

用 Portlet 技术实现数字化校园信息资源的整合

田昌鹏 张升平

(重庆工商大学网络中心 重庆 400067)

摘要 建设数字化校园是推动教育信息化的重要系统工程,而整合各种信息资源是当前研究的重点问题,其中有很多策略和技术问题需要解决。根据整合资源的需要,提出一个基于 Portlet 技术的门户解决方案,论述了门户框架的具体工作方式以及 Portlet 技术的实现原理,从而解决了数字校园建设中的系统集成,信息整合和个性化信息服务的问题。通过 Portlet 技术构建的校园信息门户,降低了门户实现和管理的成本,提高了工作效率,在数字化校园的建设中将会起着较大的作用。

关键词 门户, Portlet, 数字化校园

Use Portlet Technology to Realize Digital Campus Resources Integrating

TIAN Chang-Peng ZHANG Sheng-Ping

(Chongqing Technology and Business University, Center of Network, Chongqing 400067)

Abstract Digital campus construction is an important systemic project for modernization of education, but integrating various information resources is the priority problem studying currently, and there are lots of problems of strategy and technique to be solved. According to the need integrating resource, resolve a portal scheme based on Portlet, it discusses the frame of campus information portal and explains the mechanism of portlet, and it solves the problems of application system integration, information integration and customized information service in digital campus construction. This system reduces the cost of portal implementation and management and improves the efficiency, so it will have a great function on digital campus construction.

Keywords Portal, Portlet, Digital campus construction

1 Portal 和 Portlet

门户(Portal)是 IT 领域的新技术,顾名思义,门户是通往一套资源的入口。即,在有效的网络的环境下,把各种应用系统资源、数据资源、信息资源统一集成到一个平台之下,根据每个用户使用特点和角色的不同,形成个性化的应用界面,并通过对事件和消息的处理传输把用户有机地联系在一起。Portal 的组成可以分为三部份: Portal Server、Portlet Container、Portlet,如图 1 所示。

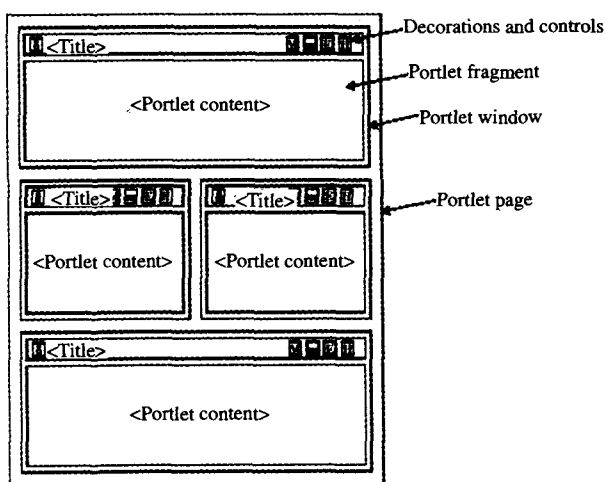


图 1 Portal 页面的结构

1) Portal Server: 一个 Portal(门户网站)就是指一个基于 Web 的系统,通常都会提供个性化设置、单点登录、以及由各

种不同来源或不同网站取得各式各样的信息,并且将这些信息放在网页之中组合而成的呈现平台,门户网站会有精巧的个性化设置去提供定制的网页,当不同等级的使用者来浏览该页面将获得不同的信息内容。

2) Portlet Container: 是提供 Portlet 执行的环境,包含了许多 Portlet 并且管理他们的生命周期,他也会永远保存着 Portlet 的喜好设置,一个 Portlet container 接收到来自 Portal 的请求后,接着将这个请求传递给存在 container 的 Portlet 执行。Portlet container 没有义务去组合 Portlet 产生的信息内容,这个工作必须由 Portal 来处理。Portal 和 Portlet container 可以放在一起视为同一个系统的组件,或者分开成为两个独立的组件。

3) Portlet: 一个 Portlet 是以 Java 技术为技术的 Web 组件,由 Portlet Container 所管理,专门处理客户的 request 以及产生各种动态的信息内容。Portlet 为可插式(pluggable)的客户界面组件,提供呈现层成为一个信息系统。这些由 Portlet 产生的内容也被称为片段(fragment),而片段是具有一些规则的 Markup(HTML、XHTML、WML),而且可以和其他的片段组合而成一个复杂的文件。而 Portlet 中的内容正常来说是与其它 Portlet 的内容聚合而成为一个 Portal 网页。而 Portlet 的生命周期是被 Portlet Container 所管理控制的。

2 门户件 Portlet 技术

我们所说的数字化校园信息资源的整合是将已有的各种应用和信息在表现层的重整,是对能力和资源面向使用者的角色所进行的重构。门户系统的建设使用户可以使用单一的

入口访问多种类型的资源信息。无论信息存储在何处、无论信息的格式如何,都可以按照用户所关注的方式个性化地收集和展现这些信息,并对这些信息进行存取。给用户提供了依据各自角色的、个性化的工作界面,通过该界面可以统一地对所关心的各种信息进行访问。

Portlet 是门户系统页面的一部分,从最终用户的角度看,它是一个能够在门户系统页面显示出来的工作区域,为用户提供和执行一项服务,和 PC 桌面的窗口相似,每一个 Portlet 都占用浏览器的一块区域来显示信息。Portlet 的内容可以简单到一个电子邮件,也可以复杂到全校范围内的包含教务教学、学生管理、人事、科研等的综合教务管理系统。从开发者的角度看,它是采用 J2EE 技术开发并且运行在门户系统 Portlet 容器中的插件,是在 Portal 中运行的小应用程序,是基于 Java 的,由容器管理的 Web 构件,用来生成动态内容,是最终用户可以在自己的门户页面上看到的一些可视化的、动态的应用组件。从内容提供者的角度来看,一个 Portlet 就是一种能够获取到所需要的信息内容的方法。从门户系统管理员的角度来看,Portlet 被设计成可以组合起来形成整个门户页面。一个 Portlet 就是一个存放内容的容器,它可以通过门户框架来注册,然后用户才可以预定。

为使各种资源和服务能正确展现在门户系统的 Portlet 中,系统的资源集成层将通过统一的资源访问接口实现后台分布的、异构的资源接入,形成统一的资源视图。门户服务支撑层将利用下层提供的各种资源接口,形成各种资源服务,但它们必须严格按照 JSR168 中 Portlet 的接口标准对这些资源和服务进行规范,以便于这些资源和服务能够有效地展现在门户系统中。Portlet 常常是应用的一些小的部分,它们并不能代替整个应用的数据显示和事务,只是常用来获取应用的数据或者信息。有些情况下,它只是一个增强生产力的工具,用户可以用来快速得到获取不同的应用系统、文档和结果内容的工具和信息。在任何情况下,Portlet 都可以基于用户的选择提供其需要的信息的实时显示。

3 采用门户技术和实现资源的整合

3.1 基于 Portlet 的门户设计

在具体现实中,我们采用了 SUN ONE Portal Server,它为最终用户提供一个门户桌面,实现对资源和应用的访问。同时,门户服务器提供了桌面(Desktop)功能。桌面是一种逻辑组件,包括 Servlet、各种提供者(Provider),由一个个频道(Channel)构成,频道的不同拼凑、频道自身的设定参数和不同的桌面方案为用户提供了个性化的体验。针对不同类型的资源,开发不同的 Portlet 并部署在 Portlet 容器中实现通过门户对资源的集成和访问。图 2 是基于门户技术的资源整合的框架实现示意图。整个框架分为界面和服务两个部分。门户的界面部分由各个预先构造的 Portlet 组成。引入了 Portlet 的门户页面类似于 PC 的桌面,而 Portlet 则类似于窗口。每个 Portlet 占用浏览器的一块区域显示信息,用户与具体的 Portlet 交互,Portlet 的显示内容由 Portlet 的内部逻辑所决定。门户的功能设计从而转化为设计实现单独的 Portlet,用户在定制自己的显示页面时,从注册的 Portlet 选择可用且有权限访问的 Portlet。

Portlet 可以由 IT 部门人员来构建,也可以由系统集成商、独立软件开发商来构建。一旦开发完成一个 Portlet 并通过测试,就可以将其注册入门户,最终用户可以选择一个他需要的 Portlet 并将其置入其门户页面。Portlet 的制作除了一些新的 Portlet 以外,还有就是为已有 Web 应用程序或应用

程序定制可以在 Portal 中使用的新界面,从而将这些应用集

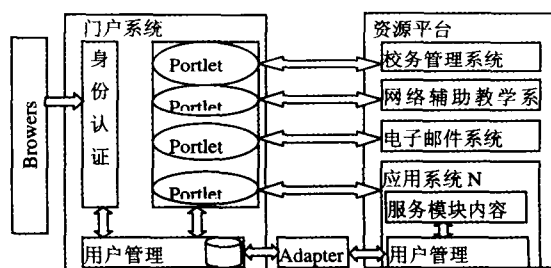


图 2 基于门户技术的资源整合示意图

成到校园信息门户中,这样的 Portlet 通常包括以下的一些应用程序,如图 3 所示。

不同的资源或系统与门户系统进行集成的方式和程度是不同的。

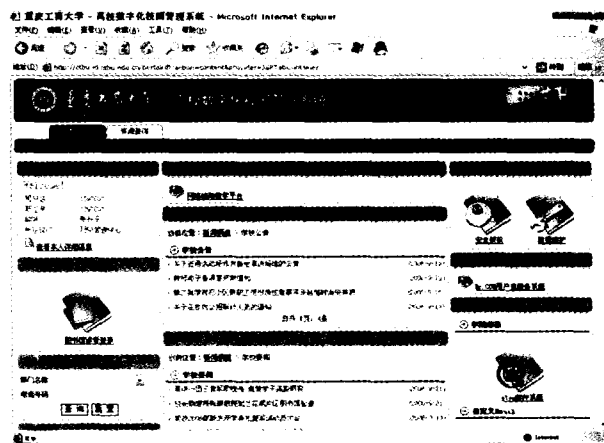


图 3 门户集成应用

1)对于采用 Web 服务器或 J2EE 标准的认证和授权方式的 B/S 系统,门户服务器可为其提供单点登录和策略控制的集成。

2)对于含有自定义认证和授权方式的 B/S 应用系统,门户服务器可在桌面上放置链接 URL 或者通过建立用户映射机制提供单点登录,被集成的系统相应地需要进行用户登录方面的修改。

3)对于采用 LDAP 标准方式存放用户数据的应用系统,门户服务器可以和他们进行单一的用户存储,达到统一认证的集成程度。

4)对于传统的 C/S 应用系统,门户服务器只能通过 Netlet 为应用系统提供安全的远程访问。

3.2 Portlet 类结构设计

为了实现一个 Portlet,需要相应类的设计,具体功能如下说明:

1)Portlet;定义为接口,表示放置在门户页面中的窗口,getContent()输出此 Portlet 的内容的 HTML 片段。

2)PortletSet;定义为接口,表示 Portlet 的集合,以集合为单位操作 Portlet。

3)PortletState;定义为类,表示 Portlet 的工作状态。

4)PortletInstance;定义为类,保存 Portlet 的持久化信息,包括门户结构描述文件中与此 Portlet 相关的内容,在页面中的布局、与当前用户相关的配置!该 Portlet 的描述信息。

5) CIPPortletInstance: 定义为类, 作为每个 Portlet 实现的公共父类, 定义了配置信息类的存储变量。

6) BasePortletSet: 定义为 PortletSet 的缺省实现。

7) PortletConfig: 定义为接口, 包含操作与该 Portlet 相关的所有配置信息的方法。

8) BasePortletConfig: PortletConfig 的缺省实现, 包含与该 Portlet 相关的所有配置信息, 包括布局、URL 信息、描述信息、布局信息、位置信息、安全信息等。

9) PortletSkin: Portlet 的色彩配置信息。

10) PortletControl: 对 Portlet 和 PortletSet 的装饰信息, 同时也管理 PortletSet 内的 Portlet 在页面中的布局。

11) PortletController: 负责布局和输出整个 PortletSet 的内容, 包括集合内的所有 Portlet 的输出。

3.3 信息门户框架处理 HTTP

客户端和 Portlets 的互动是由 Portal 通过典型的 Request/Response 方式实现, 正常来说, 客户会和 Portlets 所产生的内容互动, 举例来说, 根据下一步的连接或者是确认送出的表单, 结果 Portal 将会接收到 Portlet 的动作, 将这个处理状况转向到目标 Portlet。这些 Portlet 内容的产生可能会因为不同的使用者而有不同的变化, 完全是根据客户对于这个 Portlet 的设置。

用户在访问门户时, 通过浏览器发送 HTTP 请求, Web 服务器将收到的请求传递给 Servlet 容器, Servlet 容器调用门户引擎(门户服务器 Servlet), 门户引擎将请求转换为 Portlet 请求, 依次调用 Portlet 容器, 根据需要经过多次调用后将多个 Portlet 的输出合成起来生成结果页面(见图 4)。具体描述如下:

1) 对每一次 HTTP 请求, Portlet 容器根据传入的 ID 以及内部 PSML 配置确定要调用的 Portlet, 根据 HttpRequest 设定 PortletRequest 作为参数, PortletRequest 包含了 Portlet 运行所需要的用户信息、用户的属性、用户的类型、HTML-FormParameter 等。

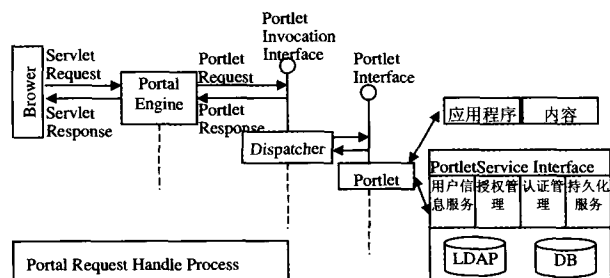


图 4 校园门户框架处理 HTTP 请求的流程

2) Portlet 的装入和实例化。Portlet 容器负责 Portlet 的装入和实例化, 这种实例化和装入可以在引擎启动时进行, 也可以延迟到该 Portlet 首次提供服务时才被实例化及装入。

如果 Portlet 尚未实例化, Portlet 容器首先必须查找定位要装载的 Portlet 类, 如果需要的话, Portlet 容器要使用通常的 Java 类装载方法从本地文件、远程文件或别的网络服务中装入相应的 Java 类, Portlet 容器实例化所装入的类。如果需要使用不同的参数实例化 Portlet 类, 则容器中存在同一 Portlet 类的多个实例。

首次装入的 Portlet, 在可以提供服务之前, 还需进行初始化。通常, 这一初始化过程包括读取持久化的信息、初始化高代价的资源(如 JDBC 连接)、以及一些只需执行一次的操作。这一过程由容器调用 init(PortletConfig)方法来完成, 实现了

PortletConfig 接口的配置对象作为方法的参数。配置对象包含用于访问由容器的配置信息提供的始化参数名称-值对, 同时配置对象也提供描述了 Portlet 运行环境的对象的访问, 这一对象实现了 PortletContext 接口。只有 Portlet 的 init 方法成功运行后, 此 Portlet 才处于就绪状态, 可以处理请求。

Portlet 经过适当的初始化之后, 就可以处理 Portlet 容器传递过来的请求。Portlet 用于响应请求的方法为 Service, 其参数包括, 类型为 PortletRequest 的对象代表请求, PortletResponse 代表响应。此方法检测请求的模式, 然后将请求分发至 doView、doEdit、doHelp、doConfig 等方法进行相应的处理。

- 多线程问题: Portlet 响应请求的时候, 容器可能会将来自多个客户的请求在任何时间发送给 Portlet, 因此 Portlet 开发者必须在编程时考虑到并行性。如果在将 Service 方法定义为 synchronized 的, 则 Portlet 容器以及之下的 Java 运行环境, 将对 Service 方法的调用串行化以保证同步操作。但鉴于 synchronized 操作会极大地影响程序的运行效率, 因此, 对于 service、doView、doEdit、doConfig、doHelp 方法推荐不要定义为 synchronized 操作。

- 例外处理: PortletException 用来表示在处理请求的过程中发生了错误, 容器应该采取适当的措施清理请求。UnavailableException 表明 Portlet 暂时或永久性的无法处理请求。如果发生永久性的不可用, 容器将 Portlet 从服务中移出, 调用其 destroy 方法, 然后释放 Portlet 实例。如果发生暂时性不可用, 容器可以在此期间不将请求发送到该 Portlet。

- 线程安全性: 请求和响应对象的实现并不是线程安全的, 因此只能在请求处理线程中使用, 如果在其他线程中访问这两个对象, 将会导致不可预知的结果。

3.4 结束服务

Portlet 容器不必一直保持 Portlet 是装载了的, Portlet 在容器的生存期间可以生存任何时间长度。当容器确定某个 Portlet 应该被移出容器, 停止提供服务, 例如, 容器要减少内存用量或是容器本身被关闭, 此时, 容器必须允许 Portlet 释放其使用的资源并持久化其状态。这一过程通过调用 Portlet 的 destroy 方法来实现。在容器可以调用 destroy(PortletConfig)方法之前, 必须让所有运行在 Portlet 的 Service 方法的线程运行结束或超过了服务器定义的时间限制。

一旦 FPortlet 的 destroy 方法被调用, 容器将不再将请求分发到该 Portlet 实例。若容器需要再次使 Portlet 有效, 则需要重新生成 Portlet 的一个新的实例。当 destroy 方法结束后, 容器就可以释放 Portlet 实例, 使之被回收。

小结 门户服务的提供是传统校园网络所不具备的, 而且从某种程度上可以说是数字化校园的高级表现形式。通过 Portlet 技术构建的校园信息门户, 可以集成已有的信息系统和 Web 应用程序, 同时, 通过可订制的门户渠道传递继承的内容、应用程序和服务, 降低了门户实现和管理的成本, 提高了工作效率, 在数字化校园的建设中将会产生较大的作用。

参考文献

- 1 许鑫, 苏新宇. 高校统一门户的设计与实现[J]. 现代图书情报技术, 2005, 5: 57~63
- 2 美国太阳微系统公司. Sun ONE Portal Server 开发指南[M]. 北京: 机械工业出版社, 2003. 1~23
- 3 美国太阳微系统公司. Sun ONE Application Server 开发指南[M]. 北京: 机械工业出版社, 2003. 2~25
- 4 美国太阳微系统公司. Sun ONE Studio Java 技术教程[M]. 北京: 机械工业出版社, 2003