

插件式考试平台实例^①

梁 武

(广东行政学院 信息技术教研部, 广州 510053)

摘 要: 基于计算机的考试日渐推广, 构建一种能灵活配置、适应性强、高效稳定而又具备二次开发能力的考试平台架构, 是此类考试大规模推广应用必须解决的问题。插件式考试平台是一种松散耦合的开放型平台, 以可更换的插件为基础, 以统一的接口规范为标准, 既满足题型的灵活配置要求, 又能增强系统的稳定性和兼容性, 提高系统开发效率。

关键词: 插件; 架构; 反射机制; 自动评分

Test Platform Based on Plug-In

LIANG Wu

(Department of Information Technology, Guangdong Institute of Public Administration, Guangzhou 510053, China)

Abstract: With the Increasing promotion of computer-based test, how to construct a flexible configuration, adaptability, highly efficient and stable test platform becomes more and more important. The test platform based on plug-in is a loose coupling open-type system which works with the removable plug-in unit and the unified interface specification, it cannot only realize the flexible configuration of test items, but also enhance the stability of the system and the compatibility.

Keyword: plug-in; refraction; scoring; architecture

1 引言

目前, 考试依然是选拔人才和评估教学质量的重要手段之一。就考试方式而言, 主要有传统的考试方式和基于计算机的考试方式两种。传统的考试方式主要有笔试、口试、面试和实际操作等几种方式。由于各种资源条件的限制, 如今全国性和全省性的大规模考试还是使用纸笔考试模式为主的传统考试模式, 这种纸笔考试模式分为编写试题、印制试题、组织考试、评卷统计等 4 个环节, 存在不少的问题: 一是人力物力资源消耗大, 效率低; 二是管理中间环节多, 保密工作困难; 三是考试作弊机率高; 四是考试重理论轻实践, 难以体现考生的实际操作能力。基于计算机的考试方式主要由供考生使用的考试终端、控制考试过程的考试服务器端、管理试题的题库端及对考试结果进行分析统计的管理端等组成, 实现考前、考中和考后的全无纸化和网络化管理, 既提高了考试效率, 也解决了传统考试方式下一些难以解决的问题, 如集体

作弊、自主学习与测试等, 深受教育界人士的喜爱。就此, 国内外已有一些课程陆续采用基于计算机的考试方式, 如全国计算机等级考试、各省区的计算机水平考试、英语考试等。不足的是这些考试系统都不同程度存在着以下的问题:

(1) 绑定考试系统与考试科目: 这些考试系统在系统开发时就已确定能考的课程和所有的题型, 或者说科目、题型等与考试系统之间的绑定是在代码级上进行的, 而不是在结构级上决定的, 日后要添加新的课程、新的题型等必然涉及大量的代码修改, 因而极大地影响系统的稳定性和可靠性。当然, 可以借助“抽象工厂”设计模式在一定程度上减少各部份之间的耦合性, 但这也无法避免系统代码的修改。笔者曾在设计“广东省计算机水平考试”和“全国英语应用能力 A、B 级考试”平台时便采用了“抽象工厂”设计模式^[1], 共享了大量的系统核心代码, 减少了各科目之间的耦合性。但考试要求的变更, 尤其是题型和

^① 基金项目: 广东省教育厅项目(粤教高函[2009]10 号 D01)

收稿时间: 2010-07-11; 收到修改稿时间: 2010-08-27

科目的变更,导致了系统代码的频频调整,还是给系统的稳定性和可靠性留下不少隐患。

(2) 采用封闭式平台结构:与时下流行的开放式系统结构相反,目前大部分优秀的考试平台都采用了封闭式的平台结构,不具备任何二次开发能力。只有平台的开发者才知道怎样给平台增加一种新的题型、一门新的课程,很难满足不同应用环境的要求,也就大大地限制了考试平台的推广应用。笔者在开发“广东省高校公共课程资源共享型学习平台”时便遇到该问题。由于该平台最终要对不同类型的课程进行管理,并同时搭建一个网上实时能力测试的环境,即学生在学习过程中可随时登录测试中心对当前所学的内容进行测试,对此,学习平台集成了一套考试系统。然而,封闭的考试平台结构很难满足各学校对不同课程和题型的要求。

因此,寻求一种能灵活配置、适应性强、高效稳定而又具备二次开发能力的考试平台架构,是考试平台大规模推广应用必须解决的问题。插件式的考试平台向外提供统一的接口规范,以可更换的插件为基础,能更好的实现系统的自由配置及功能的变更,是一个不错的选择。

2 插件及其开发技术简介

在不改动系统核心源代码的前提下更改或扩充系统已有的功能,此类技术在操作系统中早有应用,最著名的应数 Windows 操作系统中外部设备的“即插即用”功能。从底层来看,这其实就是操作系统对设备驱动程序的一种管理机制。在操作系统中有一张规范统一的表,或者称为“接口统一”的表用于维护当前系统中已有的设备驱动程序,当你增、删系统中的设备时,Windows 会利用该表自动处理与之对应的设备驱动程序,并确保使用正确的驱动程序驱动相应的设备,从而达到任意扩充或更换外部设备的目的,而这些设备驱动程序其实相当于一个个独立的插件。现在有不少著名的应用程序,如 PhotoShop、Eclipse 等,也采用了该思路对系统的功能进行扩充和调整。在 PhotoShop 中,只要遵照一定的规范,用户便可以开发自己的滤镜插

件插入到 PhotoShop 中,从而扩充系统对图像的处理能力。而 Eclipse 更明显,该平台的所有功能都是由一个个插件来实现的,所有插件由 Eclipse 统一管理,通过加插不同的插件,用户即可构建不同的应用开发环境。

插件的应用开发技术主要包括早期 windows 平台下广泛使用的 COM+接口技术^[2]和目前较为流行的接口反射技术。COM+接口技术的原理是把系统划分成一个个抽象接口(Interface),同一接口可以有不同的实现(Implement)版本,即同一接口可有不同的实现代码逻辑,系统在运行过程中通过查询相应的接口实现(Interface Implement)来完成所要的功能,通过更换不同的接口实现(Interface Implement)即可变更系统的功能。而接口反射技术则是利用编程语言所特有的反射机制(目前 java 和基于 .Net 的开发语言均支持该机制),在程序运行过程中动态获取程序集中的类型信息,并根据相应的类型信息动态创建对应的接口实例。其基本思路与 COM+技术相似,但代码实现细节较之更为简洁。例如,在 VC++.NET 中,可以通过以下的步骤实现插件的定义及使用。

(1) 定义插件接口。此处的接口名称为 IShow,并定义了需要实现的 Show 方法。

```
public interface class IShow
{
    // TODO: 在此处添加此类的方法。
public:
    String ^Show();
};
```

此处的 Show 方法只是定义了该方法的签名,如返回值、参数等,并没有具体的实现代码,或者说该 Show 方法只是一个“空壳子”,用于告诉 IShow 接口的使用者,你必须按照我的签名要求实现你自己的 Show 方法。因此,对接口的定义实质上是提供该接口的规范、不涉及具体的实现代码,而具体的实现代码则由该接口的使用者自行定义,我定标准、你去实施。

(2) 实现插件接口。由于接口的定义只是定义了

接口的规范,如果要使用它则必须提供相应方法的实现代码,并封装在符合该接口规范的类中。下例则实现接口 `PlugInA`,并用类名 `PlugInA` 表示,在所实现的方法 `Show` 中只是简单地输出一串字符串“ShowOK”。

```
public ref class PlugInA : public IShow
{
    // TODO: 在此处添加此类的方法。
public:
    virtual String ^Show()
    {
        return "ShowOK";
    }
};
```

(3) 动态实例化插件实例。实现了符合接口规范的自定义插件后,便可在程序运行过程中动态实例化该插件。通常按以下的步骤动态实例化并使用插件:

- ① 找到要实例化的插件。插件通常放在某一目录下,扩展名为 `DLL`
- ② 利用 .Net 的反射机制获取该插件的接口并实例化
- ③ 利用取得的已实例化的插件接口调用插件的指定方法

在 `VC++.NET` 中,可采用以下的代码实例化某一插件并调用该插件的指定方法。

```
//1. 获得插件文件名称(通常扩展名为 DLL),
sFile 为插件的文件名称
if (sFile != String::Empty)
{
    String ^ asmName =
System::IO::Path::GetFileNameWithoutExtension(sFile);
    //2. 利用 .Net 的反射机制,从程序集(DLL)
中,提取类,并把此类实例化
    IShow ^iShow;
    List<Type ^>
mType=Assembly::LoadFrom(sFile)->GetTypes();
    for(int i=0;i<mType.Count;i++)
    {
```

```
        Type ^t=mType[i];
        Type
^mShow=t->GetInterface("IShow");//取指定的接口
        if (mShow)
        {
            //找到,实例化
            iShow = (IShow
^)System::Activator::CreateInstance(t);
            //调用该插件的 Show 方法
            iShow->Show();
        }
    }
```

在上例中,只要用户遵照 `IShow` 接口的规范,即可根据需实现不同版本(功能)的插件,如 `PlugInA`、`PlugInB`、`PlugInC` 等,在这些插件中实现各自功能相异的 `Show` 方法,并根据不同的应用环境要求更换这些插件。尽管所用的插件改变了、功能也可能变更了,但这并不影响实例化并使用这些插件的程序(即插件管理程序),因所作的代码改变被局限在插件内部(实际上各个插件都是相互独立的,它们之间只是具有相同的接口而尔),因而插件技术在确保主程序(插件管理程序)稳定性的前提下提供了灵活扩展系统功能的方式。

3 实例研究

“广东省高校公共课程资源共享型学习平台”是广东省教育厅委托我校开发的一个资源共享自主学习平台,其宗旨是把各高校的优质教学资源共享出来,并向学生提供一个具有智能化答疑、实时化学习能力测试、社区化服务等功能于一体的全面而高效的自主学习平台。所谓实时化学习能力测试是指学生在学习过程中,可连接到能力测试服务器并根据当前所学的知识点以一定的测试方式检测自己的学习掌握情况,其中测试方式包括在线考试。在线考试完成试题选择、试题呈现、考试控制、考试评分等功能,并分析学生的作答情况,为其进一步学习提供指导性建议。这一切都是由一套完整的考试系统来实现的。

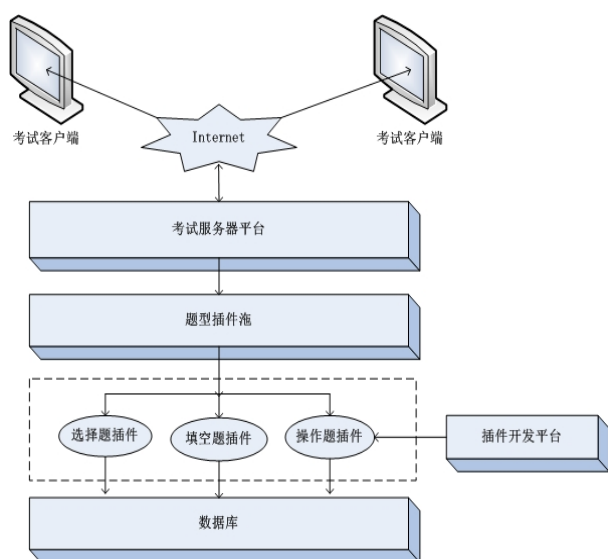


图1 “广东省高校公共课程资源共享型学习平台”
插件式考试平台架构

该学习平台涉及的课程很多，每门课程对考试的要求也各不相同，单论题型方面，有些课程只有简单的选择题、判断题和简答题等，而有些课程则有较为复杂的操作题，如 Office、Photoshop、网页设计等，甚至有些课程还要求有录音，如英语口语测试等。如果按照传统的设计思路，在一套考试系统中实现所有课程的测试要求，必将导致系统变得十分复杂而难以维护。为此，笔者设计了如图1所示的“广东省高校公共课程资源共享型学习平台”考试平台架构。

该架构采用插件技术，把原来耦合在系统中的针对各题型进行管理的模块分离出来，由相互独立的插件来完成，而系统只负责对各插件的初始化及使用。如果需要增加一种题型，只需增加一个针对该题型进行管理的插件即可，可大大降低系统的复杂性。该架构主要由考试客户端、考试服务器平台、题型插件池、题型插件、插件开发平台和数据库系统接口组成，各部份的主要功能如下：

(1) 考试客户端。由传统 WEB 浏览器和 WAP 智能终端组成，基于 Web2.0 标准，提供一个试题显示及作答的操作环境。

(2) 考试服务器平台。担当“管理者”的角色，维

护和管理各考试客户端的状态，初始化和撤销题型插件池，利用各题型插件完成相应题型的成卷、评分等操作。

(3) 题型插件池。完成所有题型插件的加载及实例化，并以链表方式把各题型插件保存在内存中，响应考试服务器平台的查询及调用请求。

(4) 题型插件。特定题型的实现逻辑，如选择题、填空题等。

题型插件是题型接口的实现(Implement)。题型接口定义了各题型必须遵守的规范，主要包括该题型的名称、如何呈现试题及如何评分等。按照该规范，各题型插件必须至少实现如下方法(Method)：

① GetStType：告诉服务器平台，本插件实现的是何种类型的试题，如选择题、填空题、操作题等。

② CreatePaper：生成由 HTML 格式描述的试题内容，该内容最终通过服务器平台传送到考生的考试客户端。

③ GetResult：实现对指定题型的自动评分。

(5) 插件开发平台。这是一个基于.Net、以框架为基础的题型插件辅助开发平台，采用自定义模板的方式，加插到 Microsoft 的 Visual Studio .Net 开发平台上，以便充分利用 Visual Studio .Net 平台的代码编写、测试、维护等强大功能，降低二次开发难度。该辅助开发平台主要提供以下功能：

① 规范插件实现定义：根据预定义的题型接口规范，自动生成插件实现(Implement)框架，编写各方法(如 GetStType、CreatePaper、GetResult 等)的标准签名(如返回值类型、参数个数、参数类型及顺序等)并提供详尽的参数注解及演示代码，以便题型开发人员忽略复杂的接口规范、把注意力集中在核心代码的编写上。

② 多语言支持：目前开发平台支持.Net 平台所支持的各种开发语言，如 VC++.Net、C#.Net 和 VB.Net。辅助平台会根据开发人员所选择的开发语言自动生成对应语言的接口框架。

③ 封装常用题型：平台还封装了部份常用题型的逻辑代码，如选择题(含单选题、多选题)、填空题、判断题等，开发人员可以直接使用这些题型的组卷、

评分等功能,也可以根据需要从这些题型派生自己的新题型或者重写默认的功能,以简化常用题型的开发。

④ 辅助功能:平台封装了大量便于简化题型开发人员编程的辅助方法,主要包括 HTML 编码(用于把输出编码成标准的用 HTML 描述的文本)、数据库访问、客户端与服务端通信等常规功能。

(6) 数据库系统接口

所有与试题有关的数据均保存在数据库中。插件开发平台提供了访问数据库的统一接口,插件开发人员可以通过该接口方便地访问底层的数据库,完成数据记录的添加、删除、查询和修改等操作,而接口底层则使用 .Net 的 LINQ 技术完成对数据库的操作。LINQ(Language-Integrated Query)是微软提供的新一代数据访问技术^[3],以简单而统一的方式访问所有数据源,不但可以访问像数据库、文件等传统的数据源,还可以操作数组、列表等数据结构及 XML。目前该接口支持 Microsoft 的 SQL Server、Oracle 等大型商用数据库。

由于系统中封装了部份常用的题型插件,使用者无须作任何修改即可使用;同时,系统还提供了开放的插件接口,只须遵循这些接口的规范,使用者便可

以利用简单的方式开发自己的新题型,并设计符合自己要求的评分逻辑,很好地解决了二次开发及题型扩充的问题。

4 结束语

采用松散耦合的插件技术来构造系统能很好地解决系统功能的随意扩展与系统稳定性之间的矛盾,尤其像考试之类的系统,在满足各种课程的不同要求时,更能体现插件技术的灵活性和高效性。如何定义插件的接口规范是设计插件式考试平台的关键,在此,笔者从实际应用出发,探讨了基于该技术的平台架构,并提供了一个简单的接口规范描述,为解决此类问题提供参考。截稿之日,利用该架构开发的学习平台能力测试系统已在广东省部份高校投入使用,运作正常,为学生的自主学习提供了实时化的能力测试功能,有力地支持广东省教学改革的发展。

参考文献

- 1 Gamma E, Helm R, Johnson R, Vlissides J.设计模式.北京:机械工业出版社,2002.
- 2 Eddon G, Eddon H.COM+组件编程技术内幕.北京:北京希望电子出版社,1999.
- 3 Ferracchiati FC.LINQ for Visual C# 2008. Apress,2008.

(上接第 131 页)

5 结束语

协同办公平台加强了部门、人员之间的交流,迅速获取信息、共享资源,为办公决策提供了大力支持,促进了办公活动的规范化和制度化。协同办公平台是企业实现现代化管理的有效手段之一,通过协同办公平台的实施,可以有效实现企业流程的规范化,大大提高企业内部的工作效率。目前协同办公平台已经投入使用,达到了功能设计要求和预期目标,取得了良好的应用效果。

参考文献

- 1 叶宇风.基于 SOA 的企业应用集成研究.微电子学与计算机,2006,23(5):211—213.

- 2 刘伯超,马晓轩,葛声.基于 Web 服务的软件服务体系结构的研究与实现.北京航空航天大学学报,2004,30(3):263—266.
- 3 程旷,张尧弼.协同办公平台项目实施的解决方案.计算机工程,2005,31(13):223—225.
- 4 单云,侯红,王萍.企业遗留系统集成平台的设计与实现.微计算机信息,2007,23:13—15.
- 5 韩祥斌,李峰,艾莉莎.基于 Struts 框架的短信办公平台软件设计与实现.重庆大学学报(自然科学版),2006,29:66—69.
- 6 杨彩,刘晓霞,贾松浩.基于 SOAP 的异构系统集成研究与实现.计算机应用与软件,2008,25(4):79—81.
- 7 李艳霞,巩九洲,黎玉琴.一种基于 Web Services 的信息集成方案.计算机技术与发展,2008,18(9):105—107.