

基于 SCORM 规范的学习管理系统的设计与实现^①

Design and Implementation of Learning Management System Based on SCORM

颜仁泉 吴丽萍 朱新华 张 红 寇丽娟

(广西师范大学 计算机科学与信息工程学院 广西 桂林 541004)

摘 要: SCORM2004 规范解决了远程教育资源在教学平台中缺乏互操作性、可重用性等问题,本文在分析 SCORM 规范中定义的“运行时间环境模型”的基础上,设计实现了一个基于此规范的学习管理系统。
关键词: SCORM 规范 学习管理系统 可共享内容对象

1 引言

随着互联网的不断发展,远程教育以其独特的优势,日益成为教育技术与计算机辅助教学中的主流发展方向。但是其中也存在着不少问题,其中最为突出的就是互操作性的问题^[1],具体表现在两个方面:(1)由于学习资源中的格式和组织方式的不同,导致学习资源的大量浪费和重复开发;(2)不同的远程教育系统之间不兼容,学习资源无法在不同的系统间进行移动。为解决上述问题,国际上有众多的组织在研究和制定远程教育技术规范^[2],比如美国 IEEE 学习技术标准委员会 LTSC 制定的 1484 标准,美国 IMS 全球学习委员会发布的系列学习技术规范,以及由美国联邦政府下的高级分布式学习 ADL 制定的 SCORM 规范。其中 ADL 2004 年发布的 SCORM 2004 Edition 给出了一套制作、使用、传播、运行与管理远程教育教学产品的完整框架。要推广远程教育技术规范,首要任务就是要建立符合规范的教学管理平台。本文使用 Jsp/ Servlet 的技术设计实现了一个基于 SCORM 规范的学习管理系统(LMS)。

2 SCORM 2004规范概要

SCORM(Sharable Content Object Reference Model)可共享内容对象参考模型。SCORM 2004 规范中^[3]包含三大部份:一是 SCORM 内容集合模型

CAM (Content Aggregation Model),其目的是提供一个公共的方法,把学习资源组合成学习内容;二是 SCORM 实时运行时间环境模型 RTE(Run-Time Environment),它提供了一种公共机制,使学习资源可以在不同的学习管理系统中重复使用和具有互操作性;三是 SCORM 序列与导航模 SN(Sequencing and Navigation),主要是对学习内容对象(SCOs 或 Assets)的发送次序进行排列,而且能够允许 SCO(可共享内容对象,能与 LMS 进行交互)传递导航请求,为学习者提供导航控制能力。在这三部分中与学习管理系统设计密切相关的是运行环境模型。

SCORM 运行时间环境模型(RTE)描述了通用的内容对象运行机制,内容对象和 LMS(学习管理系统)之间的通用传递机制 API,以及跟踪学习者体验内容对象的通用数据模型。主要包括:

Launch: 为 LMS 定义了一个通用的方法来启动内容对象。在 SCORM 中定义了两种内容对象: Assesst 和 SCO。运行处理确立了在运行的内容对象与 LMS 之间的通信机制,这种通讯通过公共的 API 进行标准化。

应用编程接口(API):是内容对象与 LMS 之间传送的信息的一种通信机制,使用 API 可以开始、结束、获取、存储数据等动作。API 执行有三种函数,包括八个方法。如下表 1 所示:

^① 基金项目:广西教育厅科研项目(200707LX113)
收稿时间:2008-11-05

表 1 编程接口函数

函数	描述	函数名
会话函数 (Session Methods)	用于通过 API 实例标记 SCO 和 LMS 之间通信对话的开始和结束。	LMSInitialize () LMSFinish ()
数据传递函数 (Data-transfer Methods)	用于通过 API 实例交换 SCO 和 LMS 之间的数据模型值。	LMSGetValue(data model element,value),LMSSetValue(data model element,value),LMSCommit()
支持函数 (Support Methods)	用于通过 API 实例辅助 SCO 和 LMS 之间传递信息。	LMSGetLast Error() LMSGet ErrorSt ring () LMSGetDiagnostic ()

Data Model (数据模型): 描述了在 SCO 与 LMS 之间传送信息数据的模型, 如 SCO 的跟踪信息、SCO 的完成状态、测试成绩等数据。在学习者会话中, LMS 必须维护来自 SCO 数据模型的状态信息。而 SCO 需要利用这些预先定义的信息在其内部设计中使用, 以便在不同的 LMS 中重复使用。

3 基于SCORM规范的LMS的实现

3.1 系统功能设计

本系统的用户主要有两类: 学习者和管理者。学习者登录系统, 选择课程学习; 管理者导入课程, 并对学习者进行管理。图 1、图 2 给出了两者所实现的功能。

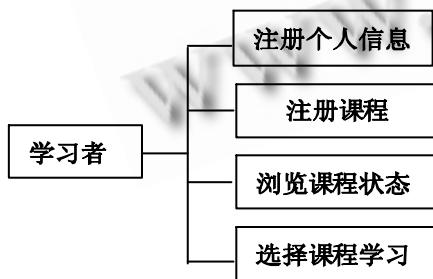


图 1 学习者功能

3.2 系统工作原理

一个符合 SCORM 规范的 LMS 的运行环境包括了服务器端及客户端两部分。LMS 服务器端包括后台数

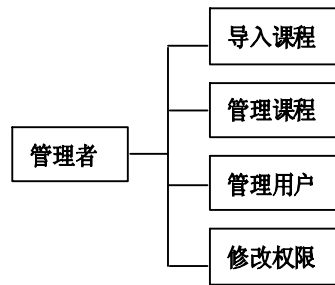


图 2 管理者功能

据的管理与通信数据的维护, 在客户端 LMS 还需要向用户提供一个 API 实例, 运行在客户端的 SCO 通过网络浏览器获取该 API 实例。API 提供了 SCO 与 LMS 沟通的管道, 通过该实例, SCO 与 LMS 进行交互, 依据数据模型(Data Model)进行通信。如图 3 所示:

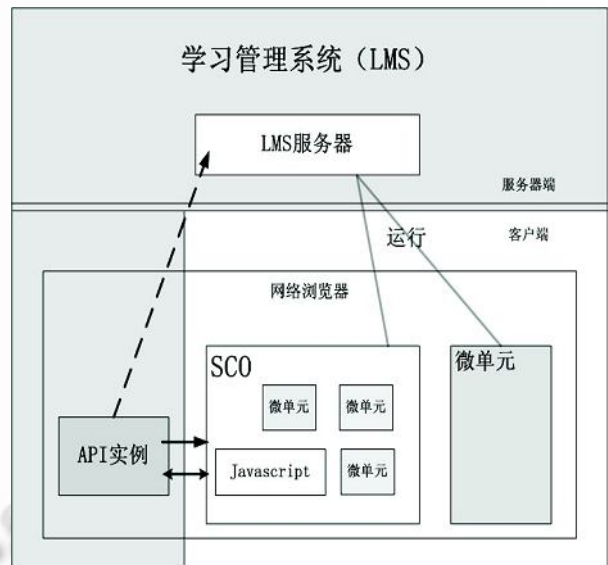


图 3 SCORM 运行时间环境(RTE)概念模型

3.3 API 适配器的设计

SCO 被启动后, 需要通过 API 适配器来实现与 LMS 进行数据模型的交互, 因此 API 适配器设计是本系统的一个关键技术。它必须实现了 SCORM 运行环境中所定义的编程接口 APIInterface 的所有函数, SCORMAPIInterface 接口函数的原型定义如下:

```

public interface SCORMAPIInterface {
    public abstract String LMSInitialize(String s);
    public abstract String LMSTerminate(String s);
    public abstract String LMSGetValue(String s);
    public abstract String LMSSetValue(String s,

```

```
String s1);
public abstract String LMSCommit(String s);
public abstract String LMSGetLastError();
public abstract String LMSGetErrorString
(String s);
public abstract String LMSGetDiagnostic
(String s);}
```

下面以 LMSGetValue 为例来说明 API 适配器的设计。

```
public String LMSGetValue(String s){
    if ( this.isLMSInitialized != true ) {
        String emptyString = new String("");
        Return emptyString;}
    String rtnVal= null;
    LMSDataModelInterface dmlInterface=new
    LMSDataModelInterface();
    rtnVal=dmlInterface.processGet(s,theSCOD
ata);
    return rtnVal;}
```

由于 SCO 是通过浏览器与 LMS 交互, 而 API 适配器的实例是由 LMS 提供的, 需要从系统启动时自动运行。Applet 是 Java 客户方小程序, 编译后可以直 接嵌入到 HTML 文件中, 在浏览器中运行^[4], 因此本 系统采用 java Applet 来实现 APIAdapter, 并在系统 主框架页面的 onload 事件中调用 init()函数部署, 其 中 IEEE 标准为远程学习系统的 API 实例对象定义了一个 强制性名字: API_1484_11。

```
function init(){API_1484_11=this.Document.
APIAdapter;}
```

3.4 SCO 与 LMS API 间接口的交互

SCO 要实现与 LMS 之间的通信, 必需建立一个 javascript 文件, 命名为"APIWrapper.js"。APIWrapper.js"是一个呼叫 Adapter API 的接口。只要在 SCO 被 launch 时, 使用它的 findAPI() 去寻找 API Adapter, 找到后就会有一个 API 的对象可用, 而这个对象会 mapping 到 LMS 所提供的 APIAdapter 上, 从而就可以使用适配器中设计的函数与系统交互。具体实现如下:

```
var nFindAPITries = 0; var API = null;
var maxTries = 500; var APIVersion = "";
function findAPI(win)
{while ((win.API_1484_11 == null) &&
```

```
(win.parent != null) && (win.parent != win))
{ nFindAPITries++;
if (nFindAPITries > maxTries)
{ alert("Error in finding API instance -- too
deeply nested.");
return null;}
win = win.parent;}
return win.API_1484_11 ; }
function GetAPI()
{ if ((win.parent != null) && (win.parent !=
win)) API = findAPI(win.parent);
if ((API == null) && (win.opener != null)) {
API = findAPI(win.opener);
if (API != null) APIVersion = API.version;
}}
```

上述 findAPI 函数和 GetAPI 函数分别表示查找与 获取 API。例如, 利用课件制作工具在 SCO 中加入这 样一段 javascript

```
javascript:
var result = doInitialize();
if (result != true) { // handle error}
result := ReadURL("javascript:doInitialize()",
100)
function doInitialize()
{ var api = getAPI();
var result = api.LMSInitialize("");
if (result.toString() != "true")
{ var err = ErrorHandler(); }
return result.toString();}
function doTerminate(){...}
function doGetValue(name){...
var value = api. LMSGetValue (name)
...}
function doSetValue(name, value){...}
...
```

这样就实现了 SCO 与 LMS 之间的通信。

4 系统数据库的设计

4.1 课件库的设计

由于系统需要在运行之前导入所需要的课程, 所 以涉及到课程内容的存放问题。系统将所有课程放置

在服务器端的 COURSE 文件夹下,并将课程包解压后的数据写入数据库,并与数据库中的课程信息一一对应。当导入课程时,查询数据库,并获得一个不存在且不重复的编号(如 A-1),建立文件夹(A-1),然后将课程解压到该文件夹下,同时把课程信息写入数据库。

4.2 普通数据库的设计^[5]

用户管理:包括系统用户注册,修改,有效性。不同权限的人功能不同,学生有选课,与删除已选课程以及更新个人信息的功能,老师具有删除与更新课件的功能。同时设多个管理员,作为 Admin 具有删除用户,管理课程的权利。

课件管理:主要是课件的上传,存储,删除更新,包括对课件是否有效的设置。

学习进度管理:用于对学生学习情况的管理。

其中 UserInfo 用于对用户的管理,Admin 用于辨别用户是否有上传课件、修改、删除课件和删除用户的功能。而 Active 验证用户是否有效。

UserCourseInfo 用于管理用户选课情况。CourseInfo 实现于课件的上传管理与更新操作。

ItemInfo 则可以认为是 UserSCOInfo 在 LMS Server 端的映射,属于学习资源信息表。通过编写一个 LMSManifestHandle 类来对该表处理,其中 data_from_Lms 表示 sco 启动时得到的初始化数据。UserSCOInfo 比较特殊,当 SCO 通过 Adapter 的打包,以 Data Model 实现了与 LMS 的通信后才起作用,主要是用来记录用户学习某门课程的学习历史。

程。其中 SCO_ID 与 imsmanifest.xml 中的 identifier 对应,表示 sco 的编号。Lanuch_address 用于保存课件的存放路径。LessonStatue 用于表示用户对 sco 的完成情况,其值只有 Complete 和 Incomplete,而 Sequece 则存放当前的学习进度,例如知识点 Rule1 (对应 identifier 为 SCO1) 就记录为 1。score 表示当前的测试成绩。

5 结束语

随着网络技术的快速发展,远程教育技术也将会进入一个高速的发展阶段。本文在兼容国际标准规范 scorm 的基本上分析及设计了一个简单的学习管理系统(LMS),目前系统已基本完成,但是还存在一些问题需要在以后的研究中解决,如协同学习的学习管理问题,以及 SCO 学习进度在 LMS 之间的互相移动问题等。

参考文献

- 1 沈中南,史元春.现代远程教育技术规范简介.计算机工程与应用,2003,39(5):66.
- 2 朱新华.基于规范的可共享智能型 Web 课间的制作.计算机工程与应用,2007,43(7):205-206.
- 3 SCORM 规范.http://www.adlnet.org/scorm.
- 4 张维中.实现基于 HTTP 的 servlet 与 applet 的通信.计算机工程与设计,2000,21(6):16.
- 5 陆峥,杨路明,宋虹.基于 SCORM 规范的教学管理平台的开发实现.计算机与数字工程,2005,33(3):83-84.