695~698
 4 Deng Yining, Manjunath B S. Unsupervised Segmentation of Color-Texture Regions in Images and Video. IEEE Transaction on Pattern Analysis and Machine Intelligence, 2001, 23(8): 800~810
 5 魏宝刚,鲁东明,潘云鹤,杨云.多颜色空间上的交互式图像分割.计算机学报,2001,24(7):770~775

2 Mohan C. Tutorial: Application Servers and Associated Technologies. ACM SIGMOD'2002, June 2002
 3 MetaGroup. J2EE App Server Evolution: Survival of the Cheapeast?, 2002
 4 Coulouris G, Dolimore J, Kingberg T. Distributed Systems Concepts and Design. 3rd Edition, Addison-Wesley, 2000
 5 Geibs K. Middleware Challenges Ahead. Computer. org, 2001
 6 Sun Microsystems. Java2 Platform Enterprise Edition Specification(v1.3), 2001