

Web 内容管理系统中的可复用展示模型

阮彤¹, 高小伍², 宋韬³

(1. 华东理工大学计算机系, 上海 200237; 2. 华东理工大学网络中心, 上海 200237; 3. 上海神计信息系统工程公司, 上海 200040)

摘要:随着 Internet 的普及, 基于内容的运营成为 Web 内容管理的重要应用领域之一, 网站在运行过程中需要频繁更改页面和内容, 其维护成本较高。针对该问题提出基于组件的内容展示模型。在该模型中, 不同类型的内容管理应用、同一应用的不同页面或同一页面的不同内容区域可以对组件进行复用。基于该模型实现通用内容管理框架(UCMP)系统的界面展示部件。UCMP 系统使网站业务人员可以通过配置方式快速部署各类 Web 网站。

关键词: Web 内容管理; 动态内容; 组件; 可扩展标记语言

Reusable Representation Model in Web Content Management System

RUAN Tong¹, GAO Xiao-wu², SONG Tao³

(1. Department of Computer, East China University of Science and Technology, Shanghai 200237;

2. Network Center, East China University of Science and Technology, Shanghai 200237;

3. Shanghai Shenji Information System Co. Ltd., Shanghai 200040)

【Abstract】 With the permeation of Internet, operation based on content becomes a key application area of Web content management. The page and content of Web sites need to be changed frequently, and its maintenance cost is high. Aiming at this problem, a component-based content representation model is proposed. Component can be reused in different types of content management applications, different pages of the same application, and different areas in the same page in this model. Interface display components of Universal Content Management Platform(UCMP) are realized according to this model. UCMP makes servicers be able to deploy different kinds of Web sites fastly through configuration manner.

【Key words】 Web content management; dynamic content; component; eXtensible Markup Language(XML)

1 概述

Web 内容管理系统被广泛应用, 其中, 最简单是基于 PHP+MYSQL 构造的网站生成工具, 用于发布新闻或企业信息, 文献[1]描述了若干种此类平台。与新闻发布平台相比, E-commerce 购物和视频点播等网站较复杂, 文献[2]将此类平台视为数据集中的动态网站加以生成。作为企业内容管理(Enterprise Content Management, ECM)组成部分的 Web 内容管理^[3]最复杂, 它与业务流程工具、数据建模工具等存在密切关系。一般而言, Web 内容管理引擎核心包括数据管理、展示管理和页面加速功能, 外围功能包括发布流程管理、推荐管理等。

就展现管理而言, 现有 Web 内容管理系统遵循的最基本原则是基于数据和展现的分离实现数据复用。由于 Web 展现技术越来越复杂, 因此很多页面效果依赖 JavaScript 和层叠样式表(Cascading Style Sheets, CSS)技巧。而拥有此类技巧的通常是美工或程序员, 而非业务人员。另外, 增加页面的过程需要复杂编程, 即增加了网站运营成本。因此, 对于 Web 内容管理而言, 展现层的复用具有重要意义。

现有 Web 内容管理采用的 2 种展现方法如下:

(1)XML+ XSL(文献[4]采用此方法)。

(2)使用占位符。在模版技术实现过程中, 不同的内容管理系统存在差别, 例如, 织梦管理系统直接把含有语义的内容标签放在模版页面中, 把需要放置内容或相关链接的地方用占位符代替, 系统在运行过程中解释这些占位符。

上述方法存在以下缺点: (1)界面可复用性低, 页面的复

用粒度在模版或页面上界别; (2)界面与数据(数据类型)的关系固定, 数据通过嵌入在 HTML 中的标签固化在界面上或通过 XSL 写明。界面模型依赖数据模型, 如果数据模型发生变化, 则界面模型必须做相应改变。

鉴于此, 本文提出一种基于组件的动态数据绑定模型, 该模型具有以下特点:

(1)一个页面不是一个完整的 HTML, 而是由元素构成, 这些元素称为页面组件。页面组件可以被自由组合, 形成一个完整的页面。

(2)页面的数据来源不是硬编码在页面中的数据, 而是通过配置方法获得。

本文在通用内容管理框架(Universal Content Management Platform, UCMP)系统中使用该模型。在 UCMP 中, 业务人员无须编程或编写 HTML, 可以通过定义内容类型、定义页面、拖拽组件等步骤定义或修改一个网站。

2 模型定义

在基于组件的动态数据绑定模型中, 一个页面可以分成几个区域, 且区域可以包含子区域。如图 1 所示, 一个页面

基金项目: 国家“863”计划基金资助项目“以用户为中心的服务门户支撑技术研究”(2006AA01Z161); 上海市科委人才计划基金资助项目“新一代卫星导航运营系统框架与模型的研究”(07XD14217)

作者简介: 阮彤(1973-), 女, 副教授、博士, 主研方向: 中间件, 内容管理与软件工程; 高小伍, 讲师、硕士; 宋韬, 高级工程师、博士

收稿日期: 2009-05-16 **E-mail:** ruantong@ecust.edu.cn

由 2 个区域 D1, D2 构成, 而 D2 包含 D3 和 D4, D3 又包含 D5 和 D6。区域和区域之间的关系可以是包含或重叠。重叠时可以区分叠放次序, 用于布局的页面称为底板。

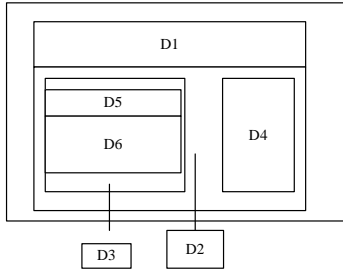


图 1 区域定义

定义 1 一个底板由一个或多个区域构成, 区域可以包含区域。

从实现角度而言, 区域可以是一个 HTML DIV, 而底板是一个包含多个区域的 HTML 页面。

每个区域可以有一个或多个组件。组件用于展示内容, 如一条新闻、新闻列表或一个导航栏目, 如图 2 所示。



图 2 页面组件

定义 2 每个区域可以不包含组件, 也可以包含一个或多个组件, 组件不允许包含组件。

为实现界面和数据的分离, 开发组件时, 不直接指定数据类型, 而是在部署站点的定义页面过程中指定。

定义 3 组件在部署期间指定展现的数据类型和数据源的过程称为数据绑定, 包括数据类型绑定和数据源绑定。

数据类型绑定表示相同界面展示的数据类型可以不同。常见排行组件如图 3 所示, 同一个界面展现的可能是文章标题、小说名称或下载歌曲的名称。

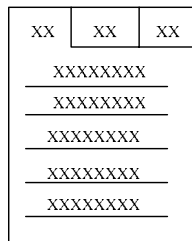


图 3 常见排行组件

实现数据绑定的主要难点如下：

(1)复杂数据与复杂界面的绑定。为了支持各种类型的内容管理应用, UCMP 后台系统支持面向对象的数据类型定义, 用户可以随意定义各类复合数据类型, 数据类型可以有包含、继承关系。上述关联关系可以在系统中加以描述。对一个界面而言, 某些元素展现的是顶层对象的属性, 而有些元素展现的可能是它包含的对象。

(2)链接与链接参数值的动态生成。所有页面都有一个唯一的 ID, 系统可以自动通过该 ID 找到特定页面。此方法的

缺点是在制作页面时, 需要假定已经存在相关页面 ID。与链接相关的页面之间需要进行参数传递, 参数个数和名称在制作页面模版时是固定的, 但数值通常是动态生成的。

(3)数据的读取。普通的内容管理应用通常由页面直接读取数据, 而在本系统中, 组件负责页面的展示, 而数据由框架负责读取, 并通过页面上下文传递给组件。系统需要提供多种方法, 允许页面/组件方便地接收各种查询数据。

3 关键技术

在本文的界面模型中, 组件可以和任何一种数据类型绑定。界面与后台数据类型的独立通过系统要求定义一个基于 XML 的中间格式完成。

定义 4 一个组件包含界面和对数据的引用 2 个部分。一个组件的实现包含一个页面文件与一个数据格式文件。

页面文件在本系统中通常由 jsp 实现, 一个简单的组件页面文件如图 4 所示。

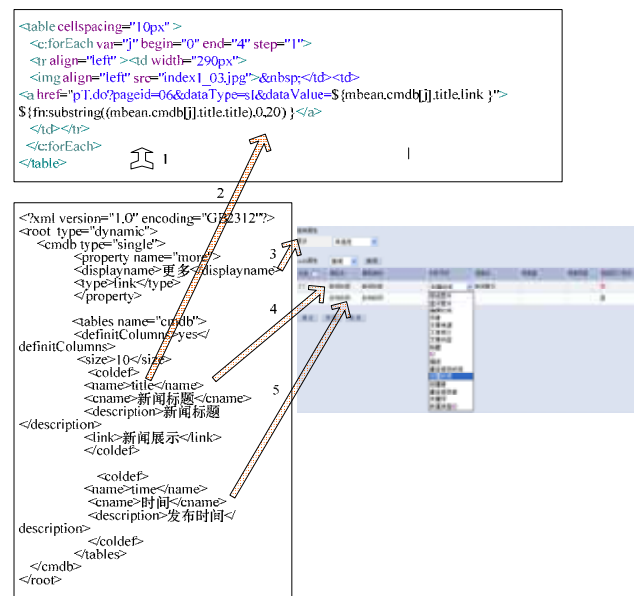


图 4 组件页面文件

在图 4 中, 上端描述的是组件页面文件, 左下部分描述了数据对象的格式——组件 XML 文件, 右下部分是系统根据组件 XML 文件自动生成的配置界面。箭头 1 所指的 mbean.jsp 页面需要展示的数据对象通过 mbean 传递过来, 数据可以只包含一个简单的变量, 也可以包含数组等其他数据。在本系统中, 前者用 <property> 标签表示, 后者用 <tables name="cmdb"> 标签表示。当组件的 XML 定义被上传到系统中时, 系统可以根据 XML 文件自动生成配置界面。需要注意 XML 文件与配置界面、jsp 页面的对应关系。例如, 在 XML 文件中, 有属性名为“更多”、“新闻标题”和“发布时间”的 3 个字段, 因此, 在配置界面中, 会自动产生对这 3 个字段的配置字段(如箭头 3~箭头 5 所指)。业务人员可以在系统中选择需要的数据类型(在图 4 中是“新闻”类型), 配置界面下拉框内列出了所有类型包含的所有字段, 业务人员可以选择需要展示的类型和相应的字段。

链接是一个页面重要的组成部分, 它体现了页面和页面之间的关系, 此类关系在本系统中被细化为组件和页面之间的关系以及字段和页面之间的关系。在组件.xml 的定义中, 每个字段都可以有一个 <link> 标签, 说明需要定义该字段对应的链接, 在系统配置界面中允许业务人员配置该界面对应的

的连接。

页面布局问题即如何把组件放到页面中的合适位置。如上所述,HTML的DIV可以定位一个对象,这种定位可以是相对的或绝对的。DIV已被越来越多地用于定位,因此,组件被打包成一个DIV,放到页面的一个区域(一个页面)中。组件和组件之间可以是重叠、并列、前后等各种关系,一个区域可以有多个组件。为了避免用户手写HTML,页面布局采用所见即所得的方式,通过拖拽完成布局。

4 模型实现

系统部署时内置3套界面,每套包含约10个组件、5个色系、3个底板、一个新闻内容类型和一个电影类型。

在UCMP系统中,一个站点的构造流程如下:

(1)类型定义。如电视连续剧包含连续剧名称、制片人等字段以及电视剧的每一集。对于电视剧每一集是否有自己的类型,根据文件是否有独立的属性而定,例如,如果每一集需要一个特定的DRM文件,则需要构造类型。

(2)数据录入。定义数据类型后,系统将自动生成对于该类型的数据增删改界面。此时,用户利用该界面录入部分数据以方便页面布局和测试。

(3)页面定义。具体如下:

1)选择页面底板。可以选择空白底板。

2)数据类型绑定。选择组件,对于每个组件,选择后台需要绑定的数据类型,如电影、新闻等,已经绑定类型的组件称为组件实例。系统提供界面,可以预览组件实例。

3)组件布局。将组件实例拖拽到底板上。

4)数据绑定。对每个组件实例,选择需要绑定的数据。数据可以有多个来源:某个频道(代表人为的、静态的分类),如韩剧、港台剧等;某个查询,内容管理系统提供一套类似于SQL的查询语言,称为UQL;系统接口,如访问最多的内容对象,可以通过系统接口获取。

整个系统基于J2EE平台,使用Spring+Hibernate框架,页面缓存使用Spring AOP技术,访问页面的各个序列如下:

(1)PageTransferServlet接收请求。页面请求格式如下:

服务器地址+pagetransfer.do+页面号+页面其他参数

(2)PageTransferServlet察看缓存里面是否有数据。如果有,则转(5),否则转(3)。

(3)PageTransferServlet读取页面定义和组件定义。

(4)PageTransferServlet通过DataReader从后台存储库中读取数据,pagetransfer把数据放入request的上下文中,并将请求转发到PageAssembler中。

(5)PageAssembler将request中的数据与组件页面文件绑定,形成一个页面。

(6)页面被返回到客户端。

5 模型的特殊情况

上述模型利用一个中间XML文件实现动态数据类型的绑定,使组件得到最大程度的复用。其缺点是组件开发人员需要定义额外的XML数据类型文件,业务人员需要定义XML到特定类型的映射。组件应用场合可能是数据类型相同而数据源不同,例如,固定某个组件展示电影类型,但可以展示不同频道的电影。因此,提出“参数化组件”的概念,在页面定义过程中,业务人员可以直接将“参数化组件”拖入页面中。

定义5 不依赖XML,直接指明了组件需要访问的数据类型的组件称为参数化组件。

类型的定义依赖内容管理系统的后台存储库。参数化组件的实现方法分为2种:(1)将数据类型定义放在XML定义文件内。如<root type="parameterized" value="movie"/></root>。为方便组件开发人员,开发人员可以不定XML,在页面文件中加入类型定义。系统支持<UCMP DataType="movie" datasource="?" sourcevalue="?" />等标签,在数据源定义时,系统会解析这些标签,并将合适的配置页面展现给业务人员。

页面的头、尾等部分通常是一些静态的HTML,并不读写内容管理后台系统的数据。

定义6 一段静态的HTML称为平凡组件。

无论参数化组件或平凡组件,从实现角度来说,都可以不定XML文件,也无需参数化过程。

在一些情况下,页面上的数据并不直接和某个具体的内容绑定,而是与内容相关的其他页面绑定。例如在菜单组件中,每个菜单项的名称和指向的链接虽然和内容有关联,但不一定一一对应。因此,自动生成菜单项,不如业务人员在配置界面中手动输入方便。

定义7 不从后台内容管理系统读取数据的组件称为静态组件。

静态组件有时用于用户自己扩充的业务,例如,一个机场的对外门户可能会展示天气预报信息,这些信息是从其他网站动态抓来的,并不通过后台内容存储库。

6 性能分析与解决方案

基于组件的动态数据绑定模型的优点是可复用性和易管理性,但使用组件模型后,数据要通过一次转换才能展示到Web界面上。另外,由于UCMP平台后台是一个可扩充的内容管理库,用户可以定制内容库的数据结构,数据结构支持面向对象的继承和包含,整个后台类似于一个动态Hibernate的引擎。上述2个问题会导致系统性能降低。在UCMP内容管理系统中,一个页面展示流程包括如下步骤:

(1)组件定义读取。

(2)后台数据读取,具体如下:

1)查询转换。系统提供一种称为UQL的面向对象查询语言,需要将UQL转换成SQL才能在实际数据库上执行。

2)数据拼装。将数据库结果转换成对象。

(3)数据转换成组件能够访问的格式。

(4)组件拼装成页面。

与一般使用模版方式的内容管理应用相比,由于多了第(3)步和第(4)步(都是内存操作),因此速度可能比一般的内容管理应用慢。而对于第(2)步,由于支持了继承和包含等一般内容管理系统不支持的操作,因此在数据较复杂、并发量较大的情况下,速度会慢很多。在无锡电信门户网站上线过程中笔者发现,该应用的数据类型较复杂,在不采用缓存等其他方法的前提下,主页的显示速度较慢。

Web内容管理应用与一般交互式应用有着本质区别。在交互式应用(如网上银行)中,用户需要实时、准确地获得各种个性化信息。而在内容管理应用中,所有用户通常共享相同的内容页面,静态性较强。因此,多数Web内容管理应用都采用预先生成静态页面的方法。简单而言,就是利用穷举算法,输入是内容管理库里的内容与某个特定动态页面,输出是一组静态HTML页面。上述处理可以提高系统性能,且内容能被分发到不同的边界服务器上。对于大型内容运营商而言,此类服务器分布在电信网络内或教育网内,保证了不

同网段内不同客户端的高速体验。

表 1 对内容管理应用与性能相关的业务特点做了分析,并给出 UCMP 内容管理系统的解决方案,具体如下:

(1)个性化需求。在内容管理系统中,用户的多数页面均相似,但也存在一些页面有个性化需求。例如,在无锡电信视频点播网站中,要求每个用户登录后,系统展示用户名、口令以及充值卡内的信息。这些信息与网站视频内容列表一起放在主页上。因此,本文采用基于 AJAX 技术的用户管理组件。

(2)大并发访问量。采用静态页面生成技术并提供页面缓存技术。前者并发性更高,后者管理更方便。所有访问过的页面根据请求参数的不同,均放到缓存当中。系统采用基于 Java 的 OSCache 软件,OSCache 允许将缓存内容放到内存或是硬盘上,因此,缓存内容的数量可以很大。UCMP 提供缓存刷新管理界面,支持管理员手动或自动刷新缓存。

(3)实时数据更新。在多数情况下,内容管理应用不需要实时更新。例如发布新闻时,Web 界面上可以等 30 s 或更长时间后再显示,这是此类应用可以容忍的。在无锡机场项目实施中,用户选择手动更新和自动更新相结合的方法。

(4)数据结构变化。对于一般的交互式应用而言,数据库的结构通常是固定的。在 UCMP 内容管理系统中,由于有一层内容存储中间层,因此允许客户增加新的数据类型。

表 1 内容管理应用与性能相关的业务特点

性能指标	普通 Web 应用	内容管理系统	UCMP 内容管理系统的实现
个性化	多	少	采用基于 AJAX 的组件,可以实现同样的 HTML 页面,不同用户展现不同内容
并发访问量	大	很大	采用静态页面生成技术,采用缓存技术
实时数据更新	多	实时少,多数准实时	设置缓存更新时间,采用 AJAX 技术
数据库数据结构变化	基本没有	较少	UCMP 支持动态数据添加,过程类似于创建一个网站

(上接第 18 页)

$$Adv_{TLPMAC}^{prf}(A) \quad Adv_{PMAC}^{prf}(D) + ((p-q)q + 2q + \binom{q}{2}) \cdot 2^{-n}$$

复杂性理论下的安全性结论基于 PMAC 在伪随机函数意义下的安全性。PMAC 的安全性在文献[4]中已经证明。TLPMAC 是一个伪随机函数,因此,具有不可伪造性。

5 结束语

TLPMAC 具有如下优点:(1)双层可并行运算结构;(2)计算某字符串向量的运算量与单独计算各分量运算量的总和相当;(3)当某字符串分量不变时,可以进行预处理,使算法在处理某向量时的计算量低于单独处理该向量各分量时的复杂度的总和;(4)处理一个向量所需调用分组密码的次数和单独处理每个分量所需调用分组密码的次数总和相等;(5)具有安全性,TLPMAC 的安全性基于 PMAC 的安全性,而普通消息认证码的安全性基于所用分组密码的安全性,由于 PMAC 的安全性基于分组密码的安全性,因此两者的实质相同。TLPMAC 可以对一组字符串进行认证,能作为输入为字符串向量的伪随机函数。

参考文献

[1] 吴文玲,冯登国. 分组密码工作模式的研究现状[J]. 计算机学报, 2006, 29(1): 21-36.

在无锡电信阿福台视频点播应用中,笔者通过 2 台普通的 PC 服务器、一层中间层、一台数据库支持无锡市所有宽带用户,目前用户数量达几十万。由于使用了缓存技术,因此中间层和数据库的 CPU 使用率都很低。

7 结束语

利用基于组件的动态数据绑定模型和后台可定制的类型系统,网站业务人员不必具有数据库和 HTML/CSS 等技能就能快速定制各种类型的网站。基于 UCMP,笔者上线了无锡电信公司的视频点播系统和无锡机场的对外门户网站,实践结果表明:(1)在用户掌握了系统的基本概念和操作后,模型给用户业务扩充带来了方便,当视频点播系统需要增加一个频道子页或在某个页面上增加一些推荐信息时,可以通过拖拽来完成。(2)不同类型的申请可能有相同组件或不同组件,如视频点播应用需要特殊的播放器组件。(3)与内容发布无关、但业务需要的内容应使用静态组件。(4)对过于复杂且区域重复性较小、页面效果要求较高的页面,应使用参数化组件。

参考文献

[1] Open Source CMS Web Site[Z]. [2009-01-03]. <http://www.opensourcecms.com/>.

[2] Schwarz M J H. Modeling and Generating Application Logic for Data-intensive Web Applications[C]//Proc. of ICWE'06. Palo Alto, California, USA: [s. n.], 2006: 77-84.

[3] IBM FileNet Site[Z]. [2009-01-03]. ftp://ftp.software.ibm.com/software/data/ECM/Bro/IBM_ECM_Site_Publisher_FED_DS.pdf.

[4] Meliksetian D, Dean S E, Weitzman L. Transforming the Content Management Process at IBM.com[C]//Proc. of the Conference on Human Factors in Computing Systems. Minneapolis, Minnesota, USA: [s. n.], 2002: 1-15.

编辑 陈 晖

[2] Sarkar P. Improving Upon the TET Mode of Operation[C]//Proc. of ICISC'07. Berlin, Germany: Springer-Verlag, 2007: 180-192.

[3] Bellare M, Canatti R, Krawczyk H. Keying Hash Function for Message Authentication[C]//Proc. of CRYPTO'96. Berlin, Germany: Springer-Verlag, 1996: 1-19.

[4] Black J, Rogaway P. A Block Cipher Mode of Operation for Parallelizable Message Authentication[C]//Proc. of EUROCRYPT'02. Berlin, Germany: Springer-Verlag, 2002: 384-397.

[5] Bellare M, Guerin R, Rogaway P. XOR MACs: New Methods for Message Authentication Using Finite Pseudorandom Functions[C]//Proc. of CRYPTO'95. Berlin, Germany: Springer-Verlag, 1995: 15-28.

[6] Bellare M, Kilian J, Rogaway P. The Security of the Cipher Block Chaining Message Authentication Code[J]. Journal of Computer and System Sciences, 2000, 61(3): 362-399.

[7] National Institute of Standards and Technology. Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode and GMAC[Z]. [2009-01-25]. http://csrc.nist.gov/publications/drafts/draft-SP-800-38D/Draft-NIST_SP800-38D_Public_Comment.pdf.

编辑 陈 晖