

网站对外资源共享 Open API 的研究与实现

高 岭,宋红军,王 海,杨建锋

(西北大学 信息科学与技术学院,陕西 西安 710127)

摘 要:通过使用资源网站提供的平台 Open API,第三方开发人员不需要巨额的硬件和技术投资,就可以在该平台上开发商业应用,使用平台上的资源,从而解决了资源网站的信息孤岛问题。文中介绍了采用 Tuscany+Spring+DAO 开发模式,由于 Spring 具有 IOC 和 AOP 的特性,所以采用 Spring 来装配组件和控制权限,Tuscany 擅长系统的集成,通过 Tuscany 把组件发布成服务,实现了松耦合的 Open API 的开发,同时客户端 SDK 的引入,在降低耦合度的同时减少了第三方开发人员的开发难度。松耦合 Open API 的实现,具有层次清晰、维护方便的特点,降低了以后对平台系统升级、改造的影响。

关键词:开放 API;Tuscany;Spring;资源共享;SDK

中图分类号:TP31

文献标识码:A

文章编号:1673-629X(2013)04-0015-05

doi:10.3969/j.issn.1673-629X.2013.04.004

Research and Implementation of Open API Based on Website External Resources Sharing

GAO Ling, SONG Hong-jun, WANG Hai, YANG Jian-feng

(Information Science and Technology Institute, Northwest University, Xi'an 710127, China)

Abstract: By using Open API platform provided by resources website, the third party developers don't require huge hardware and technology investment, they can develop commercial applications to achieve the use of the resources of the platform, so as to solve the problems of resources website information island. It introduces it is using Tuscany+Spring+DAO development moderm to achieve Open API, Spring has the IOC and AOP's features, so using Spring to assemble components and control permissions. The Tuscany specializes in systems integration, so through the Tuscany the components are published as services, and this model realizes the loosely coupled Open API development. Client SDK is introduced on the client side, it not only decreases the degree of coupling of the system, but also reduces the difficulty of development of third-party developers. The loosely coupled Open API realization with clear layer, easy maintenance, reduces the impact of upgrade and renovation of the platform system in the future.

Key words: Open API; Tuscany; Spring; resource sharing; SDK

0 引 言

Open API 是伴随着互联网技术的发展,实现信息资源共享的新途径,它减少了大量重复代码的开发工作,减少了开发成本,成为越来越多的网站对外提供服务的必然选择。下一代网络是业务驱动的网络,下一代业务的目标就是把业务应用移到网络运营范围之外,向独立的业务供应商开放市场,同时为语音、多媒体和数据提供基于 IP 和移动方面的支持^[1]。Open API 将在下一代网络中发挥巨大的作用,它的出现真正地将业务提供与网络设备分离开来,是保证下一代网络的开放式体系结构的基础^[2]。

Open API 的好处:一方面,开放了资源的外部访问、调用,提供资源混搭的机会;另一方面,也保护了资源提供者,仅提供接口用于有限数量和频度的获取^[3];同时,开发者可以方便地访问 Open API 提供的资源,并且没有维护数据资源的麻烦。随着 Web2.0 时代的到来,互联网上社交网站、博客等 Web2.0 应用的大量出现,同时 Web 应用也在不断的信息化、多样化、个性化、分享化,Open API 将成为实现资源共享的必然选择。

1 Open API 主要采用的技术分析

实现 Open API 的技术有很多,这里主要采用 SOA (Service-Oriented Architecture,面向服务的架构)的思想理念,采用 SCA (Service Component Architecture,服务构件架构)的编程模型,结合 Tuscany 的开源实现,把网站资源对外发布成服务来实现 Open API,达到资源

收稿日期:2012-07-04;修回日期:2012-10-10

基金项目:陕西省科技厅“13115 科技创新项目”(2009fwpt-06)

作者简介:高 岭(1964-),男,陕西绥德人,教授,博导,博士,研究方向为计算机网络。

共享的目的。

1.1 SOA 技术分析

SOA 是一种面向服务的架构方法,SOA 的核心概念是重用和互操作,它将企业的 IT 资源整合成可操作的、基于标准的服务,使其能被重新组合和应用^[4],借助 Web 服务可以方便、安全、高效地实现企业应用集成^[5],同时 SOA 解决了令 IT 业界和企业都非常头疼的信息孤岛和遗留系统问题^[6]。为什么要选择 SOA 呢?现在应用中的软件是架构在不同的操作系统、不同的开发语言和不同的框架基础之上,现在的软件是用来处理当前的业务,对于新的需求,从头建立一个新的业务是费时费力的。SOA 具有松耦合的特性,利用现有的应用程序和应用基础架构的投资来实现新的需求,保护了现有的投资,并且满足了新的需求。

与传统的紧耦合架构相比,SOA 的松耦合的架构更能适应需求的变化。在 SOA 中,可以把已经实现的功能发布成服务而无需关心其底层的技术实现,唯一要考虑的就是发布的服务接口;SOA 另外一个好处是可以充分利用现有的系统资源,包括遗留功能和数据库资源,新系统可以方便地把它们纳入到 SOA,而不是替换它们,这样的架构能更好地适应系统二次开发中需求的变化。

现在有许多厂商支持 SOA,例如 IBM、Microsoft、BEA、Sun、Oracle 等,并研发出一些中间件产品。主流的 SOA 企业级产品包括:Microsoft Biztalk Server、Web-Methods Business Integrator、IBM SeeBeyond 和 TIBCO 等;主流的 SOA 开源产品包括:Tuscany、ServiceMIX、CXF 和 Synapse 等^[7]。

SOA 具有如下主要特点:

(1) 服务的客户端和服务端之间应该是松耦合的。

(2) 服务接口必须进行明确定义,即客户端如何绑定到服务端的接口必须进行详细的描述。

(3) 无状态的服务设计,即服务不依赖于其它服务的上下文信息和状态,服务端在任何时候收到请求都当作是一个新的请求进行处理。

(4) 推荐采用粗粒度接口,对外提供的服务要具有比较完整的功能,这样才有意义。如果粒度比较细,需要客户端多次访问服务端才能完成一个完整的功能,这样的服务不但开发不便,而且也会增加通讯压力,这样的服务适合做内部接口,不适合做对外服务接口。

1.2 Tuscany 的介绍

目前 SOA 标准中主要包括 SCA (Service Component Architecture, 服务组件架构) 和 JBI (Java Business Integration, Java 业务集成) 两个体系。SCA 提供了一

种支持 SOA 应用程序实现的编程模型,支持组件可以采用不同的语言来编写,同时支持实现服务组件的各种技术和绑定访问组件的各种协议。SCA 由于技术无关性及众多厂商的参与,当前支持的主流厂家比较多,Tuscany 就是其中之一。Tuscany 是 Apache 的开源项目,它是 IBM、Oracle、SAP 等厂商联合成立的 SOA 标准化组织 -OSOA 支持下开发出的 SCA 框架,它既是开源界 SCA 的试金石,也是当前开源界最成熟的 SCA 框架之一^[8]。Apache Tuscany 为 SCA 应用程序的开发和操作提供了易于使用的开放源代码基础结构^[9]。

Tuscany 的优点:

(1) 通过定义服务的结构和它们之间的关系,提供了一个创建应用模块的模型,同时服务的实现与具体技术无关。

(2) 实现了业务逻辑和协议的分离,开发人员只要创建包含业务逻辑的可重用服务,就可以通过添加不同的绑定实现不同的服务以供访问。

(3) 应用程序可以很容易地根据基础设施的变化而变化。

(4) 系统已有的功能能很好地和新的组合构建一起工作,便于系统的二次开发。

1.3 Spring 与 Tuscany 的集成

Spring 具有 IOC 和 AOP 的特性, Spring 和 Tuscany 实现集成,就可以充分发挥 Spring 的特性。开发者只要实现 POJO 类,创建、管理对象和切面上的管理就交给 Spring,接下来由 Tuscany 来实现协议解耦合语言的绑定,这样一个完美的架构就实现了^[10]。Spring 实现的构件能够引用 Tuscany 的服务和引用,而不必在 Spring 配置文件中声明,可以把所有的构建管理工作由 Spring 来完成;Tuscany 擅长系统的集成,所有的系统解耦合语言绑定就由它来实现,同时 Tuscany 也不会对 Spring 内部造成影响^[4],如图 1 所示。

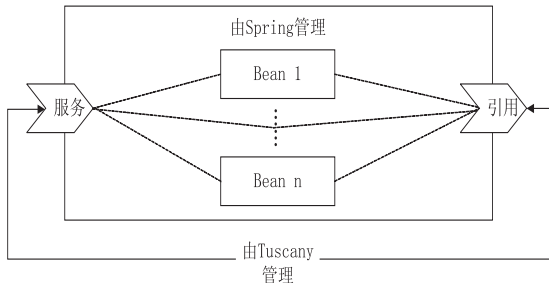


图 1 Tuscany 与 Spring 关系图

2 Open API 的实现

文中的代码是针对文档数字化与资源共享平台进行开发的,文档数字化与资源共享平台是“陕西省 13115”项目的一个子项目,该项目提供了一个省、市、

县科技档案资源有效整合、资源分类存储、以及资源加工处理的平台,该平台实现了省、市、县级资源有效共享,现在将系统中的资源以服务形式对外开放,供其它系统使用,同时也方便以后系统进行二次开发过程中的系统集成。

2.1 Open API 开发总结构图

文档数字化与资源共享平台中的资源是由文档、文章、视频、音频、图片五部分组成,分别把这五部分资源发布成服务,使其处在不同的目标命名空间中,分别为“http://document”、“http://article”、“http://video”、“http://audio”和“http://image”,各部分的总体架构如下:首先 DAO 通过数据库连接池连接 Oracle 数据库,然后由构件连接 DAO 实现各种功能,接着 Spring 把其装配成 Bean 对象,并通过 AOP 方式加入前置增强和后置增强,最后发挥 Tuscany 的特长,进行系统集成和协议解耦耦合语言的绑定,把由 Spring 装配的对象对外发布成 Web Service 供访问。

图2是访问 video 资源的序列图,其它资源的访问过程基本相同。

2.2 把系统的资源发布成服务

下面介绍如何把视频资源发布成服务,其它资源的构建方法类似。首先编写构件所需要的各种实现 DAO 类,主要是要实现操作数据库的功能,video 的 DAO 类为 VideoDAO.java;然后定义操作类接口 VideoOperationInterface.java,同时把接口声明为“@ Remotable”;接着定义接口的实现类 VideoOperation.java,通

过调用 DAO 实现各种功能;接下来由 Spring 进行装配,下面是 video 的 Spring 配置文件 springContext.xml:

```
<sca:service name = " VideoOperationService" type = " video.
spring. VideoOperationInterface" target = " VideoOperation"/>
<bean id = " VideoOperation" class = " video. spring. VideoOper-
ation"> </bean>
<aop:config>
<aop:aspect ref = " AOPRole">
<aop:pointcut id = " rolePointcut" expression = " target ( video.
spring. VideoOperation) and execution( * * (..))"/>
<aop:before method = " before_set_role" pointcut-ref = " role-
Pointcut"/>
<aop:after method = " after_del_role" pointcut-ref = " rolePoint-
cut"/>
</aop:aspect>
</aop:config>
<bean id = " AOPRole" class = " video. sca. AOPRole">
</bean>
```

Spring 配置文件中<sca:service>将 bean 对象开放为一个 SCA 服务,同时这个服务也要在 SCA 的.composite 配置文件中声明,并且名字要相同。<aop:config>面向切面编程引入了 AOPRole 增强类构件,目标类为 VideoOperation,当执行该类下所有的方法时进行增强,前置增强为 before_set_role,后置增强为 after_del_role。

下面是 Tuscany 通过 soa.composite 把 Spring 装配的内容发布成服务。

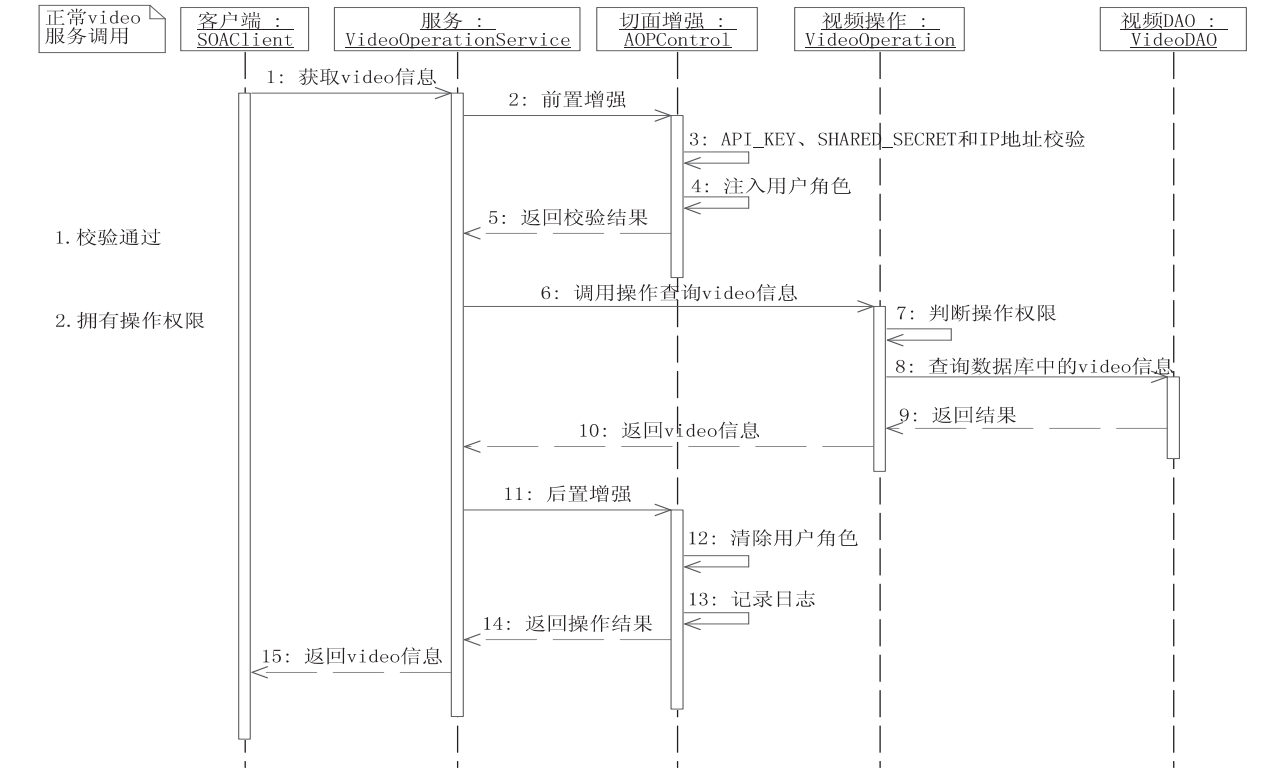


图2 video 资源访问序列图

```
<composite xmlns = " http://www. osoa. org/xmlns/sca/1. 0 "
targetNamespace = " http://video" name = " video">
    <service name = " VideoOperationService" promote = " VideoOp-
eration">
        <interface. java interface = " video. spring. VideoOperationInter-
face"/>
        <binding. ws uri = " http://localhost:8089/SOAService/video-
listWebService"/>
    </service>
    <component name = " VideoOperation">
        <implementation. spring location = " video/spring/springCon-
text. xml"/>
    </component>
</composite>
```

<component>标签用于声明由 Spring 装配的构件, <binding. ws>标签用于将服务绑定为 Web Service,uri 属性给出了发布的 URL 地址,localhost 为本机地址,实际部署的时候要换成服务器的 IP。

接下来还要编写服务的启动类 StartSOA. java,主要是通过 SCADomain. newInstance (" audio/sca/soa. composite")加载“soa. composite”组合构件文件来启动服务。

2.3 Open API 的权限控制

权限控制采用的是 RBAC (Role – Based Access Control)的方式,基于 RBAC 访问控制模型是目前公认的解决大型企业的权限管理问题的有效方法。RBAC 可以通过角色来分配用户的权限,在给用户分配权限时,只需要把一种或多种角色分配给用户,用户就会拥有这些角色对应的权限,具有重复率低、控制方便的特点。文档数字化与资源共享平台包括文档、文章、视频、音频、图片五种资源,每一种资源对应查询、添加、修改、删除四种权限,相当于具有 16 种角色,然后把每一种资源的权限用一种角色来表示,这样每个用户对应的五种资源的访问权限就可以用五种角色来表示。图 3 是本系统的权限控制图:

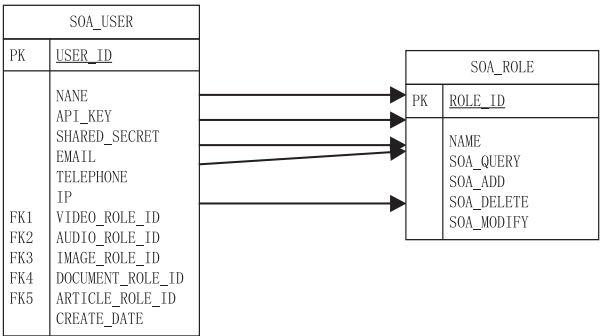


图 3 用户和角色关系对应图

2.4 Open API 的安全控制

客户端发出的请求和返回的数据必须确保是安全的、经过验证的事务^[11]。使用文档数字化与资源共享

平台的 Open API,必须进入相应的网址进行权限申请,系统会自动返回 API_KEY 和 SHARED_SECRET, API_KEY 是系统自动产生的十位字母数字混合编码, SHARED_SECRET 是系统自动产生的 32 位字母数字混合编码。系统在创建用户的时候会自动把用户的 API_KEY、SHARED_SECRET 和角色等其它信息存储到数据库 SOA_USER 表中,为了安全 SHARED_SECRET 在数据库中的存储是经过 MD5 加密后的内容。为了更好地进行访问控制,用户在申请 API_KEY 和 SHARED_SECRET 的时候必须提交使用资源的客户端 IP 地址,在客户端访问服务器端资源的时候,服务器端在进行 API_KEY、SHARED_SECRET 校验的同时进行 IP 地址校验,如果与申请时填写的信息不符,则拒绝访问。因为 Spring 具有 AOP 的功能,所以把所有的验证操作都放在构件的前置增强 after_del_role 方法中,如果验证通过,则根据用户具有的角色注入相应的权限,否则拒绝访问并返回错误信息。如果验证通过,在访问完成并返回信息的同时,通过构件的后置增强方法 after_del_role,解除用户的权限,防止非法用户对信息的访问。

2.5 Open API 的应用

要调用服务器端提供的服务,必须编写对应的客户端,传递 API_KEY、SHARED_SECRET 和访问服务的其它参数。下面是调用服务访问 video 前十个资源的客户端的主要代码:

```
RPCServiceClient serviceClient = new RPCServiceClient();
Options options = serviceClient. getOptions();
EndpointReference targetEPR = new EndpointReference ( " ht-
tp://localhost:8089/SOAService/videolistWebService" );//设定服
务提供者的地址
options. setTo( targetEPR );
QName opGetHello = new QName( " http://video", " video_
num_list" );//设定要调用的方法
Class[ ] returnTypes = new Class[ ] { Video[ ]. class };
String test [ ] = { " dezmljx7qd ", "
ef7fb02ef2c802f665c7a7d14ad5934c", "10" };//传入的参数
Object[ ] opGetHelloArgs = new Object[ ] { test };
//设定调用的方法的参数值
Object[ ] response = serviceClient. invokeBlocking( opGetHel-
lo, opGetHelloArgs, returnTypes );
Video[ ] result = ( Video[ ] ) response[0]; //得到调用的结
果
```

Tuscan y 是基于 Axis2 实现对外发布 Web Service, 所以客户端也基于 Axis2 进行调用^[4]。其中 Video 是和 VIDEO 数据库表对应的 POJO 类,服务器端返回的是 ArrayList<Video>类型的结果,然而 ArrayList<Video>的返回类型 Tuscan y 是不支持的,通过设置 Video[] 返回类型进行强制类型转换便可解决问题。其中 Vid-

eo 类存储着一个视频信息的名称、存储位置、视频标号等信息,这样就可以在其它地方调用平台资源了。同时为了方便程序员的开发,系统提供了客户端 SDK (Software Development Kit),SDK 包装了客户端访问服务器端的细节,提供了详细的 Open API 开发文档、示例代码、支持性的技术注解和支持客户端开发的 JAR 包,程序员在项目开发过程中只要导入 JAR 文件,就可以像使用普通的接口一样进行应用开发,客户端到服务端的操作细节由 JAR 文件来完成。

3 结束语

文中阐述了采用 Tuscany+Spring+DAO 模式,实现文档数字化与资源共享平台 Open API 的开发步骤。系统采用了松耦合的开发模式,文档数字化与资源共享平台在自身内部升级中对 Open API 的影响很小,在不改变数据库原结构和资源存储路径的情况下,甚至没有影响;Open API 的发布和构件的实现之间也是松耦合的,构件功能的变动和增减对 Open API 发布影响甚小;同时客户端 SDK 的引入,屏蔽了 Open API 对开发人员构建的新系统的影响。同时,实现了 API_KEY、SHARED_SECRET 和客户端 IP 地址的集中验证,访问权限的集中注入和解除,方便权限的变更。共享平台 Open API 不但方便在其它地方访问平台的资源,而且方便平台系统在以后升级、改造过程中的系统集成,同时对其它网站实现 Open API 具有一定的参考价值。

(上接第 14 页)

拟的全过程,着重介绍了投影网格,基于统计模型和 FFT 合成的实时海浪的模拟,尤其是基于统计模型和 FFT 合成,不但从理论上详细地给出推导,而且最后给出实验的过程及结果,以及最后简单地介绍了光照模型。实验结果还是非常不错,但为了更进一步增强整个海水的真实感,可以加入阴影,以及考虑实现交互性的海浪。

参考文献:

[1] 李向伟. 一种海水实时绘制方法的研究与实现[EB/OL]. 2005-10-18. <http://www.paper.edu.cn/index.php/default/releasepaper/content/200510-188>.

[2] Fournier A,Reeves W T. A simple model of ocean waves[J]. Computer Graphics,1986,20(4):75-84.

[3] Reeves W T. Particle System-A Technique for Modeling a Class of Fuzzy Objects[J]. Computer Graphics,1983,17(3):359-376.

[4] Johanson C. Real-time water rendering-Introducing the pro-

参考文献:

[1] 李余琨,杨平,朱桑权. 支持开放的 API 接口的增强型业务[J]. 计算机工程与应用,2004,10(4):63-68.

[2] 刘鹏,顾军,周勇. 面向下一代网络的开放式 API 技术研究[J]. 计算机技术与发展,2006,16(2):56-58.

[3] 裴珊珊,叶小梁. 国外 Open API 发展现状及趋势研究[J]. 情报科学,2009(12):1896-1900.

[4] 邓子云. SOA 实践者说:分布式环境下的系统集成[M]. 北京:电子工业出版社,2010.

[5] 余名高,贾秀峰,林坤江,等. 基于 Web 服务的企业应用集成[J]. 计算机技术与发展,2007,17(5):55-57.

[6] 凌晓东. SOA 综述[J]. 计算机应用与软件,2007,24(10):122-124.

[7] 梁爱虎. SOA 思想、技术与系统集成应用详解[M]. 北京:电子工业出版社,2007.

[8] Laws S,Combellack M,Feng R,et al. Tuscany SCA in Action[M]. Stamford:Manning Publications Co.,2011.

[9] Segal A,Beisiegel M,Delfino J S. Deploy an SCA application using the Tuscany domain manager[EB/OL]. [2008-10-02]. <http://www.ibm.com/developerworks/webservices/library/ws-sca-tuscany/index.html>.

[10] Ramalingam R. Design and develop SCA components using the Spring Framework,Part 1: The trifecta: Spring, SCA, and Apache Tuscany[EB/OL]. [2009-10-06]. <http://www.ibm.com/developerworks/webservices/library/os-springsca/index.html>.

[11] 刘泉,周颖,汪雪梅. 安全扩展的 UDDI API 实现架构[J]. 武汉理工大学学报,2003(10):77-80.

jected grid concept[D]. Lund:Lund University,2004.

[5] Mitchell J L. Real-time Synthesis and Rendering of Ocean Water[R]. [s.l.]:[s.n.],2005:2-4.

[6] 夏新华,潘志庚. 基于统计模型的海水运动仿真[J]. 计算机仿真,2005,22(1):62-63.

[7] Chiu Y F,Chang C F. GPU-based Ocean Rendering[C]//2006 IEEE International Conference on Multimeadia and Expo.. [s.l.]:[s.n.],2006:3-4.

[8] Rost R J. OpenGL 着色语言[M]. 天宏工作室译. 北京:人民邮电出版社,2006.

[9] 王纲,季振洲,张泽旭. 大范围动态海浪实时渲染[J]. 哈尔滨工业大学学报,2012,44(3):60-61.

[10] Shreiner D,The Khronos OpenGL ARB Working Group. OpenGL 编程指南[M]. 李军,徐波译. 第7版. 北京:机械工业出版社,2010:78-80.

[11] Perlin K. An image synthesizer[J]. SIGGRAPH Comput. Graph.,1985(19):287-296.

[12] 吕文静. 基于真实感图形技术的虚拟海面场景研究[D]. 天津:河北工业大学,2007.

网站对外资源共享Open API的研究与实现

作者: [高岭](#), [宋红军](#), [王海](#), [杨建锋](#)
作者单位: [西北大学 信息科学与技术学院, 陕西 西安 710127](#)
刊名: [计算机技术与发展](#)
英文刊名: [Computer Technology and Development](#)
年, 卷(期): 2013(4)

本文链接: http://d.g.wanfangdata.com.cn/Periodical_wjtz201304006.aspx