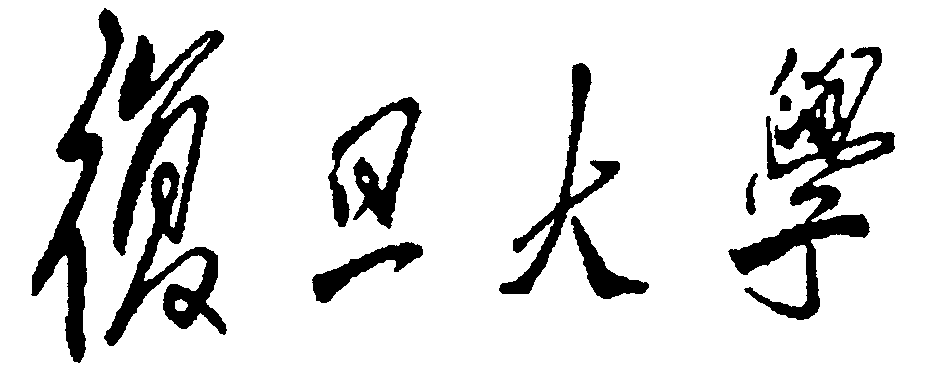
|  |  |
| --- | --- |
|  | 学校代码：10246 |
|  | 学号： |



|  |
| --- |
| 硕士学位论文 |

（专业学位）

|  |
| --- |
| 基于KVM虚拟化技术的私有云教学系统的构建  **Private Cloud Teaching System**  **based on**  **KVM Virtualization Technology** |

|  |  |
| --- | --- |
| 院系： | 软件学院 |
| 专业： | 软件工程 |
| 姓名： |  |
| 指导教师： |  |
| 完成日期： | 2018年2月28日 |

# 目录

[摘要 1](#_Toc509688545)

[ABSTRACT 2](#_Toc509688546)

[第一章 绪论 4](#_Toc509688547)

[1.1 论文背景 4](#_Toc509688548)

[1.2 国内外相关研究和现状 5](#_Toc509688549)

[1.3 论文的内容与意义 6](#_Toc509688550)

[1.4论文结构 8](#_Toc509688553)

[第二章 教育私有云系统与KVM虚拟化技术 10](#_Toc509688554)

[2.1 教育私有云系统概述 10](#_Toc509688555)

[2.2 KVM虚拟化技术 13](#_Toc509688558)

[2.3 其他相关技术介绍 17](#_Toc509688561)

[2.4 本章小结 20](#_Toc509688565)

[第三章 教育私有云系统需求分析与架构设计 21](#_Toc509688566)

[3.1 私有云教学系统的需求分析 21](#_Toc509688567)

[3.1.1 云计算在教育领域的现状 21](#_Toc509688568)

[3.1.2非功能性需求分析 22](#_Toc509688569)

[3.1.3 功能性业务需求分析 22](#_Toc509688570)

[3.2 私有云教学平台总体设计 25](#_Toc509688571)

[3.2.1系统总体架构设计 25](#_Toc509688572)

[3.2.2虚拟化层和资源调度层设计 27](#_Toc509688573)

[3.2.3 业务应用层设计 28](#_Toc509688574)

[3.3网络拓扑架构设计 30](#_Toc509688575)

[3.4本章小结 32](#_Toc509688576)

[第四章 教学私有云系统的实现 34](#_Toc509688577)

[4.1 教学资源虚拟化 34](#_Toc509688578)

[4.1.1 硬件虚拟化的实现 34](#_Toc509688579)

[4.1.2 教学云桌面的实现 40](#_Toc509688580)

[4.2 教育私有云中间层的实现 43](#_Toc509688581)

[4.2.1 注册和垃圾回收的实现 43](#_Toc509688582)

[4.2.2 中间层通信机制的实现 46](#_Toc509688583)

[4.3 教育私有云应用层的实现 51](#_Toc509688584)

[4.3.1 云桌面访问控制的实现 51](#_Toc509688585)

[4.3.2 教学云桌面制作和备份的实现 53](#_Toc509688586)

[4.3.3 私有云客户端的实现 55](#_Toc509688587)

[4.4成本效益及系统性能分析 56](#_Toc509688588)

[4.4.1 成本效益分析 56](#_Toc509688589)

[4.4.2 可靠性和可用性分析 57](#_Toc509688590)

[4.5 本章小结 59](#_Toc509688591)

[第五章 总结与展望 60](#_Toc509688592)

[5.1 总结 60](#_Toc509688593)

[5.2 展望 61](#_Toc509688594)

[致 谢 62](#_Toc509688595)

[参考文献 63](#_Toc509688596)

[论文修改说明 65](#_Toc509688597)

# 摘要

目前传统的教育行业中，基于多媒体资源的教学模式在国内大部分地区已经常态化, 教师利用个人电脑授课成为主流，众多学校为此都建有教育专网及多媒体教室，由此带来硬件管理复杂性、计算资源利用率不足的问题普遍存在。基于虚拟化技术的云计算系统是解决上述问题的有效手段之一。

为满足教育行业云计算基础设施建设的需要，促进教学应用的统一管理和教学资源的安全共享，满足面向未来教室的新型教学模式的探索，本文提出了一个基于开源技术的私有云教学系统。该系统包括采用KVM（Kernel-based Virtual Machine）虚拟化技术的虚拟资源管理系统、云服务管理系统、教学终端、后台管理系统等几部分。虚拟资源管理系统实现了物理硬件的虚拟化，能为日常教学提供虚拟桌面和镜像存储服务；云服务管理系统解决了虚拟资源的自动调度的决策问题；教学终端和后台管理系统为教职工和IT管理人员提供了系统使用和管理的入口。

本文基于一个真实的，在高校多媒体教室部署实施的私有云系统，以及在实施过程中的遇到的真实需求与难点，分析了目前云计算在教育行业落地的现状和挑战，梳理出私有云在校园落地的规划思路，阐述了系统的功能性需求和性能指标。随后详细描述了系统的架构设计与核心模块组成及其软件实现的过程。针对原有系统存在的缺陷，设计并实现了将多媒体教室电脑结合KVM虚拟化技术作为边缘计算节点融入到整个云计算系统的实施方案，完成计算资源由中心机房向网络边缘的扩展，进一步降低私有云建设的前期投入和管理难度。同时，针对原有项目在教学场景下，消息处理能力不足的问题进行改进，并分析了改进以后系统的可用性与可靠性。分析结果表明该系统能够满足云计算辅助教学的基本需求，有效提升现有硬件的利用率、解决教学电脑的管理复杂和资源浪费的问题。它还能为老师授课活动提供可定制、统一管理而且运行稳定的云桌面服务，从而有利于真正实现云计算系统建设在教育行业的落地。

关键词 KVM，虚拟化，私有云，云桌面，IAAS

# ABSTRACT

In traditional education field, teaching based on multimedia re-sources has become more and more popular in most parts of our country. Teachers usually use personal computers in their class. Many universities have built many multimedia classrooms. Therefore, the complexity of hardware management and the problem of insufficient utilization of computing resources are widespread. Cloud computing system based on virtualization technology is one of the effective solutions to solve above problems. To promote the unified management of education applications and resources, as well as explore a new teaching models in the future, this thesis describes a private cloud computing teaching system based on open source technology. The system includes virtualization management system, cloud management system, user client and administrator management system , which is based on KVM(Kernel-based Virtual Machine) technology. Virtualization management system implemented the virtualization pool for hardware, which can provide virtual desktop services for daily teaching. Cloud management system controls and manages virtual resources. User client is the interface for user to access virtual desktop as well as administrator management system provide management for faculty and IT administrators.

This thesis describes the architecture design and software implementation for the core modules of this system. The personal computer in multimedia classroom is integrated into the cloud computing system as a computing node via KVM. The solution of computing resources from server to the network edge highly reduces the investment and management complexity. At the same time, this thesis solve a problem of insufficient message processing ability. The analysis results show that it can meet the needs of cloud computing system to assist teaching. It also improves the utilization of existing hardware in most effective way. Furthermore, it provides a customizable cloud desktop for teachers, which is helpful for an education institutions to build a proper cloud computing system in the correct direction.

Keyword　KVM，Virtualization，Private Cloud，Cloud Desktop，IAAS

# 第一章 绪论

## 1.1 论文背景

云计算的相关技术，如虚拟化、分布式集群、容器技术、云存储、云安全等，经过十多年的发展取得了长足的进步。云计算从当初概念普及逐步转变为实际生产力服务于各行各业。企业开始重视云计算对自身行业领域效率提升、成本节约上带来的变化，纷纷力求通过云计算技术取得行业内的竞争优势。

在教育领域，教育信息化建设在国家相关政策的支持下不断深化。2015年国务院发布的《关于促进云计算创新发展培育信息产业新业态的意见》中明确指出：“到2020年，云计算应用基本普及，云计算服务能力达到国际先进水平，掌握云计算关键技术，形成若干具有较强国际竞争力的云计算骨干企业。”[1]。这使得与垂直领域紧密结合，大力发展云计算产业，加快云计算布局得到国家政策上的扶植。国内教育行业信息化建设速度不断加快。2017年4月全国教育信息化工作会议总结了十八大以来我国在教育信息化方面取得的卓越成效。会议上提到过去五年我国多媒体教室的比例从40%增加到了目前的80%以上，学校网络的网速和带宽性能和过去相比大幅改善，有很多地区已经开始启动教育城域网、无线教育网、校园网及未来教室的试点[2]。硬件及网络基础设施不断完善使得教育信息化创新的基本条件越来越有利。教育行业成为许多先进技术实践落地的前站。过去三年，教育行业的虚拟化与云桌面市场在持续快速增长。一方面，学校的服务器性能、网络环境和硬件设施稳步完善，使得云计算实施风险显著降低；另一方面，利用云计算和虚拟化技术也为教育领域 “未来教室”，教育资源的互联互通，硬件资源的有效利用，新型教学模式的深入探索提供了必要的技术支撑。

本文的项目背景是为华东某高校的多媒体教室和学生机房提供私有云解决方案。该高校目前拥有300间左右的多媒体教室，全部配备了基于还原卡系统的中控设备，但是电脑的性能和配置存在差异。校园内铺设了独立的教学专网，带宽性能是千兆主干网，百兆到终端。软件环境更为复杂，教室电脑中提供两套Windows XP系统，一套上课使用，一套备用。所有老师上课需要的单机版软件，全部都安装在一个操作系统上。软件更新时，必须关闭还原卡功能，等安装完毕再打开。基于C/S（Client/Server）或是B/S（Browser/Server）架构的教学应用都分布在不同中心机房的不同服务器上，形成信息孤岛，很难共享。这都给老师的使用和IT人员的维护工作带来了不便。由于高校的大部分教学应用都运行在教学专网，很难将其搬迁至商用的公有云服务，高校本身也对公有云服务的稳定性、安全性和采购价格存在疑虑，所以希望通过自建一套的定制化的私有云系统解决其软硬件资源维护困难，利用率低、管理复杂等问题。该项目背景在国内高校中具有一定的典型性和代表性，有一定的研究价值。

## 1.2 国内外相关研究和现状

首先从虚拟化和云计算技术发展的角度看，国外商用的虚拟化技术主要有VMware和微软Hyper-V。在开源的技术中，原来主流的Xen脱离红帽Linux后，发展越来越缓慢，2017年AWS(Amazon Web Service)也开始提供基于KVM的虚拟机服务，使得KVM已经取代Xen后来居上成为主流的平台虚拟化开源技术。据统计，截至2017年底超过70%以上开源虚拟化应用采用KVM。国内云服务提供商大部分都基于OpenStack + Linux + KVM的架构上来构建云计算系统。近年来，开源的云计算平台涌现了大批新的开源技术和应用，如Spark、Docker、TensorFlow、Kubernetes等，大大促进了云计算生态的发展。

从公有云和私有云的发展现状看，过去三年以亚马逊AWS为首的国外公有云厂商发展势头强劲。截至2017年底，亚马逊AWS、微软Azure、Google和IBM已经占据60%-70%的公有云市场。与此同时国内的私有云发展明显好于公有云。原因之一是国内企业对公有云的付费方式和信息安全仍然存有疑虑，加之国家大力推行对政务、金融、教育、交通、制造等行业云计算基础设施建设的扶植政策，使得包括教育行业在内的各行业逐步加大对私有云基础设施建设的投入。

当前校园的硬件设施具备了云计算落地教育领域的前提条件，加之云计算技术更加成熟，云计算已经适合在教育领域开展商业化尝试。从教育行业相关研究和应用的角度看，基于云平台的各类软件服务发展迅猛，但是云基础设施的研究发展缓慢。从国外IBM提出的智慧教育开始，基于谷歌APP的在线教育资源，还有国外很多高校已经开始将教学视频上传至Youtube进行在线教育。同时,基于云计算的MOOC（Massive Open Online Courses）教学也发展迅猛。国内有慕课网、MOOC学院等网站提供云教育的服务。有些高校开始部署基于云计算平台发展起来的LES（Learning Management System）学习管理系统。国外比较著名的提供软件即服务的LES平台有NEOLES。上述基本上都是基于云计算软件即服务方式向客户提供的教育业务应用服务。

与云上教育应用飞速发展形成鲜明对比,针对教育行业云计算基础设施建设的研究还处于起步阶段，理论研究不少，但是具体实施起来，由于教育行业本身的研发能力有限，往往需要通过供应商采购商用云计算平台解决方案，然后在此基础上搭建自己的教育应用，因此云计算实施起来依赖于供应商，成本非常高。

由于云平台的多样性，国内外对教育行业私有云基础设施建设的研究主要有两方面。一是云平台的兼容性。私有云虚拟化后带来的平台的封闭架构不利于支撑合作共享的行业。因此，如何在不同云计算平台下提供通用的访问接口，使之逐步走向开放的平台架构。二是连接协议的标准化。虚拟化平台的访问协议有微软的RDP(Remote Desktop Protocol)协议、VMWare公司的PCoIP协议和本文所采用的红帽公司SPICE（Simple Protocol for Independent Computing Environment）等。协议多样性带来了兼容和管理的难题，因此急需统一访问标准。自2017起教育行业致力于研究如何部署以用将VDI(Virtual Desktop Infrastructure)模式和VOI(Virtual OS Infrastructure)模式融合的下一代云桌面技术，既能提供高性能的稳定性，还满足移动教学的需求。

教育行业的云计算基础设施研究主，要围绕基于虚拟资源池运行各类教学应用为主，即将原来部署于物理机上的服务迁移至虚拟服务器上继续提供服务，降低IT管理的难度，提高硬件资源使用率。真正利用开源的私有云技术,进行面向教育领域的云计算基础设施建设的实施案例并不多见，本文将讨论在此基础上为教师提供一个与日常教学任务紧密结合的云计算服务，即可以定制并且统一管理的教学云桌面服务。

## 1.3 论文的内容与意义

### 1.3.1论文的内容

高等院校面临的挑战是硬件资源分散在不同教学楼的各个教室，维护人员需要亲临现场解决各类软硬件问题，存在大量重复劳动且效率低下。如故障未能及时处理，需要临时调换教室，会影响授课效果，耽误教学任务。教室电脑的操作系统和应用软件环境与老师私人电脑存在很大的差异，无法满足不同课程体系下软件环境多样性的需要，导致老师时常需要自带电脑上课，教学桌面难以个性化定制也无法统一管理。

云桌面是市场上针对上述问题的一个普遍有效的解决方案。而云桌面的基础是虚拟化和云计算技术。该项目原来实施的云计算系统基于中心机房的服务器虚拟化技术，将服务器上的物理硬件虚拟成多个虚拟桌面向外提供服务。多媒体教室电脑通过远程桌面访问协议使用虚拟桌面进行教学。这样做的优势是IT管理人员的主要工作由维护分散在各间教室的电脑转向维护中心机房的私有云系统，降低了维护成本。弊端是需要增加服务器及网络的硬件投入满足多媒体教室中运行虚拟桌面的硬件要求，然而和企业不同，学校多媒体教室并不需要经常提供7 X 24小时的不间断服务，大部分情况下，服务器的计算能力被闲置和浪费，而且远程桌面访问模式依赖于整个网络的可靠性，一旦断网将导致整个系统无法向教室提供虚拟桌面服务，造成教学事故。

针对原有系统的上述不足,必须结合课堂教学的使用场景对系统进行定制和改造。本文利用KVM虚拟化技术与多媒体教室电脑结合，将其改造成私有云系统中的一个临时的计算节点统一管理起来，为教室提供云桌面服务。将计算能力由中心机房集中处理向有一定计算能力的网络终端转移，不但充分利用了现有硬件资源，同时分担了机房服务器的计算压力。最小化情况下，学校只需采购一台刀片服务器就能部署一个基本的私有云系统,显著的降低了实施云计算系统的采购成本。由于计算能力的边缘化需要重新考虑私有云中间层各个模块之间的协作和通信，本文对各其各个服务模块间的通信机制进行了重新设计和实现，以满足课堂教学的并发峰值压力的需求。针对网络故障的情况，本文设计实现了离线模式，通过用户管理和访问控制使得云桌面可以推送并缓存到指定的电脑上运行，提高了整个系统的高可用性，有效的解决了断网情况下服务被中断的问题，保证教学事务的正常进行。

本文首先详细介绍了项目的背景、云计算的基本概念以及实现本系统所使用的核心技术。通过几种虚拟化技术对比，介绍了KVM虚拟化技术对学校硬件资源进行虚拟化改造带来的优势，例如资源的充分利用、易维护、可弹性伸缩、高可用性、节约成本等。

其次，归纳了教育行业云计算项目实施的现状和挑战。对教学场景进行梳理，分析整个系统的非功能性需求分析和功能性需求，确定系统设计指标。从私有云系统构建个性化的教学环境来支撑教学业务的角度，进行系统架构、软件模块和网络拓扑设计。总结了多媒体教室、学生机房等教学场景下基于私有云构建新型教学桌面给教育业务带来的好处。

然后，对私有云教学系统的虚拟化层、中间层、应用层访问控制和备份等重要功能进行详细设计和实现。重点论述了与教学环境结合最紧密的私有云中间层的实现。本系统设计参考了经典的私有云计算开源框架Eucalyptus（Elastic Utility Computing Architecture for Linking Your Programs To Useful Systems）的架构，并对其中间层作了重新实现与改进，进一步提高可靠性、高可用性，使其能够满足商业环境的实际需求，促成云计算在教育行业落地。

最后，总结了私有云教学系统对教学的带来的影响，展望了云计算在教育行业未来的发展趋势。

### 1.3.2 论文的意义

首先，我国对高等教学、中小学教学、职业教学以及网络教学的信息化发展十分重视，甚至在某些方面已经跨入世界领先水平，如在线授课、互联网移动教学等。然而在学校传统教学方面，国内各地区的发展还不平衡，东部地区高校信息化建设十分成熟，网络基础设施相当完备，而一些西部地区还十分落后。如何使教学信息互联互通、打破信息孤岛是教育领域的重点发展战略之一。云计算作为实现这一战略的基础经历了十几年的发展，却很难真正在许多学校实践与落地。本文的所有需求分析和实现均来自于高校云计算建设项目的真实案例，创造性的利用KVM虚拟化技术与校园闲置计算资源相结合，实现云计算边缘化，可以为众多教育单位多媒体教室和学生机房的云计算系统实施方案提供切实可行的参考意义。

其次，教育领域一直是云计算服务提供商争夺的重要垂直细分市场之一。本文通过分析多媒体教室教学这个实际的场景，对利用新技术改进传统课堂教学手段进行研究,设计并实现了一个低成本、高可用、可伸缩、易维护的私有云系统，它是云计算技术与教育垂直领域密切结合的产物。本文以课堂授课的实际场景为基础，对原有的私有云开源框架进行了重新设计和实现，例如虚拟机管理方式，资源调度方式等，并创新的增加了离线模式，本地缓存等功能满足教学场景对系统稳定性的需要。可以对云计算技术应用在现代教育技术方面的研究提供宝贵的实践经验。

最后，本文所描述的私有云平台实现部分，主要面向但不仅限于教育行业，它也可作为其它行业云计算的信息化建设的一种解决方案，为其他行业的云计算落地提供重要的参考依据。将技术和某个具体行业的实际需求相结合，有利于在实践中不断发现新需求，继续打磨好技术，也有利于让技术进一步助推业务，让技术与业务相辅相成获得共同发展。

## 1.4论文结构

本文一共分为五章，具体结构如下：

第一章，绪论。介绍本文的写作背景、内容和意义。

第二章，教育私有云系统与KVM虚拟化技术。介绍项目背景及其相关技术，梳理了教育行业云计算平台建设的现状。比较了不同虚拟化技术，重点阐述了KVM虚拟化技术，剖析其工作原理，同时介绍了相关的其他技术。

第三章，教育私有云系统需求分析与架构设计。从功能性需求和非功能需求两方面进行需求分析，论述基于KVM技术的私有云系统的架构设计、基本功能设计和网络拓扑设计。

第四章，教育私有云系统详细设计与实现。根据第三章系统架构设计，对私有云系统的各个模块展开详细设计和实现。重点论述了教学资源虚拟化、中间层、应用层中作者亲自参与的软件模块的实现。

第五章，总结与展望。对本文的写作进行总结，论述了本系统对教育行业实施云计算项目产生的影响。展望了云计算在教育信息化领域里的发展趋势。

# 第二章 教育私有云系统与KVM虚拟化技术

## 2.1 教育私有云系统概述

### 2.1.1云计算的定义

各大厂商对云计算的定义略有不同，但普遍接受的一种解释是“云计算是分布式处理、并行处理、网格计算、效用计算、虚拟化、负载均衡等传统计算机技术和网络技术发展融合的产物”[3]。按照服务提供者与使用者的地域关系可分为公有云、私有云和混合云。

公有云运行在使用者外部的网络上，由云计算供应商提供计算和运维服务。通过审计和计费方法，用户从供应商按需购买IT基础设施和软件运行环境。由于业务在使用者外部网络运行，服务提供商要保证服务的安全性和高可用性等，而使用者不必自建机房，只需要租用计算服务来搭建自身业务。

私有云与公有云正好相反，服务运行于用户内部网络和服务器等基础设施上，完全由用户来管理和控制整个云计算服务的生命周期。与传统中心机房建设相比，采用云计算可以有效地较低管理复杂度和运维成本。同时由于数据保存在用户内部，信息安全性上比公有云更可控。缺点是对IT维护人员的素质要求比较高[4]。

混合云是“公共云”结合“私有云”一种方式。一部业务运行在私有云，另一部分由公有云承载。通常非核心业务由公有云负责，而核心业务运行在私有云上。对于教育领域来说核心业务就是教学，最好采用私有云的方式管理。

采用云计算技术首先要面临的问题是选择公有云还是私有云解决方案。表2-1中比较了公有云和私有云影响云计算采购决策的几项因素。从表中可以得出的结论是，公有云的服务质量有更多的多变性和不确定性。核心业务和私有数据部署在私有云上，可以满足业务灵活多变和快速响应，非核心业务由公有云托管，可以降低部署成本，最终以混合云的方式共同提供服务。

|  |  |  |
| --- | --- | --- |
| 影响采购决策的因素 | 公有云 | 私有云 |
| 数据安全 | 存储于外网，依赖于供应商的安全服务保障。 | 存储于内部网络。依赖于自身IT信息安全架构和管理制度。 |
| 前期投入 | 直接按需购买服务。不用关心硬件和基础设施的构建。 | 需要自己设计和实施。需要品谷运行环境，采购必要的硬件设施。 |
| 系统可靠性 | 依赖供应商数据中心的稳定性。 | 自己掌控。硬件层面依赖于服务器和网络的服务能力，软件层面依赖于系统设计的可靠性。 |
| SLA（Service Level Agreement） | 必须同意接受服务供应商的服务协议。一旦服务中断和数据泄露，自己无法掌控恢复时间和备用方案，另外，索赔流程复杂。 | 由系统架构决定，并有管理员针对自身系统做跟踪、日志和审计，进行风险管理和故障排除。 |
| 业务应用自由度 | 受到服务供应商的管控。但也降低了IT管理的难度。 | 按需定制。可以按不同需求定制适合自身特有的系统。需要IT人员对自身系统比较了解。 |
| 网络质量 | 受服务供应商和第三方网络运行商的制约，如电信、移动等。 | 受自身网络基础设施和局域网带宽影响。可以自己掌控。 |
| 承载业务 | 非核心业务，如移动应用等 | 以核心业务为主 |

表2-1 公有云和私有云采购决策因素

云计算四层模型与服务类型的对应关系如图2-1所示。通常将虚拟化层和基础设施层统称为基础设施即服务，因此，云计算提供的服务分为以下三类：

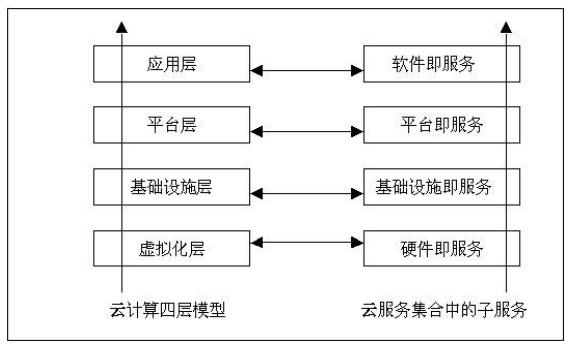


图2-1 云计算服务层次

软件即服务（Software as Service, 简称SaaS）是将业务以网络服务或者基于网络服务的客户端提供给前端用户使用。它的计算和处理以及软硬件设施的维护由云计算平台提供，用户免费或者按需购买服务即可。例如国外的Salesforce和国内的百度网盘等。

平台即服务（Platform as Service，简称PaaS）是将应用开发平台以服务的方式提供给用户。用户可以在云计算的环境下开发和发布自己的业务系统。云服务商需要提供开发环境，测试环境和发布手段等。代表产品有微软公司的Azure。

基础设施即服务（Infrastructure as Service，简称IaaS）是将运行操作系统所需要的中央处理器、网络服务、存储设备等的计算设施提供给用户。底层的基础设施对用户来说是透明的，由云平台商统一维护并按标准进行计费。用户按照自己的业务需求申请计算资源，并在此基础上部署自己的操作系统和软件环境[5]。代表厂商有亚马逊AWS和阿里云等。

### 2.1.2 私有云教学系统项目背景

目前高校授课经常在采用中控系统、配备电脑、接连投影仪等设备的多媒体教室，以PPT演示以及流媒体点播的多媒体方式进行。大部分学校多媒体教学手段已非常普及，成为支撑整个授课的基本需求。本项目的私有云教学系统是一个针对各类院校、中小学多媒体教室里教学现状和未来教室的发展方向，以KVM虚拟化技术为核心，基于中间层虚拟机管理模块以及远程访问协议为基础的私有云系统。该系统对服务器硬件进行了虚拟化，并为了老师和学生提供了可以按需定制的教学云桌面，为学校多媒体教室提供了可以离线使用的教学环境。管理员通过后台管理工具对系统进行随时随地的集中管理。

云计算特别适合高等教育这个应用领域。云计算本质是各种资源的更高层次的共享，是对硬件资源的合理分配和有效利用。采用云计算能够充分整合高校现有资源,包括用于教学的电脑、应用系统和课件等，并且能够以很低的成本为每个教师度身定做一个符合实际需要的教学环境，其他IT技术很难满足。师生是一个传道、授业、解惑的关系，云计算辅助教学是云计算技术在这种关系上一种新型应用模式，是学校利用云计算服务构建个性化教学环境，提升教学和学习效率的手段。要使教育行业能充分获得云计算的好处，或者说要使云计算在教育领域落地，就要先从云计算的基础IaaS建设做起。

下面进一步阐述教育领域实施云计算时，为什么要优先考虑私有云解决方案。第一小节公有云和私有云的对比中认为私有云更适合承载核心业务，学校的核心业务之一就是教学工作，以及围绕在教学业务周围的，为教学服务的业务系统。如果教学系统的运行完全依赖一个商业公有云服务企业的正常运转，安全风险会比较大。例如，前两年谷歌网站关闭事件导致在此之上的教学系统在瞬间都无法访问；最近Facebook公司的服务漏洞造成5000多用户信息的泄露。运营商的网络也无法保证在正常教学时间100%可用，一旦网络故障就会造成教学事故，影响正常的教学秩序。此外，提供服务的质量和自由度、可控性都受到了第三方服务提供商的捆绑。时间久了，服务商如果发生变化，如企业转型、公司倒闭等，会对业务系统维护带来不确定的风险，例如，国内360云盘关闭服务直接导致用户需要迁移大量的私人数据。因此，为学校单独构建的私有云系统，能够保障数据安全和服务质量的有效控制。学校对基础设施拥有绝对的控制权，能更从容的部署教育应用。

采用私有云的劣势是需要充分评估学校的硬件和网络环境是否满足云计算机系统的运行要求。需要自建或租用中心机房，采购服务器等硬件并对其进行维护，从基础设施维护的管理的成本和前期采购成本都要高于公有云方案。

本项目针对上述问题实现了基于开源软件的IaaS解决方案。该方案是一个私有云系统的纯软件实现，可以部署和运行于支持X86虚拟化架构的硬件平台。能充分利用现有机房的服务器进行部署可以有效降低私有云的采购成本。此外，项目为后台管理软件可以降低部署以后的管理成本。项目架构上吸取了开源软件Eucalyptus架构设计思想，Eucalyptus是一套开源的云平台的管理中间层软件，其本身并不能完全满足教学环境的需要。本文对原有的项目进行了改进。首先采用虚拟化技术，利用对自动注册和垃圾回收功能的实现，将教室电脑改造成临时加载到私有云系统的计算节点，降低服务器的采购成本。在系统内部的私有云中间层高性能通讯接口设计上，采用基于Apache的Thrift和RabbitMQ消息队列技术，加强系统运行的可靠性，使其有具有良好的弹性扩展能力。用户接口设计上与当前教学模式对接，与教师和学生使用习惯吻合；实施上不与学校教学和科研现状冲突，减少引入新系统后的学习成本为前提；以享受云计算的资源合理分配，弹性可伸缩、高可用性等诸多特性为目标。

## 2.2 KVM虚拟化技术

### 2.2.1 虚拟化技术

虚拟化技术主要分为平台虚拟化、资源虚拟化和应用虚拟化三大类[6]：

1. 平台虚拟化主要针对操作系统虚拟化，如桌面和服务器虚拟化。
2. 资源虚拟化主要是对网络、存储等系统资源的虚拟化。
3. 应用虚拟化主要是针对应用程序的虚拟化，例如仿真器，模拟器等。

本文主要讨论平台虚拟化技术，即通过虚拟化控制软件将运行操作系统所需要的物理硬件虚拟出来。平台虚拟化可根据虚拟化实现方式的不同分为全虚拟化、半虚拟化和硬件辅助虚拟化技术。下面依次介绍这三种类平台虚拟化技术：

全虚拟化(Full Virtualization)是指将真实的物理硬件虚拟化成直接可以运行操作系统的一整套完整的虚拟物理硬件。该技术架构如图2-2所示：

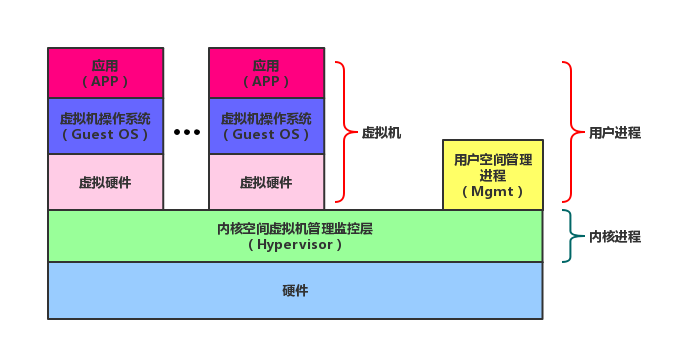


图2-2 全虚拟化架构示意图

图中控制虚拟硬件的软件被称作虚拟机管理层（Hypervisor）。在虚拟机上运行的操作系统被称为客户机（Guest OS）而管理和运行虚拟机的操作系统被称为宿主机（Host OS）。全虚拟化技术由于提供了完整的虚拟硬件，因此其特点是操作系统不需要做任何修改就能运行在虚拟机里。由于全虚拟化技术是通过Hyper-visor访问底层硬件，因此性能上不如真实物理机，但是要快于纯软件的硬件模拟。是非常适合应用于云桌面实现的底层技术[7]。

半虚拟化(Paravirtualization)类似全虚拟化技术，也使用Hypervisor读写底层硬件。架构如图2-3所示：

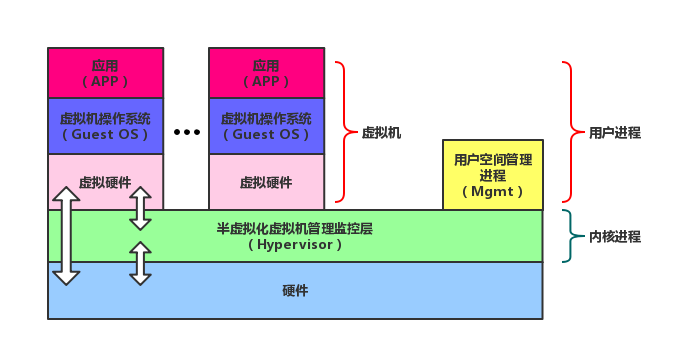


图2-3 半虚拟化架构示意图

半虚拟化技术需要修改客户操作系统，在其中集成虚拟化的进程，使得客户操作系统能与自身运行的Hypervisor进程协作。由于半虚拟化需要客户操作系统做一些修改配合Hypervisor，因此半虚拟化性能会略好于全虚拟化技术。代表项目有Xen[8]。

硬件辅助虚拟化(Hardware-assisted virtualization)是指操作系统在运行时直接依靠硬件虚拟机来完成虚拟化的过程。该技术的架构如图2-4所示：

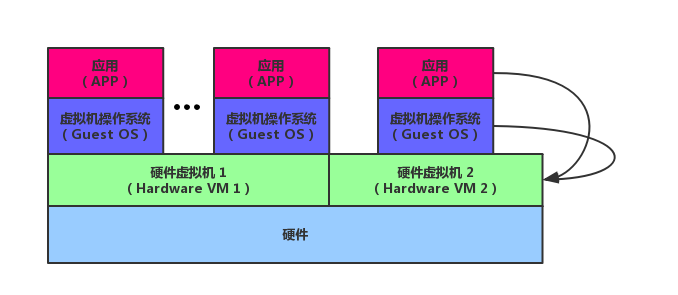


图2-4 硬件辅助虚拟化架构示意图

硬件辅助虚拟化不但能够提高全虚拟化的效率，而且半虚拟技术的Xen也通过该技术做到能支持Windows，MacOS等非开源的操作系统的运行。X86平台上的硬件辅助虚拟化项目主要是Intel-VT和AMD的AMD-V。

无论哪种平台虚拟化技术，其思想是利用硬件平台上富裕的计算资源执行更多的计算任务。目前市场上的大多数X86架构的服务器和个人电脑都已经将Intel-VT或AMD-V作为默认的标准配置集成在硬件平台上。在下一节中，本文将进一步介绍当前主流的开源全虚拟化技术KVM。

### 2.2.2 KVM虚拟化技术

KVM是X86平台上的开源的全虚拟化技术方案。红帽公司的Linux 放弃Xen而从2.6.20内核开始集成KVM。它的特点是系统内核通过加载kvm.ko内核模块，使内核自身变成了一个虚拟机的Hypervisor。KVM内核模块利用/dev/kvm块设备显示和调用虚拟化的硬件，利用QEMU（QEMU是Linux下虚拟化硬件的纯软件实现）用户进程的一部分功能作为客户操作系统接口[9]。KVM虚拟化架构如图2-5所示：

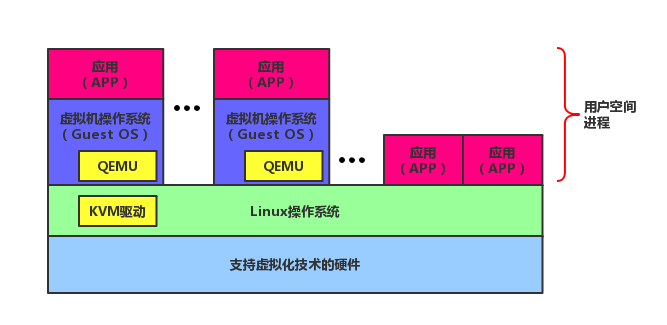


图2-5 KVM虚拟化技术架构示意图

Linux操作系统有内核模式和用户模式。KVM增加了一个新的执行模式: 客户模式。客户模式被用于执行虚拟机中所有非输入/输出客户代码。相对于其他虚拟化技术，KVM最大优势在于[10]：

1. 技术成熟且普及率高。截止2017年占领了70%以上开源虚拟化市场。
2. 代码完全开源，使用免费。虚拟机个数无许可证限制。
3. 支持快照、热备份和热迁移。
4. 和QEMU结合后有不错的外设访问能力。支持外设热插拔。

讨论KVM虚拟机化技术不得不提到另外两项与之紧密结合的技术：QEMU和Libvirt。

QEMU也是一个Linux的开源项目，它是一个通过纯软件方式实现的虚拟化平台，可以模拟X86、AMD64等许多架构的处理器。还可以虚拟显卡、声卡、网卡、键盘、鼠标、USB外设等设备。QEMU是运行在用户空间中的一个硬件的纯软件模拟，因此运行的性能不是很好，QEMU有一个和KVM结合的代码分支，用以提升虚拟硬件的读写性能，并借助QEMU的虚拟机管理模块来管理虚拟机设备，这就是qemu-kvm[11]。在进行基于KVM虚拟化平台的构建时，必须先安装qemu-kvm相关组件。

Libvirt是管理虚拟机及虚拟网卡、磁盘等其它虚拟设备的一组软件套件。它是Linux操作系统的进程，由libvirtd守护进程、virsh命令行工具和一套API库组成。Libvirt支持的hypervisors有KVM、Xen、LXC、OpenVZ、VirtualBoxLibvirt等[12]。它提供虚拟机的启动、停止、暂停、保存、恢复和迁移的管理服务，它还管理物理的和逻辑的网络接口，创建和存储多种类型的虚拟机镜像文件等。

## 2.3 其他相关技术介绍

### 2.3.1 Eucalyptus私有云框架

Eucalyptus是一种用于构建兼容亚马逊Web服务接口的私有云和混合云的软件[13]。如图2-6所示，Eucalyptus主要以下五个部分组成：

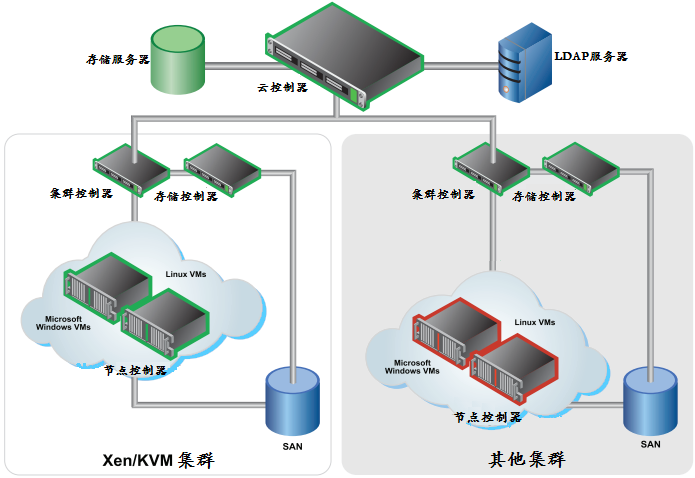


图2-6 Eucalyptus组件架构图

云控制器CLC（Cloud Controller）负责响应客户端请求和最高层次的资源调度。

存储服务器Walrus是一个兼容Amazon S3访问接口的存储服务，主要作用是保存虚拟机镜像和系统的用户数据。

存储控制器SC（Storage Controller）是兼容Amazon EBS接口的存储块设备服务，作用是存放业务数据。

集群控制器CC（Cluster Controller）负责监控和协调一个集群内的计算资源，并控制集群内的网络流量，它是支持分布式管理以及横向扩展。满足私有云伸缩性的需要。

节点控制器NC（Node Controller）运行在计算节点上，由CC管理，用以监控计算资源状态，并负责制作、启动和关闭虚拟机等操作。

以上组件可以根据需要在同一台物理服务器上安装，也可以分别安装在单独的服务器上。如果是采用KVM虚拟化，在每一个NC上需要预先安装Libvirt和qemu-kvm模块。Eucalyptus的虚拟网络通常需要分配两个IP地址范围，第一个范围是私有的，仅用于Eucalyptus系统服务本身。第二个范围是公开的，可以路由到最终用户和虚拟机。Eucalyptus直接运行于Linux内核之上，它有商用和开源两个版本。

本文的私有云系统IAAS层参考了开源版本的Eucalyptus的设计框架并且作了优化和部分去Eucalyptus化的重构工作，例如用户认证、虚拟机管理等。放弃了Eucalyptus原有组件间的通讯方式，取而代之以基于Thrift和RabbitMQ的消息传递来提高整个系统的高可用性，从而满足国内校园面向教学的私有云建设需求。

### 2.3.2 基于Thrift的RPC技术

Apache Thrift是一个开源的RPC（RemoteProcedureCall）框架，用来进行可扩展且跨语言服务的开发。它结合了功能强大的软件堆栈和[代码生成](http://baike.baidu.com/view/8477718.htm)引擎，用以构建在 C++、Java、Python、PHP、Ruby、Perl、Haskell、C#、Cocoa、JavaScript、Node.js、Smalltalk这些编程语言间无缝结合的、高效的服务[14]。Thrift允许你定义[数据类型](http://baike.baidu.com/view/675645.htm)和服务接口以作为输入文件，[编译器](http://baike.baidu.com/view/487018.htm)生成代码用来方便地生成RPC[客户端](http://baike.baidu.com/view/930.htm)和服务器通信的无缝跨编程语言。Thrift消息占用的字节数较少，适用于有大量数据传输的场景，有利于降低网络带宽和系统负载[15]，其架构如图2-7所示。

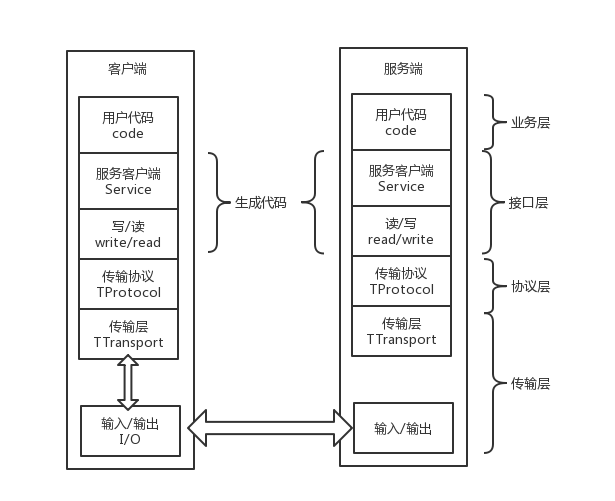


图2-7Thrift架构示意图

本系统根据Thrift语法定义了云计算服务API（Application Programming Interface，应用程序编程接口），并根据这些API来实现了系统的功能和服务。同时还提供了包含这些API服务类库，可以根据这个类库自己开发业务系统来调用云计算服务。编程语言的支持也非常丰富，可以选择与业务最符合，开发成本最低的一种。相比于市场上同类产品可以降低二次开发成本，提升开发效率。

本文中和Thrift配合起来使用的是RabbitMQ消息队列协议。RabbitMQ是高级消息队列协议（AdvancedMessageQueuingProtocol，AMQP）的一个开源的实现。AMQP是面向消息传递的中间件而设计的一个应用协议。消息中间件主要的作用是帮助得中间件各项组件间解耦。RabbitMQ的服务器端由Erlang语言实现，并且支持多语言的客户端，例如AJAX、Ruby、ActionScript、PHP、C、.NET、JAVA等。经常用于分布式系统的存储、转发消息。他具有很好的高可用性、可扩展性，因此也被广泛应用于具有分布式特性的云计算中间层中[16]。

### 2.3.3 基于SPICE的远程桌面连接技术

SPICE是红帽公司的远程桌面的管理渲染协议。和KVM一样SPICE也具有开源的许可协议，运行在Linux上。如图2-8所示SPICE可分为四种不同的组件：协议（Protocol）、终端（Client）、服务器（Server）和客户进程（Guest）。

1. 协议是其他三个组件之间通信的规范。
2. 终端负责发送数据并且转换从虚拟机传回的数据，以便进行交互。
3. 服务器是为了在SPICE协议下共享VM而供虚拟化技术（例如KVM）使用的类库。
4. 客户进程包含了所有运行在虚拟机里的必要软件[17]。例如图形设备驱动和VDI代理。

SPICE比RDP用户体验要好，更适合多媒体教学的场景。借助SPICE访问协议不但可以显示32位颜色渲染的虚拟桌面，支持自动全屏和自动分辨率调整，通过QXL驱动SPICE拥有出色的图形性能，还可以处理每秒多达30多帧的视频，播放分辨率高达1080P的高清视频，此外，它还能支持IP电话的双向音频、支持视频电话和视频会议的双向视频。这些优势使得用户无论是在PC机，还是在受客户机，甚至是移动终端，都可以通过SPICE客户端访问虚拟机获得接近于物理机一样的反应速度和显示效果。通过系统客户进程中SPICE的VDI代理软件，除了满足键盘、鼠标等外设的通讯需要外，还能支持USB外设的重定向（一种USB设备连接在物理机的USB端口上，却可以被虚拟客户机中的操作系统访问读写的技术）。

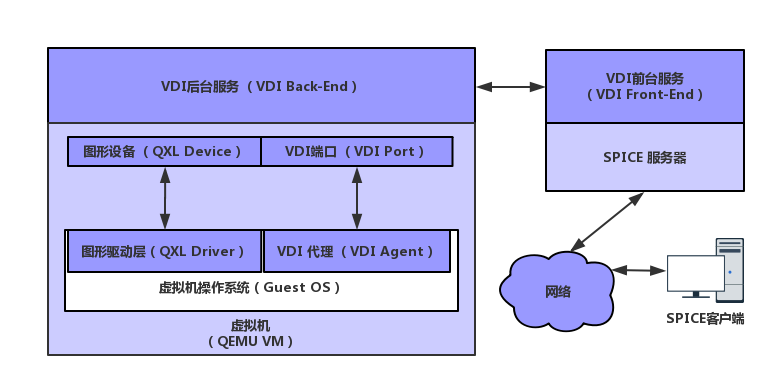


图2-8 SPCIE通信示意图

从SPICE的上述特性可以看出，它可以为教学环境下的课件展示、视频播放、USB外设使用等提供不错的用户体验。相比其他远程桌面技术更加适合于教学场景。本文中的私有云教学系统采用SPICE协议作为客户端与虚拟机连接的手段。

## 2.4 本章小结

本章从介绍云计算的基本概念、的分类和服务模型。从决策层面对比了采用公有云和私有云的不同点。并基于此分析了教育行业采用私有云的原因，进一步阐述了本文所在项目的背景。

随后分别从系统底层至上层，详细阐述了本文的系统实现中涉及的软件技术。本章介绍了目前主流的虚拟化技术，并重点说明了KVM虚拟化技术原理和特点。然后介绍了私有云开源框架和改进该框架的消息队列技术。最后，阐述了业务层对虚拟资源的远程桌面访问协议SPICE的原理和特点。

# 

# 第三章 教育私有云系统需求分析与架构设计

## 3.1 私有云教学系统的需求分析

### 3.1.1 云计算在教育领域的现状

云计算虽然在教育领域有过一些的成功案例，例如，将它部署在某些学生机房进行探索性试验，但是目前来说并没有得到大规模应用，对现有教育资源往云端迁移还需要通过实践来检验。云计算似乎很难在教育领域开花结果、形成效益，其原因可能是多方面的，本文认为主要有以下几点：

第一，前期投入巨大。云计算需要对现有教学中的基础设施进行改造，需要协调学校内不同组织和部门。往往不是单独某个学院或者年级所能决策的。需要获得全校的乃至教育部门的批准，前期调研论证的投入巨大。

第二，实施收效缓慢。新技术的引入往往带来对传统生产方式的改变。教育行业的生产方式就是教书育人，虽然教育部门积极探索新教学模式，例如电子书包、网络授课、未来教室、智慧学堂等，但目前普遍接受的教书育人的方式还是以课堂教学结合课后学习为主，原有的模式不可能在短期内发生巨大的颠覆，必须有个循序渐进的过程，与企业相比相对而言不可能很快的收到成效[18]。

第三，部署环境复杂。校园的信息化基础设施比较复杂，要面对各种各样不同品牌和性能的服务器与电脑，还要面对校园内不同局域网环境。更要面对校园内现存的多样化教学应用。存在大量的信息孤岛，互联互通有实际的难度。

第四，很难做到共赢。私有云建设，特别是IAAS层私有云建设，属于校园现有环境中的基础设施的改造，虽然可以降低维护人员管理成本，但从教学业务层面上讲，不容易有立竿见影的效果。教师们因为仅仅是基础设施的变化主观上就接受和推动实施的进行。事实上，由于基础设施的变化而带来的对传统工作模式的冲击，反而会得到老师的某些抵触与抱怨，没有政策或行政上的推动，不容易推行下去。

因此，本文认为教育行业私有云系统的规划思路需要从以下角度去考虑实施[19]。

1. 满足教书育人的教学需要。
2. 充分利用原有的基础设施资源。
3. 为用户带来方便的同时，降低学习新系统的培训成本。
4. 降低前期的采购成本和后期硬件的维护成本。
5. 功能上具有可扩展性，降低二次开发成本。
6. 降低学校IT管理人员的运维难度。

### 3.1.2非功能性需求分析

根据上面介绍的规划思路，在进行总体设计时要考虑下面的非功能性需求：

1. 充分利用现有资源。校园现有资源是指计算资源、网络资源和存储资源，包括中心机房的服务器、磁盘阵列、局域网、交换机、教室里的台式电脑和教职员工的工作电脑。系统设计时要加以利用，降低采购成本。私有云服务需要占用校园网络带宽向外提供服务，因此，必须满足不同时间不同环境下网络承受及服务响应能力。
2. 用户界面简单明了。硬件虚拟化服务设施的管理对于普通用户来说是透明的，从传统平台迁移至云平台时，尽量做到平滑过渡，用户在使用虚拟资源时，几乎感觉不到虚拟机和物理机区别，保证不变和良好的教学体验。客户端界面设计要接近师生普遍接受的教课和听课习惯，减少用户初次使用的学习成本。
3. 易于维护。在技术选型上，尽量使用比较主流而且稳定的开源技术，能够获得良好的可持续的技术支持，保证上层业务能够顺利进行。系统架构设计时，要能满足现有及将来软硬件环境下的部署与升级，不能设计成依赖与某些硬件的套件，避免与硬件绑定，减少后期维护成本和升级的压力[20]。软件架构设计时，各个模块之间采用MVC架构，功能模块之间通过Web服务或者RPC进行松耦合设计隔离开来，保证其良好的可扩展性。
4. 安全性需求。电脑硬件分散，网络环境复杂，加之大家习惯使用个人笔记本、U盘、移动硬盘进行学习交流，都为病毒的传播、网络攻击和信息的泄漏提供了便利，容易造成系统故障甚至崩溃。将操作系统和教学应用向云计算平台转移时要仍然考虑系统安全性。采用虚拟化技术要保证各个虚拟机在自己的进程空间和虚拟网络中运行，相互独立，互不干扰，有效的外部防止攻击。运用快照技术取代还原卡，提高系统的阻止病毒传播的能力。即便感染了病毒，也可以让操作系统迅速的恢复。通过虚拟机的备份和迁移技术，消除系统崩溃隐患，即使系统崩溃也不会造成影响，保障数据安全。

### 3.1.3 功能性业务需求分析

1）软件范围与运行环境

在进行功能性需求分析前，首先阐述该系统的软件范围和运行环境。本项目的软件范围是通过私有云提供的计算服务，为学校大量多媒体教室、学生机房构建一套完整的教学桌面管理系统，以取代原来教室和机房中，基于还原卡系统或是无盘工作站，安装部署操作系统和应用软件的传统IT管理模式。

运行环境包括硬件环境和软件环境。硬件环境需要学校中心机房至少提供一台基于X86结构并且能够运行Linux操作系刀片服务器，用以安装云计算各个服务模块。还需要提供若干台服务器作为计算节点的集群，计算节点必须支持Intel-VT和AMD-V硬件虚拟化。软件环境分为客户端和服务器端的运行环境。本系统基于B/S和C/S两种架构实现，因此客户端软件的运行环境需要支持安装JRE（JAVA Runtime Environment）和Web浏览器。服务器端的软件运行环境需要支持运行Linux系统和KVM虚拟化进程。

综上所述，该系统的开发基于Linux操作系统的开源框架和技术，优点是对软硬件几乎没有特殊的要求，市场上的主流硬件和学校现有的设施基本都能满足，降低了采购的难度，还具有非常好的可移植性，弊端是不能像商用软件那样购买良好的售后服务和技术支持，对学校IT人员的维护能力会带来挑战。因此，在安装程序设计和系统管理模块界面和功能设计时，必须充分考虑如何降低IT人员的使用难度，尽量通过Web API调用的方式向外提供GUI(Graphical User Interface)操作接口。还需要考虑管理员可以随时随地通过电脑和移动设备开展工作。

2）外部接口

本系统的用户分为三类：教职工、学生、系统管理员。教职工包括老师、学校领导及行政人员等。系统管理员也属于教职工，但在本系统中是特殊用户。

根据角色定义，用户分为超级用户、高级用户、普通用户。超级用户只有一个。学生只能是普通用户，教职工可以是高级用户或者是普通用户三种角色，系统管理员必须是超级用户。根据上述角色划分，从软件功能定义的角度，本系统为不同角色提供不同级别的服务。外部接口需求上需要对不同角色提供不同的访问接口，系统管理员的接口基于浏览器访问后台管理功能。高级用户和普通用户的通过客户端访问系统提供虚拟机：

1. 基于浏览器的用户界面。主要针对管理员，提供对该系统中用户管理，系统监控、云计算服务的管理、系统维护、虚拟机制作管理等的操作界面。
2. 基于客户端的用户界面。分为普通用户和高级用户两种情况，高级用户同时拥有普通用户所有功能。针对教职和学生中的普通用户，提供用户登录认证，信息查询和对云计算资源的申请与访问的界面。针对高级用户还增加了虚拟机制作、管理、备份等功能。

本系统不同角色的用户用例图如图3-1所示：

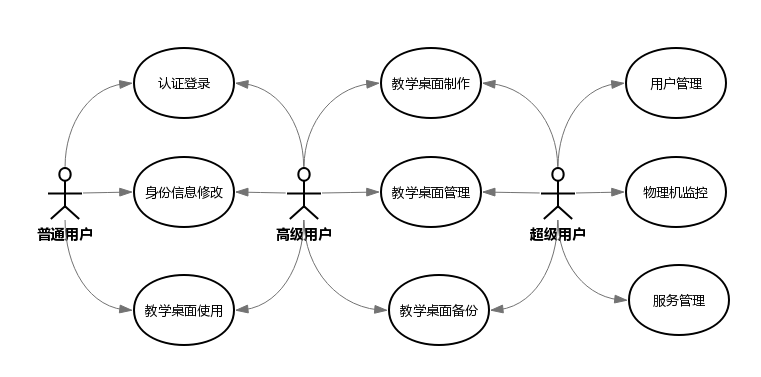


图3-1 依据不同角色的用户用例图

3）功能需求

功能需求分析包含业务层、中间层和基础设施层的功能需求。

1. 物理硬件虚拟化。这是基础设施层的功能需求，是构建云计算系统的前提。需要能够充分利用物理服务器闲置的计算能力，通过虚拟化技术转变为向外提供服务的虚拟机，并且提供虚拟机的运行平台。虚拟机允许对硬件进行访问，例如挂载硬盘、网卡等，还需要支持对外设的访问，例如USB2.0设备，光驱等。虚拟机性能的配置有定制化的需要，例如管理员可以根据服务器运行状况，或者教学应用的业务需求，配置CPU核数，内存大小，磁盘容量。运行办公软件的虚拟机和运行视频处理软件的虚拟机，在性能上应有所不同。虚拟机运行时能够支持多种访问协议。为了满足校园应用的运行需要，虚拟机需要支持主流X86操作系统的安装。
2. 物理机监控和服务管理。这也是基础设施层的功能需求。对于物理服务器运行状态进行汇报和监控，为管理员提供查询状态，重起和关闭服务的接口，及时汇报硬件资源的占用情况，例如CPU使用率、网络负载等。如有故障应能及时报警并通知管理员，通过日志管理记录服务器的运行状态以方便操作回溯和故障排除。服务的管理是指本对系统的软件模块的监控和管理，帮助管理员了解云计算各类服务的实时运行状态，可以及时挂起、停止或重启每一项服务。
3. 虚拟资源管理与调度。属于中间层的功能需求。虚拟资源的管理包括对虚拟资源的申请和释放，虚拟机的启动与关闭等，向业务层提供制作和存储各类虚拟机的手段。虚拟资源的调度需要系统可以根据物理硬件的资源使用情况，将虚拟机调度到计算节点上启动运行。
4. 用户权限管理。用户管理是业务层的需求。用户管理要结合虚拟机的调度策略，在公用教室中的用户登录要相互独立，即不同用户使用同一台电脑时只能使用属于该用户自己的虚拟机资源，而不能访问其他人的资源。同时能基于角色的定义来控制用户对系统各项服务的访问权限。管理员可以管理用户的信息和访问权限。用户管理是该教学系统的核心功能之一，只有系统管理员才能有权限对该功能进行操作。 用户管理存储校内用户的资料，也包含了用户的增加和删除、用户信息的修改、用户认证、用户角色定义和授权、用户批量导入和导出等操作。
5. 教学桌面的按需定制。也属于业务层需求。教学桌面是每个老师特有的工作桌面。目前教室电脑里是IT人员安装的标准的操作系统，系统上配备了还原卡，下次上课自动返回初始状态，每间教室的软件环境完全一样。如果老师在上课时，需要使用一些没有预先在教室电脑安装的软件时就非常不方便，此时老师必须携带自己的电脑到教室。教学桌面要能够按照各院系老师的不同需要定制后分发到教室运行。
6. 教学桌面的离线访问。同样属于业务层需求。原来教室里的操作系统直接运行于教室电脑上中并不依赖于网络情况。云计算的各项服务均运行在学校局域网上，依赖于网络的稳定性。在那些无法容忍服务中断的场景，例如课堂教学正在进行中，如果网络故障耽误个10-20分钟，即使维护人员及时修复，也已经造成了教学事故。离线访问要求即使断网，系统各项服务不能中断，而是能够以类似脱机的方式继续运行，等待网络恢复后自动切换回在线模式，这个功能需求使得云计算系统的可用性大大提高。

## 3.2 私有云教学平台总体设计

### 3.2.1系统总体架构设计

根据云计算将资源按需分配的思想，首先要将系统的硬件、网络和存储虚拟化后形成计算资源池管理起来，然后对资源的申请进行合理的分配和使用。教育私有云系统首先应充分利用校园里现有电脑、机房服务器、网络和存储服务器等条件建立虚拟资源池统一管理起来，即基础设施虚拟化。然后，通过后台管理工具和统一访问接口向各个院系的教职工和学生提供各种的教学服务，例如云桌面服务、云存储服务等。基于上述思想，将该系统划分为云计算服务模块与其配合使用的客户端模块，各个部分通过学校局域网链接起来组成一个可伸缩的校园私有云计算环境。通过用户授权和加密传输的方式，校内的云计算资源可以被校外师生，IT维护人员安全的访问。其系统逻辑架构如图3-2所示：

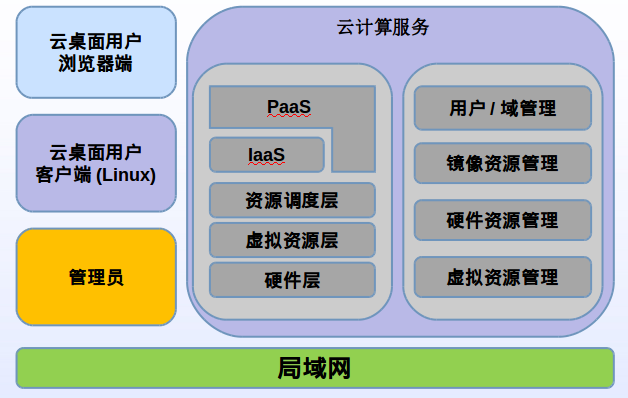


图3-2 私有云教学系统逻辑架构

该逻辑架构图参考美国国家标准局云计算平台软件框架模型，结合教育应用环境的特点优化后得出。云平台的提供的服务是在局域网内部。用户可以通过浏览器或是基于Web服务的客户端软件访问后台服务。云计算服务分为图3-1中左侧的基础服务和右侧的管理服务。

基础服务基于云计算的三层架构。硬件层是指学校计算中心机房的服务器和多媒体教室的电脑及其它相关物理硬件设施。虚拟资源层是通过虚拟化技术，将硬件资源变为虚拟硬件，例如虚拟中央处理器，虚拟网络，虚拟存储。IaaS层是通过Hypervisor对硬件层虚拟化后建立虚拟资源池，并提供统一的管理访问接口。PaaS层是对IaaS层服务的进一步封装。不同点在于，IAAS负责对虚拟化层的调用，而PaaS层调用IaaS层负责向教学应用提供软件环境和业务算法等层面的支持。例如一个教学应用的负载均衡服务。

管理服务层通过调用基础服务的IaaS和PaaS层的接口实现应用层对系统的各项管理功能。基本的管理有：

1. 用户管理。一个基于轻量级目录访问协议的用户账户和角色管理。
2. 镜像资源管理。镜像本质上是一个比较大的二进制文件。它涉及到两个方面的管理问题：文件存储和访问控制。在该系统中虚拟机镜像资源从访问权限划分为私人级、院系级和全校级三个层次。校内的任何师生都可以访问全校级镜像。被本院系的教职员工只能访问院系级镜像。私人级只能被这个镜像文件的所有者访问。
3. 硬件资源管理。把学校的物理服务器和PC机集中管理，对其运行状态和资源使用情况进行监控、日志记录和汇报。
4. 虚拟资源管理。把私有云计算节点上的虚拟资源集中管理起来，例如虚拟机的运行状态，占用资源情况。提供管理员管理虚拟资源的API访问接口。

### 3.2.2虚拟化层和资源调度层设计

虚拟化层设计主要是利用KVM虚拟化技术将物理机的计算资源，如CPU、内存、硬盘等虚拟化后，以接口调用的方式提供给上层使用。硬件虚拟化从其提供的服务分为两类：桌面虚拟化和服务器虚拟化。

* 桌面虚拟化将教室里的PC机、教师的办公桌面、学生桌等由本地迁移到校园私有云的云端服务，通过VDI虚拟桌面访问接口访问。
* 服务器虚拟化则将通常在校园中心机房提供的校园信息化服务，例如，选课服务、校园卡充值服务、教务系统、邮件服务等由服务器迁移到云端。

在软件设计上，虚拟化层作为独立运行的进程部署在服务器和PC机上。根据电脑硬件配置，判断当前PC是作为虚拟机运行平台，还是作为类似于瘦客户机的VDI访问终端。作为面向教育行业的私有云系统，需要和高校及中小学的现状相结合。目前高校基本上都有自己的中心机房，但并不是所有中小学都有自己的中心机房。学校的规模各不相同，桌面使用数量不同，对课堂教学的需求也不相同。有些学校希望先从某些院系的多媒体教室尝试云计算平台；有些则希望从学生机房开始实施；有些希望在未来教室方面进行实践。但也有一些共性的要求，就是保证上课时服务不能间断，教学资源数据要安全，网络负载要低，管理要智能化，操作不能不能太复杂。因此，软件上将其设计为可以独立运行的逻辑模块，各个模块间通过Web服务和RPC进行通信。这样既可以部署在同一台物理服务器，也可以部署在不同的服务器上，提高部署的灵活性。

在虚拟层上面是对虚拟资源进行管理的资源调度层。资源调度层设计参考了Eucalyptus框架（参见2.4.1）主要分为以下几个组成部分：

中央控制器CLC。主要完成对分布式集群的控制和管理，对计算能力的要求不高。中央控制器中集成了云存储服务功能，因此对磁盘IO（Input/Output，即输入输出）和网络IO的要求较高。每个私有云平台只有一个中央控制器。

集群控制器CC。每个集群都有一个集群服务器负责对集群中的计算节点进行管理，在云计算平台环境下可以部署一个CC或若干CC以实现云计算的伸缩性。例如虚拟桌面集群，虚拟服务器集群等。

存储控制器SC。它是计算资源虚拟化的重要组成。负责虚拟机镜像的注册，保存和下载。负责将镜像文件用户数据等存储工作转交中心机房的存储服务器完成。

节点控制器NC。部署在集群的计算节点上，每个计算节点对应于一个节点控制器，负责管理和监控该节点上的虚拟机运行，并向集群个控制器CC汇报状态。每个集群至少有一个计算节点和节点控制器。

LDAP服务器。LDAP是轻量级目录访问协议的缩写。LDAP服务器保存了整个云计算系统正常运行所有的配置参数，管理整个系统的核心数据。它以单独物理服务器的方式运行，必须得到最高级别的硬件层面的灾备。LDAP服务器提供管理员对云计算系统配置参数进行修改的在接口服务。

### 3.2.3 业务应用层设计

业务逻辑层是与用户关系最贴近的软件层，涉及到教育行业用户如何访问并使用该系统提供的服务和资源。用户分为普通用户和管理员两大类，从业务应用包括用户管理，云桌面使用，云桌面管理，系统状态监控等。用例图如图3-3所示：

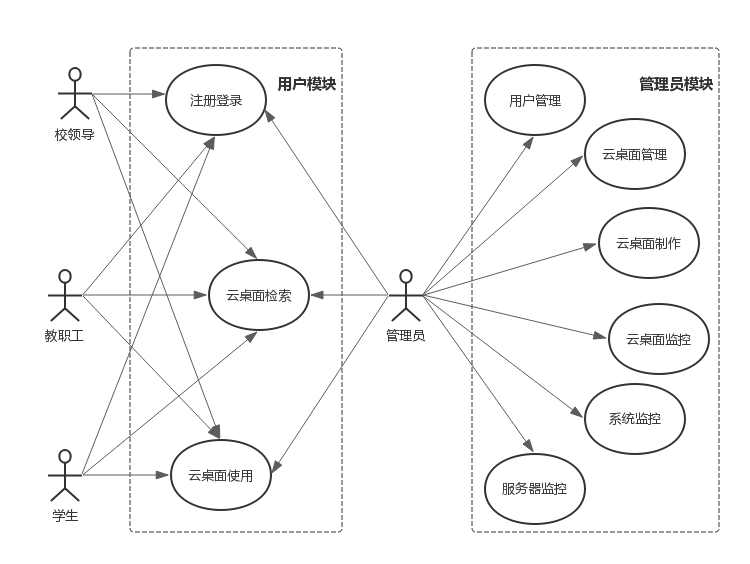


图3-3业务层用户用例图

1）用户管理

指用户的认证和授权管理。包括用户注册和登录、用户信息查询和编辑、用户授权、角色定义、用户帐户的增删改查。用户的权限分为普通用户、高级用户和超级用户。其中，用户登录针对老师利用启动教室中控系统直接进入操作系统的使用习惯，设计为单点登录和自动登录。自动登录指在系统启动以后，以下列方式之一自动进入私有云桌面访问客户端：

* 管理员预设的方式。例如某个院系公共帐号、中小学年级组公共帐号。
* 以前一次登录的帐号自动登录。
* 跳过登录流程直接打开客户端。

以上登录方式的优先顺序可进行参数配置。如果老师需要用私人帐号进行授课，那么打断上述过程转为用户登录界面输入用户名和密码进行登录。在公共教室上课，老师在该教室的登录信息不会被缓存，但确保了信息安全。以上登录方式和配置参数保存在后台LDAP服务器上。用户账户密码经过加盐哈希保存在数据库，用户登录认证过程通过SSL加密传输保证账户信息安全。

2)云桌面使用

云桌面指学校多媒体教室的教学桌面、学生机房的电脑桌面以及教职员工的工作桌面，配合预先安装在桌面里的应用，完成如同物理机操作系统所提供的软件环境，如文档处理、视频播放等。云桌面的使用包含查询、启动、连接、关闭、桌面制作，删除等操作。对云桌面的操作可以基于Windows版、MacOS 或者是Linux版的客户端、以及Web浏览器，满足教师的不同硬件条件和使用习惯。

云桌面的使用包括查询、启动和关闭操作。查询是指用户登录系统后，根据用户权限列出其所能使用的云桌面。例如，一个物理系老师打开系统后，能发现其能使用的云桌面是一组由全校级别云桌面、物理系云桌面和其私人云桌面的组合，而不能访问其他院系和其他老师的私人的云桌面。在用户界面上显示该云桌面操作系统版本，已安装的应用软件等基本信息。用户从中选择一个合适的云桌面启动，待启动完成，客户端利用预设的连接方式连接（例如，RDP或者是SPICE协议），将云桌面推送到用户端。使用完闭，可以像原来PC机关机一样关闭云桌面。整个过程对用户来说最重要的一点是对虚拟资源的使用过程是透明的，后面的具体细节都由资源调度层和虚拟化层完成，用户应该感觉到他在使用一台真实的物理机。

云桌面的连接。连接方式分别对应部署模式（参见3.3小节）的三种方式分为VDI（Virtual Desktop Infrastructure）方式、本地方式和混合方式。

VDI方式的桌面运行在服务器端，客户端通过VDI协议获取桌面画面呈现给终端用户，并通过网络进行鼠标、键盘事件通讯。特点是所有计算任务在服务器上进行，客户端不需要很高的硬件配置。缺点是对网络带宽和服务器计算能力有一定的要求，并且在网络故障的情况下会停止服务。

本地方式是根据教学环境的特点，对VDI方式的一种补充。云桌面从服务器端缓存到客户端运行，客户端软件访问的是本地运行环境的虚拟桌面，机房服务器云计算系统负责虚拟机的调度但不负责运行。其特点是在桌面使用过程中对网络带宽和稳定性的要求和原来一样，即使断网也可以继续运行。不足客户端PC必须支持虚拟化并具备一定空闲的存储空间，而且对PC的硬件配置有一定要求；其次，缓存需要消耗一定时间，需要在PC机闲置时运行，例如上课前，必须等待缓存完毕才能使用，不适合实时性要求特别高的业务。从本质上讲，它将云计算的计算任务边缘化，将虚拟机的调度和运行分离，调度由中心机房服务器负责，而运行交给局域网中具有一定计算能力的PC机。既降低网络负载和服务器的负担，又解决了在网络故障的情况下，VDI模式不能工作的弊端。本地模式是十分适合实时性要求不高的多媒体教室教学场景的一种模式。目前大多数院校教室中控系统的PC硬件架构都具备虚拟化能力。通过还原卡安装的操作系统和桌面应用都比较有限，因此也具有富足的存储空间。同时，在电化教室上课并不是24小时不间断的过程，有足够的时间可以利用非上课时间段将服务器上的云桌面缓存下来。因此实施过程中，应将本地模式作为私有云教学系统客户端的首选方式。

混合方式是上述两种模式的合并。在当前校园环境下，准备部署私有云平台的客户往往是从现有的硬件资源进行迁移，已经拥有了一定的PC机和服务器。可以根据当前硬件使用状况统计哪些PC机可以作为本地模式使用，哪些不符合本地模式的硬件要求。对于不符合要求的，进一步验证是否可以当作廋客户机而采用VDI的方式。在充分利用现有PC后再计算还需采购买多少PC机，这样就能够做到资源利用的最大化。对教育行业客户而言，混合模式是性价比较高方式，但同时相对而言前期准备要更加的充分，实施起来相对更复杂一些。

## 3.3网络拓扑架构设计

1.网络拓扑设计

网络拓扑如图3-4所示，这个网络拓扑的优点体现在：

计算部分和存储部分是松耦合的。对用户来说私有云系统部署以后并不意味着从此就被产品绑定。相反，私有云教学系统是一个开放的纯软件系统，能运行在任何支持X86硬件架构的服务器上。服务器、终端设备存储部分可以从不同的供应商采购。例如，学校已经购买了存储设备，那么不用重复投资而可以直接纳入该系统里面来。

计算部分是通过分布式集群方式管理的。可以根据学校业务的实际情况，逐步将计算资源加入到系群里来，充分发挥云计算可伸缩性的优势，使云计算建设可以分批、逐步、渐进式实施，而避免一次性投入巨大。

私有云IaaS层松耦合，支持多种虚拟化方式。可以通过配置在不同的集群中使用不同的虚拟化技术。例如，在一个集群里面采用开源的虚拟化技术KVM，而在另一个集群又可以采用商业虚拟化技术VMware等。用户根据自己的业务需要采购适合本学校的自己的虚拟化软件，进一步增加灵活性、降低成本。

桌面虚拟化和服务器虚拟化相互独立。可以由集群中一组工作集群实现桌面虚拟化，而另一组实现服务器虚拟化。方便用户在一套私有云系统框架内同时提供桌面虚拟化服务和服务器虚拟化服务。

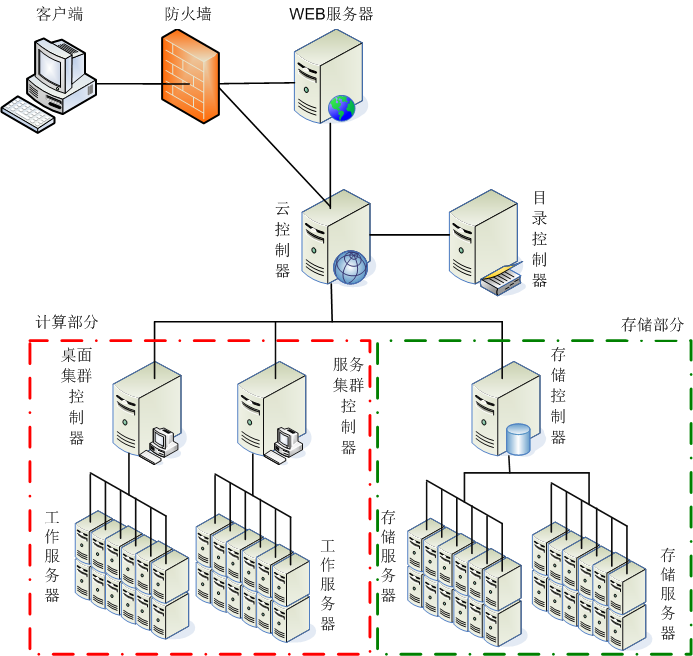


图3-4网络拓扑示意图

2.部署模式设计

集群的部署模式是指在云计算系统中计算节点的部署方式。针对校园网络环境下的硬件配置，在设计时采用离线模式和在线模式，或者两者皆有的混合模式，来满足不同的定制化需求，如图3-5所示，上半部分是在线模式，下半部分为离线模式。

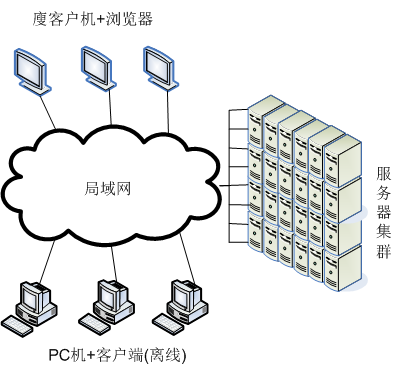


图3-5计算节点部署模式示意图

1. 在线模式。特点是所有的工作负载都交给后台的服务器来处理；前端的终端设备可以是很简单的廋客户机+浏览器。这个模式的优点很明显：用户使用非常简单，安全性好；同样，这个模式的缺点也很明显：前期投入大；对带宽要求高，甚至有可能要重新进行网络规划；浪费了现有的那些已经购置好但还未到报废期的PC机。
2. 离线模式。离线模式的最重要的优势在于，即使网络故障，离线模式下的终端仍旧可以正常使用，离线模式采用的是PC机+服务器的部署方式（这里PC机也可以用平板电脑来代替）。离线模式的本质是PC或平板电脑承担了真正的计算任务，服务器只是负责管理和资源调度。这会对PC机和平板电脑的硬件有一定的要求。客户端软件可以保证用户即使在离线状态下，也可以完成登录、启动并使用虚拟机等工作。当然，这时用户可调配的资源肯定没有在线的时候多。但离线模式把云平台的服务从要么100%，要么0%，改善为要么100%，要么>0%这样一种程度。离线模式的另一个优点是客户可以根据自己的实际情况，考虑到底是买服务器还是把现有的PC机充分利用起来。考虑到现在PC机基本上都很强大，节省下来的费用还是非常可观的。

## 3.4本章小结

本章阐述了由需求分析到系统架构设计的整个过程。

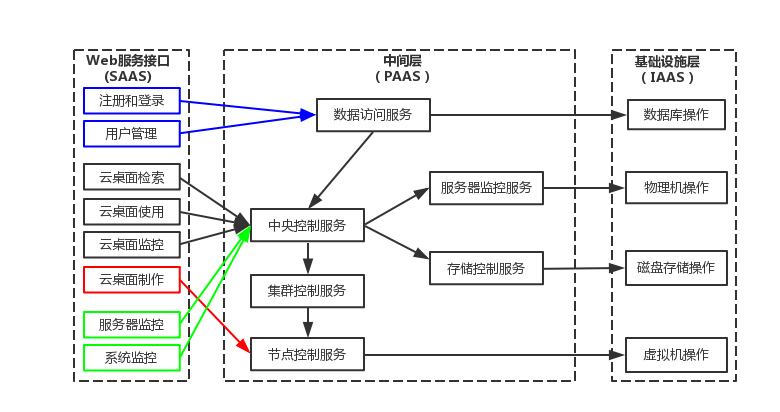
首先，从几个方面分析了云计算在教育行业难以落地的原因，并对此总结了实施云计算系统的总体规划思路。在进行基于云计算的校园信息化改造时，应该结合学校实际情况，充分利现有硬件资源，减少不必要的前期投入和后期维护成本。通过对教学领域的非功能性需求的梳理，发现必须设计一套可伸缩、高可用、易维护、并且成本不高的可定制的私有云系统。

接着，通过对教职工、学生、IT维护人员等系统干系人的分析，结合在教室教学这个最具代表性的业务场景、提出了私有云教学系统在校园落地首先要满足的功能性需求，即面向教育行业的核心业务，以教授知识为目的，为不同老师提供可按需定制的拥有统一教学环境的云桌面。

最后，从需求出发进行总体架构设计，阐述了整个系统中虚拟硬件层、中间管理层和业务逻辑层由底向上的设计要点。并且在网络拓扑设计中进一步阐述本系统针对教学提出的两个特有的部署模式——离线模式和在线模式。

# 第四章 教学私有云系统的实现

本文要讨论的系统实现是基于对原有私有云系统不足的改造，对虚拟化层，中间层和应用层相关模块进行了重新实现或者修改，以适合教学模式下对私有云系统的需求。原有系统的实现按照云计算服务的IaaS，PaaS，SaaS三层架构划分。各部分运行在服务器的Linux操作系统之上，通过KVM技术将服务器硬件资源虚拟化后，为上层应用提供计算单元和存储服务。各个软件模块如图4-1所示：



4-1 私有云教学平台软件模块示意图

系统总体结构的核心是中间层的设计，采用松耦合的模块化设计，其逻辑组成如图4-1所示包括中央控制服务、集群控制服务、存储控制服务、节点控制服务、数据访问服务等。各个逻辑模块相互分离，既可以同时部署在同一台物理机上，也可以相互组合分别部署在同一局域网内的不同物理器上。例如，对于一个小规模的学生机房，可以将中央控制服务、集群控制服务、存储控制服务、数据访问服务都安装在一台服务器上，而节点控制服务安装在机房学生电脑上。下面将从教学资源的虚拟化、中间层的实现和客户端的实现来具体阐述本文都原有系统改进。

## 4.1 教学资源虚拟化

### 4.1.1 硬件虚拟化的实现

想利用云计算技术解放学校硬件的生产力并降低管理的复杂度，首先是通过硬件虚拟化对学校的硬件资源设施进行改造。学校内的硬件资源一般分成部署于中心机房的服务器、分散在教室或者教师所使用的个人电脑两类。在此之上是基于硬件资源的业务系统，例如学生选课系统、图书馆管理系统等，还有例如教学课件、教学视频、科研数据等学校的教学资源。教学资源大量分散校园网的服务器及个人电脑中，因此要将教学资源从物理主机迁移到私有云的虚拟主机上，其核心是针对服务器和个人电脑这两类计算资源进行虚拟化改造，即服务器虚拟化和桌面虚拟化。

从KVM虚拟化技术实现的角度，以上两类虚拟化的本质上是相同的，都是通过虚拟化技术将底层物理硬件与操作系统、软件分离，使得一台物理机上可以划分出多台逻辑主机，从而建立一个虚拟计算资源池，并通过虚拟机的管理软件，以按需申请的原则来分发计算能力。

学校硬件虚拟化可以按以下四个步骤实现：

第一、搜集教学场景并进行分析与归类，整理出不同场景下所需要的操作系统和软件环境。

第二、找出学校里支持虚拟化的闲置硬件资源，用以充当私有云的计算节点。

第三、通过KVM技术在闲置硬件上建立虚拟资源池。

第四、通过KVM在虚拟资源池上建立能满足不同教学需求的云桌面。

前两点属于前期调研和需求阶段要解决的问题，后两点是软件设计上要考虑的因素。从软件实现的角度对硬件虚拟化层的具体实现主要由物理硬件层、加载KVM内核的Linux操作系统层、运行于操作系统用户空间的qemu-kvm、LibKvm等模块以及对这些模块封装以后向IAAS层提供的创建、启动、关闭虚拟机操作的一系列API接口组成。一个安装了硬件虚拟化相关模块，可以作为虚拟资源池向私有云中的客户端提供计算服务的计算机，我们称为该私有云系统中的一个计算节点。硬件虚拟化是通过计算节点来实现的。

教育私有云系统中的一个计算节点内部框架如图4-2所示。从图中可以看到，通过虚拟化技术，一个计算节点可以在一台物理主机上拥有多个虚拟机的实例，他们都相互独立运行在宿主操作系统Linux上。每个计算节点还包括管控虚拟机启动和关闭等操作的服务进程，它们运行在图4-2中的“其他用户进程中”。每个虚拟机的操作系统里运行着原来位于物理机上的教学应用和教学内容。

KVM内核与用户空间的QEMU进程相互配合，为教学桌面所使用的操作系统提供了虚拟硬件的运行环境，实现了教学桌面迁移到中心机房的计算节点上运行并统一管理起来。通过QEMU之上的远程桌面访问协议SPICE，实现客户端访问与外设的重定向等和交互操作。

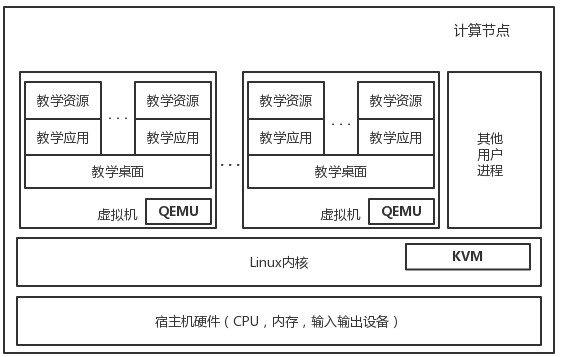


图4-2 计算节点软件框架示意图

根据KVM虚拟化的特点（参见2.2节），对于教学私有云中的一个采用KVM虚拟化技术的计算节点，其硬件配置的最低要求是：

CPU和主板架构支持Intel-VT或者AMD-V，并且要开启BOIS的虚拟化设置开关。Intel-VT和AMD-V是Intel和AMD公司支持X86虚拟化技术的相应CPU指令集。在Linux下可以通过以下命令判断硬件是否支持Intel-VT或者AMD-V：

#grep vmx /proc/cpuinfo （INTEL芯片）

#grep svm /proc/cpuinfo （AMD芯片）

其次，Linux内核从2.6.2开始支持KVM，所以要安装具有KVM内核的Linux操作系统，其内核版本要高于2.6.2。

本系统包装了计算机硬件和操作系统检测的API。在安装过程中调用该API对硬件环境的虚拟化支持情况进行自动检测。

可以看到，在学校里满足以上条件而能用来充当计算节点的机器并不少，它们可以是学校机房的服务器集群，也可以是安装在多媒体教室的个人电脑，甚至是老师的笔记本电脑。其中有一部分在很大程度上被浪费和闲置。机房服务器计算能力较强，但利用率不足，可根据其硬件状况，如CPU的核数，内存的大小和磁盘的容量，结合目前服务器上承载的服务负载来确定其向外提供计算服务的能力；多媒体教室的电脑作为计算节点，计算能力较弱，不适合做大规模运算，但可以作为单一的计算单元加入私有云的集群管理起来；对于老师的电脑，大部分都是Windows操作系统，而KVM基于Linux操作系统，还必须争得老师本人的同意，因此虚拟化改造的可行性不大。综上所述，本文在系统实现时对硬件虚拟化层作如下的规定：

1. 允许机房服务器上同时运行多个虚拟机实例，作为计算节点的主要生产者。
2. 具备虚拟化条件的多媒体教室的电脑作为计算节点，只运行一个云桌面实例，作为计算节点的辅助生产者。同时也作为私有云客户端，即消费者，使用本节点或机房节点的虚拟化资源。
3. 不具备虚拟化条件的电脑、瘦客户机和移动设备等只作为私有云的客户端，即消费者，来使用其他计算节点上的虚拟化资源。
4. 老师的个人电脑只作为消费者，通过客户端来使用私有云计算节点上的虚拟化资源。

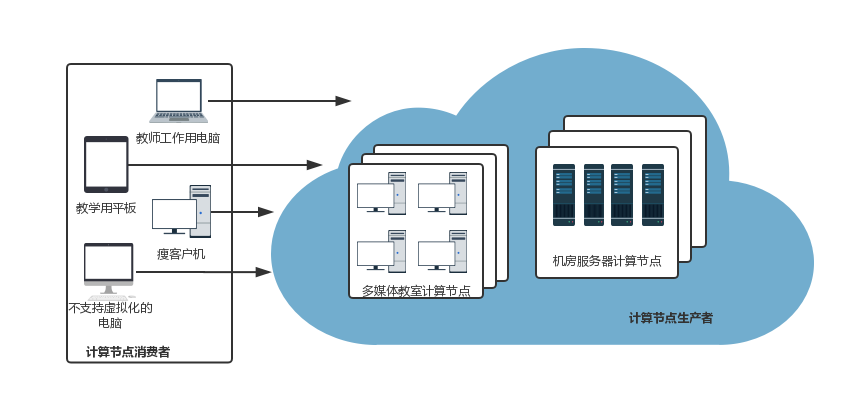


图4-3计算节点生产者和消费者示意图

根据上述规定，一个校园应用场景下私有云计算节点的生产者和消费者关系如图4-3所示。多媒体教室计算节点与机房服务器计算节点不同。机房服务器一般是24x7，即每周7天，每天24小时不间断运行，而多媒体教室的电脑是上课时才开机，下课就关机了，因此不可能将多媒体教室计算节点当作持久向其他消费者提供服务的计算节点。为了避免学校计算资源浪费，本系统将多媒体教室计算节点实现为仅可以向本教室提供虚拟机服务的节点，并且在同一时刻只允许运行一个虚拟机实例。一个典型的多媒体教室计算节点由通过以下模块来实现：

宿主机硬件。支持X86虚拟化的主机硬件。由于还需要在其上运行Linux操作系统，除了支持Intel-VT或AMD-V 以外，为了保证虚拟机运行流畅应满足以下最低配置：

1）硬件环境 (表4-1)

|  |  |
| --- | --- |
| 主板 | 支持VT-x虚拟化技术 |
| CPU | 支持VT-x虚拟化技术，主频2.0GHz以上 |
| 内存 | 推荐4GB，最低2GB |
| 硬盘 | 100GB 以上 |
| 显卡 | 支持Linux驱动的主流显卡 |
| 网卡 | 支持Linux驱动的主流网卡 |

表4-1硬件配置表

2）软件环境（表4-2）

|  |  |
| --- | --- |
| 操作系统 | Ubuntu 10.04 Desktop LTS amd64 |
| 浏览器 | MS IE 7.0以上, Firefox 14.0以上 |
| 软件 | JRE 1.6.0 |

表4-2软件配置表

3）网络环境（表4-3）

|  |  |
| --- | --- |
| 有线带宽 | 千兆主干网，百兆到教室PC机。 |
| 无线带宽 | 300M以上Wi-Fi AP |
| 其他要求 | 教室PC机能访问机房服务器  学校无线网络能访问教室PC机 |

表4-3网络配置表

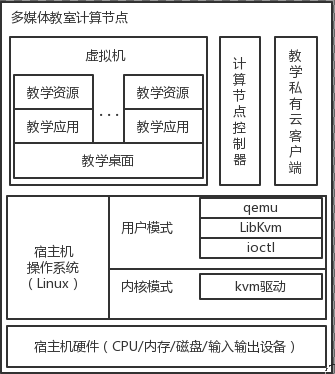
多媒体教室计算节点的逻辑实现架构如图4-4所示：

宿主机操作系统。集成KVM内核模块的Linux操作系统。推荐的操作系统为Ubuntu14.04 LTS X86/64桌面版。KVM驱动运行在Linux内核模式下。

QEMU模块。Linux用户模式下的虚拟化的纯软件实现，通过KVM对上层提供的API库LibKvm以及输入输出的访问控制模块ioctl，进行虚拟机向宿主机的IO读写操作。

计算节点控制器。私有云系统中负责启动、关闭、监控、清除计算节点上的虚拟机的IAAS层服务模块。底层是通过调用qemu-kvm的API来实现。

教学私有云客户端。负责用户的登录和对虚拟机操作。既可以访问机房服务器上的虚拟机，也可以访问本教室计算节点上的虚拟机。



4-4 多媒体教室计算节点架构示意图

多媒体教室计算节点为基于私有云系统的课堂教学带来以下好处：

1. 虚拟机运行在教室的电脑上，不占用服务器计算资源，使得资源利用率最大化。
2. 客户端与虚拟机的通讯直接在本地进行，降低了上课同时进行的时候校园网络带宽压力。从而使得云计算系统构建的网络基础设施成本得到控制。
3. 为网络环境不稳定时能够进行离线模式下的教学提供了技术上的保证，使得教学不会因为网络问题而中断，提升用户体验。
4. 使得向老师提供一个统一的教学云桌面成为可能（将在4.2.2进行进一步阐述）。

它的不足是虽然可以基本满足正常的教学工作，例如播放PPT，播放教学视频等，但由于受到电脑配置优劣的影响，计算性能可能有所差异，需要在部署时根据其配置对虚拟机参数设置作一定优化。可以用设置硬件检测的阀值来实现。例如，判断电脑启动后宿主机所占用的内存不高于40%的时候，才可以将它当作一个私有云的计算节点启用。此外，中心机房的主干网络至少是千兆级，而教室网络往往只有百兆甚至更低，当虚拟机第一次启动时，将虚拟机调度到教室电脑比调度到中心机房内部的计算节点要慢，迟滞时间主要取决于校园网络带宽和教室电脑磁盘读写能力。

所以，需要在私有云系统的实际部署环节，根据不同的教学场景或者教学应用结合部署成本，灵活的选择采用机房服务器节点模式还是多媒体教室节点模式。例如，如果校园网络环境以及机房服务其性能都非常好的情况下，可以多部署一些计算节点在中心机房；如果多媒体教室的电脑配置比较高，而中心机房服务器负载能力不足的情况，尽量采用多媒体教室计算节点的方式。在教学规模变化的时候，只需要通过相应的增加和减少计算节点的数量，以及计算节点上虚拟机的数量来实现私有云的可伸缩性。

### 4.1.2 教学云桌面的实现

在一个典型的教学场景中，以下因素都有可能影响授课正常进行。

1. 教师备课在自己的电脑上,而上课却是在多媒体教室的电脑上。但是老师电脑和教室电脑操作系统和软件环境的不同，如软件配置或版本问题。
2. 老师使用自己的笔记本到多媒体教室上课，必须自行解决连接其他硬件设备（如投影、话筒）等问题。
3. 教学PPT通过U盘拷贝，潜在的病毒威胁会干扰到整个教学网络环境。
4. 很多学校的中控都有还原卡系统，出现故障往往需要IT人员马上赶到教室进行处理，对IT管理带来不少压力。

在上述情况中，老师的个人电脑往往既充当着课堂教学时使用的工作电脑，又是平时备课、科研、作课题、访问或开会等情况下的个人电脑，老师的个人电脑在不同场景下需要安装不同的应用，需要存放不同资料，更可怕的是一但个人电脑损坏或丢失所造成的无法挽回的损失。基于KVM虚拟化技术的私有云就是针对以上情况，预先为不同场景提供不同的操作系统和软件环境，从而为老师提供一个统一的教学桌面。它是通过教学镜像和教学云桌面来实现的。

教学镜像。虚拟化后的操作系统以二进制文件的形式存储在私有云平台存储管理模块中，我们将这个二进制文件统一称为镜像（Image）。我们把从操作系统安装光盘或者USB通过KVM虚拟化技术安装以后制作成的一个基本镜像，称为原始镜像（OI），这个制作过程称为原始镜像制作。在原始镜像的基础上，安装了有基础软件以及云桌面使用所必要服务进程的新镜像称作该原始镜像的黄金镜像（Golden Image）。例如，一个电化教室典型的黄金镜像应该包含Windows操作系统、Office办公软件等。用于教学的黄金镜像被称为教学镜像。教学镜像是一个关键镜像，是学校桌面虚拟化的重要资源，需要保证其存储安全，防止丢失损坏，教育私有云平台的教育镜像是存储在分布式文件系统上，例如可以采用HDFS（Hadoop Distributed File System）技术的存储节点上，并由存储控制器进行管理。

教学云桌面。一个镜像的运行实例，其运行某个在私有云系统的计算节点上，并能和客户端交互实现某种特定业务的桌面，称为云桌面。云桌面是一个被私有云上的消费者使用的计算资源，是个人电脑桌面被虚拟化后的保存或运行在私有云系统中虚拟机。在本系统中对教学云桌面的描述是针对当前教学工作的需求，预设了所有能满足当前教学事务需求的应用软件及其正确配置的环境，可代替原来教学软件环境及操作系统，供私有云系统调度给客户端使用的桌面。

采用教学云桌面的好处是，IT人员可以直接在黄金镜像的基础上根据不同的上课场景或者上机实验的要求制作出多样化的、定制化的教学云桌面，却不必每次都从基本的操作系统安装开始，节省了大量的重复劳动，方便维护和提高管理的效率。同时，只要虚拟机的运行环境一致，老师就可以使用一个经过IT人员预先测试，并且统一而稳定的教学桌面，进行课堂教学，使得老师把工作重点放在教学和研究等核心业务上，而不用过多关心IT设施。教学云桌面是整个私有云教学系统向师生提供服务的基础，是与教师关系最紧密的云计算服务。

下面进一步阐述如何实现教学桌面的统一。所谓统一，即老师在备课时使用的桌面环境和其在教室上课时保持一致。不会因为一些与教学无关的软件配置问题给教学造成障碍。其设计思想是通过KVM的虚拟化技术，创建教学云桌面并发布到私有云上，老师在家中的电脑上通过客户端启动教学云桌面进行备课。在多媒体教室里老师也能使用相同教学桌面上课。老师相互之间虽然在同一间教室上课，却各自使用各自的桌面，互不干扰，也避免了将个人电脑带到教室充当工作电脑的麻烦。其工作流程是：

1. 系统认证通过后，初始情况下每个老师都有一个默认的基础教学镜像，包含基础的软件，例如办公软件，邮件客户端等。
2. 教学云桌面启动后，私有云系统会将其调度到某个空闲的计算节点，通过计算节点控制器启动虚拟机，并且系统会记录是哪个计算节点。
3. 启动成功后，通过虚拟桌面远程访问协议（VDI）访问该云桌面。
4. 在启动的云桌面中进行备课等相关工作，直至关闭云桌面并由保存老师的相关修改。
5. 上课时老师以同样的账户登录启动云桌面。私有云系统会查找最近一次启动该云桌面的节点，通过计算节点控制器启动虚拟机。

以上方案实际上是给老师分配了一个专属的工作桌面，并能通过客户端在不同的地方、不同的电脑上使用相同的桌面。为了满足上述流程，在实现时需要对计算节点上的KVM虚拟机做如下配置：

1. 将KVM虚拟机的硬盘格式设置成稀疏格式。KVM的硬盘格式总体上分为两种固定格式和稀疏格式。稀疏格式是虚拟机启动后，使用多少，分配多少，支持快照，降低存储空间。固定格式是在虚拟机创建时就设置好的，寻址和写入速度快，但占用空间大。个人云桌面在磁盘IO可以接受的情况下，为了降低服务器存储的容量压力，需要设置为稀疏格式。在KVM的配置文件中可以进行硬盘格式的修改：

"<driver name='qemu' type='qcow2' cache='none'/>"。

1. 打开KVM对USB外设重定向的访问设置，并且要支持热插拔。实现时要在编译qemu的时候，要增加 “CONFIG\_USB\_OHCI = y”，并且以包含USB的设备控制器（USB Controller）的方式启动虚拟机。
2. 虚拟机的网络模式设为桥接模式（Bridge），使得其在网络配置上接近局域网中的一台真实的计算机。如果IP地址有限的情况下，也可设置为网络地址转换模式(NAT)。
3. 将镜像设置为可篡改模式（COPY-ON-WRITE）。KVM镜像有可篡改和不可篡改两种方式。不可篡改方式（虚拟机文件以raw方式存储）是指当虚拟机关闭后，对虚拟机镜像所进行的所有改变都被丢弃，完全恢复期启动前的状态，功能类似教室电脑中的还原卡系统。反之，可篡改模式（虚拟机文件以格式qcow2方式存储）是运行时的所有修改都会被保留，下次启动时，还可以从以前状态继续，类似于真实PC机的使用方式。显然，这里应该将其设置为可篡改模式。
4. 打开镜像快照的支持。所谓镜像快照是指在云桌面运行的某一个时间点，将其所有状态保存下来的一种数据拷贝和备份技术。快照生成以后，可以在任何时候启动某个时间点的云桌面。可以有效防止云桌面发生错误时，可以返回到最近一次正确的状态。创建快照底层通过qemu命令 “qemu-img snapshot-create” 来实现。

此外，上述流程中云桌面的启动是通过计算节点控制器NC来实现的，云桌面的调度是通过中央控制器CLC和集群控制器CC来实现，云桌面的存储是通过存储控制器SC来实现，云桌面的访问是通过支持云桌面访问协议的客户端来实现。这些将在下面的章节作进一步的阐述。

## 4.2 教育私有云中间层的实现

在完成对硬件资源虚拟化之后，用户并不能直接使用运行在分布式集群的計算节点上的云桌面，在虚拟化平台层基础上需要实现一个中间管理层，对虚拟资源集中进行统一的管控和合理的调度，并且以API类库或Web 服务的方式向应用层提供服务，这一层类似于云计算的PAAS层。它是整个私有云教学系统的管理核心，基于校园局域网络环境传输，关系到整个系统所能提供服务的能力和SLA（Service Level Agreement），设计上一定要具有高可用性。

中间层会对虚拟资源池中的计算资源和存储资源使用状况进行分析，对资源进行合理的分配，对于学校来说最基本的资源就是日常教学事务中的教学云桌面。通过KVM虚拟化技术对教学桌面虚拟化并制作成镜像以后，会被存储在私有云系统的存储模块中，由存储控制器进行控制。中间层会负责统一处理用户对的教学云桌面的创建，调度、启动、关闭、释放资源等请求。中间层还要负责以上过程中云桌面的运行状态的监控，一旦发生故障要及时进行资源的备份和动态迁移。

中间层的实现参考了Eucalyptus框架，如中央控制器CLC、集群控制器CC、节点控制器NC、存储控制器SC以及LDAP服务器等。所有教室同时开始上课时，要在很短的一段时间完成云桌面的调度和启动；下课后电脑闲置时也必须及时释放虚拟计算资源。面对在教学过程中对资源请求的峰值压力，原有Eucalyptus无法满足这类需求，本文对其进行了改进，增加了自动注册和垃圾回收模块，摒弃原有的通信机制，取而代之以基于Thrift/RabbitMQ的RPC通信，这些都来自于高校实际部署和使用后的经验与教训。

### 4.2.1 注册和垃圾回收的实现

自动注册和垃圾收集都是针对多媒体教室教学这一特定场景提出的改进方案。下面将分别对其设计与实现进行阐述。

1. 自动注册的实现

中间层的各项服务要相互调用，必须知道对方服务类型（例如是一个NC服务，还是一个CC服务）、IP地址以及环境配置等信息，将这些信息保存到LDAP服务器管理起来的过程叫注册。在中间层的所有服务安装过程中，通过运行于Linux的一个后台精灵进程调用初始化模块自动向LDAP控制器发起注册，初始化模块是每个服务启动时第一个被加载的模块，监控安装和注册的状态，当所有检测和配置都正确以后，才会继续加载后续的模块。自动注册流程需要调用的接口函数如表4-4所示：

|  |  |
| --- | --- |
| 接口 | 描述 |
| reg\_is\_registered(servID,hostIp) | 判断是否注册。servID为node；hostIp为本机Ip |
| reg\_get\_ccname\_conf() | 获取集群控制器CC的信息 |
| reg\_init\_node\_info(nodeInfo) | 将节点信息注册到LDAP服务器上。nodeInfo中保存节点信息 |
| is\_online() | 判断本节点是否在线 |

表4-4 自动注册接口函数

节点信息和集群信息都会以XML格式缓存在本机Linux操作系统的/etc目录下：

<nodeInfo>

<hostIp>192.168.99.100</hostIp>

<clusterName>teaching-demo</clusterName>

…

<isLocal>TRUE</isLocal>

</nodeInfo>

<clusterInfo>

<clusterName>teaching-demo</clusterName >

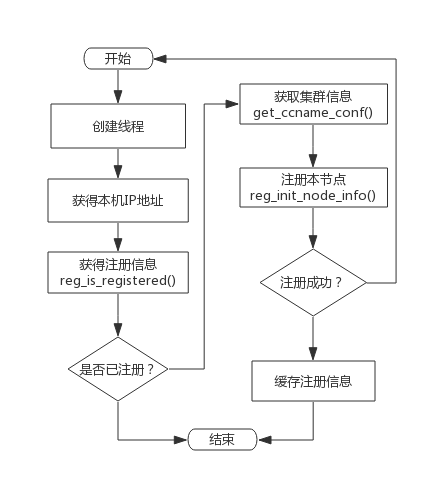
<hostIp>192.168.99.100</hostIp>

<HYPERVISOR>kvm</HYPERVISOR>

</clusterInfo>

注册信息缓存在本地是为了实现断网情况下，仍然可以以离线模式启动虚拟机提供云桌面服务。

自动注册的流程图如图4-5所示：

  
图4-5 自动注册流程图

私有云系统的自动注册，自动化了原来需要人工干预的服务注册过程。为私有云的伸缩性提供了方便。需要添加计算节点的时候，只要启动NC便会自动向LDAP添加一条记录，整个系统的中央控制节点和集群控制节点就能自动找到新增加的NC。反之亦然，如果私有云中某个NC故障或者被人为移除了，LDAP上该节点的记录也会被自动删除。

自动注册是将多媒体教室作为私有云系统计算节点的重要功能，因为教室电脑不可能像机房服务器24小时不间断运行，所以在上课启动电脑时，会自动向私有云系统报告并注册自己，关机时私有云系统又会自动将其从可调度的节点上删去。

此外，自动注册有利于减轻校内IT人数的工作压力，系统不需要再进行繁琐的配置，只需要安装相应的服务，等待自动注册完毕即可马上投入使用，提高了工作效率。

2．垃圾回收的实现

在实际使用中会存在客户端还未来得及关闭虚拟机，就由于人为因素或者网络和硬件原因导致客户端异常退出，此时虚拟机并不知道，还会继续运行，从而成为僵尸虚拟机。为了及时释放私有云上宝贵的计算资源，需要建立基于心跳机制的垃圾回收模块，自动清理残留的僵尸虚拟机。其设计思想是通过定期检查由客户端向发送计算节点NC发送的心跳请求，设置一个阀值，例如10分钟后没有再次接收到客户端心跳时信息时，认为客户端已退出，立即清除由该客户端启动的虚拟机。判定虚拟机是否需要清理的算法为：

1. 定时获取本地虚拟机运行状态和物理机的系统时间。
2. 查询该虚拟机所属客户端的最近一次心跳的时间。
3. 检查心跳信息没有超过阀值时，将1）中物理机的系统时间更新为最新的心跳时间，返回1）；否则，则认为客户端已经退出。
4. 如果1）中的虚拟机运行状态已经关闭，直接到5）。否则，认为找到一个僵尸虚拟机。垃圾回收模块马上调用NC相关函数停止虚拟机，释放节点服务器的计算资源。
5. 从LDAP配置服务器中删除该虚拟机的相关信息。

还可以对上述算法做优化，例如从第4）步不直接停止虚拟机，而是先置一个停止标志，再次返回第1）步，如果第二次还是执行到第4）步，那么系统原因造成的判断失误的可能性不大，可以基本确定僵尸虚拟机形成，应立即停止它。

对于多媒体教室教学场景下使用本系统提供的教学桌面时，我们不能对老师或者学生行为作约定，需要通过软件的方式去规避可能出现的异常情况。分散在各个教室的个人电脑情况也很难预料，因此垃圾回收机制也为部署了私有云系统的学校管理人员带来了方便。

### 4.2.2 中间层通信机制的实现

系统如何选择某一个计算节点去启动云桌面的策略和过程，被称为云桌面的调度。当客户端调用中央控制器CLC接口启动教学桌面时，系统从存储控制器中选取相应的镜像，CLC根据调度策略选择集群中某个计算节点,用KVM接口创建该镜像的一个虚拟机运行实例，并将的桌面图像和键盘、鼠标事件通过SPICE协议与客户端交互。用户体验像在使用本地电脑，但实际上运算和操作都是在计算节点虚拟环境中。云桌面的调度依赖于应用层和虚拟化层之间一个中间层，它向应用层提供对云桌面进行调度和管理的API接口。

CLC服务模块是教学桌面调度的核心，也是整个私有云中间层的核心。教学桌面调度的实现分成两个层面，第一，客户端与CLC之间的交互；第二，CLC与其他底层私有云服务之间的交互。

首先阐述CLC的客户端接口。CLC负责接受应用层上客户端的各种请求操作并返回处理结果和中间状态。具体来说它要负责向外提供基于WEB服务的系统管理，验证和接受其他服务模块节点注册与注销（参见4.2.1），认证并授权在计算节点上运行的虚拟机的请求，管理教学桌面镜像以及用户私有云存储空间。

中央控制器处理客户端的请求并返回结果。中央控制器和应用层的交互分别通过后台管理接口和用户访问接口实现：后台访问接口需要为管理员提供管理员身份认证；用户的增删改查；私有云系统相关的配置；虚拟机镜像管理，如可以暂停某个虚拟机的使用，或者对其进行访问权限进行修改；用户私有云存储空间的配置；查看的教学桌面镜像列表；选定一个镜像作为基础镜像，重新制作镜像，并且提交新的镜像到镜像管理子系统。用户访问接口为用户使用私有云系统的入口。中间层接口如图4-6所示。

当启动多教室的计算节点（参见4.1.1）时，计算节点会扫描CLC服务器，利用自动注册模块向CLC的节点控制子系统发送注册请求，在LDAP服务器上完成注册（参见4.2.1）。客户端从云控制服务器获取已注册且被授权使用的虚拟桌面镜像列表。然后用户通过用户客户端启动教学桌面的请求。集群控制子系统收到请求后将教学桌面调度到该计算节点上。以上处理由中间层各个模块配合，必须做到迅速而且稳定，不能发生故障而造成教学事故。Eucalyptus原有的处理方式无法满足教学场景下对系统稳定性的需求，所以必须对它们之间的交互性能做出改进。

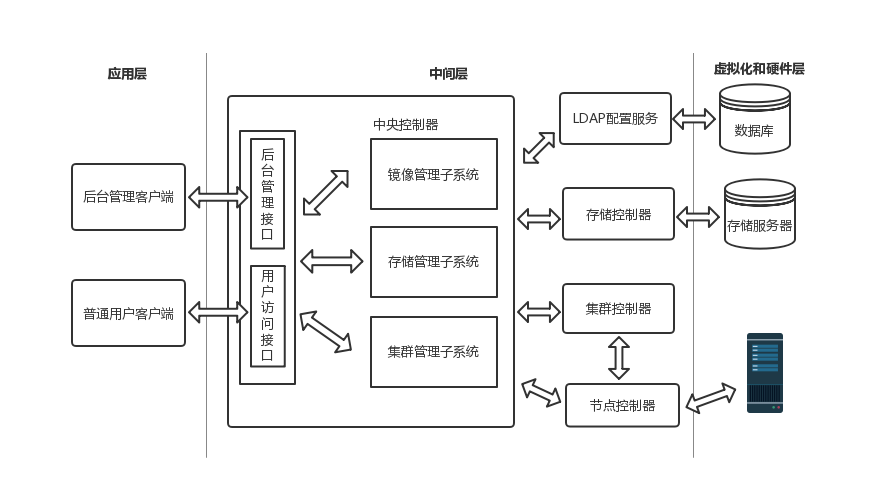


图4-6 中间层接口示意图

Eucalyptus框架的交互过程是CLC负责到LDAP配置服务器上检查用户请求是否合法，如果不合法直接报错退出，并返回错误结果给web服务器。如果合法则根据目前集群的状态，按照预先设置好的调度的算法，选择一个存在空闲计算节点的集群，向该集群的集群控制器CC发送启动虚拟机的请求。集群控制器CC收到请求以后，将根据CLC指定的计算节点的节点运行状态和剩余计算能力，将镜像调度到节点上，交给节点控制器NC。节点控制器NC根据CC的请求运行虚拟机，并且将虚拟机的状态实时汇报给集群控制器CC。再由CC向中央控制器CLC汇报。CC采用的调度策略如下：

1. Round-robin 即轮换策略。每次找一台最空闲的服务器来承担新任务。
2. Greedy 即贪婪策略，即选择最忙的服务器，直至其工作负载至上限为止。

难点在于整个过程中，老师在启动桌面的用户体验要和启动普通电脑基本相同。Eucalyptus在当前网络负载很重或者NC非常繁忙时，如果CLC发出启动虚拟机的指令，却不能及时收到包括虚拟机运行状态、网络配置信息、资源消耗情况在内的任何当前状态，将直接导致CLC指挥CC继续向一个已经没有资源的NC继续发送运行虚拟机的调度后长时间得不到响应，进而造成CLC长时间等待而教学桌面始终无法启动的故障。这些在教学过程中是绝对不允许发生的，因此本文针对上述问题对Eucalyptus的原有消息处理机制进行了如下重新实现。

首先引入RabbitMQ消息处理来取代旧的通信机制。同一台服务器内进程间通信采用基于Thrift的RPC方案。具体实现为在NC，CC和CLC服务模块之间增加两个独立运行模块NC消息生产者（NC-Sensor）和CC消息消费者（CC-Adaptor）来专门负责CC和NC之间消息传递。NC-Sensor和NC必须部署在一起，与此类似，CC-Adaptor和CC也部署在相同的服务器上。

当NC的NC-Sensor与CC的CC-Adaptor通信时，Eucalyptus原有的通信方式仍然可用，但通信方式只能采用两者中的一种，选择哪种方式可以通过LDAP配置服务中的相关配置信息来决定。CC与CC-Adaptor交换信息的通信方式为RPC(默认使用Thrift，也可扩展将来扩展为其他RPC方式)。NC-Sensor和CC-Adaptor通过AMQP消息中间件相互通信，与AMQP通信的消息为一个标准的W3C Message/RFC822格式数据包。NC和CC原有数据格式不变，数据的打包与解包工作均在NC-Sensor和CC-Adaptor中进行。客户端与CLC通讯，以及CLC与CC通信时仍然采用原有机制，CC和NC的消息中间层通信遵循AMQP协议，基于RabbitMQ实现。为了表述方便，此处仅列出直接与RabbitMQ通信的成员（图4-7）

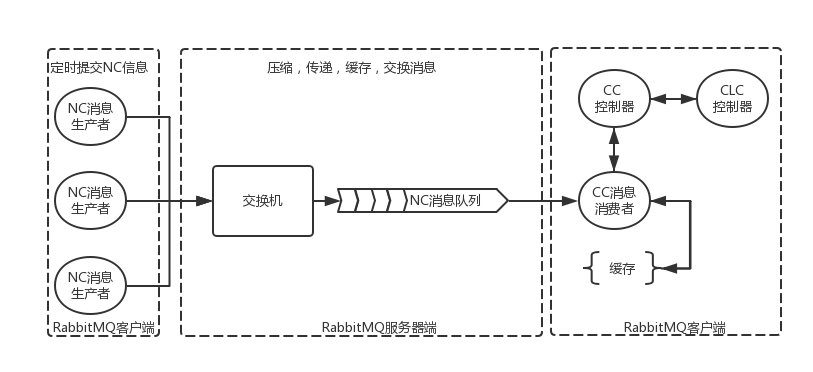


图4-7 NC/CC消息传递示意图

中间部分由RabbitMQ实现，包含交换机和消息队列，这两者都在RabbitMQ的服务器端，也被称作中间人（Broker）。两边是客户端，遵守发布/订阅模型，分为消息的生产者和消费者两种类型，它们访问同一个RabbitMQ服务器端。

实现时，消息队列被定义为NC-Raw，消息生产者定义为NC-Sensor，消息消费者定义为CC-Adaptor，同时在CC-Adaptor中设置缓存以加快CC对数据提取请求的快速响应。NC-Sensor 和CC-Adaptor的API接口设计说明如下：

1）NC-Sensor 的API 接口设计（表4-5）

|  |  |  |  |
| --- | --- | --- | --- |
| API | 输入 | 输出 | 处理伪码 |
| data-collection | 无 | 原始数据包 | 设定定时器定时触发。收集虚拟机状态、资源和网络信息数据，生成原始数据包。 |
| data- encapsulate | 原始数据包 | W3C数据包 | 把缓存中的原始数据据解析并重新打包成W3C( World Wide Web Consortium)数据包。 |
| Send-data | W3C数据包 | RabbitMQ Data | 把W3C数据包转换为RabbitMQ格式发送给指定RabbitMQ交换机。 |

表4-5 NC-Sensor接口设计

2）CC-Adaptor的API接口设计（表4-6）

|  |  |  |  |
| --- | --- | --- | --- |
| API | 输入 | 输出 | 功能 |
| set-amqp | 配置信息 | RabbitMQ返回信息 | CC-Adaptor创建交换机, 创建队列NC-Raw，设定路由规则。 |
| data-receiver | NC-Raw | W3C数据包 | 监听NC-Raw队列，如有数据到达，实时接收并存放在缓存中。 |
| de-encapsulate | W3C数据包 | 原始数据 | 解析W3C数据包，还原为NC的原始数据并存放于缓存中。 |
| response-cc | 请求指令 | 缓存数据 | 监听CC请求。 如有请求到达，从缓存中读取数据，返回给CC。 |

表4-6 CC-Adaptor接口设计

NC-Sensor和CC-Adaptor都是系统常驻进程，并遵照RabbitMQ的规定，本系统的交换机、消息队列及路由规则都由CC-Adaptor创建，交换机对NC-Raw绑定的路由规则为所有消息均存入消息队列中NC-Raw，NC-Raw中保存来自NC的原始信息。消息处理算法如下：

1. NC-Sensor定时采集NC的状态，并将状态打包成消息后提交给交换机。交换机按路由规则把消息存放到NC-Raw中。
2. CC-Adaptor实时监听服务器端的NC-Raw队列，插入或更新缓存中的NC状态。
3. 当CC收到来自CLC查询请求时，不访问NC而是直接从缓存中读取数据。
4. 当CC通过缓存获取数据出现错误时，CC直接向NC查询状态信息并更新缓存。

以上是NC和CC之间通过RabbitMQ通信的一个简单实现，随着业务增长，其他模块也可以采用类似的方法来实现，如CLC和CC之间，CLC和SC之间等等。

采用以上方案改造以后，通过定时采集方式解决了因NC在繁忙时无法及时响应来自CC请求的问题，使得整个系统在同一时间内支持调度的节点数大大提高。不同计算机间采用了中间件传递消息后，可靠性要高于Web服务方式，提高了系统整体高可用性（HA）。同时，各个CC-Adaptor和NC-Sensor作为常驻进程相互独立运行，松耦合的设计保证了使得局部的错误不会扩大到整个子系统中。从而提高了系统的可靠性。其次，由于NC-Sensor与 CC-Adaptor的多对一通信方式为发布/订阅模式，为上层服务提供了预先准备好的服务，通过缓存机制事先保存以供上层读取，降低了处理超时的可能性；系统处于消息处理的动态平衡状态，从而提高了整个系统的稳定性。此外，所有消息处理事件均写入日志文件中，方便监控。一旦出错，通过异常处理保障服务不会中断。这也进一步保证了整个中间层的稳定性。

## 4.3 教育私有云应用层的实现

客户端是基于B/S架构的，因此是以网页的方式进行交互来使用云桌面，这里重点阐述在客户端层面用云桌面的访问控制、云桌面的制作和备份等几个主要功能。

### 4.3.1 云桌面访问控制的实现

前面提到，本文把一个典型的教学桌面分成校级、院系级和私人级三个不同层级。这三个层级的访问约定如下：

1）任何一个合法用户都使用校级和私人级资源。

2）任何一个合法用户只能使用自己所属院系的资源。

以上访问控制对教学桌面可以被谁使用的进行限制，但并没有约定教学桌面如何被使用。换句话说只解决了一般的访问控制，还需要增加动态访问的规则，该规则基于{用户，镜像ID，物理机，属性}：

* 用户：教学桌面必须被特定用户使用，或者任何用户皆可使用该教学桌面。
* 镜像ID：必须使用某个特定的教学桌面，或者使用任何教学桌面都可以。
* 物理机：教学桌面必须调度到某个物理机，或者调度到任何物理机器都可以。
* 属性：是对由{用户，镜像ID，物理机}这三个条件约束下的一个云桌面的相关配置，主要内容为：
  + 虚拟资源。如虚拟CPU、虚拟内存、虚拟硬盘的大小。
  + 网络参数。例如是采用NAT（NetworkAddressTranslation）、Bridge还是Host Only的VLAN模式，使用静态IP还是动态IP、网关信息等。
  + 磁盘模式。例如是否以COW（COPY ON WRITE）的方式使用。
  + 外设访问。例如是否支持USB设备，是否支持并口/串口设备等。
  + 热迁移/热备份。是否需要打开热迁移和热备份功能。
  + 快照配置。是否开启支持功能，快照数量。

以上定义约定了本系统教学桌面从启动、运行到关机的生命周期涉及到的所有配置及访问控制。如果把取任意值设为0，取特定值设为1，那么可以定义出以下8种规则（表4-7）：

|  |  |  |  |
| --- | --- | --- | --- |
| 规则名称 | User | Image | Server |
| Rule-111 | 1 | 1 | 1 |
| Rule-110 | 1 | 1 | 0 |
| Rule-101 | 1 | 0 | 1 |
| Rule-100 | 1 | 0 | 0 |
| Rule-011 | 0 | 1 | 1 |
| Rule-010 | 0 | 1 | 0 |
| Rule-001 | 0 | 0 | 1 |
| Rule-000 | 0 | 0 | 0 |

表4-7虚拟机规则定义表

在对这8类规则的定义作详细说明（表4-8）之前，本系统规定这些规则的优先级设定为从Rule-111到Rule-000，优先级从高到低。

|  |  |
| --- | --- |
| 规则名称 | 规则说明 |
| Rule-111 | 指定的用户，在指定的服务器上，按照指定的参数运行指定的虚拟机镜像文件 |
| Rule-110 | 指定的用户，可以在任何服务器上，按照指定的参数运行指定的虚拟机镜像文件 |
| Rule-101 | 指定的用户，只能在指定的服务器上，按照指定的参数，运行任何虚拟机镜像文件 |
| Rule-100 | 指定的用户，必须按照指定参数运行虚拟机实例，无论虚拟机镜像文件和服务器 |
| Rule-011 | 指定的虚拟机镜像文件，只能在指定的服务器上，按照指定的参数运行，无论用户是谁 |
| Rule-010 | 指定的虚拟机镜像文件，必须按照指定的参数来运行，无论用户和服务器的具体内容 |
| Rule-001 | 在指定的服务器上，任何用户，运行任何镜像文件，都必须按照指定的参数来 |
| Rule-000 | 没有限制 |

表4-8虚拟机规则详细说明表

如果两个规则同时应用到同一个虚拟机实例，那么以优先级高的规则为准。这些规则一旦建立，在后台服务启动过程中，会自动的根据当前的用户数量，虚拟机镜像文件数量，以及工作服务器的数量，计算出所有的可能的虚拟机实例。例如，

如果只设置了一条Rule-111规则，那么计算结果是1个虚拟机实例；

如果有10个用户，8个校级虚拟机镜像文件，5台工作服务器，那么，在只配置了Rule-000的情况下，总共能够计算出10\*8\*5=400个虚拟机实例。每个用户在登录后，可以看到1\*8\*5=40个可用的虚拟机。其它规则照此类推。

### 4.3.2 教学云桌面制作和备份的实现

云桌面制作通过三种方式实现，制作完成后的镜像文件存储在私有云系统的存储服务器上并向LDAP服务器登记镜像信息以便中间层查询和调度：

直接从光盘制作。虚拟机加载光盘并运行安装程序，在制作过程中可以更换光盘文件。这在大型操作系统安装的过程非常有用。

从已有的教学镜像制作。这是教学系统中使用最频繁的方式。管理员可以先从光盘制作一个黄金镜像。然后黄金镜像为基础，根据需求安装应用软件，给学校定制符合个性化需求的教学桌面。

从P2V工具制作。有些操作系统已经找不到安装程序或者安装过程过于复杂，可以先用P2V（Physical to virtual）工具把整个电脑硬盘克隆下来做成镜像文件提交到服务器上。这种方式对将服务器迁移至云平台非常有效。

云桌面至制作完成后还需增加备份的功能满足教学资源的存储安全的需要。教学桌面备份实现是针对某台计算节点NC来说的。管理员可以备份NC上的某个特定的虚拟机或者所有虚拟机。当服务器出故障时，可以把备份重新恢复到新的服务器上。在实现时通过中央控制器CLC的调度在两台计算节点NC间进行。NC上处理备份事物的数据结构定义如下：

struct nc\_backup\_transaction{

string state; //需向CLC汇报的状态

string userName; //从CLC传递的参数中得到

string imageID; // 从CLC传递的参数中得到

Thread threadID;//由NC创建）

I64 progress; //需向CLC的transaction中写

}

在Thrift中备份处理被定义成一个原子事务，要么成功，要么回滚。以上信息被Thrift包装成下列数据结构传递给CLC：

struct thd\_backup\_transaction{

1: string transaction\_id // 事务ID

2: string state // 备份状态

// 'INIT' : 初始化状态，等待开始备份

// 'DUPLICATING': 正在备份

// 'BACKUP\_FINISH': 备份完成

// 'BACKUP\_FAILED' : 备份失败

3: string user\_name // 使用虚拟机的用户名

4: string image\_id // 要备份的镜像ID

5: string instance\_id // 要备份的虚拟机实例ID

6: string machine\_name // 虚拟机名称

7: string node\_ip // 所在计算节点 IP

8: i64 progress // 备份进程 -1或者0-100，-1 未开始

}

下面继续阐述基于Thrift的RPC通信机制下CLC和NC计算节点之间备份的实现方法。在CLC侧的实现流程是：

1. 当ClcThriftServer启动时，创建事务表。
2. 创建循环线程，删除事务表中已经完成备份或恢复操作的事务。
3. 当用户或系统发起备份请求时，判断CLC的事务表是否存在该虚拟机的信息，如果没有创建一个事务，并将状态置为“等待”。
4. CLC读取一条事务信息，判断事务是否处于“等待”状态，如果是，根据这条信息中NC的IP信息调用备份的处理函数开始备份。

在NC一端与之对应的处理为CLC通过RPC通信启动NC端的备份函数，然后NC端的备份函数创建并调用备份线程，线程实现算法如下：

1. 判断该虚拟机的缓存文件是否备份，如果没有则开始备份。
2. 备份该虚拟机的目录中的增量文件。
3. 备份过程中通过Thrift调用CLC状态函数更新备份进度。
4. 备份完成后通过Thrift调用CLC的状态函数将事务表中该虚拟机的备份记录改为“已完成”。

在实现中很有可能用户会临时终止备份，因此还需要实现备份中途终止的情况，具体的流程为查询CLC上的备份事务表，是否有该虚拟机的备份事务。如果事务的状态不是“备份中”，删除该事务，结束退出。接着通过调用NC上Thrift的备份终止函数来停止该虚拟机的备份，并删除已备份的残留文件。如果NC上停止备份成功，通过Thrift调用CLC的状态更新函数在事务表中删除该备份操作的记录。

虚拟机备份的底层实现机制是在NC上调用KVM接口实现的。KVM虚拟化技术本身支持热迁移和热备份。热迁移又叫动态迁移，即在虚拟机运行的时候，将整个运行状态完整记录下来，通过网络快速的保存到不同硬件平台上，在整个过程中用户不会有任何察觉。在热备份中，运行虚拟机实例的机器，定时的把虚拟机及其运行空间保存到某个存储设备，如果主机崩溃，备份机器立刻从存储中读取先前保存的虚拟机装载到新的硬件上。在热迁移中，KVM可以在很短的时间把虚拟机的内存和存储内容完整的搬到另一台物理机重现，用户基本上感觉不到服务的中断。利用KVM的这一特性加之中间层对其有效的管理，对整个系统高可用性的实现至关重要。

### 4.3.3 私有云客户端的实现

私有云客户端基于Linux的开源远程桌面访问协议SPICE实现。本系统中的客户端分为客户端软件和Web客户端。它们针对不同教学环境下的不同用户。客户端软件主要针对多媒体教室，学校机房中需要公共使用的电脑,运行在宿主机Linux操作系统上，通过API调用访问私有云中间层接口访问虚拟机，为老师和同学提供一套可以访问教学云桌面的入口。Web客户端基于B/S架构，通过Web服务器调用私有云中间层接口系统资源，通过用户权限的设置，为管理员提供管理整个系统的入口和老师在课后或家中访问教学桌面的入口。

通过私有云PaaS层提供的接口的实现客户端对私有云操作系统相关功能的调用，一个用户登录客户端以后看到的典型用户界面如图4-8所示：



图4-8 客户端云桌面列表界面

客户端对后台访问的主要接口设计如表4-9所示：

|  |  |
| --- | --- |
| 接口 | 功能 |
| cli\_domain\_user\_logon | 用户登录认证,返回用户Id。 |
| cli\_is\_registered | 判断本机是否在私有云注册 |
| cli\_init\_node\_info | 注册本节点到系统，自动注册时调用 |
| cli\_get\_client\_data | 查询用户信息 |
| cli\_get\_category\_list | 查询院系，所属部门信息 |
| cli\_get\_image\_list | 查询镜像列表 |
| cli\_get\_image\_info | 查询某个镜像列表和描述信息 |
| cli\_get\_node\_info | 获取计算节点信息 |
| cli\_start\_vm | 启动云桌面 |
| cli\_stop\_vm | 停止云桌面 |
| cli\_start\_backup | 启动备份 |

表4-9 客户端访问后台主要接口

客户端和虚拟桌面的连接以及对键鼠、外设的宿主机向虚拟机重定向是通过SPICE客户端实现，SPICE客户端被集成进了私有云客户端软件，当客户端软件调用查询虚拟机运行状态的接口发现虚拟机已经正常启动后，就会调用SPICE客户端与计算节点上的SPICE服务器端通信，将虚拟机桌面显示在宿主机的显示器上。同时，SPICE客户端还负责截取宿主机操作系统中的的鼠标、键盘及外设的输入输出事件，通过SPICE协议与虚拟机中的SPICE的VDI Agent服务进行通讯。

可以看到，老师用同一个私有云用户账号登录后，可以得到一个属于自己的云桌面运行环境，和他人互不干扰。配合4.3.1节的访问权限的讨论，不论在家中还是学校，不论是上课还是下课，老师都能在自己的教学桌面里进行教学工作。管理员不必跑到教室去维护软件环境，只需要在私有云上发布云桌面，分配给老师即可。

## 4.4成本效益及系统性能分析

### 4.4.1 成本效益分析

私有云将物理硬件虚拟化后，承载的服务能力来自于底层硬件的性能。在本系统中，就是集群里每个计算节点服务器上运行虚拟机的数量。每台服务器承载虚拟机数量如下：

(CPU核数×CPU数量－宿主机占用核数)×单核承载云桌面数

那么，

计算节点服务器采购数量 = 云桌面数量÷每台服务器承载虚拟机数量

采用原有系统中服务器虚拟化的方式，以一所学校拥有300间教室为例，假定所有教室都是用云桌面上课，采购具有2个C8核CPU的机架式刀片服务器作为计算节点，宿主机占用2个核运行操作系统核软件，每个核提供2个云桌面运行使用，那么，

节点服务器采购数量 = 300÷((8×2－2)×2) ≈ 10.714 = 11（台）

本文将具有虚拟化能力的教室电脑作为私有云计算节点以后，服务器的压力大大降低，学校采购服务器的成本也随之降低。如果有300教室中有150个可以作为计算节点，那么，

节点服务器采购数量 = (300－150)÷((8×2－2)×2) ≈ 5.357 = 6（台）

在上述例子中，在部署私有云系统时，可以直接为学校节省40%左右的采购成本。如果考虑到后期机房建设、电费、维护费用、人力成本等，节省下来的成本相当可观。

如果将剩下的150台无法作为计算节点的电脑，更新为具有虚拟化能力的电脑，那么只需要采购普通电脑来代替以服务器为主的计算节点。普通电脑的成本通常来说比服务器要低很多，而且更换起来也比服务器简单。采用基于多媒体教室计算节点的私有云系统后，可以根据学校的规模，资金投入状况灵活定制私有云的建设。

此外，从项目实施策略上，按照上述成本效益分析，校园环境下的教学环节中，计算密集型应用比较有限，大部分是面向普通教学的操作系统桌面的使用需求。因此，如果是计算密集型应用，例如3DMAX，MATLab等可以考虑部署服务器上，而剩下的大部分面向课堂教学的应用，都应该考虑尽量由多媒体教室或者机房的电脑来承担。根据小节的分析，本系统能够充分利用校园的硬件，特别是现存的普通电脑，前期投入成本大大降低，使得资金有限的学校更加容易部署私有云系统。

### 4.4.2 可靠性和可用性分析

对一个基于云架构的系统，除了满足用户的基本功能需求外，系统的其他性能指标，例如可靠性和可用性是保证系统足够健壮来应对正常和非正常环境下的挑战。下面我们从这两个方面来对系统进行性能方面进行定量或定性分析。

1．可靠性分析

系统的可靠性（Reliability）是指系统在正常和非正常情况下完成所设定功能的能力。从系统框架来看，该系统给的组成主要由一个中央控制器和许多由其控制调度的计算节点组成。

1）计算节点的可靠性。计算节点根据设计方案，既可以接受中央控制器的控制运行，也可以不依赖于中央控制器独立运行，即离线模式，如果某个计算节点失效，虚拟机会被调度到其他节点运行；如果失效时某个虚拟机正在运行，则可以通过备份和热迁移调度到其他节点继续运行。只要系统内计算节点的计算能力大于这个系统所要提供的云桌面所需的计算能力，在硬件不出现问题的条件下可靠性可以到良好的保证。

2)中央控制器的可靠性。中央控制器是整个系统的关键，也是单点失效点(single point of failure)。如果计算节点已经在运行了，中央控制器崩溃不会影响节点的正常运行；如果还未运行，那么计算节点会直接进入本地离线模式。这时,对中央控制器的可靠性要求就转化为是否能够及时的检测到控制器崩溃并重新启动。系统框架中的监控程序会不停的检测中央控制器是否还在工作，如果一旦发现有问题会立刻重启中央控制器。有了层的保证，中央控制器也可以达到一个非常高的可靠性。

2．可用性分析

系统的可用性(availability)是指在一段时间段内，系统的功能持续可用的百分比。从下列指标进行分析：在一个正常的工作日内，计算节点和中央控制器的可用性是多少。

1）计算节点的可用性

当采用多媒体教室电脑作为计算节点的使用模式时，只有一个虚拟机运行，并且该计算节点通过自动注册模块注册到私有云前已经对运行的硬件环境进行了测试。上课时候查询中央控制器并采取离线模式运行的方式保证了在没有硬件故障的情况下，计算节点几乎是100%可用的。

2）中央控制器的可用性

中央控制器的可用性计算方法：

(24 - (崩溃到被检测到的时间间隔 + 重新启动的时间)) ÷ 24 小时

在实验环境下，中央控制器重新启动的时间约为30秒，相对于重新启动的时间，崩溃到被检测到的时间几乎可以忽略不计。我们假设云控制器崩溃的频率为每天一次，那么中央控制器的可用性为:

100 - (30÷24×60×60) × 100 = 99.965%

而在实际运行中，不可能发生每天崩溃一次的情况。这说明中央控制器的实际可用性还会更高。

从以上的分析可以看出，整个系统在提供满足需求的功能时，它们的可用性也非常高，此外，中央控制器作为单独的服务进程，其可用性的高低不会完全影响整个系统的可用性。从普通用户的角度看整个私有云系统的可用性几乎接近99.99%。

## 4.5 本章小结

本章在第三章需求分析和架构设计的基础上，详细设计了私有云教学系统包含的各个主要模块和相关处理流程。本章节主要阐述了作者对原有系统进行的改造升级。

首先，从硬件层至业务层，本章由底向上详细说明该系统核心模块开发的详细设计及其软件实现过程。包括基于KVM虚拟化技术的计算节点的详细设计与实现，阐述如何将教室电脑作实现为一个虚拟化的计算节点来降低部署成本。

然后，进一步阐述了通过基于Thrift RPC通信和RabbitMQ消息队列缓存技术，的改进私有云中间层各模块调度和通信流程。

接着，从应用层为满足向教师提供一个可按需定制的教学桌面的需求，在业务逻辑层的实现中重点阐述了云桌面访问控制规则的算法，说明如何通过从RabbitMQ来实现系统的备份，以增加系统可用性。 阐述了客户端的实现方法。

最后从成本效益、可用性、可靠性几方面分析了本文对原有系统改进后所带来的整体的提升，使之能更好的为课堂教学提供成低廉，运行稳定的虚拟桌面服务。

# 第五章 总结与展望

## 5.1 总结

教育信息化的核心问题是如何将信息技术和教育行业的基本职责教书育人结合起来，即如何利用信息技术更好地为教学服务，创造出更多有利于推进深化教育改革的应用。而教育创新离不开IT基础设施的改进和发展。本文介绍了KVM虚拟化技术，并设计了一个建立在虚拟化技术之上的教学平台。研究与分析了如何实现一个能满足现有和未来教学业务的可扩展、高可用的私有云系统。通过私有云保证了校园教学资源的安全，通过虚拟化技术将教学桌面虚拟化后迁移到云上，节约了管理成本。中国各行各业正在进入“互联网+”时代。以云计算、移动互联、大数据技术等为代表的互联网技术正在深刻影响着人们的生产生活方式。教育领域也必定会受到此股浪潮的推动而向前发展。

本文从IT基础设施建设的角度，研究了如何利用KVM虚拟化技术对现有教育平台的基础设施进行改造与升级。将电脑的计算资源虚拟化而形成运行在私有云上的云桌面，带来的是教学模式的转变，包含两个层次：

层次一、教师使用教学电脑模式的转变。教师备课用自己的电脑，而上课却使用多媒体教室的教学电脑。不同操作系统和软件配置以及各种版本问题，经常影响教学正常进行。老师直接带笔记本到多媒体教室上课，会遇到网络环境参数设置等问题。此外，教学课件的传递往往通过U盘，潜在的病毒威胁会干扰到整个校园网络环境。私有云可以为每个教师提供统一的云桌面。这样教师能够专注于教学过程和教学内容的改进，而不会被IT配置的其他的事情所分心。

层次二、教师课程资源整合和使用模式的转变是教师授课的核心，所有的教学资源都应该围绕着课程来组织、管理和使用。这里教学资源包括：教师授课的课件、学生的作业、第三方的参考资料以及同样上这门课的其他教师的课件等。这就要求教师从单干转变为协作，教学资源的组织从分散转变为一个统一的、有序的模式。这就是常说的以课程为核心的基于WEB的教学应用。云计算平台在部署和运行这类应用的优势是强大的，可扩展虚拟存储能力能够提供安全可靠的存储，即使出现硬盘故障的情况，教学资源也不会丢失。教学资源的统一管理和存储，为在此之上的各种教学应用提供了原始数据，教学应用只需要关注如何提炼、重新组织和表现这些原始数据。这意味着教学应用只是教学资源的消费者，而生产者和所有者是云计算平台。

## 5.2 展望

在国家积极推进和政策扶植下，各行业正努力将云计算技术运用于自身领域，教育行业作为国家的特殊的产业，也在积极寻求创新和发展的新思路。随着当今教育理论和教学手段的不断进步，今天看上去比较先进的教学应用可能终将会被淘汰，代之以更科学、合理的新应用。但教学资源本身一旦形成后是基本不变的，而且会有长久保存价值的。变的只是对这些资源消费利用的方式，即建筑在新的教育理念上的新应用的最终目标和实现步骤。

首先，将一朵朵相互独立的教育小云汇聚成为一朵教育大云。这时，教育数据和教育资源的共享也达到了一个前所未有的高度，不再有各系统之间的割裂和数字信息孤岛。这也要求高校在信息化建设的过程中，要以松散耦合的系统构建方式，先搭好云计算平台。

其次，按照具体教学应用需求以及当前的资金投入，分阶段地建立好每个教学应用系统，最终达到构建一个够用、实用和好用的社会化教学系统的目标。

云计算辅助教育的最终目标是通过信息、资源和应用整合形成不分地域、国家和时间的数字校园。其核心任务是构造课程体系，建立最适合人类的学习方式和最完整的资源，改进学生如何学习和学到何种知识。这时的教与学完全融合为以适应自主学习为主的模式之中，而云计算技术能够帮助高校信息化建设逐步演化到最终阶段。

# 致 谢

首先要向我的导师XXX老师表示由衷的感谢。在论文开题阶段为我耐心地答疑解惑，在论文的写作过程中细心地帮助我指导写作方向，耐心梳理写作的思路，并且及时地指出细节上的不足。XXX的不断鼓励让我在整个论文完成阶段受益匪浅。课堂上XXX为人师表，能带动大家以积极而活跃的方式参与到与课程相关的讨论中，举一反三，融会贯通，使我明白从技术以外的更高层面来看待和认识软件工程的重要思想，及其未来的发展趋势，令人印象深刻。在此我要感谢XXX老师在我学习过程中对我提供的帮助。

其次，感谢我的家人。感谢他们在我的求学过程中对我学业的支持和鼓励，感谢我的爱人牺牲自己的时间，使我能全心全意的投入其中，并且激励我不断前进。

最后，感谢给予我指导和帮助的各位老师及同学们。

# 参考文献

[1]毕夫. 超级经济引擎的云计算[J]. 上海企业, 2015(2):44-45.

[2]杜占元. 深化应用 融合创新 全面深入推进教育信息化——在2017年全国教育信息化工作会议上的讲话[J]. 浙江教育技术, 2017(3):3-10.

[3] 杨金花. 云计算关键技术的探讨[J]. 电子设计工程, 2012, 20(15):86-88.

[4] Sandra Sergi Santos. 私有云计算的优点和选项[J]. 2012.

[5] 冯雪, 丁建军, 汪岳. 企业IAAS层私有云规划建设[J]. 信息系统工程, 2012(2):42-43.

[6]Wikipedia.Virtualization[OL].<https://en.wikipedia.org/wiki/Virtualization>

[7] Hong C H, Kim B J, Kim Y P, et al. Performance Prediction and Evaluation of Parallel Applications in KVM, Xen, and VMware[M]// Euro-Par 2014 Parallel Processing. Springer International Publishing, 2014:99-110.

[8] Varrette S, Guzek M, Plugaru V, et al. HPC Performance and Energy-Efficiency of Xen, KVM and VMware Hypervisors[C]// International Symposium on Computer Architecture and High PERFORMANCE Computing. IEEE, 2014:89-96.

[9]Sharma A, Ahmad A R, Singh D, et al. CloudBox — A virtual machine manager for KVM based virtual machines[C]// International Conference on Next Generation Computing Technologies. IEEE, 2017:588-594.

[10]何坤源.Linux KVM虚拟化架构实战指南。人民邮电出版社。2015年7月.48-50

[11]姚华超, 王振宇. 基于KVM-QEMU与Libvirt的虚拟化资源池构建[J]. 计算机与现代化, 2013(7):26-29.

[12]ArchLinux.Libvirt[OL].https://wiki.archlinux.org/index.php/Libvirt

[13]Marten Mickos. Eucalyptus Cloud-computing Platform [OL].<https://github.com/eucalyptus/eucalyptus.2017>

[14] 田翠珍. 基于Thrift框架RPC的研究与实现[J]. 信息与电脑, 2016(1):83-84.

[15] Surhone L M, Tennoe M T, Henssonow S F, et al. Apache Thrift[M]. Betascript Publishing, 2010.

[16] David Dossot.RabbitQM Essentials.Packt Publishing. 2014年4月.第一版. 13-15

[17] 王建宇. SpiceProtocol与虚拟桌面架构VDI技术[J]. 信息系统工程, 2016(4):70-70.

[18]程聪.云计算+职业院校与中小企业.清华大学出版社。2015年10月，第一版.67-70

[19]熊昕,熊茂华,晏细兰.创新型实践教学私有云平台的设计与研究[J].中国教育信息化, 2015(21):84-87.

[20] 郑爱民,满青珊,孙亭.一种基于云的智慧城市系统架构[J].中国电子科学研究院学报,2014,9(3):226-233.

# 论文修改说明

针对评审老师在论文中指出的客观问题再次进行调查研究和认真思考，为了更好的阐明论文题目所要表达的内容，随后对论文的相关章节进行了如下修改：

1. 重写论文摘要的中英文部分，修改了对论文主题的描述和简单介绍。增加了对本论文中论述项目各个组成部分及其功能的概述。增加了本论文对原有系统改进和体现本人工作的描述。并且检查了英文摘要的表述修正了语法问题。使摘要可以更加准确的体现整篇论文核心内容，符合论文摘要的写作要求。
2. 对第一章的“内外相关研究和应用”移动到1.2节，并且进行了大量的修改，增加了国内外基于云计算的教学应用介绍，并且提出了个人的思考。在第一章的“论文内容”中增加了现有项目的背景描述。介绍了项目的现状以及实施过程中发现的问题与不足。然后，进一步针对这些问题描述了本论文对原有项目的改进，体现作者在论文中的工作与贡献。
3. 第二章的第一小节增加了对公有云和私有云的对比与思考。重写了私有云教学系统的介绍，分析了问什么要在教育行业采用云计算以及私有云，私有云的优势与不足。增加了项目背景的内容，针对原有项目的不足，简单描述了本系统对私有云缺点的改进，以及作者在项目中对原有项目的贡献。加入了作者自己的观点和在本文中实现的软件系统的工作介绍。在第二章最后增加了“本章小结”。概要的总结了该系统所涉及的技术的基本原理和特征。
4. 在第三章3.1节的需求分析中，结合教育行业特点更加深入的阐述了非功能性需求的描述，删除了与本文无关的需求；重新撰写了3.1.3小节功能性需求，增加了用户用列图，并从软件的范围、运行环境、用户接口和业务功能几方面按照软件工程的要求阐述了本论文的功能性需求 。使本章内容能够指导后续的软件实现过程。在第三章最后增加了“本章小结”。简单总结了私有云系统设计方面的内容。
5. 修改了而第四章中描述不够严谨的部分。删除和修改了第四章中操作说明式的表述。删除和修改过于工程化的描述和口语化的描述，增加了对问题的论述和个人的思考。增加了自动注册模块的流程图和接口定义。在应用层实现的内容里增加了对客户端设计和实现的阐述。增加了对采用了本文对私有云系统的改进后的成本效益分析。最后增加了“本章小结”。简明扼要的阐述的私有云系统实现方面的概要。
6. 更新了论文的目录，并重新检查了论文的格式，使其符合研究生论文的写作要求。