

一项结合云计算和 PKI 的数字图书馆系统框架设计

叶小榕¹, 邵 晴²

1. 中国科学技术信息研究所, 北京 100038

2. 中国互联网络信息中心, 北京 100190

摘要 结合云计算和 PKI 技术, 设计了一个数字图书馆系统框架。根据云计算和 PKI 技术的概念、技术特点, 依托 Google 提供的 AppEngine 云计算开发平台, 集中其技术优势, 设计开发了云计算和 PKI 结合的数字图书馆系统框架并提出框架具体部署方法。运行结果证明: 该系统扩展了数字图书馆的信息搜集、分析、检索等能力; 实现了证书登录、管理员录入保存图书信息、用户查询图书信息等几大模块, 已经拥有了基本的图书馆功能, 可以为数字图书馆提供完备的安全保障和可信的网络服务。今后将在此基础上, 充分发掘云计算平台 AppEngine 潜力, 逐步实现较全面的数字图书馆系统服务。

关键词 数字图书馆; 云计算; PKI; AppEngine

中图分类号 TP311.13, N99

文献标识码 A

文章编号 1000-7857(2010)09-0063-05

A System Framework Design of the Digital Library with the Combination of Cloud Computing and PKI Technology

YE Xiaorong¹, SHAO Qing²

1. Institute of Scientific and Technical Information of China, Beijing 100038, China

2. China Internet Network Information Center, Beijing 100190, China

Abstract This paper presents a design of a digital library system framework, which can take advantage of both cloud computing and PKI technology. According to the concepts and technical characteristics of the cloud computing and PKI technology, and based on the AppEngine development platform of the cloud computing provided by the Google company, a digital library system framework is designed and developed and its detailed and specific implementation is presented. The operative tests of the digital library system framework demonstrate that this system significantly expands the functions of information-gathering, analysis, and retrieval capabilities of the digital library. This system framework enjoys basic library functions, as are achieved in some modules, where the users can login the digital library by their security certificates, and the administrators of the digital library can input and save the information of the books, and the readers of the digital library may query and retrieve the data of the books. Furthermore, this system can provide comprehensive security and reliable network services for the readers. In the future, this system will adequately explore the technological potential of the Google's AppEngine development platform of the cloud computing on this basis, and gradually implement more comprehensive services of the digital library.

Keywords digital library; cloud computing; PKI; AppEngine

0 引言

数字图书馆近年来取得了长足的进步, 但随着信息资源、读者群的迅速增加, 图书馆数字技术滞后, 需求和服务之

间的距离阻碍了数字图书馆的进一步发展。首先是存储和网络问题。随着数字图书馆的电子资源数量激增, 对数据存储的容量和可靠性要求越来越高, 而且随着通过网络访问数字

收稿日期: 2010-04-14

作者简介: 叶小榕, 助研, 研究方向为计算机编程、数字图书馆系统研究, 电子信箱: yeelfine@sina.com

图书馆的读者越来越多,对网络访问能力、并发处理能力的要求也随之提高。其次是安全问题。互联网普及的同时也带来了安全问题,现今钓鱼网站层出不穷,黑客木马防不胜防,通信被窃听、信息泄密也时有发生,给用户造成重大经济损失^[1]。

建立数字图书馆需要投入大量资金,同时还要不断提高自身技术,以保证系统日常的维护、升级。但一般图书馆的财力、物力、技术有限,往往对建立数字图书馆所需费用都难以承受。本文根据云计算和 PKI 的概念、技术特点,在原有图书馆设施和技术基础上,依托 Google 提供的 AppEngine 云计算开发平台,集中其技术优势,设计开发出云计算和 PKI 结合的数字图书馆系统框架并提出了框架具体部署方法,为实现较全面的数字图书馆系统服务提供了可行的途径^[2]。

1 云计算、PKI 概念及特点

云计算是从“网格计算”、“分布式计算”到“互联网计算”逐步发展起来的。它结合了虚拟化、高性能存储等技术,利用互联网将分散的信息资源和计算能力整合起来,供用户方便地访问、分享、使用和管理,可以提供在线文档、信息搜集、科学计算、数据挖掘、数据存储等多种服务,并能实现按需服务、按需付费,达到高效率低成本的目的^[3]。

首先,云计算有超大规模的计算和存储能力,可依靠虚拟化技术,使“云”能够随时扩展,满足不断变化的应用需求;其次,云计算能提供各种在线服务而不受用户主机性能的限制,如在线文档、在线杀毒等,甚至包括像超级计算这样的任务也能完成;第三,云计算提供按需付费,用户按照所订购的服务付费即可,节约了成本;最后,云计算提供个性化服务,通过对海量数据的搜索、挖掘,分析用户个性化行为模式,满足多元化服务需求^[4]。

PKI(Public Key Infrastructure, 公钥基础设施)是利用公钥理论和技术建立的提供安全服务的基础设施,能为网络应用提供数据加密、数字签名、身份识别,以及所必需的密钥和证书管理等安全服务^[5]。在数字图书馆中,PKI 主要从身份识别和加密通信两方面解决了数字图书馆的安全问题,其整套的框架体系包括:PKI 策略,数字证书,认证中心(Certificate Authority, CA),注册中心(RA),查询和吊销系统等^[6],可实现信息安全的 4 大特性:机密性、真实性、完整性和不可抵赖性。具体地讲,数字证书是 PKI 中最基本的协议,由 CA 签发,可以进行身份验证的电子证书,用来在互联网中证明自己的身份和识别对方的身份。证书包含持有人身份、加密算法、公钥、有效期等加密信息。其中 DN(Distinguished Name)标识了证书持有者的身份,例如 DN="CN=张三,O=YYY 大学,S=北京市,C=CN",使得服务器或其他用户均可有效地验证识别证书持有者身份。同时,PKI 通过非对称加密算法实现数据加密通信功能,使得用户和服务器之间、服务器与服务器之间,能够建立加密的可信通道。现 PKI 已广泛应用于安全认证、保密通信领域,已成为当前信息安全技术的核心组成部分,很

多安全协议均建立在 PKI 基础之上,如:SSL、S/MIME、IPSEC、VPN 等。

2 结合云计算和 PKI 的数字图书馆的优势

首先,云计算可降低数字图书馆的建设和维护费用,不再需要进行大规模的 IT 投资,只需较少的费用租用云计算的数据存储、检索、网络接入等服务即可,实现按服务付费;其次,云计算增强了数字图书馆按需变化的业务处理能力,在访问高峰时可获得更多的网络带宽和计算能力,在闲时可以将其归还,实现业务动态管理;另外,云计算利用庞大的数据资源和计算能力可为数字图书馆提供个性化的用户服务,在保证隐私的情况下,通过行为模式分析用户的浏览、借阅和检索记录、访问页面等信息,提供更完善的个性化服务。

PKI 则是对数字图书馆提供安全保证。PKI 签发的数字证书,使读者和数字图书馆服务器之间相互验证,既能确认读者的合法身份,防止冒名顶替,同时也使读者避免误登录钓鱼网站;同时,PKI 通过对称和非对称加密技术,在用户和数字图书馆之间建立起安全保密的可信信道,最大限度保障用户数据的机密性。

3 结合云计算和 PKI 的数字图书馆系统框架设计

根据已有的云计算框架,充分利用云计算和 PKI 的优势,设计了结合云计算和 PKI 的数字图书馆框架。按照该设计,数字图书馆提供了极其丰富的信息资源,云计算提供了整体的支撑框架,PKI 则在多个层次提供安全模块服务^[7-8](图 1)。

云计算包括 3 个层次的服务:基础设施即服务(Infrastructure as a Service, IaaS),平台即服务(Platform as a Service, PaaS)和软件即服务(Software as a Service, SaaS)。

1) IaaS 提供了面向 IT 基础设施的服务和管理。包括 CaaS(Communication as a Service),DaaS(Data as a Service)和

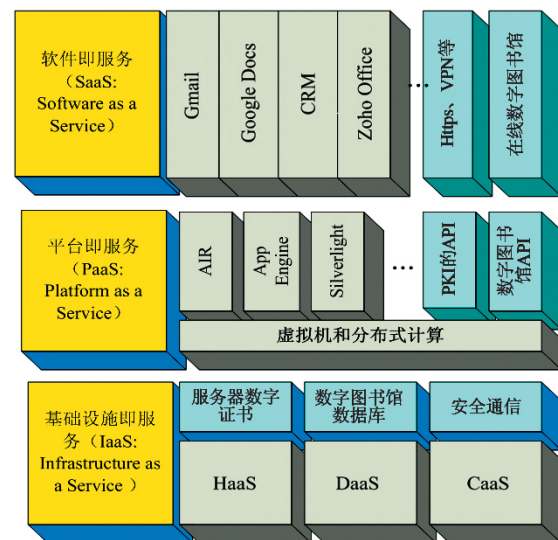


图 1 系统模块架构图

Fig. 1 System module architecture

HaaS(Hardware as a Service)等。数字图书馆主要担当了 DaaS 的角色,为整个云计算提供数据资源。而 PKI 则主要为 CaaS 服务,提供加密安全通信服务,包括 SSL、VPN 等,同时也为服务器提供数字证书服务。

2) PaaS 是指通过互联网提供的面向开发人员的开发平台、API 接口和中间件等。开发人员可以基于此平台构建丰富的 Web 应用服务。已经发布的 PaaS 包括 Salesforce 的 Force.com、Adobe 的 AIR、Microsoft 的 Azure 和 Google 的 AppEngine^[9] 等。数字图书馆 API 接口实现录入、保存、检索、下载等功能。PKI 提供证书校验、身份识别等功能。本文主要设计开发基于 AppEngine 平台的数字图书馆服务程序。

3) SaaS 指向用户提供的按需应用服务,使开发人员可以快速构建大规模、面向最终用户的服务程序。数字图书馆和 PKI 在此层能够提供完整的图书馆应用和安全服务。

3.1 系统设计和开发的实例

AppEngine 是 Google 推出的云计算开发部署平台,它使开发人员可以将应用程序轻松地部署到 Google 云上,后台服务框架和存储过程完全由 AppEngine 负责,从而实现“一次部署,全球运行”。本文以 AppEngine 为平台,初步设计开发了基于云计算和 PKI 的数字图书馆应用系统 MyCloudLibrary,实现了用户登录、图书信息录入保存、用户检索查询等图书馆系统所需的基础功能。

MyCloudLibrary 设计了图书馆管理员和普通用户两个角色。图书馆管理员实现图书信息管理,包括查询、保存、删除等基础功能;用户实现图书信息查询和图书借阅两个功能;两者共用证书登录模块,负责管理数字证书登录(图 2)。

重点介绍 MyCloudLibrary 3 个主要功能模块:管理员的录入保存功能、用户的图书信息查询功能和本地的证书登录功能。

系统的结构模型主要分为 3 个部分:证书登录、管理员录入保存信息(云计算存储),以及用户查询图书信息(Lucene 检索)(图 3)。

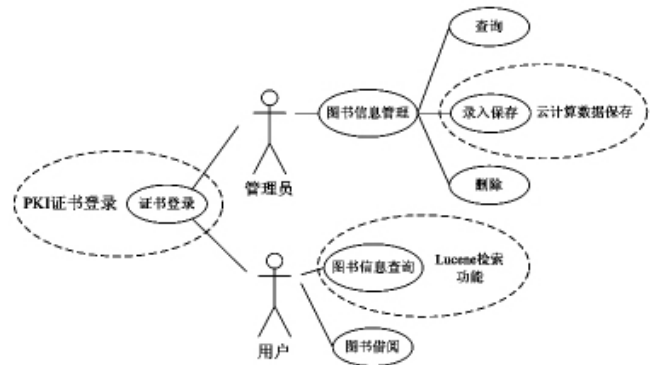


图 2 系统用例图

Fig. 2 A case of system flowchart

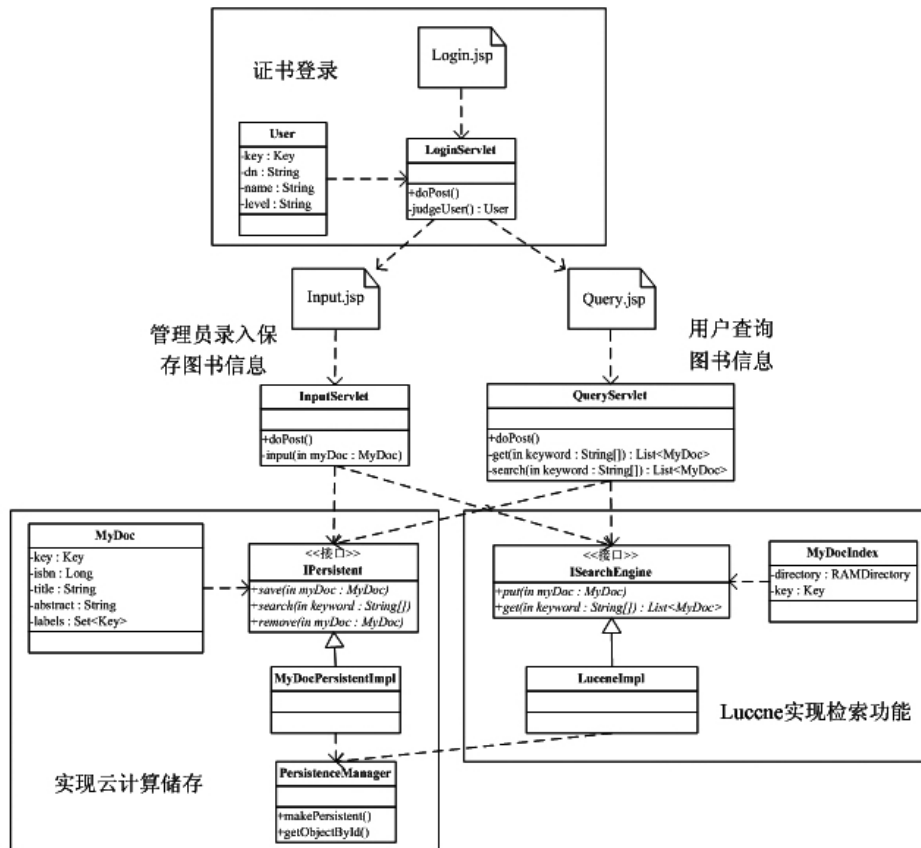


图 3 系统结构类图

Fig. 3 System structure class diagram

1) 证书登录

虽然现在的 AppEngine 还未支持证书登录的方式,但当云计算得到广泛使用,并应用在高安全性领域时,PKI 体系仍然是最安全最稳健的选择。因此,MyCloudLibrary 暂时采用本地生成、安装证书的方式,实现证书登录并识别登录者身份。

首先生成部署数字证书。MyCloudLibrary 在本地使用 tomcat 作为服务器端,采用 jdk 自带的 keytool 工具生成 3 张证书,分别是服务器证书、管理员证书和普通用户证书。3 张证书均由同一个根证书签发,可以实现互相信任识别。

服务器证书 DN 为“CN=www.mycloudlibrary.com,O=云计算数字图书馆,S=北京,C=CN”,其中通用名 CN 标识为域名 www.mycloudlibrary.com,单位名称 O 标识为“云计算数字图书馆”。然后修改 tomcat 配置文件 server.xml,其中 clientAuth 为“true”,表示需要客户端证书登录,server.keystore 为服务器证书文件,保存在服务器指定目录下:

```
<Connector port="8443" protocol="HTTP/1.1"
    SSLEnabled="true"
    clientAuth="true" sslProtocol="TLS"
    keystoreFile="./server.keystore" keystorePass="password"
    truststoreFile="./server.keystore"
    truststorePass="password" />
```

管理员证书 DN 为“CN=admin,O=云计算数字图书馆,S=北京,C=CN”,CN 为 admin 标识其身份为管理员。用户证书 DN 为“CN=张三,S=北京,C=CN”,CN 为用户姓名,标识其为一个普通用户。两张证书保存在管理员和用户的浏览器或者 USB 盘中。

其次,实现用户证书登录。用户登录 login.jsp 提交登录请求后,浏览器会弹出证书选择窗口,用户选择登录证书后,服务器证书和用户证书间将完成彼此的验证识别过程,从而确认各自身份,建立加密的可信通道,此过程对用户完全透明。当通过验证后,LoginServlet 可以获取用户证书的 DN 值,并根据 DN 判断用户身份,提供不同的操作权限:

```
X509Certificate[] certs=
    (X509Certificate [])request.getAttribute ("javax.servlet.
    request.X509Certificate");
```

```
String dn=certs[0].getSubjectDN().toString();
```

```
User user=judgeUser(dn);
```

2) 管理员录入保存信息

管理员在登录通过后,通过 input.jsp 页面录入图书信息,填入图书的 isbn、title、abstract、labels 等信息(可扩充),InputServlet 根据传入的参数生成 MyDoc 图书对象:

```
MyDoc myDoc=new MyDoc(isbn, title, abstract, labels);
```

然后通过 IPersistent 接口的 save() 函数提交数据库,简单的两行代码即可实现数据存储:

```
PersistenceManager pm=PMF.get().getPersistenceManager();
pm.makePersistent(myDoc);
```

AppEngine 平台可以通过 JDO (Java Data Object) 作为数

据的持久化存储接口,代码中的 PersistenceManager 是 JDO 提供的调用接口。在内部,JDO 会调用用开源的 DataNucleus 持久化引擎负责数据保存。而用 DataNucleus 来实现 AppEngine 云计算数据存储,十分简单,只需修改 JDO 的配置文件 jdo-config.xml 即可,不需要再指定数据库地址等信息。

```
<persistence-manager-factory name="transactions-optional">
    <property name="javax.jdo.PersistenceManagerFactoryClass"
        value = "org.datanucleus.store.appengine.jdo.DatastoreJDOPersistenceManagerFactory"/>
```

```
<property name="javax.jdo.option.ConnectionURL" value="
appengine"/>
```

可以看到,云计算的数据保存十分简单,几乎不需要写 sql 语句,也不需要关心数据具体的保存位置。

3) 用户查询检索图书信息

MyCloudLibrary 采用了开源的 Lucene 实现简单的搜索功能,可对选择的关键词建立索引,提高搜索速度。

首先建立索引库。在管理员录入图书信息时,调用 ISearchEngine 接口的 put() 函数将关键词提交到搜索引擎,且其图书索引记录 docIndex 也保存到了云端。本文中,仅设定标题(title)作为关键词添加到索引库,以后可根据需要添加其他关键词。

```
Document doc=new Document();
doc.add(new Field("title",myDoc.getTitle(), Field.Store.YES,
Field.Index.TOKENIZED));
iwriter.addDocument(doc);
docIndex.setKey(key);
pm.makePersistent(docIndex);
```

其中,doc.add() 负责添加关键词,docIndex 作为图书索引记录也通过 JDO 持久化保存。该搜索引擎支持对图书信息的快速检索查询,比仅用数据库本身的查询更为迅速。

其次,通过索引库和云端数据库实现用户快速检索。用户通过页面 query.jsp,输入关键词检索图书标题,提交到 QueryServlet 后,由接口 ISearchEngine 调用函数 get() 查询:

```
IndexSearcher isearcher=new IndexSearcher(docIndex.getDirectory());
```

```
Query query=MultiFieldQueryParser.parse(new String[] {key-
words},new String[] {"title"}, new StandardAnalyzer());
```

```
Hits hits=isearcher.search(query);
```

```
for (int i =0; i<hits.length(); i++)
```

```
{
```

```
MyDoc myDoc = pm.getObjectById(MyDoc.class, hits.doc
(i).get("id"));
```

```
resultList.add(myDoc);
```

```
}
```

系统通过 MultiFieldQueryParser 的 parse() 函数设定待检索关键词在检索库中查询,然后根据查询结果集从云端数据库取出图书信息,显示给用户,完成整个图书信息检索过程。

MyCloudLibrary 通过本地测试运行良好后, 就可通过 AppEngine 的插件上传到 Google 云计算平台, 通过 AppEngine 实现结合云计算的数字图书馆。

3.2 系统部署

结合云计算和 PKI 技术的数字图书馆系统部署如图 4 所示, 它包括了 PKI 模块, 云计算和数字图书馆等几部分。

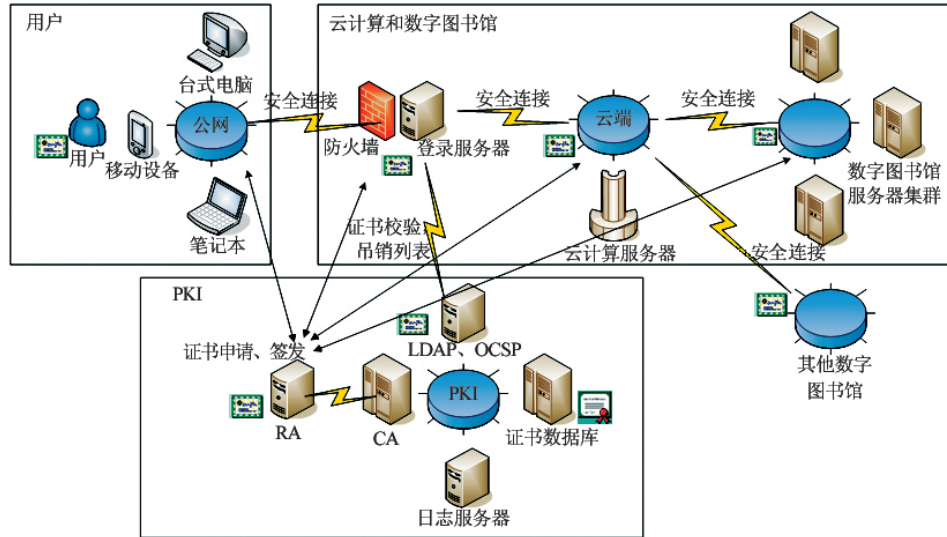


图 4 系统部署图

Fig. 4 System deployment

1) 云端和数字图书馆部分

数字图书馆可直接作为云计算的数据仓储, 为云计算提供丰富的数据资源。并且, 云计算可进行数据挖掘进行二次开发, 实现个性化需求。

2) PKI 部分

PKI 为整个系统提供安全保障, 包括两部分: 首先为服务器和读者签发数字证书, 实现双方的互相信任; 另一部分是为读者与服务器之间, 服务器与服务器之间建立起加密的可信通信。

3) 用户部分

读者申请并获取数字证书后, 可以将数字证书存放在自己的终端中, 用户在访问数字图书馆时, 选择相应的数字证书登录即可, 安全又简便。

4 结论

本文初步设计开发了与云计算和 PKI 相结合的数字图书馆系统, 实现了证书登录、管理员录入保存图书信息、用户查询图书信息等几大模块, 已经拥有了基本的图书馆功能, 今后将在此基础上充分利用云计算平台 AppEngine, 逐步实现较全面的数字图书馆系统服务。

参考文献 (References)

- [1] 关振胜. 公钥基础设施 PKI 及其应用 [M]. 北京: 电子工业出版社, 2008.
Guan Zhensheng. Public key infrastructure PKI and applications [M]. Beijing: Publishing House of Electronics Industry, 2008.
- [2] 杨明芳, 袁曦临. 云计算环境下的数字图书馆 [J]. 图书馆建设, 2009

(9): 7-12.

Yang Mingfang, Yuan Xilin. *Library Development*, 2009(9): 7-12.

- [3] 李永先, 梁旭伦, 李森森. 云计算技术在图书馆中的应用探讨[J]. 江西图书馆学刊, 2009(1): 105-106.

Li Yongxian, Luan Xulun, Li Sensen. *The Journal of the Library Science in Jiangxi*, 2009(1): 105-106.

- [4] 卢晓娟. 云计算与未来图书馆数字信息资源建设[J]. 四川图书馆学报, 2009(2): 23-24.

Lu Xiaojuan. *Journal of the Library Science Society of Sichuan*, 2009(2): 23-24.

- [5] 荆继鹏, 林璟铨, 冯登国. PKI 技术[M]. 北京: 科学出版社, 2008.

Jing Jiqiang, Lin Jingqiang, Feng Dengguo. PKI technology [M]. Beijing: Science Press, 2008.

- [6] 李明柱. PKI 技术及应用开发指南[BE/OL]. [2009-10-17]. <http://www.ibm.com/developerworks/cn/security/se-pkiusing/index.html>.

Li Mingzhu. PKI technology and application developer's guide[BE/OL]. [2009-10-17]. <http://www.ibm.com/developerworks/cn/security/se-pkiusing/index.html>.

- [7] 郭乐深, 张乃靖, 尚晋刚. 云计算环境安全框架 [J]. 信息网络安全, 2009(7): 62-64.

Guo Leshen, Zhang Naijing, Shang Jingang. *Netinfo Security*, 2009(7): 62-64.

- [8] 谢四江, 冯雁. 浅析云计算与信息安全 [J]. 北京电子科技学院学报, 2008, 16(4): 1-3.

Xie Sijiang, Feng Yan. *Journal of Beijing Electronic Science and Technology Institute*, 2008, 16(4): 1-3.

- [9] Google Corporation. 在 Google 的基础架构上运行您的网络应用程序 [BE/OL]. [2009-10-17]. <http://code.google.com/intl/zh-CN/appengine/>.

Google Corporation. Run your web apps on google's infrastructure[BE/OL]. [2009-10-17]. <http://code.google.com/intl/zh-CN/appengine/>.

(责任编辑 王芷)