

中间件与 EJB 构件技术研究

柴春轶, 于雪燕, 胡金初

(上海师范大学 数理信息学院, 上海 200234)

摘 要: 目前由于多方面的因素,越来越多的企业涌入了合并的潮流,中间件技术显得越发重要。文中首先从中间件技术谈起,然后讲述了支持服务器端中间件技术开发的平台 J2EE,最后着重详述了 J2EE 的核心 EJB 构件技术。可见新的应用特点对中间件技术的发展提出了新的挑战,也决定了中间件技术未来几年的应用发展方向,为了满足各方面的需求,中间件技术要呈现出丰富多彩的格局。

关键词: 中间件;构件;J2EE;EJB;MDB;JMS

中图分类号: TP311.5

文献标识码: A

文章编号: 1673-629X(2006)06-0163-03

Research of Middleware and EJB Component Technology

CHAI Chun-yi, YU Xue-yan, HU Jin-chu

(Shanghai Normal Univ., Shanghai 200234, China)

Abstract: Nowadays, because of various factors, more and more corporations swarm into the incorporative tideway, so the technology of middleware becomes more important. Firstly, introduces the middleware technology. Secondly, tells of the Platform J2EE that sustains server develop middleware technology. At last, amplifies the EJB component technology that is the core of J2EE. So, the characteristic of new applications put forward new challenges, also determine the development of orientation of future application to the technology of middleware, for the sake of the various demands, the technology of middleware must present rich and colorful patterns.

Key words: middleware; component; J2EE; EJB; MDB; JMS

1 中间件技术

1.1 中间件的概述

中间件(Middleware)是基础软件的一大类,它处于操作系统软件与用户的应用软件的中间^[1],在操作系统、网络和数据库之上,应用软件的下层。总的作用是为处于自己上层的应用软件提供运行与开发的环境,帮助用户灵活、高效地开发和集成复杂的应用软件。

在众多关于中间件的定义中,比较普遍被接受的是 IDC 的表述^[2]:中间件是一种独立的系统软件或服务程序,分布式应用软件借助这种软件在不同的技术之间共享资源;中间件位于客户机服务器的操作系统之上,管理计算资源和网络通信。IDC 对中间件的定义表明,中间件是一类软件,而非一种软件;中间件不仅仅实现互连,还要实现应用之间的互操作。

1.2 中间件的分类

中间件的种类很多,根据中间件在系统中所起的作用和采用的不同技术,大致划分为以下 5 类。

(1)数据库中间件(ODBC, JDBC);

(2)远程过程调用中间件(Remote Procedure Call, RPC);

(3)面向消息中间件(Message Oriented Middleware, MOM);

(4)基于对象请求代理的中间件(Object Request Broker, ORB);

(5)事务处理中间件(Transaction Processing Monitor, TPM)。

1.3 中间件的十大优点

中间件的十大优点:缩短应用的开发周期;节约应用的开发成本;减少系统初期的建设成本;降低应用开发的失败率;保护已有的投资;简化应用集成;减少维护费用;提高应用的开发质量;保证技术进步的连续性;增强应用的生命力。

1.4 中间件的工作原理

分布式应用程序需要一定的数据或服务,而这些数据或服务在这样一个分布式系统中可能处于运行不同操作系统和特定数据库管理系统的服务器中。此应用程序中负责寻找数据的部分只需访问一个中间件系统,由中间件完成到网络中查找数据源或服务,进而传输应用程序的请求、重组答复信息,最后将结果送回应用程序的任务,如图 1 所示。

收稿日期:2005-10-05

作者简介:柴春轶(1980-),男(满族),辽宁葫芦岛人,硕士研究生,研究方向为网络与多媒体技术;胡金初,教授,研究方向为网络与多媒体技术、并行计算。

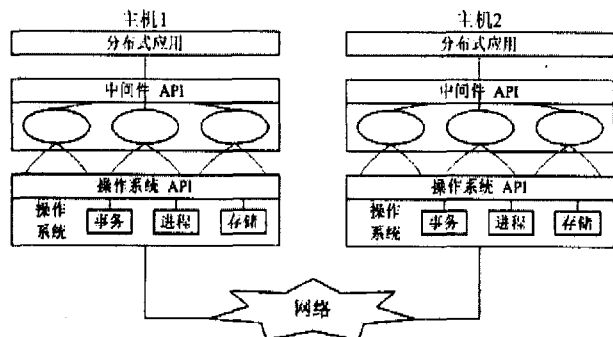


图 1 中间件的工作原理

2 中间件技术开发的平台

2.1 J2EE(Java 2 Platform Enterprise Edition)平台

Sun 公司联合 IBM, Oracle, BEA 等大型企业应用系统开发商共同制定了一个基于 Java 组件技术的企业应用系统开发规范,该规范定义了一个多层企业信息系统的标准平台,旨在简化和规范企业应用系统的开发和部署,可见, J2EE 是一种更理想的集成平台和规范。

2.2 J2EE 平台架构

它是基于组件-容器搭建的^[3],如图 2 所示。

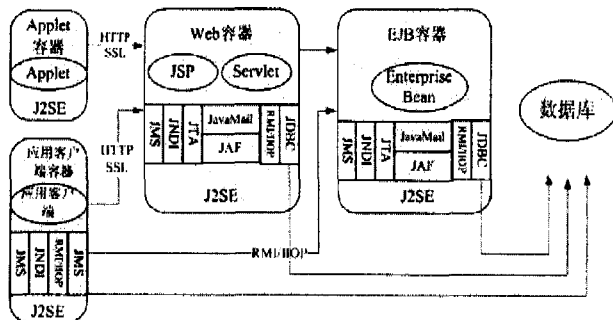


图 2 J2EE 平台架构

可见 J2EE 平台是由 Applet 组件、Application 客户组件、Web 组件及 EJB 组件和它们各自所必需的容器构成。J2EE 中组件之间不能直接访问,而是通过容器提供的协议和方法来相互调用。

3 服务器端构件技术 EJB(Enterprise JavaBean)

3.1 EJB 容器

EJB 是 J2EE 平台的核心,其核心思想是将商业逻辑与底层的系统逻辑分开,使开发者只需关心商业逻辑,由 EJB 容器实现目录服务、事务处理、持久性、安全性等底层系统逻辑。EJB 容器还负责 EJB 的部署、发布和生命周期管理。比较流行的 EJB 容器有 IBM 的 WebSphere; BEA 的 WebLogic Server; Sun 公司的 iPlant。

3.2 EJB 组件架构

EJB 不是一个产品,而是 Java 服务器端服务框架的规范,软件厂商根据它来实现 EJB 服务器,使得应用程序开发者可以专注于支持应用所需的商业逻辑,而不担心周围框架的实现问题,EJB 组件架构模型如图 3 所示。

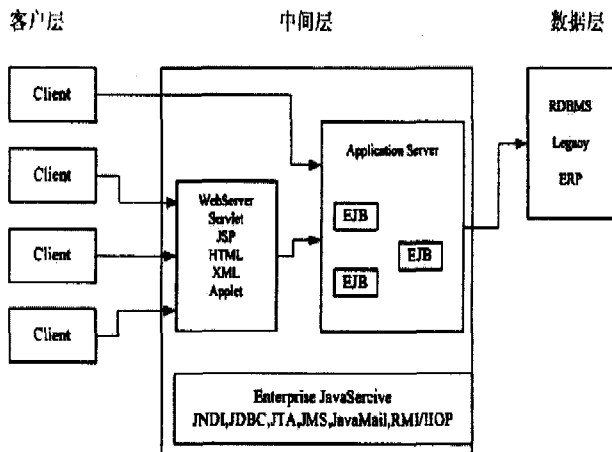


图 3 EJB 组件架构

3.3 EJB 开发过程

一个 EJB 开发者开发的 EJB 通常由 Bean 实现对象、远程接口、Home 接口和部署描述文件构成^[4]。Bean 实现对象仅实现业务逻辑,而服务逻辑是通过基于 XML 的部署描述文件来完成,该文件精确描述了所需的事务、安全等服务,最终由容器根据描述提供所需服务逻辑。显然,基于构件的分布式应用与传统分布式应用的区别主要在于对服务支持逻辑的分离。以事务处理为例,传统应用要求在对象实现中对事务服务接口进行直接编程,实现事务控制。而在基于构件的应用中,Bean 开发者只需声明所需服务,无需实现任何事务控制代码,在部署时由 J2EE 平台自动生成相应代码,并在运行过程中隐式实现构件所需的事务控制。

另外,Bean 实现对象没有任何体现分布特性的代码,所以 EJB 开发者无需面向中间件的 API 进行网络编程,从而简化了开发过程。远程接口定义了 Bean 实现对象向外部客户提供的服务接口信息。Home 接口定义了外部客户用于创建、删除以及查询 EJB 的接口信息。在应用部署时,J2EE 平台将根据 Home 接口和远程接口等信息自动生成 Home 对象和 EJB 对象。

EJB 对象实现了远程接口,对外接收客户发向 Bean 的服务请求,对内将请求转发给 Bean 实现对象。分布对象 EJB 对象,作为 Bean 的请求代理,一方面使得 Bean 间接具有网络特性,可被远程访问;另一方面,也通过截获请求从而向 Bean 提供了隐式的管理和服务。作为管理构件的工厂对象,Home 对象实现了 Home 接口。因为 EJB 对象作为 Bean 的代理,直接面向客户,而真正运行于容器中的 Bean 实例的创建和删除,由容器根据对资源的管理而独立实现。EJB 对象和 Home 对象的分布特性最终使 EJB 成为一个分布对象,可被远程创建并访问。在开发的最后阶段,EJB 开发者将 Bean 实现对象、远程接口、Home 接口和部署描述文件打成一个 Jar 格式的压缩包,称为构件包,EJB 以构件包的形式流通、交易和被组装重用。

构成应用的 EJB 通过应用部署,最终运行在 EJB 容器中。在部署过程中,J2EE 平台将根据远程接口、Home

接口以及部署描述文件等信息,自动生成 EJB 对象、Home 对象以及其它辅助对象,并完成所需外部资源的绑定等工作,从而使构件在容器的支持下正常响应远程客户的请求,获得声明的所需服务,并接收容器其它方面的管理。至此,基于 EJB 的应用进入正常运行状态。

3.4 EJB 的工作原理

结合上文的开发过程可以看出 EJB 的工作原理,如图 4 所示。

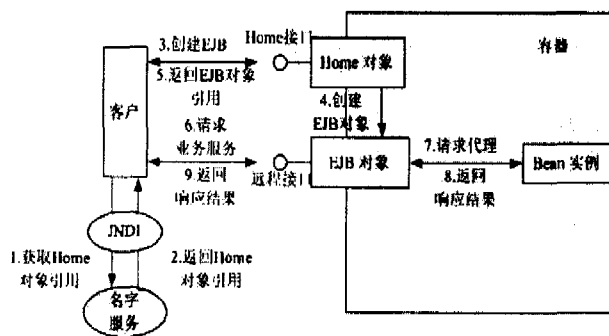


图 4 EJB 的工作原理

3.5 EJB 分类

(1) 会话 Bean(Session Bean)。

Session Bean 是一种代表客户程序执行操作的 EJB^[3],在 J2EE 应用程序中被用来完成一些服务器端的业务操作,如访问数据库,调用其它 EJB 组件。会话 Bean 分为两种:无状态会话 Bean 和有状态会话 Bean。无状态会话 Bean 不能维持一个调用客户的状态,在一个方法调用过程中,它可以维持调用客户的状态,但是,当方法执行完时,状态不会保持,并且无状态会话 Bean 被立即释放,所以可以支持大量的客户调用。有状态会话 Bean 可以一对一地维持某个调用客户的状态,并且在不同的方法中维持这个状态,由于对每一个并发用户,必须有一个对应的有状态会话 Bean。显然在多用户并发执行条件下,无状态会话 Bean 运行效率要高于有状态会话 Bean。

(2) 实体 Bean(Entity Bean)。

实体 Bean 代表数据库或另外一个企业应用系统中的数据对象,如代表数据库的一行记录。由于实体 Bean 是底层数据库记录的映象,和数据库记录保持同步,所以它是持久的,其状态不依赖于服务器而存在,允许共享访问。按持久性划分,实体 Bean 分为 Bean 管理持久化和容器管理持久化两种。前者指有 Bean 开发者自己管理 Bean 和

它所代表的数据库记录同步;后者指由容器自动管理 Bean 和它所代表的数据库记录同步,不需要开发者编写 SQL 语句。

(3) 消息驱动 Bean(Message Driver Bean, MDB)。

会话 Bean 和实体 Bean 都是基于 RMI-IIOP 这种同步调用机制进行通讯的,而 MDB 是基于 Java 消息服务(Java Message Service, JMS)技术的异步消息通讯机制通讯的^[5]。类似会话 Bean, MDB 也用于实现业务过程,但是 MDB 是 JMS 的监听器,一般是客户把消息发布到 JMS 目的地,然后 JMS 提供者和 EJB 容器协作,把消息发送给 MDB,即对 MDB 的访问只能通过向其发送消息这种异步方式实现。

3.6 EJB 3.0

上述主要是 EJB 2.0 规范,刚刚出现的 EJB 3.0 规范的变化可以大致分成 2 大部分:

1) 使用了以注释为基础的 EJB 编程模型。

2) 提供了新的实体 Bean 持久化模型, EJB QL 也有许多重要的改变。

4 结束语

中间件技术的应用在当今是必不可少的,其应用也很多,但都存在着一定的缺点,比如 J2EE 只能局限于 Java 语言的开发;EJB 2.0 规范虽然能帮助开发者缩短开发时间,但也存在很多弊端,所以 EJB 3.0 规范应运而生,但还不够成熟,有待完善。

参考文献:

- [1] 王安保,郭建星,王晓静,等. 中间件技术在程序设计中的运用[J]. 微机发展, 2003, 13(6): 60-64.
- [2] Wagner G. Agent - Oriented Analysis and Design of Organizational Information Systems [A]. In: Proceedings of Fourth IEEE International Baltic Workshop on Databases and Information Systems [C]. Lithuania: Vihus, 2000.
- [3] 张宏展,蔡宗琰,吴欣. 实战 J2EE 与 Weblogic Server 应用开发[M]. 北京:电子工业出版社, 2004.
- [4] 赛蕾,袁臻,刘冬梅. 基于构件技术的中间件技术 J2EE [J]. 计算机科学, 2004, 31(6): 13-17.
- [5] Artiges M. BEA Weblogic Server 8.1 大全[M]. 袁毅,等译. 北京:机械工业出版社, 2004.

(上接第 162 页)

- [3] IBM, BEA, Microsoft, TIBCO Software. Web Services Reliable Messaging Protocol [EB/OL]. <http://www-900.ibm.com/developerworks/cn/webservices/ws-rm/ws-rm-reliablemessaging.pdf>, 2003.
- [4] Mani A. Understanding quality of service for Web services

[EB/OL]. <ftp://www6.software.ibm.com/software/developer/library/ws-quality.pdf>, 2002.

- [5] 贾晖,吴江,张佩云. 基于 Web Services 的电子商务模型分析[J]. 微机发展, 2005, 15(3): 66-68.