

图书馆之城技术平台ULAS研究

□ 蔡晖 / 深圳图书馆 深圳 518036

摘要: ULAS是深圳图书馆在深圳建设“图书馆之城”目标的大环境下,依据原有的dILAS系统应用经验和设计思路重新设计和创新的产物,是传统图书馆理论升级到Lib 3.0、服务核心完全确立下的产物。文章从6个方面将dILAS和ULAS进行纵向比较,并且针对国内其他图书馆自动化软件,介绍了其主要的功能特点和技术特点,同时介绍了研发中出现的问题及解决方法。

关键词: 图书馆之城, 图书馆自动化, ULAS, dILAS

DOI: 10.3772/j.issn.1673-2286.2010.06.007

引言

“东边借书西边看”已经日益成为现代都市人生活的迫切需求,图书馆理应为市民提供更优质的城市化统一服务。随着深圳图书馆之城项目的开展和实施以及Web 2.0理论研究的深入,深圳图书馆原有的dILAS系统已无法满足高速发展的业务需求。深圳图书馆针对新时代的新需求,主要采用B/S模式,吸取dILAS应用的成功经验,并拓展dILAS的设计思路,重新设计并完成了全新的ULAS(United Library Automation System)系统,目前已经完成流通业务系统的升级和改造,并取得了显著成果。

1 ULAS与dILAS的比较

ULAS与dILAS相比,虽然系统架构和代码几乎完全重写,但ULAS的设计思路仍然从dILAS完全继承并深化。因此,ULAS具有dILAS的一切功能,也延续了其在图书馆自动化和数字图书馆建设方面的优点。

1.1 定位上的不同

dILAS定位于实现图书馆传统业务的自动化和数字图书馆建设。而自从深圳图书馆在2006年全面开展“城市街区24小时自助图书馆”项目,深圳市也制定打造“图书馆之城”的宏伟目标后,dILAS已无法满

足日益增长的图书馆城市化需求。在新环境下,ULAS不仅仅着眼于为“城市街区24小时自助图书馆”项目提供坚实后盾,更考虑到深圳市图书馆与各级区馆的合并工作,为实现统一系统平台、统一服务规则、统一资源加工,真正实现“图书馆之城”的目标而努力奋斗。可以说,ULAS是为“图书馆之城”而生的。与dILAS突出数字化建设(Digital)不同,ULAS更加突出“U”(United)这个字母,即为实现各图书馆资源联合共建、联合服务。

美国霍普金斯大学的张甲教授在讲座“Future Library System”中曾指出,未来图书馆系统必须是以服务为主的图书馆系统设计、能进行统一资源的管理、多元化的元数据控制、全球性的书目数据制作共享以及可分散独立使用的系统功能,能在读者需要的时间和场所提供图书馆服务。ULAS秉承这一理念,不仅在传统图书馆自动化业务和数字图书馆构建方面完全继承了dILAS的优点,在服务深化方面更拓展到了更加广阔的领域。ULAS是建立在dILAS和Lib 3.0理论基础上的,为深圳打造“图书馆之城”服务的,拥有更加深化的服务模式的全新系统。

1.2 开发技术及维护方式不同

dILAS采用3层系统架构,服务器端程序主要采用C++编写,g++编译器部署,客户端主要采用Visual C++6.0编写C/S模式应用程序。在图书馆之城环境下,

各分馆间的通讯如果仍然采用C/S模式，就必然要求更高的网络通信质量。但限于分馆的人力财力，同时考虑“图书馆之城”以后规模扩大的预期，完全使用光纤构建整体的高速Intranet并不现实。因此，ULAS放弃了原dILAS相对复杂的3层架构，全面采用J2EE平台并主要进行B/S模式的开发。ULAS在慎重调研和考察的基础上，选择Spring+Hibernate+Ajax（jQuery）的轻量级开发框架构建系统。

dILAS的C/S模式导致了其维护升级方式的繁琐，每次应用程序的升级和更新都会导致用户的客户端程序的更新，但在“图书馆之城”中，这样的维护方式显然是不合理的。针对这一问题，除了不得使用C/S模式管理的电子阅览室功能模块外，ULAS的几乎所有业务都采用了瘦客户端的B/S方式，客户端仅需安装一些用于硬件设备控制的ActiveX控件，其他的功能都由服务器端实现，日常的维护工作无须对用户用机进行更新。ULAS采用J2EE框架，并针对应用功能划分不同维护模块，服务器端每个模块的更新只需要十几秒时间，并具备更新后自动通知使用者的能力。在深圳图书馆的应用实践中发现，系统升级和更新几乎不会对馆内正常业务的运行产生影响。

1.3 接口的开放方式不同

dILAS采用Open API的方式提供对其他系统的接口。但在图书馆之城平台下，不可避免地要引入多个第三方系统来弥补空缺。Open API在多次引入第三方系统后，已经暴露出其学习时间长、整合项目实施难度大、接口不规范、业务不完整等等缺陷。为此，ULAS采用了多种开放式交互协议提供接口，包括标准Web Service接口、LDAP接口、NCIP协议、SIP2协议等等。目前，深圳图书馆的ULAS已和易瑞数据库统一平台、无线网认证、存包柜管理、读秀系统等多个系统实现了无数据拷贝的系统整合。由于采用通用的标准接口，项目实施的难度大幅降低。因此，ULAS比dILAS具有更好的开放性。

1.4 功能上的不同

dILAS主要侧重于传统图书馆业务的自动化及数字图书馆的建设，而ULAS则秉承Lib 3.0概念，以服务为中心，不仅为“城市街区24小时自助图书馆”项目提

供了强有力的后台支撑，也在统一认证、统一检索、Web 2.0应用、短信服务、无线网服务、自助借还服务、参考咨询服务、预约送书服务等方面不断推陈出新，不断研究“图书馆之城”的业务运营模式，为打造“全市一个公共图书馆”的目标制定全新的服务规则和服务模式。可以说，dILAS还是基于传统图书馆学的“采编流刊检”一体自动化模式，而ULAS则将重点完全转移到服务上来，为全市“图书馆之城”的目标提供理论和技术支撑。

1.5 稳定性和排错性能不同

dILAS采用API方式对数据库进行操作，没有事务管理机制，因此数据不匹配的现象时有发生，如可能发生读者预付款缴纳欠款因网络原因造成预付款已扣而欠款未缴的情况。而图书馆之城如果搭建于Internet上，其网络稳定性无法保障，而且各分馆与中心图书馆的空间距离将使这一问题愈加突出。ULAS采用Spring框架的声明式事务管理方式，对每一个完整的事务操作一次提交，完全避免了数据不匹配现象的产生。Spring+Hibernate作为比较成熟的J2EE轻量级框架，其稳定性和高负载性能也非常突出。

ULAS在设计研发时考虑到大部分的网络不稳定、系统不稳定的情况，对服务器CPU、内存、存贮使用状况、服务运行状态、自助图书馆运行状态等系统状况进行实时监控，将产生的故障及时短信通知给相关人员。同时ULAS采用LOG4J日志开发模块进行日志管理，保证系统日志的安全性和规范性，也为系统故障迅速排查提供了有力的支持。

1.6 数据结构不同

虽然dILAS和ULAS都使用Oracle数据库，但dILAS于2006年初在深圳图书馆使用以来就暴露了其数据结构不甚合理的弊端。ULAS研发时总结经验教训，提出了数据结构的修改方案，主要包括：(1)dILAS的读者表采用XML元数据格式存贮，由于其格式比较固定，扩展性不大，产生了不少问题；ULAS考虑到Marc和DC并不适用于除了书目数据之外的业务数据存贮，将除了书目数据之外的数据表全部采用关系表处理，提高系统效率，简化应用程序；(2)dILAS的流通信息在馆藏库中记录，已经在使用中产生了许多问题；ULAS按

照面向对象的设计原则，将馆藏数据和流通数据完全分离，避免dILAS将馆藏和流通信息共用一表带来的种种弊端，为“图书馆之城”的多流通模式提供支撑；(3)dILAS仅有一个流通事务表，为了适应“图书馆之城”后的业务量激增的需求，将事务表按年进行分表处理，将较少使用的事务数据存入历史表备查，提高系统的整体效率。与dILAS不同，ULAS在数据结构设计上大幅度简化，将更多的应用直接交给Oracle处理，充分发挥Oracle和服务器的性能，而在图书馆业务应用不同于商业应用的地方则下足工夫，保证系统的稳定和高效。

2 ULAS的技术特点

J2EE架构的软件框架多如牛毛，那么，ULAS又是如何根据图书馆业务的特点选择合适的软件平台搭建系统，体现其技术优势的呢？

2.1 系统架构

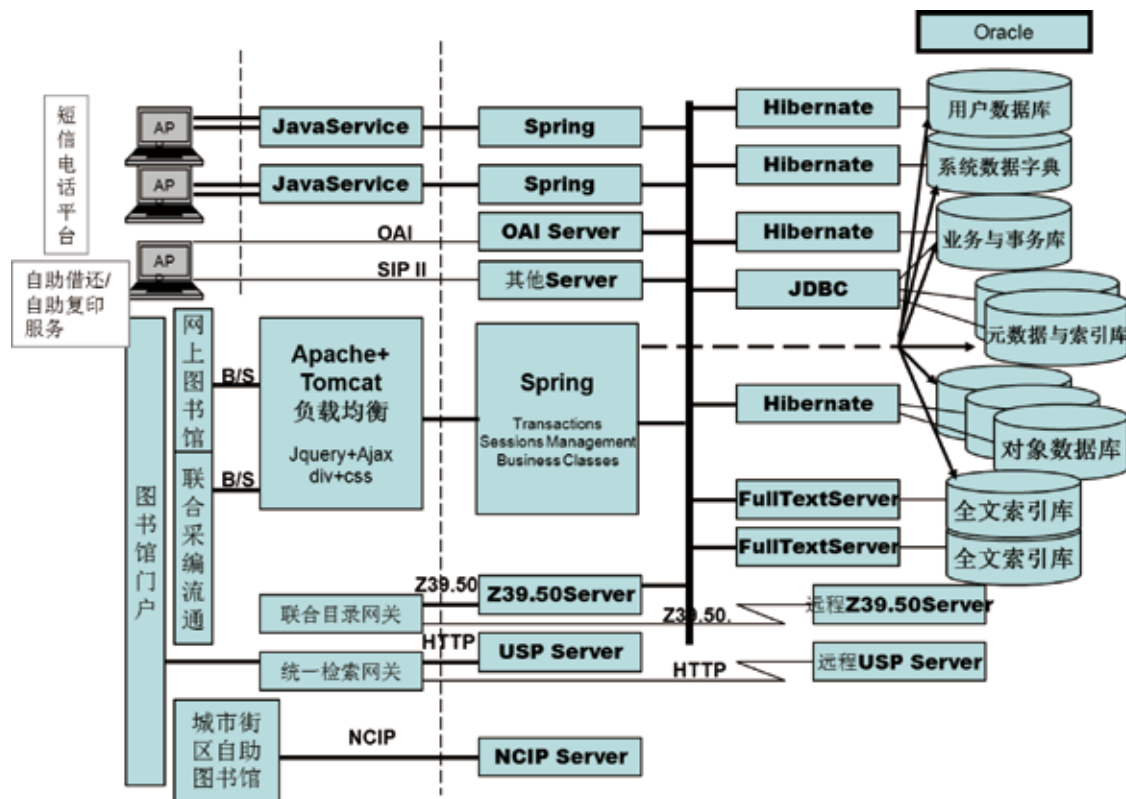
如前所述，总体采用B/S瘦客户端模式构建，充分

发挥Oracle的强大数据处理能力解决检索和日常业务应用需求。ULAS是采用Spring中间层思想、完全遵循Oracle数据设计思路的，能充分发掘服务器资源潜力的系统。

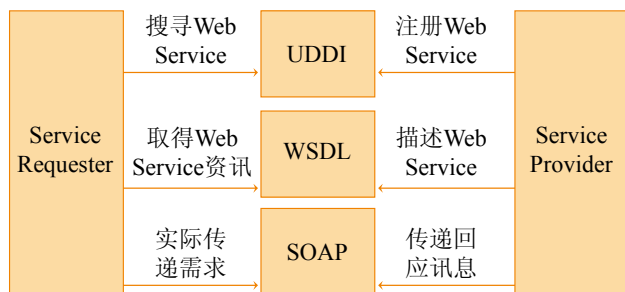
图书馆的主要的实体比较简单，为馆藏、读者、馆员3者。因此采用了Oracle+Spring+Hibernate的经典轻量级框架，继承其开发周期短、效率高、代码健壮、维护方便等优点。

2.2 Web Service技术

在当前图书馆城市化趋势已成必然的背景下，如何在图书馆间部署应用进行统一服务成为图书馆系统急需解决的问题。在J2EE平台上，实现这一目标的技术有EJB和Web Service两种，长远看来，采用分布式应用的EJB优势更加明显，无论在应用承载量和故障修复等方面都有集中式管理的Web Service无法比拟的优势。但图书馆毕竟并非企业，现阶段城市间的各图书馆也不是真正意义上的行政联合体，仅仅是希望进行统一服务的松散联盟，各图书馆的人才资源和设备资源也参差不齐并且难以调剂余缺。这种情况下，基



于大型企业级应用设计的EJB在实际应用中就会障碍重重。因此，在经过审慎调研后，ULAS采用集中式管理的Web Service作为图书馆之城的应用部署方案，即JavaBean（业务处理部分）统一集中在中心馆并由中心馆统一管理，各分馆的应用请求通过Web Service接口向中心馆提交应用请求，同时自行设计并维护其应用界面及服务器设备。这样就在图书馆之城中实现了MVC（Model、View、Controller）模型，通过作为Controller的Web Service将业务应用（Model）和界面UI（View）完全分离。



Web Service同时也是ULAS的与第三方系统交互的窗口。除了提供图书馆学标准的接口协议以外，Web Service是ULAS的标准二次开发接口。Web Service不仅是图书馆之城各分馆之间的信使，也是ULAS与图书馆已存在的系统以及购置的资源服务系统等异构系统之间的翻译家。

2.3 ActiveX控件技术

B/S模式目前还无法完全取代C/S的地位，主要在于B/S方式下浏览器的安全机制更高，如Javascript无法直接操作硬件、无法读取本地文件、大文件上传无断点续传机制等等。而解决这些问题的唯一途径就是ActiveX控件。ULAS采用证书服务技术管理众多的ActiveX控件，用户只需要在证书服务器上下载相应的证书即可完全享受ULAS的ActiveX控件的自动安装、自动升级服务。

实际上，ActiveX控件技术能解决更多的问题，dILAS的Marc和XML编辑器技术可以作为控件完全嵌入ULAS的浏览器中，ilas和dILAS的用户无须重新学习新的Marc和XML编辑方式。

3 ULAS的功能特点

ULAS是基于Lib 3.0理念和SOA模型的，以服务为中心构建的图书馆之城应用系统。相比传统概念上的图书馆，“图书馆之城”的各级图书馆在资源和服务上都做到了整合和统一，实体信息资源量、内需和外需等方面都有了质的飞跃，与此同时，读者只需要一张读者证即可享受全城的图书资源和服务，不同于馆际互借，真正的大流通将给读者带来更加贴心的服务和感受。因此，相比国内其他同类的图书馆自动化软件，她拥有众多的为适应新型服务模式而开发的新功能。

3.1 ULAS提供对“城市街区24小时自助图书馆”的全面支撑

城市街区自助图书馆是深圳市建设“图书馆之城（2006-2010）五年规划”重点建设项目，目前深圳市已经完成布点40台，2010年预计布点达到200多台。不同于ATM式的自助借还机，自助图书馆是真正意义上的图书馆，不仅可以完成办证、查书、借还书等基本功能，还是深圳图书馆的预约服务实体，读者可以选择将预约的图书送至最近的自助图书馆，以方便取书。自助图书馆的每一个请求（包括办证、借还书等）都通过NCIP协议实时发送至ULAS，由其进行统一处理。

ULAS不仅对“城市街区24小时自助图书馆”提供全面的业务支持，同时与城市街区24小时自助图书馆中心管理系统实现无缝衔接，提供对自助图书馆的24小时运行状态监控，其故障信息和预警信息正是通过ULAS的短信平台发送到工作人员的手机中。

3.2 ULAS全部采用B/S模式，是“图书馆之城”建设的技术平台

在“图书馆之城”中，分馆只需登录ULAS的网站即可进行办证、借还书、交押金等传统图书馆“采编流刊检”的日常业务，其功能强大的ActiveX控件库提供对多种外设，包括各种型号的打印机、RFID设备、身份证阅读设备等的支持。瘦客户端的管理模式无须专人进行软件升级和更新维护，不仅大幅度降低了软件维护成本，相对于C/S模式，B/S模式也能更好地适应Internet的网络环境，提供更稳定的服务。

同国内其他主要使用B/S方式开发的图书馆软件相比, ULAS在硬件设备的支持上更加完善, 丰富的ActiveX控件使得ULAS能完成C/S方式能做到的所有功能。同时, J2EE架构提供多级分布式管理模式, 尽管采用开发维护成本较低、学习门槛较低的轻量级框架, ULAS仍然可以借助Apache和Tomcat的现有技术搭建多级分布式的服务集群。

3.3 ULAS集成对RFID、智能书车、自助借还机等多项新技术的支持

ULAS提供对多种技术的支持和接口, 提供对书商、其他图书馆进行数据交换。这些接口包括短信服务技术、电话语音服务技术、流媒体技术、打印控制、自助借还技术、身份证阅读器技术、自助缴款技术等。

在新技术应用中, ULAS更加从人性化、自动化角度出发, 尽量减少工作人员的工作量, 如图书借还时, RFID读写允许同时放置书和证, 一次性完成借还; 再如电子阅览室管理, 读者可自行完成等待、刷证上机的自动化流程, 无须工作人员做更多操作, 可以将主要精力用在阅览室监管上。

3.4 ULAS提供完善的预约送书服务解决方案

为保证图书馆预约送书服务质量, 深圳图书馆采用预约送书馆藏(自助图书馆馆藏)和开架阅览馆藏完全分离的方式。ULAS的预约送书系统正是在此基础上产生。“图书馆之城”的馆藏是个统一体, ULAS的预约系统能够根据系统规则, 在中心馆和分馆之间进行图书筛选, 使读者真正可以预约到“图书馆之城”的全部图书。

ULAS的预约送书分为服务点送书、送书到家、自助图书馆送书和服务, 这三者虽然预约请求提交方式相同, 都是由读者直接在网上提交, 但找书下架和通知处理却完全不同, 前两者基本由工作人员处理, 而另一者则由工作人员和机器共同处理。ULAS能很好地处理这三种预约方式并取得了成功的应用经验。

3.5 ULAS采用完全自主研发的电子阅览室自动化管理系统

ULAS的电子阅览室管理系统不仅采用更多的管理方式——如远程截图、远程控制、自动开关机、禁止程序列表等功能, 还采用了自动化管理思想。从深圳图书馆应用电子阅览室管理软件的经验来看, 馆员不是网吧管理员, 不应将主要精力用于管理机位上, 而应主要用于读者咨询和上机监管上。为此, ULAS的电子阅览室系统进行了周密设计, 基本实现了自助上机管理, 刷卡、分配座位、打座位号单、入座、计时、提醒等功能全部自动完成。馆员可以集中精力进行读者咨询工作, 并利用远程监控功能进行读者监管, 防止不正当读者进行涉黄、涉毒、涉赌和反动言论的行为。

3.6 ULAS采用自主研发的短信平台

ULAS与中国移动广东分公司合作, 采用长短信技术研发短信平台, 具有查书、查借、续借、缴费、定制短信公告服务、挂失、绑定手机号、到期提醒、短信参考咨询等多项功能。同时ULAS提供完善的短信跟踪功能, 从业务操作到状态报告都有详细的日志记录, 一个未收到到期提醒短信的投诉仅须几分钟时间即能确定问题的所在。

4 小结

ULAS是深圳市“图书馆之城”的产物, 她不仅汇集了深圳乃至全国图书馆用户的宝贵经验, 也吸取了Web 2.0乃至Lib 3.0的思想精髓。ULAS追踪世界科技的最新发展趋势和潮流, 以服务为中心, 遵循图书馆城市化设计理念构建。但我认为图书馆城市化仅仅是ULAS在新时代下的第一站, 当图书馆联合共建、联合服务, 成为图书馆之城、图书馆之省甚至图书馆之国, 构建出真正的大型联盟式图书馆, 充分发挥其海量实体信息资源和全面聚类方式的优势, 建立完全统一的Web 2.0服务模式, 就能具备与搜索引擎、数据库商相当的实力, 在信息领域真正地占领一席之地。

参考文献

- [1] 陈玮华,马自卫. 基于J2EE的数字图书馆统一检索系统的研究与实现[J]. 现代图书情报技术,2006(6):17-21.
- [2] 刘秋梅,郑耿忠. 基于J2EE和XML的数字图书馆系统设计与实现[J]. 情报杂志,2006(7):22-26.
- [3] 郭晓利,曲朝阳,刘旭东. 基于J2EE的数字图书馆设计[J]. 现代情报,2006(4):102-104.
- [4] 杨宝洪,赵光标. J2EE在图书馆管理信息系统中的应用[J]. 现代情报,2006(9):90-92.
- [5] 何海钊,王虎奇,黄榜彪. J2EE在图书馆信息参考咨询服务系统中的应用[J]. 科技情报开发与经济,2007(17):88-90.
- [6] 冯文龙. 基于J2EE和XML的电子图书馆设计与实现[J]. 信息技术,2005(12):35-37.

作者简介

蔡晖 (1978-), 男, 深圳图书馆馆员。通讯地址: 广东省深圳市福田区福中一路2001号。E-mail: chui@sina.com

Research on the United Library Automation System

Cai Hui /Shenzhen Library, Shenzhen, 518036

Abstract: ULAS (united library automatic system) is an innovative product in the environment of building library networks, which is based on the design idea and practice of dILAS. ULAS upgrades the traditional library theory to Lib 3.0, which establishes the service kernel. The paper compares dILAS with ULAS from six aspects. Then it introduces ULAS's main functions and technical features. Meanwhile, it introduces the problems in the process of R&D and its resolutions.

Keywords: Library network, Library automation, ULAS, dILAS

(收稿日期: 2010-03-16)