

基于 Weblogic 面向服务架构的信息门户开发实践

高明 (东北财经大学信息工程学院 辽宁 大连 116025)

摘要:本文立足于对企业信息门户的体系结构的研究,提出一个基于面向服务体系结构(Service Oriented Architecture, SOA)设计思想的企业信息门户。并通过实例展现了基于 Weblogic Portal 和 Workshop 集成开发环境构建的信息门户的开发流程。该门户具有清晰的四层架构:门户应用桌面、应用和服务层、业务逻辑和组件层、企业资源整合层。

关键词:企业信息门户 面向服务 Portlet Weblogic

1 面向服务架构

面向服务架构(Service - Oriented Architecture, SOA)是一种面向企业级服务的系统架构,在基于 SOA 架构的系统中,具体应用程序的功能是由一些松散耦合且具有统一接口定义方式的组件(服务组件, Service)组合构建起来的,并提供一个抽象的服务层,对服务使用者隐藏了服务的实现细节,具有可复用、灵活和可扩展等诸多优势。

SOA 并不是一个新概念,20 年前, Gartner Group 就提出了 SOA 的概念,其基本思想是以服务为核心,将企业的 IT 资源整合成可操作的、基于标准的服务,使其能被重新组合和应用。今天, SOA 已从概念模型变为业界的热点技术、客户未来的投资重点,这不仅归功于 Web 服务标准及相关软件技术的成熟,更重要的是众多业内厂商包括 BEA, IBM, HP, Oracle, Microsoft 等对 SOA 的倾力支持。

2 Weblogic 面向服务架构平台概述

20 世纪 90 年代,企业纷纷实施 Web 应用项目和构建多元化门户,在许多情况下,此类门户的部署未曾考虑到业务逻辑的重用、单点登录或通用的用户界面等问题。问题日趋复杂,并最终面临 Web 灾难蔓延的结果,在这种境况下企业 IT 部门必须尝试维护数以千百计的离散门户和 Web 应用。而当前最好的方法就是——企业门户合理化。很多领先的软件和方案提供商提供了 BEA WebLogic Portal 之类统一的、简化的和可扩展的门户平台,能够从容的应对企业的 Web 灾难蔓延。

BEA WebLogic Portal 是独立、可靠的解决方案,能够安全、高性价比的创建和维护一个统一的门户基础结构,而且能够与合作伙伴 Documentum、HP、Hyperion 的内容管理、硬件、系统集成和商业智能紧密集成。

下图是 Weblogic Portal 的门户解决方案。Weblogic Server 是传统的 J2EE 容器,提供了最基础的服务如 EJB, JNDI, JMS, Servlet/JSP 等,它工作在其下的 J2SE 实现上,这里它采用了经过二次开发优化过得的 JVM——JRockit,一般用于正式运行中,开发中则一般使用 Sun 的 J2SE SDK1.4,以便利于调试。Application Framework 是 Weblogic 开发平台的重要扩展,它内置了比较完整的企业开发框架(可从 Apache Beehive 获取更多信息),并将这个框架集成在 Weblogic Workshop 中,它提供了一些对企业数据和通用业务的基本类模块和一些业务流程,大大提高了开发的效率,降低了重复开发带来的资源、人力浪费。最上一层是 Weblogic Integration,它主要用于集成企业遗留信息系统,业务流程,并且采用了开放的、标准的 ebXML 和 RosettaNet 协议作为电子商务对话的语言,提供了 B2B 的信息集成和商务协作功能。Weblogic Portal 在这些基础构件的支持下,提供了门户桌面的开发框架,并且有效的将各种集成的信息、业务流程个性化的展现给不同的门户用户。

3 基于面向服务架构的企业信息门户

下图是一个基于 Weblogic Portal 和 Workshop 集成开发环境构建的 B2E (business - to - enterprise, 商务对企业)企业信息门户。该门户具有清晰的四层架

构:门户应用桌面、应用和服务(主要是 Web 应用和 Web 服务应用)、业务逻辑和组件、企业资源(内部和外部)。企业内部的员工、客户和合作伙伴通过统一的

品库存信息、订单管理信息,客户关系管理信息,电子邮件,文件系统,支付系统等有机的整合起来。

典型门户结构包括以下几个方面:

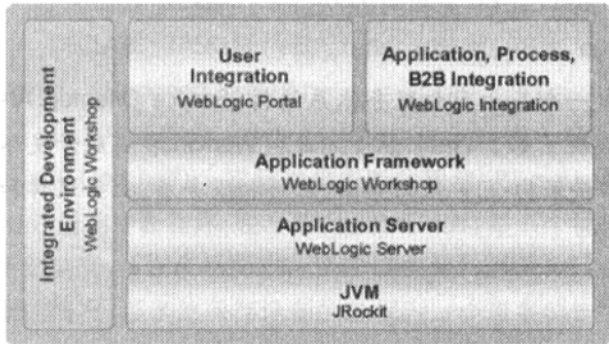


图 1 Weblogic Portal 门户解决方案体系结构

(1) 门户应用桌面包括雇员桌面、客户桌面、合作伙伴桌面。

(2) 应用和服务(主要是 Web 应用和 Web 服务应用)包括订单接口的 Web 服务、订单向导的 Web 应用、订单审批的 Web 应用。

(3) 业务逻辑和组件包括订单服务复合逻辑、订单管理业务流程、客户系统业务流程、产品库存管理业务流程、订单报表业务流程。

(4) 企业资源(内部和外部)包括:

内部资源: SAP 订单应用系统、商务系统、支付系统、客户服务系统、定价组件、邮件系统、文件系统、IT 基础架构

外部资源: 运输跟踪 Web 服务、信用报告 Web 服务、税金计算 Web 服务、合作厂商 Web 服务。

下面将主要从该门户应用提供的订单管理系统进行分析,并对主要的实现进行了深入剖析和扩展。



图 2 门户组件

4 基于 Weblogic Portal 和 Workshop 的门户管理生命周期

企业可以通过 Weblogic Portal 和 Workshop 实施构建和管理门户的完整生命周期:

(1) 架构设计。Weblogic Portal 和 Workshop 提供了 IT 基础架构设计的最佳实践模型、公司外观和混合业务逻辑,这些均由使用 Weblogic Workshop 与传统 J2EE 工具的传统 J2EE 开发人员提供。这些资源被封装在 Weblogic Workshop 应用与项目模板中。例如,内部门户通常需要的功能,像显示企业信息系统中的信息的 portlet,可以包含在一个应用或项目门户模板中,而面向用户的商业功能可能位于另一个应用或项目门户模板中。

(2) 开发。业务应用开发人员以这些模板为起点,为其添加书(Book)、页面(page)和 portlet 功能。其中可能包括一组基本的访问者活动(campaign),或者包括到 Weblogic Workshop 中轻松可用的业务流程及 Web 服务的链接。业务应用开发人员与门户界面设计人员合作,对诸如页面与 portlet 这样的

Portlet 实例 - PortletInstance	
Portlet 实例属性	
标题	Order Entry
实例标签	portlet_1_1
Portlet URI	/pageFlows/main/Controller.portlet
主题	
方向	
默认状态为最小化	false

图 3 订单条目管理 portlet 的实例属性

用户认证方式进入专门为自己设计的个性化的网页界面,察看相关信息,诸如产品、库存、工作流程、电子邮件、专家文章等等,不同的角色之间可以很容易的通过门户平台实现协同工作,在后端的应用服务和业务逻辑层的支持下,将该企业的各种资源: SAP ERP 中的产

门户资源进行改进,尽量符合由 IT 设计师提供的最佳实践,从而开发出可以用于组装大量门户的门户模板。

除并修改销售内容。最后,访问者可以管理自己的门户页面,创建新的门户页面并添加他们需要的特定内容和功能。

5 门户应用桌面层

5.1 门户 Web 应用程序的体系结构

WebLogic Portal 体系结构具有较强的灵活性和可扩展性。无论是在 WebLogic Workshop 中生成门户资源和模板,还是使用 WebLogic Administration Portal 创建和管理门户,都可以先处理单个组件,然后再由门户框架将它们组合起来。

构成门户的组件

在 WebLogic Portal 中,门户的定义是一个 XML 文件。XML 文件在门户设计器生成门户时自动创建,门户设计器作为 WebLogic Workshop Portal Extensions 的一部分提供。门户文件包含组成特定门户实例(如书册、页面、Portlet 和外观及感觉组件)的全部组件。许多组件之间具有分层关系。例如,书册包含页面,而页面包含 Portlet。文档结构图示显示了门户文件中的组件之间的关系。以下是组成门户接口的组件的描述。

5.2 门户开发工具和服务

WebLogic Portal 包括许多功能强大的工具和服务,它们使门户开发快速简便。

(1) 设计器——WebLogic Workshop Portal Extensions 包括门户设计器及 Portlet 设计器,可提供图形拖放功能和基于网格的属性设置,可以快速简便的创建完善的门户。

(2) Java 控件——WebLogic Workshop Enterprise Edition 包括许多功能强大的 Java 控件,开发者无需从事低级编码,从而减少了要写入的代码行(进而减少错误数)。Java 控件提高了开发人员开发应用程序的工作效率。开发人员可以将用户配置文件控件插入 JSP 或 Java 页面流,并选择检索及更新用户配置文件信息的适当方法,确定用户是否存在,或根据搜索参数检索用户列表,而无需编写代码以直接访问 API 或 J2EE 资源或使用 JSP 标记。

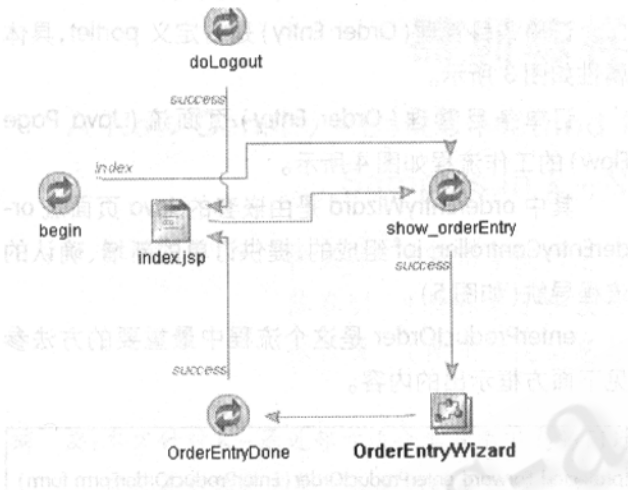


图 4 订单条目管理(Order Entry)页面流的工作流程

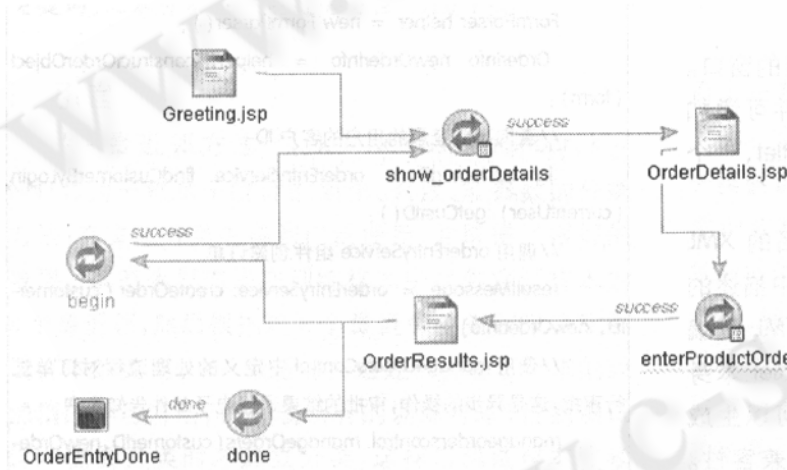


图 5 orderEntryController.jsp 流程

(3) 组装与部署。业务单元或 IT 部分的门户管理人员使用基于浏览器的工具来组装和配置门户,出发点是由业务应用开发人员提供的门户模板中的资源。通过使用页面和 portlet,可以组装多个称为书的页面组,并在提供的最佳实践指导方针中选择外观。

(4) 管理。通过使用委派管理,可以将不同的管理角色委派给各个门户管理员组。例如,内容提供者角色可以被定义为只允许改变特定的内容集合。内容提供者的例子之一是销售内容提供者,他可以添加、删

(3) JSP 标记——WebLogic Workshop Portal Extensions 包括 JSP 标记库,减少重复编码。

(4) 内容管理——WebLogic Portal 包括功能强大的内容管理功能,可以在一个虚拟内容仓库中集成并管理多个内容管理系统。开发人员可以查询仓库,检索并显示门户应用程序的个性化内容。虚拟内容仓库在 WebLogic Administration Portal 中设置并管理,支持完全开发生命周期。

其它服务包括:

统一用户配置文件、个性化、交互作用管理和行为跟踪、API 及移动设备等等。

5.3 门户桌面的核心构件:Portlet

Portlet 是门户 Web 应用程序的基本组成部分,它使应用程序子集的行为能作为单个单元进行管理。Portlet 以一组关联文件的形式存在,通常为 XML 和 JSP。在 WebLogic Workshop IDE 中,可以在设计视图图中对 Portlet 进行可视化编辑,并可在设计视图和源代码视图中编辑 JSP。

Portlet 是呈现应用程序、信息和业务流程的窗口。Portlet 可以彼此通信,可以使用 Java 控件,并可添加到 Java 页面流中。一个页面可以有多个 Portlet,一个 Portlet 也可以有多个实例。

在结构上,Portlet 是具有 .portlet 扩展名的 XML 文件所描述的对象集合。框架使用该文件中描述的元素在运行时渲染 Portlet,在最终生成的 HTML 汇编中的特定点应用任何许可权限和自定义。Portlet 本身可以使用 JSP、页面流或可选支持文件,并且可以生成符合 JSR 168 标准的 Portlet,以实现 Portlet 兼容性。Portlet 可以使用现有的 Web 应用程序和内容(ASP、JSP、HTML、XML,等等)。

Weblogic Portal 已经实现了符合 JSR168 规范的 portlet 容器,利用了 Weblogic Application Server 在故障恢复、可扩展性、安全性及 portlet 的热部署等领域中的全部功能。Portlet 容器将管理 portlet 生命周期中的所有阶段。

Portlet 容器支持 Portlet Preferences 的概念,一个具体的 portlet 实例可以用作多个用户的逻辑实例,而且每个用户可以拥有多个实例,这样这些用户就能定制与他们的逻辑实例相关联的行为及外观。

6 应用和服务层

6.1 订单条目管理(Order Entry)的实现:

订单条目管理(Order Entry)是自定义 portlet,具体属性如图 3 所示。

订单条目管理(Order Entry)页面流(Java Page Flow)的工作流程如图 4 所示。

其中 orderEntryWizard 是由嵌套的 Java 页面流 orderEntryController.jspf 组成的,提供订单的新增、确认的流程导航(如图 5)。

enterProductOrder 是这个流程中最重要的方法参见下面方框示出的内容。

```
protected Forward enterProductOrder( EnterProductOrderForm form)
{
    //利用 FormParser()把 form 表单对象构造成业务中定义的订单
    信息对象 OrderInfo
        FormParser helper = new FormParser();
        OrderInfo newOrderInfo = helper. constructOrderObject
        (form);
        //查询当前登录的用户的客户 ID
        int customerID = orderEntryService. findCustomerByLogin
        (currentUser).getCustID();
        //调用 orderEntryService 组件创建订单
        resultMessage = orderEntryService. createOrder ( customer-
        ID, newOrderInfo);
        //调用 manageOrdersControl 中定义的处理流程对订单进
        行审批,这是异步的操作,审批的结果通过电子邮件告知客户
        manageorderscontrol. manageOrders( customerID, newOrde-
        rInfo);
        resultMessage = " Thank you. Your order is being pro-
        cessed, expect to receive an email confirming your order within 24
        hours. ";
        return new Forward(" success" );
}
```

其中的 orderEntryService 是位于业务逻辑和集成逻辑服务层的一个服务组件,它提供了 Web 应用服务所需的一些业务方法,对企业的各种数据资源进行了有效的集成,并且能够支持同步和异步方法的调用。

(下转第 74 页)

参考文献

- 1 Christopher C. Shilakes and Julie Tylman, Enterprise Information Portals, Merrill Lynch, 16 November, 1998.
- 2 IBM 公司, Guide to WebSphere Portal 5.0, IBM Developer Works 网站, September 2003.
- 3 BEA 公司, BEA Portal brochure, BEA, 2003 年.
- 4 Marshall Lamb, 专家访谈: Marshall Lamb 谈 WebSphere Portal 编程, IBM Developer Works 网站, 2004 年 2 月.
- 5 张应利、毛小娟, 企业信息门户: 通往网络经济的桥梁, 经济论坛, 2002 年, 第 12 期.
- 6 DeveloperWorks, 专题: SOA—面向服务的体系结构, IBM Developer Works 网站, 2004 年 3 月.
- 7 Daniela Rudrof, Andre Tost, 用 Web 服务和 J2EE 集成企业应用程序, IBM Developer Works 网站, 2002 年 11 月.
- 8 李卫红、白杨, EIP 的功能与实现技术研究, 情报科学, 2004 年 2 月, 第 22 卷第 2 期.
- 9 Craig A. Berry, John Carnell, Matjaz B. Juric, 实用 J2EE 设计模式编程指南, 电子工业出版社, 2003 年 1 月第一版.