ABSTRACT

This paper discusses the advantages of database-backed websites and describes the model for a library website implemented at the University of Nottingham using open source software, PHP and MySQL. As websites continue to grow in size and complexity it becomes increasingly important to introduce automation to help manage them. It is suggested that a database-backed website offers many advantages over one built from static HTML pages. These include a consistency of style and content, the ability to present different views of the same data, devolved editing and enhanced security. The University of Nottingham Library Services website is described and issues surrounding its design, technological implementation and management are explored.

**摘要**

本文讨论了数据库支持的网站的优势，并描述了使用开源软件PHP和MySQL在诺丁汉大学实施的图书馆网站的模型。 随着网站规模和复杂性的不断增长，引入自动化来帮助管理它们变得越来越重要。 建议数据库支持的网站提供许多优于静态HTML页面构建的网站的优点。 这些包括样式和内容的一致性，呈现相同数据的不同视图，下放编辑和增强安全性的能力。 描述了诺丁汉大学图书馆服务网站，并探讨了围绕其设计，技术实施和管理的问题。

1．Introduction

As websites grow in complexity it is important to choose the right technology to deliver the information. Here we discuss the advantages of database-backed websites and our model for a library website at Nottingham University using open source software, PHP and MySQL. Dynamically-generated websites are easy to manage and offer significant advantages over those built with static HTML.

**1.简介**

随着网站复杂性的增加，选择合适的技术来传递信息非常重要。 在这里，我们使用开源软件PHP和MySQL讨论数据库支持网站的优势以及诺丁汉大学图书馆网站的模型。 动态生成的网站易于管理，与使用静态HTML构建的网站相比具有明显的优势。

* 1. Redesign of the Library website

During the reorganisation of the University of Nottingham Library Services website (http://www.nottingham.ac.uk/library) in 1999, the decision was made to improve the design by minimising the number of options presented to the user on the home page. As well as options to find out more about the library, many library home pages offer a number of different choices to users, such as ‘subject guides’, ‘networked CD-ROMs’ and ‘online databases’. These choices are often based on the format of the resources (CD-ROMs, websites, printed items). Monica Brinkley points out that this is “undoubtedly the least likely way the user will approach an information need, and should not be the primary route to the information they seek”1.

1.1重新设计图书馆网站

在1999年诺丁汉大学图书馆服务网站（http://www.nottingham.ac.uk/library）重组期间，决定通过尽量减少在家中向用户提供的选项数量来改进设计 页。 除了了解有关该库的更多信息的选项外，许多图书馆主页还为用户提供了许多不同的选择，例如“主题指南”，“网络CD-ROM”和“在线数据库”。 这些选择通常基于资源的格式（CD-ROM，网站，印刷品）。 Monica Brinkley指出，这“无疑是用户接近信息需求的最不可能的方式，而不应成为他们寻求信息的主要途径”1。

In an attempt to reduce the potential for user confusion, it was decided to bring these resources together under one option: ‘Subject resources’ (the other home page options are currently ‘Online catalogue’, ‘Services and information’, ‘Opening hours’, ‘Manuscripts and Special Collections’ and ‘Contact us’). The pages in the Subject resources section of the website offer a subjectbased view of important resources, including bibliographies, databases and web gateways. Resources are listed together regardless of their format (although the format and access details of the resource are still made clear to the user for each resource). Staff and students know intuitively the subject in which they are interested and this section provides a simple gateway to the resources (as shown in Figure 1).

为了减少用户混淆的可能性，决定将这些资源集中在一个选项下：“主题资源”（其他主页选项目前是“在线目录”，“服务和信息”，“开放时间” ，“手稿和特别收藏”和“联系我们”）。 网站主题资源部分的页面提供了重要资源的主题视图，包括参考书目，数据库和网关。 资源一起列出，无论其格式如何（尽管资源的格式和访问详细信息对于每个资源仍然清楚）。 工作人员和学生直观地了解他们感兴趣的主题，本节提供了一个简单的资源门户（如图1所示）。

* 1. Static versus dynamic

It is fairly straightforward to produce web pages of lists of resources, their URLs and descriptions for different subject areas by using static HTML. The simplest way to produce such pages is to use a visual 'what-you-see-is-whatyou-get' HTML editor, such as Microsoft FrontPage or Macromedia Dreamweaver. However, this is not an activity that scales well. Once there are several hundred resources, many of which may appear on a number of pages, inconsistencies begin to develop and maintenance becomes a time-consuming task

1.2静态与动态

通过使用静态HTML生成资源列表的网页，其URL和不同主题区域的描述是相当简单的。 生成此类页面的最简单方法是使用可视化的“你看到的是什么”的HTML编辑器，例如Microsoft FrontPage或Macromedia Dreamweaver。 但是，这不是一个可以很好地扩展的活动。 一旦有数百个资源，其中许多可能出现在许多页面上，就会出现不一致的情况，并且维护成为一项耗时的任务

The alternative is to use dynamic pages that are created on demand by pulling the requested information from a database. This offers several major advantages:

– a database, with a set of defined fields, helps to ensure consistency of content between records;

– one resource can appear on several different pages, however, since there is only one database entry behind these, consistency is ensured;

– unlike static HTML files, it is simple to search the database (for title keywords, for example);

– it is possible to present different views of the same data, producing subject pages as well as those based on format for users to browse;

– style is separated from content, it is therefore easy to ensure consistency and changing the style requires minimal effort (changes only need to be made once to the template);

– editing can easily be distributed and controlled via web forms; – security is easy to manage: password fields can be hidden to certain types of users (such as those viewing the pages from off-campus) if required

另一种方法是使用通过从数据库中提取所请求的信息来按需创建的动态页面。这提供了几个主要优点：

- 具有一组定义字段的数据库有助于确保记录之间内容的一致性;

- 但是，一个资源可以出现在几个不同的页面上，因为这些资源后面只有一个数据库条目，所以确保了一致性;

- 与静态HTML文件不同，搜索数据库很简单（例如，标题关键字）;

- 可以呈现相同数据的不同视图

主题页面以及基于用户浏览格式的主题页面;

- 样式与内容分离，因此很容易确保一致性并且需要最少的努力来改变样式（只需要对模板进行一次更改）;

- 可以通过网络表单轻松分发和控制编辑;

- 安全性易于管理：如果需要，可以向某些类型的用户（例如从校外查看页面的用户）隐藏密码字段。

For these reasons, it was decided to develop a web-enabled database to form the basis of the Subject resources section of the Library Services website. After some research into ‘off the shelf’ products, such as ROADS, it was also decided that they were overly complex for our needs and that a simple ‘inhouse’ solution would be preferable. ROADS (Resource Discovery and Indexing software) is a set of software tools that was developed as part of an Electronic Libraries Programme project to enable the set up and maintenance of web-based subject gateways (see http://www.ilrt.bris.ac.uk/roads). An inhouse system was seen as having the advantages of being simpler to install, maintain and use, without having to rely on external support. A locally developed system is also highly customisable: it can be designed to meet a particular need and further developed as needs change.

出于这些原因，决定开发一个支持Web的数据库，以形成图书馆服务网站的主题资源部分的基础。 在对“现成”产品（如ROADS）进行一些研究之后，还决定它们对我们的需求过于复杂，而且简单的“内部”解决方案将更为可取。 ROADS（资源发现和索引软件）是一套软件工具，是作为电子图书馆计划项目的一部分开发的，用于设置和维护基于网络的主题网关（见http://www.ilrt.bris.ac.uk/roads）。 内部系统被认为具有安装，维护和使用更简单的优点，而不必依赖外部支持。 本地开发的系统也可高度定制：它可以设计为满足特定需求，并在需求变化时进一步开发。

2. Technologies

To deliver a database-backed website, three main technologies are required: a web server, a database and a mechanism for these two to interact and incorporate database content in the HTML pages. We chose to run the Apache web server (see http://www.apache.org), the most popular web server on the Internet2 on a UNIX server. For a relational database, we chose MySQL (http://www.mysql.com) which is stable, has a very fast query response time on simple tables, is reliable and ANSI SQL92 compliant.3 To glue these two together (see Figure 2), we chose the increasingly popular server-side HTML embedded scripting language PHP (see http://www.php.net). PHP was initially developed as a personal project, hence the original name Personal Home Page. It is now a major open source server-side HTML embedded scripting language and is in use at over two million websites, including Volvo (see http://www.volvo.com), Mitsubishi Motors (see http://www.mitsucars.com), W3C (see http://www.w3.org) and HumanitiesWeb (see http://humanities web.org). PHP is available as an Apache module, and is therefore very effi- cient, and works well with MySQL (as well as a variety of other databases).

**2.技术**

要提供数据库支持的网站，需要三种主要技术：Web服务器，数据库和这两者在HTML页面中交互和合并数据库内容的机制。我们选择运行Apache Web服务器（请参阅http://www.apache.org），它是UNIX服务器上Internet2上最受欢迎的Web服务器。对于关系数据库，我们选择了稳定的MySQL（http://www.mysql.com），在简单的表上具有非常快的查询响应时间，可靠且符合ANSI SQL92.3。将这两者粘合在一起（参见图2），我们选择了越来越流行的服务器端HTML嵌入式脚本语言PHP（参见http://www.php.net）。 PHP最初是作为个人项目开发的，因此原始名称为Personal Home Page。它现在是一种主要的开源服务器端HTML嵌入式脚本语言，并且在包括沃尔沃在内的200多万个网站中使用（参见http：// www .volvo.com），三菱汽车（见http://www.mitsucars.com），W3C（见http://www.w3.org）和HumanitiesWeb（见http：// humanities web.org）。 PHP作为Apache模块提供，因此非常高效，并且与MySQL（以及各种其他数据库）配合良好。

PHP is very easy to integrate with HTML. Blocks of PHP code can simply be inserted in a standard HTML document as required. Before the web server sends the requested file to a user’s browser it executes any PHP code within the page and incorporates the relevant output. For example, the PHP code becomes Today's date is: August 1st, 2000. One of the most useful features of PHP is the way it handles HTML forms. Any form element automatically results in a variable with the same name being created on the target page. All the elements in the form are therefore available as variables and ready to incorporate in your HTML output or include in a database query. PHP can perform mathematical calculations, manipulate strings of data and (as in this case) query databases. It can work with many predefined functions, including XML support, e-mail capabilities (including IMAP functions) and automatic session handling

PHP很容易与HTML集成。 可以根据需要将PHP代码块简单地插入到标准HTML文档中。 在Web服务器将所请求的文件发送到用户的浏览器之前，它会执行页面中的任何PHP代码并合并相关的输出。 例如，PHP代码<？ 打印“今天的日期是：日期（”M d，Y“）”; ？>成为今天的日期是：2000年8月1日。 PHP最有用的功能之一是它处理HTML表单的方式。 任何表单元素都会自动生成具有相同变量的变量。在目标页面上创建的名称。 因此，表单中的所有元素都可以作为变量使用，并且可以在HTML输出中包含或包含在数据库查询中。 PHP可以执行数学计算，操作数据字符串和（如本例中）查询数据库。 它可以与许多预定义的功能一起使用，包括XML支持，电子邮件功能（包括IMAP功能）和自动会话处理。

All of these popular products are open source and available free of charge (see http://www.opensource.org). There is also support from the user community available in a variety of ways on the Internet. For example, there are excellent tutorials on PHP and MySQL at both the DevShed (see http://www. devshed.com) and Webreview (see http://webreview.com) sites, and active discussion boards on PHP at the PHPBuilder (see http://www.phpbuilder.com) and Faqts (see http://www.faqts.com/knowledge-base/index.phtml/fid/51) sites. The installation and setup of these products is not described here as this is done elsewhere.4 However, it is possible to set up a standard Intel PC as a web/database UNIX server using Apache, PHP and MySQL in just a few hours. We chose to use Linux, a free version of UNIX. This combination of software running on a Linux server proved to be easy to install, tightly integrated, and stable. The whole system can also be assembled for very little cost. The Linux Documentation Project at Manchester Computing service provides a useful source of information about Linux (see http://www.mcc.ac.uk/LDP).

所有这些受欢迎的产品都是开源的，免费提供（见http://www.opensource.org）。用户社区也提供互联网上以各种方式提供的支持。例如，在DevShed（请参阅http://www.devshed.com）和Webreview（请参阅http://webreview.com）站点以及PHPBuilder上有关PHP的活动讨论板上都有关于PHP和MySQL的优秀教程。 （见http://www.phpbuilder.com）和Faqts（见http://www.faqts.com/knowledge-base/index.phtml/fid/51）网站。这里没有描述这些产品的安装和设置，因为这是在别处完成的。但是，可以在几个小时内使用Apache，PHP和MySQL将标准的英特尔PC设置为Web /数据库UNIX服务器。我们选择使用Linux，一个免费的UNIX版本。这种在Linux服务器上运行的软件组合证明易于安装，紧密集成且稳定。整个系统也可以很少的成本组装。曼彻斯特计算机服务的Linux文档项目提供了有关Linux的有用信息源（请参阅<http://www.mcc.ac.uk/LDP>）。

3. The ‘Subject resources’ database

3.1 Requirements

The Subject resources database contains metadata for various resources. The database was constructed in response to a number of specific requirements. The first requirement was that the various resources should be consistently described using a number of metadata elements. These were title, alternate title, author, URL, password/access details, and description (some of which would be optional) which equate to five key elements of the Dublin Core (see http://purl.org/DC). Secondly, it should be possible to associate each resource with one from a given list of ‘sub-headings’ ( as listed in Appendix 1) and one or a number of ‘format/category’ details (as listed in Appendix 2). Thirdly, it should be possible to associate each entry with one or a number of different subjects, so that the same basic entry could be displayed under one or a number of subject pages. Fourthly, there was a requirement that different subject-specific comments could be associated with the same resource entry on different subject pages. For example, this would allow different comments to be made about Web of Science on the Chemistry page than under Economics. For each subject, it would also be required to assign a resource a ‘status’ of ‘key resource’ if appropriate

**3.“主题资源”数据库**

3.1要求

主题资源数据库包含各种资源的元数据。数据库的构建是为了响应一些具体要求。第一个要求是应使用许多元数据元素一致地描述各种资源。这些是标题，备用标题，作者，URL，密码/访问详细信息和描述（其中一些是可选的），相当于Dublin Core的五个关键元素（请参阅http://purl.org/DC）。其次，应该可以将每个资源与给定的“子标题”列表（如附录1中所列）和一个或多个“格式/类别”详细信息（如附录2中所列）相关联。第三，应该可以将每个条目与一个或多个不同的主题相关联，以便可以在一个或多个主题页面下显示相同的基本条目。第四，要求不同的主题特定注释可以与不同主题页面上的相同资源条目相关联。例如，这将允许对化学页面上的Web of Science进行不同于经济学的评论。对于每个主题，如果合适，还需要为资源分配“关键资源”的“状态”。

Fifthly, there was also the requirement that the different entries in the database could be presented to the user in different ways. It should be possible to search for resources by title or title keyword. In addition, it should also be possible to browse a list of resources, divided under meaningful sub-headings, under a given subject. Subjects available largely correspond to the departments and schools in the University. It should also be possible (where appropriate) to find resources by format.

第五，还要求数据库中的不同条目可以以不同方式呈现给用户。 应该可以按标题或标题关键字搜索资源。 此外，还应该可以在给定主题下浏览在有意义的子标题下划分的资源列表。 可用的科目主要对应于大学的系和学校。 也应该（在适当的情况下）按格式查找资源。

Finally, it was also required that maintenance of the database entries should be easily carried out by a number of subject librarians. Adding new resources and editing existing ones should be a straightforward task.

最后，还要求由一些主题图书馆员轻松地进行数据库条目的维护。 添加新资源和编辑现有资源应该是一项简单的任务。

3.2 Data analysis

In a relational database multiple tables of data relate to each other through special key fields (this is in contrast to a flat file database which contains a single table of data). The main advantage of a relational, as opposed to a flat file, database is that data duplication, and therefore potential inconsistancies, are eliminated. In response to the requirements listed above, seven tables were designed within the Subject resources database to contain the metadata. These are outlined below and illustrated in Figure 3. – resource - resource details (title, URL, description etc); – subject - subjects and their introductions; – subheading - subheadings that resources are listed under; – format - common formats and access methods; – resource\_subject - defines which resources appear on which subject pages; – subject\_subheading - subject specific sub-heading introductions; – resource\_format - defines which formats/access methods apply to each resource

3.2数据分析

在关系数据库中，多个数据表通过特殊键字段相互关联（这与包含单个数据表的平面文件数据库形成对比）。 与平面文件数据库相比，关系数据库的主要优点是消除了数据重复，因此消除了潜在的不一致性。为了响应上面列出的要求，在主题资源数据库中设计了七个表来包含元数据。 这些概述如下并在图3中示出。

- 资源 - 资源详细信息（标题，URL，描述等）;

- 科目 - 科目及其介绍;

- 副标题 - 列出资源的小标题;

- 格式 - 通用格式和访问方法;

- 定义哪个资源出现在哪个主题上页;

- 主题特定的子标题介绍;

- 定义适用的格式/访问方法每个资源。

Some common formats of material (such as networked CD-ROMs) and access methods (such as ‘ATHENS username and password required’, or ‘Access from the Nottingham campus only’) require links to further information. (ATHENS is a UK authentication system used for accessing a number of databases see http://www.athens.ac.uk). Separating these options off, in a separate format table, enables many resources to include or link to one copy of this information which ensures consistency. Similarly the lists of sub-headings and subjects are located in separate tables.

一些常见的材料格式（例如网络CD-ROM）和访问方法（例如“需要ATHENS用户名和密码”，或“仅来自诺丁汉校园的访问”）需要链接到更多信息。（ATHENS是英国认证系统 用于访问许多数据库，请参阅http://www.athens.ac.uk）。 在单独的格式表中将这些选项分离，可以使许多资源包含或链接到此信息的一个副本，从而确保一致性。 类似地，子标题和主题列表位于单独的表中。

3.3 The web pages

Users can browse the Subject resources by subject (from the menu page illustrated in Figure 1) or search for a particular resource by title. A typical subject page is shown in Figure 4 and comprises of: – the title of the page;

– a general subject description;

– a menu of sub-headings together with a key to the icons used;

– key resources listed at the top of the page followed by all the resources associated with that subject, sorted alphabetically by sub-heading.

3.3网页

用户可以按主题浏览主题资源（从图1中所示的菜单页面），或按标题搜索特定资源。 典型的主题页面如图4所示，包括：

- 页面标题;

- 一般主题描述;

- 一个子标题菜单以及所用图标的键;

- 页面顶部列出的关键资源，后跟与该主题相关的所有资源，按子标题按字母顺序排序

When the user selects a given subject from the Subject resources menu page, the subject page is constructed ‘on the fly’ from the database using PHP. Theresults page is passed the required subject-id and a query of the subject table produces the full subject title and introduction for the page. A second query identifies all the required resource\_ids related to this subject from the resource\_subject table. All the details for each resource are then pulled from the resource table and printed out to the HTML page. The styles of the page itself are determined by the use of cascading style sheets (which are used throughout the Library Services website).

当用户从“主题资源”菜单页面中选择给定主题时，主题页面将使用PHP从数据库中“即时”构建。结果页面传递了所需的subject-id，主题表的查询产生了完整的主题标题和页面的介绍。第二个查询从resource\_subject表中标识与此主题相关的所有必需resource\_id。然后从资源表中提取每个资源的所有详细信息并打印到HTML页面。页面本身的样式由使用级联样式表（使用它们）决定

Each resource entry is displayed as illustrated in Figure 5 and consists of:

1. The resource title (clickable for appropriate electronic resources)

2. A set of icons indicating the resource format (such as print, CDROM, web) and access (such as ATHENS, campus access only) details

3. The URL (for web resources)

4. The description

5. Any special access requirements (passwords are only displayed to users accessing the page from on campus)

6. The subject specific evaluation (optional).

整个图书馆服务网站）。每个资源条目如图5所示，包括：

1.资源标题（可点击适当的电子资源）

2.一组图标，指示资源格式（如打印，CDROM，Web）和访问（如ATHENS，仅限校园访问）详细信息

3. URL（用于Web资源）

4.描述

5.任何特殊访问要求（密码仅显示给

用户从校园访问该页面）

6.主题特定评估（可选）。

3.4 Administration

Subject librarians can all update the database via a series of password protected web forms (illustrated in Figure 6). This is simple to do and is a very efficient way of devolving the content creation to a large number of people whilst maintaining a consistent style.

3.4管理

学科馆员可以通过一系列受密码保护的网络表格更新数据库（如图6所示）。 这很容易做，并且是一种非常有效的方式，可以将内容创建转移给大量人员，同时保持一致的风格。

4. Electronic journals

The above model may appear very specific but it is actually extremely flexible and can easily be applied to other projects. We have used a similar approach to generate our index of electronic journals (http://www.nottingham.ac.uk/ library/ejournals/). This service offers subject-based or alphabetical lists of titles, and users can also search for particular titles. In addition, we have added a message field, to inform users about a change in the service, which can apply to an individual title or a whole package of titles from a particular supplier.

**4.电子期刊**

上述模型可能看起来非常具体，但实际上非常灵活，可以很容易地应用于其他项目。 我们使用类似的方法来生成我们的电子期刊索引（http://www.nottingham.ac.uk/library/ejournals/）。 此服务提供基于主题或按字母顺序排列的标题列表，用户还可以搜索特定标题。 此外，我们还添加了一个消息字段，以告知用户服务的更改，该字段可以应用于特定供应商的单个标题或整个标题包。

5. Conclusion

As library websites migrate from being about the service to being part of the service they inevitably grow in complexity. In this environment, some degree of automation is advantageous otherwise maintenance becomes difficult to manage. In many cases, automation can improve the content quality and navigability of the site. Preliminary evidence suggests that the Subject resources database is popular with both library staff and users. The former find it easy to maintain and useful in their information skills training and enquiry work. The latter are happy to work with a subject-based one-stop-shop approach to the quality information resources available to them.

**5.结论**

随着图书馆网站从服务转变为服务的一部分，它们不可避免地变得复杂。 在这种环境中，某种程度的自动化是有利的，否则维护变得难以管理。 在许多情况下，自动化可以提高网站的内容质量和可导航性。 初步证据表明，主题资源数据库受到图书馆工作人员和用户的欢迎。 前者发现在信息技能培训和咨询工作中易于维护和使用。 后者很乐意使用基于主题的一站式方法来获得他们可获得的高质量信息资源。