

摘要

随着多媒体通信技术的发展,多媒体技术和计算机有效地结合了起来,多媒体远程培训逐渐成为了一种流行的学习和培训方式。由于中国在世界经济的舞台上逐渐扮演者越来越重要的角色,优质人才的需求必然导致传统教育方式的变革。传统的书本的教育方式已经不可能满足人们对于更丰富更方便的教育方式的需求。而远程培训为学习者提供了更为弹性的学习条件,使得人们更容易突破时间和空间的限制,帮助人们更方便的学习,有助于促进教育社会化和学习社会化,同时,网络提供的良好的交互性也能够让学习者更好地吸收所学的内容。从企业和机关的角度出发,基于 Internet 的远程培训有效的减少了员工因为参加培训而产生的出行成本,同时增加了培训投资的总效益,也扩大了培训的反馈和培训的频率,使得培训效果更好,企业还能根据实际需求为员工制定更为灵活的培训计划。多媒体远程培训系统普遍通过在线直播培训和结合多媒体课件的点播培训提供了同步培训和异步培训两种主要方式。由于企业和机关的网络架构在远程培训部署的过程中骨干网存在着带宽瓶颈的问题,所以如何针对此种网络架构在培训系统中实现有效的调度策略以及如何实现远程培训的同步培训和异步培训功能,有着重要的理论和现实意义。

本文首先在对多媒体远程培训系统进行需求分析的基础上,提出了系统的功能模型,架构模型以及该系统适用的网络架构。然后对于多媒体远程培训系统中视频直播同步培训的基本原理进行了阐述,提出了以流媒体服务器为网络中心的多路视频推送直播模式;基于时间轴模型和事件驱动机制,对多媒体课件制作中多媒体文档转换、视频与多媒体文档同步、视频编辑等技术进行了研究,提出了一种多媒体课件的制作方法;对内容分发网络(Content Distribution Network)进行研究的基础上提出了一种基于内容分发网络(CDN)的视频文件调度策略。最后根据系统的实际使用需求在 Asp.net, Winform, Silverlight 三个技术平台上使用统一的 C#语言对多媒体远程培训系统进行了实现。

关键词: 远程培训; 网络直播; 多媒体课件制作; 视频调度

Abstract

With the development of multimedia communication technology, multimedia technology and computer technology combines and distance training is becoming more popular than ever before. China is playing an increasingly important role in the stage of the world economy, which will lead the demand for people of high caliber to a change in the way of traditional education. The traditional way of education using books have been impossible to meet the needs of more convenient way of education. However distance training makes it easier to break through time and space constraints, to help people more easily learning, which contribute to education socialization and learning socialization. From the perspective of enterprises and institutions, Internet-based distance training, effectively reduce the travel costs of the staff who attend the training, while increasing the total benefits of investment in training and expanding the training feedback and training frequency. Enterprises can also make a more flexible training programs based on actual demand for staff. Distance training system generally provides synchronous training and asynchronous training in two main ways through live online training and on-demand training combined with multimedia courseware. Due to the shortage of network architecture for enterprises and institutions in remote training deployment, how to achieve streaming media scheduling strategy, synchronous training and asynchronous distance training system in such network architecture has important theoretical and practical significance.

Firstly, based on the distance training system requirements analysis, the structure model and function model of the system has been proposed. Then, on the basic of principles of live synchronous training, a streaming media server for push broadcast model has been proposed, then a way of multimedia courseware producing based on the timeline model and event-driven mechanism, after researching the multimedia courseware conversion, synchronization of video and multimedia courseware and video editing has been proposed, a streaming media

scheduling strategy based on principle of CDN has been proposed after some work on the CDN distribution network. At Last, System has been achieved on Asp. Net, Winform, and Silverlight three technology platforms using unified C# language.

Key Words: Distance Training; Live Broadcast; Multimedia Courseware Producing;
Streaming Media Scheduling

第1章 绪 论

1.1 课题背景及意义

多媒体通信技术突破了计算机、通信、传统出版业的界限，将多媒体技术和计算机相关技术有机的结合在了一起，是多媒体技术和计算机，通信等技术相结合而逐渐发展出来的新的边缘技术^[1]。多媒体通信技术向人类提供了各种各样拥有全新体验的强大信息服务，如：网络教育，实时视频会议，以及远距离医疗等。多媒体通信技术将极大地提高人类对多媒体的体验和使用的效率，改变生活的方式，是 21 世纪人们通信的基本方式。

远程培训是随着多媒体通信技术而出现的一种全新的学习方式，它为学习者提供了更加弹性的学习条件，有助于促进教育社会化和学习社会化。如今许多国家十分重视远程培训，努力扶持并发展这一教育方式，为推进网络化的教育方式的普及与发展投入大量资金，并不断扩大远程培训的普及化程度和培训的层次以及内容。

随着中国在世界经济的舞台中扮演着越来越重要的角色，人才市场的开放化与国际化将不可避免。优质人才的需求必然带来教育体制的变革。终身教育对我们来说，已经不是纯粹的理论而已经是正在发生的现实。传统的书本式的教育方式已经不可能满足对于教育的巨大的需求，我们迫切需要更丰富的教育方式和手段来满足人们日益增长的教育需求。而基于 Internet 的网络培训的出现，有效地解决了这个问题。网络使得人们更容易突破时间和空间的限制，帮助人们随时随地地学习，让更多的人共享优秀的教育资源^[2-3]。同时，网络提供的良好交互性让人们更好地吸收所学的内容。

在机关和企业中，以企业文化和技术为主题的远程培训也占据着企业培训中越来越重要的地位。从企业的角度出发，远程培训有效地减少了员工因为参加培训而支出的出行成本，而且提高了培训投资的总效益，也扩大了培训的反馈以及培训的频率，使得培训的效果更好，企业还可以根据员工的工作时间表制定更为灵活的培训时间表^[4]。远程培训以其低成本，个性化以及良好的培训效果而成为企业培训的新宠，越来越多的企业开始采用远程培训进行员工培训。

远程培训涉及了与网络相关的若干种培训方式。直播培训的方式能够及时的将培训的内容通过视频直播呈现给员工，保证员工在第一时间获得所需要接受的信息，保证了实时性。多媒体教室中的讲师能够通过多个摄像头间切换视频输出来保证直播观看者在讲师讲课过程中更好的观看体验。而视频点播与多媒体课件结合的方式则改变了传统网络培训只能使用简单文字和图片进行培训的不足，使得网络培训的形式更加丰富。多媒体化的网络培训通过视频，图片等多方面去刺激学习者，加深了学习者对于新知识的印象，提升了整体的培训效果。直播培训和点播培训这两种培训方式有着各自的优势和不足，在典型的培训系统中由于需求不同都有着各自存在的必要。

机关和企业的网络架构普遍有着自身的特点，在实际运行远程培训系统的过程中骨干网的带宽瓶颈是一个普遍存在的问题，更换底层网络架构，增大带宽是一个有效的方案，但是由于耗时过长，投资相对较大，并不是一个值得推荐的方法，所以远程培训系统需要针对这种网络环境与特点进行一种有效的流媒体调度策略来保证系统在不改变底层网络架构的情况下稳定运行。

1.2 国内外研究状况与进展

远程培训是 20 世纪随着通信技术的发展而产生出来的新生事物，远程培训的研究具有相当短暂的历史。1969 年召开的第 8 次国际教育发展会议上，查德斯(G.B.Childs)发表了关于远程培训相关的国际调查报告。可以说他的报告是最早涉及远程培训的相关研究报告。

1988 年在挪威首都奥斯陆召开的世界教育会议上，远程培训成为会议的主题，文献中有 40 多项直接涉及到关于远程培训的研究，其他许多论文也与此有关。1992 年在泰国曼谷召开的第 16 次国际教育发展会议上，有 35 篇论文涉及了远程培训的问题。

作为一个研究领域，远程教育的理论研究制约着其实践中的发展，特别是关于远程教育自身的定义阐述以及其理论构建十分重要。没有理论基石和特征的一致认同，远程教育的研究很难取得有效进展。总体而言，国外远程教育研究的发展具有四个基本阶段：

1 第一阶段：术语研究阶段

远程教育作为一种抽象概念为欧洲所接受。

2 第二阶段：定义研究阶段

一些研究者对于远程教育的概念阐述有着浓厚的兴趣，尤其是关于远程教育的特性定义。国外研究者对于远程教育给予了不同的特性定义，其中较突出的是科根(Keegan)关于远程教育五个特征的界定：(1)学习过程中教师和学习者的准持久性分离(2)一个教育组织的影响既体现在学习材料的计划和准备中，同时体现在对于学习者的服务上(3)技术媒体的应用使得教师和学习者成为整体(4)双向式的信息交流使学生从对话中获得收获(5)学习过程非集体化，是单纯的个人行为。

3 第三阶段 对象研究阶段

远程教育在上世纪 90 年代取得了较大的发展。远程教育不再被视为一种纯粹的学术性学科，而是教育人员的一个研究领域。

4 第四阶段 研究的重心

远程教育领域的出现是因为传统教育体制有其自身局限性，现在多媒体通信技术在人们生活和教育中扮演着越来越重要的角色。这种事实导致远程教育研究从教育研究体系逐渐分离出来，并发展成为一个单独的研究领域。这个领域的重点在于如何满足学习者个性化的需要，现在网络技术如何有效地和传统媒体进行结合，发挥其应有的价值^[5-6]。

美国的部分高校已经开设了基于 Internet 的远程学习课程，如斯坦福大学，华盛顿州立大学，加利福尼亚大学，有些公司和教育基金也对网络教育提供了资金支持。开发远程培训教材比单纯的书写教材复杂，需要许多不同类型的专业人才互相配合，需求持续的信息输入和修改，这是一个不断增量的过程。培训教材需要不断更新才能与最新的技术相适应。而编制新的以技术为基础的培训教材需要集体的努力，在时间，经费和设备等方面的投资远远超过目前用于编写教科书的投资。而高校有人才，经验和设备，最有条件涉足这一增长的市场。当前美国每年对教育和培训的投入有 10000 亿美元，并在不断增长，但是增长的领域不在仅仅局限于传统的大学教育，而是成人继续教育。企业中的工程师和知识工作者由于需要追赶各自领域的发展速度，所以这些专业人士都把网络教育当成继续教育的主要手段。美国哥伦比亚大学师范学院院长阿瑟·列文日前发表谈话指出，新科技和互联网所产生的强大冲击波有可能导致美国现行的大学体系彻底解体。

远程教育在中国的发展总共经历了三次重大变革，第一次是函授教育，第

二次是 80 年代兴起的广播电视教育，在 90 年代，随着网络通信技术的发展，产生了以 Internet 为基础的远程教育。网络的发展和技术的提升为远程培训系统的建立提供了实现可能^[7]。

在第三代远程教育的初期，由于当时网络技术和计算机硬件的限制，直播授课并为广泛使用，多媒体课件也只是包含网页和图片的 Web 页面，学员进行网络培训只是对 Web 页面的简单浏览，系统对学员的学习过程并不能进行跟踪记录。随着计算机技术的成熟，网络技术的改善，直播授课渐渐流行，多媒体课件渐渐的加入了视频音频的素材，但是缺乏两者之间的同步控制。

Evans 和 Haase 经过研究总结了远程培训的优缺点，远程培训的优点主要在于（1）增加培训投资的总效益；（2）减少了员工在培训中需要支付的各种出行成本和时间；（3）总量上增加了培训时间但是减少了纯粹的培训时间；（4）增大了培训范围和增加了培训频率；（5）可以按照工作时间表依据员工各方面情况来制定个性化的培训时间表；（6）员工可以在家中进行远程培训；（7）协作学习可以增加员工的团队意识和协作精神（8）有利于开拓国内外市场。而远程培训也有一些不足：（1）没有切实可行的员工评价系统；（2）没有不同方案的成本评估和教学设计思想不成熟。

在今天，与传统的教育模式相比，远程教育的研究者将远程教育的实践工作和理论研究并重，这种研究趋势的发展并不是偶然的。远程教育研究热潮兴起主要有着以下几个原因：（1）远程教育为学员提供的反馈机会相对较少，有必要开发远程教育方案评价的正规程序。（2）远程教育其实是很多专业的分工，因此，远程教育的体系组织，管理变得更加重要。（3）远程教育的热潮大多基于社会目标和特殊群体需要的目标。教育产出评估越来越受到官方的重视，由于经济发展的外在压力，国家对远程教育的成本效益越来越强调。

如今，随着网络技术的进一步发展，直播授课成为了一种非常有效的培训学习方式，而视频，音频等大颗粒信息单元成为了多媒体课件的主体，同步控制机制使得多媒体课件同视频音频的同步播放成为可能。系统对学员的学习过程能够进行记录并进行分析。

1.3 本文所做工作

本文以企业网络架构下的多媒体远程培训系统作为研究对象，着重针对企

业网络架构下的多媒体远程培训系统的视频直播，多媒体课件制作，视频文件调度等关键技术进行研究。主要研究内容如下：

1) 以流媒体服务器为网络中心，通过流媒体服务器作为直播发布点的体系架构模型

早期的直播培训系统需要用户自己来选择需要浏览的直播摄像头，由于本身网络架构的缺陷造成网络摄像头端的网络瓶颈影响了系统整体的使用。该系统通过研究将视频流推送到流媒体服务器上并建立公共发布点和视频编码技术解决了这个问题，同时可以保证在不中断直播流的同时在多个摄像头中进行切换，给直播培训提供了更好的体验。

2) 基于直播产生的毛坯文件的多媒体课件制作模型及其实现技术

对直播产生的毛坯文件重新进行处理，基于时间轴模型制作多媒体课件，引入事件驱动机制保证点播的时候的视频与多媒体课件的同步。

3) 针对企业网络架构实现远程培训系统的骨干网带宽问题提出了基于 CDN 分发网络的视频文件调度模型，以及后台分发与复制模型。

作为针对上述理论体系的具体应用，结合实验室项目需要，设计并实现了多媒体远程培训系统。这也是本文的重要的技术研究成果。

1.4 本文结构和组织

论文共分为六章，内容概要如下：

第1章 阐述了本课题的研究背景，研究意义以及国内外的研究状况和进展，针对现阶段的研究现状，提出了本文的研究重点，简要的阐述了论文所做的工作，以及论文的组织结构。

第2章 对远程培训系统的系统需求进行了详细说明，并对其进行了功能分析。利用软件工程结构化理论化理论结合远程培训系统的业务功能对系统的结构进行了分析建模，提出了功能模型和架构模型。

第3章 对远程培训系统中的直播模块的涉及的基本原理，功能设计以及实现过程进行阐述，提出了以流媒体服务器作为网络直播中心的适用于直播的网络架构。

第4章 对远程培训系统中的多媒体课件制作模块中涉及的基本原理，功能设计及其实现过程进行了阐述，引入时间轴模型和事件驱动模型解决了多媒体

课件在线显示以及多媒体课件元素之间的在线同步问题。

第 5 章 对远程培训系统中的点播调度涉及的基本原理，调度策略及其实现过程进行了阐述，提出了一种基于 CDN 分发网络的使用于企业网络的视频文件点播调度的策略。

第 6 章 对系统整体代码实现的应用平台进行了相关讨论，同时对程序实现过程中的分层架构进行了阐述，最后对系统的实现情况进行了阐述。

第 7 章 对本文所做的工作进行总结，并对后续要进行的工作进行了声明。

第 2 章 多媒体远程培训系统架构体系设计

2.1 系统需求分析

多媒体远程培训系统针对企业内部网络架构实现了企业内部员工的多媒体同步和异步培训的功能。提供了实时视频直播学习，后期多媒体课件制作，视频点播学习等多媒体远程培训系统应有的功能。由于企业内部网络架构相对于传统 Internet 的网络架构有一定区别，而且多媒体远程培训一般通过远程视频的方式进行，系统针对这种特殊性对 Internet 上采用的优化技术进行了修改，使其更好的适应企业内部的网络架构。

多媒体远程培训系统首先通过多媒体教室中安装的若干个摄像头将多媒体教室课堂的实时视屏分发给想对视频进行收看的所有用户，完成直播功能，用户可以通过实时视频及时的获取信息。与此同时在直播过程中会将分发给用户的视频数据复制一份，作为一个视频文件保存在多媒体教室的 PC 机上。产生的视频文件会作为毛坯文件，在直播完毕之后，可以将得到的毛坯文件与课堂上讲师所用到的比如 PPT，Word 等多媒体资源进行整合，做成体验性更好的多媒体课件，将其上传到服务器上，供没能及时看到视频直播的用户或者是想重新进行二次培训的用户进行观看。

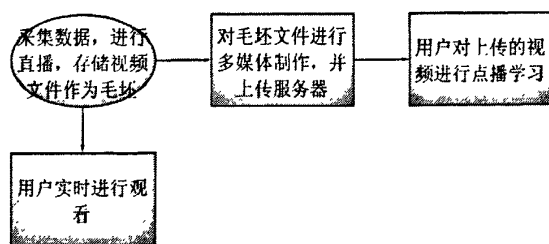


图 2-1 多媒体远程培训系统的基本使用流程

很多企业网络具有的特点就是中心服务器部分的带宽普遍比较有限，这种网络架构特点在很大程度上制约了远程培训系统所能发挥的作用。本系统在完成多媒体远程培训流程的同时，针对企业网络的这个特点，在借鉴 CDN 内容分发网络技术的核心思想上，实现了一种适用于企业网络架构的流媒体调度

策略，有效地保证了多媒体远程培训系统在大访问量下的系统稳定性。

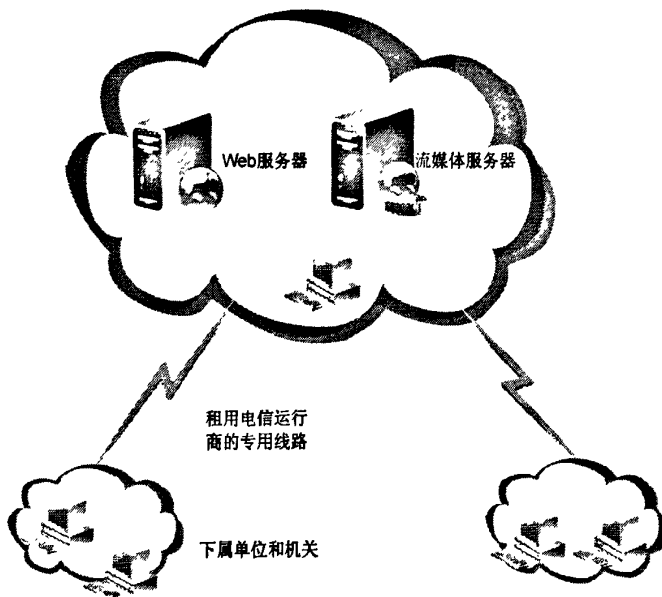


图 2-2 多媒体远程培训系统适用的典型企业网络架构图

图 2-2 是一个典型的企业内部的网络架构图，企业内部网络中心部署了 Web 服务器和流媒体服务器等相关核心的服务器，然后各个下属子机构通过租用电信运营商的相关的网络线路同网络中心的服务器进行连接。由于租用的成本原因导致企业内部骨干网络的带宽有限，当大量用户同时访问视频这种较占用带宽的多媒体资源时，就很容易造成网络拥塞。多媒体远程培训系统需要对此采取一定的调度策略来对此情况进行优化。

2.2 系统功能设计

根据系统需求分析可知，本系统需提供如下的应用功能：(1)系统管理，包括用户管理，视频管理，网络硬盘管理；(2)视频直播；(3)多媒体课件制作；(4)点播调度；(5)资源分发与管理 (6)资源重定向

将上述各功能进行抽象，采用模块化的方法对功能进行封装，系统的功能模块有提供管理模块，包含用户管理，视频管理等，视频直播模块，多媒体课件制作模块，随机点播调度模块，如图 2-3 所示。

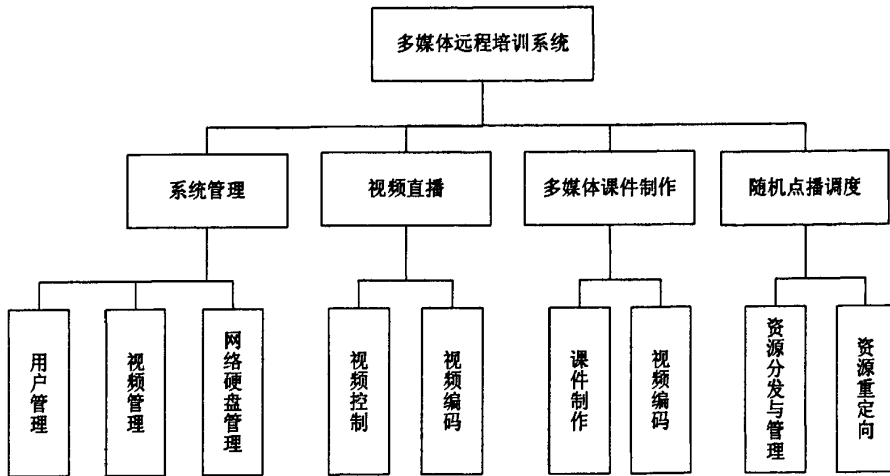


图 2-3 多媒体远程培训系统功能模块划分

2.3 系统架构设计

由上述分析可知，该系统的主要目的就是实现同步和异步的多媒体远程培训，同时解决企业内部网络架构本身对于多媒体远程培训系统的局限性带宽问题。

上述过程可以分解为以完成不同任务为目标的若干功能子系统，根据层次化理论，将用户置于系统操作最顶层，整个系统的将架构模型如图 2-4。

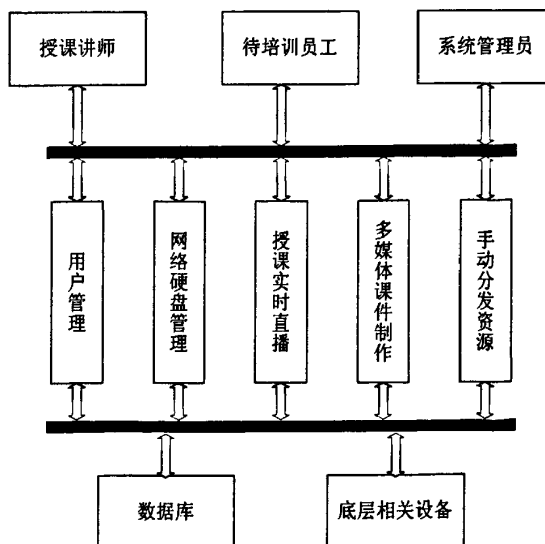


图 2-4 系统架构图

2.3 本章小结

本章主要阐述了适用于企业网络架构的多媒体远程培训系统的架构体系。对系统进行了基本的需求分析，并给出了系统适用的网络架构图，结合系统的业务流程，采用软件工程的结构化理论对其进行了功能模块的划分，最后给出了系统的架构模型，为后期系统的实现提供了理论基础。

第3章 基于流媒体服务器的多路视频推送现场直播

3.1 基本原理

3.1.1 视频编码技术

由于视频信号数字化后的数据带宽很高,计算机很难对之进行存储和处理。视频编码技术即通过特定的压缩技术,通过算法将数据带宽降低,这样就能比较方便的将视频信号保存在计算机中并对其进行处理^[8]。

目前应用比较普遍的视频编码方案有三种: MPEG-4, H.264, VC-1, 下面分别对这几种编码格式进行简单的介绍:

MPEG 它的英文全称为 Moving Picture Expert Group, 即运动图像专家组格式。MPEG-4 之前的编码技术主要采用的是第一代压缩编码技术, 即把视频序列按时间先后分为一系列的帧, 每一帧图像又分为若干块进行运动补偿和编码, 这种编码方案有着诸多不足。MPEG-4 标准在 1999 年正式提出, 采用了基于内容编码的编码方案。MPEG-4 代表了基于模型/对象的第二代压缩编码技术, 利用了人眼视觉特性, 抓住了图像信息传输的本质, 从轮廓、纹理思路出发, 支持基于视觉内容的交互功能, 这适应了多媒体信息的应用由播放型转向基于内容的访问、检索及操作的发展趋势。MPEG-4 提出了 AV 对象的概念, 对象是指一个场景中能够访问和操作的实体, 对象的划分可以根据其独特的纹理, 形状等作为划分依据, 音频视频已经不再是第一代压缩编码中的图像帧, 而是一个个 AV 场景, 这些不同的场景由不同的 AV 对象组成。MPEG-4 编码标准的基本内容就是对 AV 对象进行高效的编码、组织、存储和传输, AV 对象编码就是 MPEG-4 的核心编码技术。MPEG-4 是为了播放流式媒体的高质量视频而专门设计的, 通过压缩和传输数据, 在带宽有限的情况下以求使用最少的数据获得最佳的图像质量^[13]。MPEG-4 最有吸引力的地方在于它能够保存接近于 DVD 画质的小体积视频文件。

H.264 是由 ITU-T 视频编码专家组 (VCEG) 和 ISO / IEC 动态图像专家组 (MPEG) 联合组成的联合视频组 (JVT, Joint Video Team) 提出的高度压

缩数字视频编解码器标准。该标准是目前最广泛使用的高清晰视频格式。H.264 是在 MPEG-4 技术的基础之上建立起来的, 所以也被称为 MPEG-4 AVC, 其编解码流程主要包括 5 个部分: 帧间和帧内预测(Estimation)、变换(Transform)和反变换、量化(Quantization)和反量化、环路滤波(Loop Filter)、熵编码(Entropy Coding)。H.264 对 I 帧的编码是利用空间相关性实现的, 以前的编码标准只是利用了一个宏块内部的相关性, 忽略了块之间的相关性。H.264 的帧内帧内预测编码是基于块的, 对于亮度分量, 块的大小可以在 4×4 和 16×16 之间选择。对于色度分量, 有 4 种预测模式, 每种预测模式都对应不同方向上的预测。H.264 的帧间预测编码在宏块划分上, 采用了 $8 \times 8, 8 \times 16, 16 \times 8, 16 \times 16$ 这 4 种模式。H.264 支持多达 16 帧的的参考帧预测, 允许在缓存中的多个图像中选取一个 P 帧或者两个 B 帧作为参考帧, 虽然多参考帧使得缓存容量咋家, 加大了编解码器的复杂度, 但是可以改善运动估计的性能, 提高解码器的错误恢复能力。为了消除在预测和变换过程中引入的快效应, H.264 采用了消除快效应滤波器, 与块效应滤波器不同的是, 经过滤波后的图像需要放在缓存中进行帧间预测, 而不仅仅用来重建图像时改善质量。由于该滤波器位于解码环中, 因此又被叫做环滤波器。H.264 对帧内或帧间的预测的残差进行变换编码, 以 4×4 为变换基本单元, 由于变换块的尺寸较小, 运动物体的划分更精确, 这样不但变换计算量小, 而且在运动物体边缘处的衔接误差也大为减小。H.264 将强了对各种信道的适应能力, 采用“网络友好”的结构和语法, 有利于对丢包的处理。H.264 的技术有着诸多先进之处, 比如统一的 VLC 符号编码, 高精度, 多模式的位移估计, 基于 4 块的整数变换, 分层的编码语法等。这些措施使得该编码方式有着很高的编码效率, 在同等的图像质量, H.264 的数据压缩比能比当前 DVD 系统中使用的 MPEG-2 高 2~3 倍, 比 MPEG-4 高 1.5~2 倍^[12,14]。H.264 在具有高压缩比的同时还拥有高质量流畅的图像, 正因为如此, 经过 H.264 压缩的视频数据, 在网络传输过程中所需要的带宽更少, 也更加经济。H.264 的码流结构网络适应性强, 增加了差错恢复能力, 能够很好地适应网络应用。

VC-1 即视频编解码方案一(Video Codec One), 基于微软 WMV9 编码技术, 2003 年正式提出, 于 2006 年正式被 SMPTE 批准成为国际标准。该压缩技术整合了 MPEG 及 H.264 的优点, 是两者之间一个较好的平衡。VC-1 的帧内预测采用的是传统的 MPEG 标准的预测格式, 帧间预测参考帧仅由前 I 帧的

重建帧构成，煎炒了缓存容量，使得解码器的复杂性大大降低。VC-1 的 B 帧预测编码采用前向后向和补偿的方式，为了提高编码的性能，采取了三个措施，B 帧编码位置只与两个参考帧相关而与时间无关，采用 B 帧的帧内编码方案，在出现场景切换时提高了编码的整体效率，运动矢量采用前向预测前向，后向预测后向的原则，并分别使用缓存运动矢量，改善了缓存质量，提高了运动矢量的连续性。VC-1 的滤波器采用了两项技术，一是采用了环路滤波器，二是采用了重叠平滑技术。其变换的基本单元不是采用的固定的矩阵，二是使用了自适应变换。这种灵活措施可使其在高码率还是低码率下都有着良好的性能。VC-1 只有 4 种动作补偿 (motion composition)，从压缩比无法胜过 H.264。但是由于算法比相对简单，VC-1 在压缩时间上，明显比 H.264 短了许多，复杂度约只有 H.264 的 50%^[11]。

现在对这三种主流编码技术进行一下综合比较，由于 MPEG-4 在压缩比上没有优势，已经渐渐的被 H-264 技术进行取代，所以主要是对 H264 和 VC-1 进行比较^[9-10]。在压缩比上，H-264 的表现较 VC-1 要更好一些，但是高压缩比带来的必定是对硬件的较高要求，整体考虑多方面因素，VC-1 具有更好的均衡性，所以决定选取 VC-1 作为直播输出源的编码方式。

3.1.2 流媒体技术

多媒体文件一般较大，由于网络带宽的限制，如果下载到 PC 机往往需要的时间比较长，总体来说这种方式的延迟较大，也大大限制了多媒体的应用。流式传输的出现有效的解决了这个问题，流式传输时，声音和影像实时的发送到 PC 机，并在 PC 机上创建一个缓冲区，播放前先下载一部分作为缓冲，用户不需要等到整个文件下载完毕，而只需要经过几秒的启动延迟就可以观看多媒体视频。而在用户进行观看的同时，文件未下载的部分将在后台继续进行下载到缓存之中。与完全下载的方式相比，这种对多媒体文件边下载边播放的流式传输不仅使用户的启动延时大幅度地缩短，而且对系统缓存容量的需求也大大降低^[15-16]。

流式传输除了能够发送编码的 A/V 文件以外，还可以通过采集服务器实时采集现场音视频，推送到流媒体服务器端，实时播放。因此，流媒体除了能够更好的承担如下载一样的多媒体点播服务外，更能够应用在现场直播、电视转播、突发事件报道等多种对实时性传输要求较高的领域^[17-19]。

实现流式传输有两种方法：实时流式传输（Realtime streaming）和顺序流式传输（Progressive streaming），如视频为实时广播，可以使用流式传输媒体服务器或者使用 RTSP 这样的专门设计的实时协议，如果使用 HTTP 传输，文件则是顺序流传输。

1) 顺序流式传输

顺序流式传输是媒体文件顺序下载，用户只能观看已下载的那部分而不能跳到未下载的部分，由于 HTTP 协议本身的限制，这种传输方式也不能在传输期间根据带宽情况进行调整。一般情况下，HTTP 服务器即可发送这种形式的文件，所以这种方式也称为 HTTP 流传输，

2) 实时流式传输

实时流式传输和顺序流式传输不同，它采用专门的流媒体服务器和传输协议，实时流媒体支持随机访问，可以对观看的内容进行快进和后退。实时流式传输需要特定流媒体服务器，如 Darwin Streaming Server、Helix Server 与 Windows Media Server。这些服务器允许你对媒体发送进行更多级别的控制。实时流式传输还需要特殊网络协议，如：RTSP(Realtime Streaming Protocol)或 MMS。

流式传输相对于传统的下载后播放大幅度的减少了启动延时，而且由于所有的内容都下载到缓存之中，所以大大减少了所需的空间。在多媒体网络传输中，采用流媒体传输是比较好的选择。

3.1.3 网络架构分析

用户对多媒体教室内的教学场景有若干种方式进行实时观看，一种是直接向多媒体教室中的摄像头发起数据请求，现在一般的网络摄像头都允许直接通过 Web 对摄像头的实时画面进行查看。典型的网络拓扑结构如图 3-1 所示。

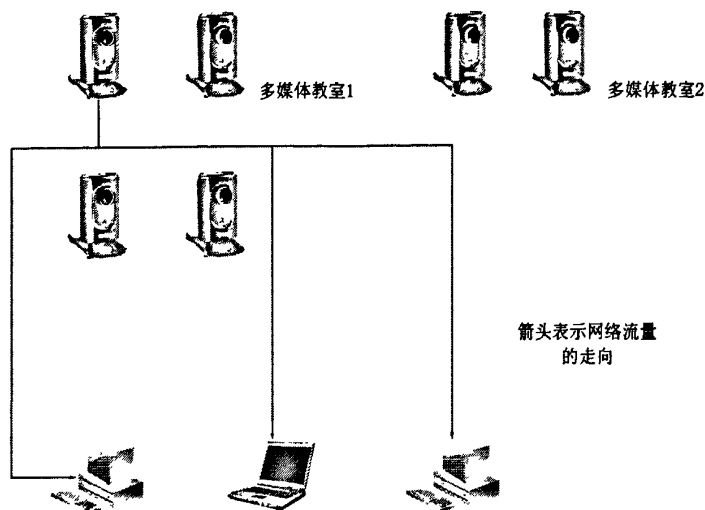


图 3-1 直接通过摄像头进行直播的网络架构图

这种网络架构存在几个缺点：

- 1) 对于选取哪个摄像头需要用户自己手动的切换，而且切换的过程中势必会对用户观看直播的用户体验造成影响，使其不能专注于观看视频本身。
- 2) 网络摄像头的网络连接普遍采用以太网连接，常用的是 10M 或者 100M 的以太网网线进行连接，相对于核心骨干网普遍采用的光纤组网方式，如果访问用户过多容易在此处造成网络堵塞，影响系统整体性能。

考虑到上述不足，在网络架构中加入了流媒体服务器，摄像头的网络流量直接推送给位于网络中心处的流媒体服务器，如果有多个用户需要查看摄像头的直播的视频，流媒体只需要将摄像头发送过来的流量进行分发即可满足需求，流媒体服务器处于网络中心的位置，造成网络堵塞的可能性相对于上一种架构来说要小，系统整体性能更加优化。

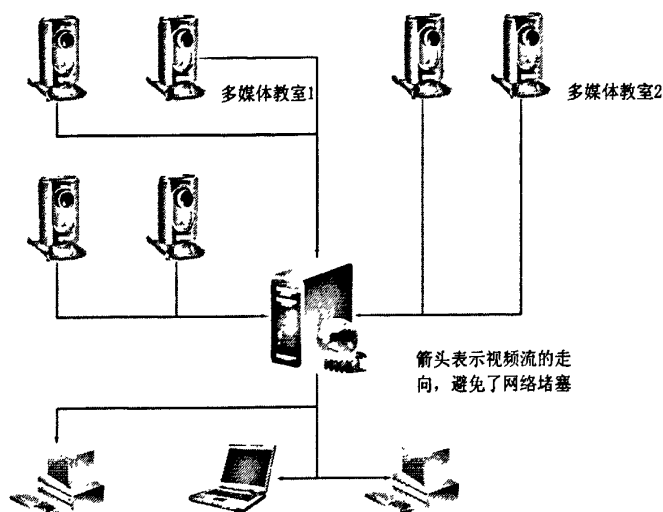


图 3-2 以流媒体服务器为中心的网络架构图

3.1.5 流媒体服务器

流媒体服务器是整个流媒体应用的核心，流媒体服务器的主要功能是对流媒体内容进行缓存，调度和传输，而整个应用的性能体现很大部分决定于服务器的性能和服务质量。所以流媒体服务器是整个流媒体服务的基础也是重要的组成部分。

现在比较流行的流媒体技术解决方案主要有 Real Neatwork 的 Helix Server, Apple 的 Darwin Streaming Server, 和 Microsoft 的 Windows Media Server。

Real 公司是因特网多媒体数据流技术和市场的创立者和领导者。目前，全球超过 2 亿的个人计算机用户通过其 RealPlayer 软件。Helix 是其推出的流媒体服务器产品，该产品不仅支持自身的 rm 等媒体格式，还广泛支持微软的 wmv 等视频格式，是公认的优秀的媒体软件。Helix Server 采用 RTSP 协议传输视频，推荐采用 Real Player 作为前端播放器^[20]。

Darwin Streaming Server 简称 DSS, 是 Apple 公司提供的开源流媒体服务器程序。整个程序采用 C++编写，设计上遵循简单，模块化，高性能等设计原则，并且开放源代码。可以运行在 Windows NT 和 UNIX 实现上，包括 Linux, FreeBSD 等操作系统。支持多种媒体格式和 RTSP 协议。

Microsoft 由于其平台植入的便利性在流媒体服务器上有着很大优势。Windows media server 支持 MMS RTSP 这两种多媒体传输协议。Windows Media Server 一般情况下是以组件的形式集成在微软操作系统中，用户不需要购买其

他第三方的软件^[21]。

对三种流媒体服务器进行综合比较,虽然技术细节每个公司都有不同,但是功能上基本相同,流媒体服务器的一般功能都能满足。而从软件成本和管理成本上来分析,微软的流媒体服务器由于和系统结合更为紧密,系统管理更为方便,软件成本基本不需要另外支出,是一个较为理想的选择。

流媒体代理服务器可以代替流媒体服务器接受用户的请求,主流的流媒体服务器都提供代理。通过在用户较多的区域部署代理服务器,该区域内的用户直接通过代理服务器请求数据,代理服务器然后到源服务器上请求数据后进行分发,能有效的减轻源服务器的负载。由于直播过程中用户所获取的数据流都相同,所以代理服务器只需要具备基本的缓存功能即可满足需求^[22-23]。

3.1.5 传输协议

RTSP (Real Time Streaming Protocol), 实时流传输协议定义了如何有效的通过 IP 网络传输多媒体数据,该协议处在 TCP/IP 网络的应用层。由于 HTTP 要求数据传输的正确性,所以传输层协议采用的是 TCP,这样导致 HTTP 并不是很适合有时间限制的多媒体传输,相比之下 RTSP 更适合于多媒体数据的传输,而且和 HTTP 的无状态不同,RTSP 是有状态的协议,通过维护一个 Session 来维护状态的改变,在体系结构上位于 RTP 和 RTCP 之上,它使用 TCP 或 RTP 完成数据传输。该协议是一个基于文本的协议,在 Client 和 Server 建立和协商实时流会话^[24]。

RTP 实时传输协议是在 Internet 上进行媒体数据传输的一种协议,为了实现一对一或者一对多的同步传输和提供时间信息,就需要采用 RTP 协议。由于 RTP 的典型应用建立在 UDP 传输之上,不过也可以在 TCP 等其他协议上使用这个协议。RTP 协议本身只是确保数据的实时性和完整性,并不对传输的顺序以及传输可靠性提供保障。如果是在 UDP 上,RTP 协议本身并没有提供流量控制或者拥塞控制,所以,RTCP 对传输质量进行监测的同时 RTP 实施数据的传输。RTP 为数据提供了端对端的实时传输,比如在单播或组播网络下的视频数据。^[25-26]

RTCP(RTP Control Protocol),实时传输控制协议的作用是管理传输的质量,就是在传输的同时相互交换信息,在 RTP 会话建立的时候,参与传输的双方周期性的传输 RTCP 包,这个数据包中包含了所有相关传输的信息,比如丢失

的包的数量，数据包大小等等。所以 RTCP 通常被用来对传输流量或者有效负载进行动态调整，同时与 RTP 协议配合有效的控制传输速率，所以比较适合传输实时数据。

3.2 现场直播功能设计

基于上述基本原理，该系统直播功能设计如下：

在 3.1.1 中已经讨论使用流媒体服务器作为分发中心的网络架构，因为直接向摄像头发起网络请求的话可以共同观看的用户数等于摄像头的带宽/一路直播视频所占的带宽，而如果加入了流媒体服务器的话，可以共同观看的用户数等于流媒体服务器的接入带宽/一路直播视频所占的带宽，由于流媒体服务器位于企业网络架构中的中心位置，接入带宽相对于单个网络摄像头是更加充裕的，所以带大用户量访问的情况下造成网络拥塞的情况相比之下更难发生，除此之外，流媒体服务器相比于网络摄像头对网络摄像头对于流媒体做了更多的优化，所以关于视频直播的网络架构选择以流媒体服务器为中心的网络架构是更有优势的。

符合上述流媒体服务器为中心的网络架构的数据传输方式有两种模式：摄像头推送模式和流媒体服务器拉取的模式。摄像头推送模式是指数据的传输由摄像头端主动发起，通过在流媒体服务器上建立公共发布点，将摄像头捕获的数据实时地推送到流媒体服务器上。而流媒体服务器进行拉取则是流媒体服务器主动向摄像头发起数据请求，然后将接收到的数据进行分发。这两种方式各有优点和缺点，推送模式更为灵活，并且结合程序可以完成多个摄像头往一个流媒体的发布点上推送数据，进行切换。而拉取模式的部署则更为简单，根据系统需求，在这里的数据传送模式决定采取摄像头端的推送模式。

在多媒体教室中安装有多个高清摄像头，这样在教师授课时能够从多个角度获取授课的实时场景，PC 机上的软件对多个摄像头的数据进行采集，将各个摄像头的视频流作为输入，向远处的流媒体服务器进行身份验证，当身份验证通过以后，就可以向远端的流媒体服务器进行数据推送。PC 机上的控制软件选择输入中的某一路视频流作为输出将其推送到位于中心的流媒体服务器上，在推送的同时能根据授课教师的需要实时的在这些输入流中进行切换。方便现场教学，这样能够有效的减少无用的视频流量，提高网络利用率。

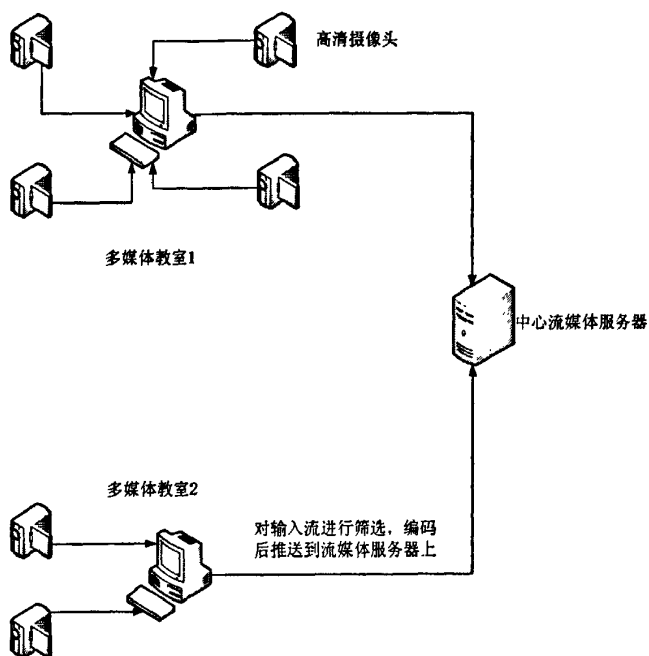


图 3-3 直播数据推送网络架构图

图 3-3 是视频直播的网络架构图，位于多媒体教室的 PC 机将多媒体教室中的高清摄像头进行管理，并从其中选取一路需要进行推送的视频数据将其推送到中心流媒体服务器上，推送过程中，位于多媒体教室中的老师可以随时根据需要在几路视频数据中进行切换，使得推送到中心流媒体服务器上的数据进行变换。

在视频推送进行的同时，直播接收用户首先利用浏览器登录到直播系统 Web 服务器进行身份认证，如果身份经过认证，则系统可以通过查询服务器状态返回当前正在直播的发布点的 IP 地址和端口号。当用户获得直播发布点的 IP 地址和端口号之后，用户向中心流媒体服务器发送数据请求，获取数据以后对数据进行解码并显示。

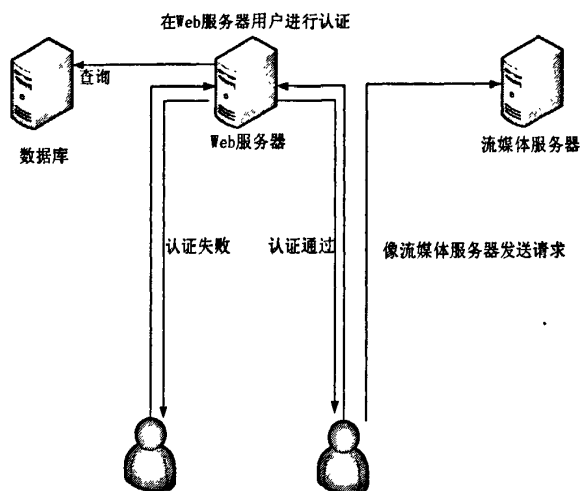


图 3-4 直播数据接收网络架构图

上述过程已经可以完成整个视频直播的功能，现在整个系统的带宽瓶颈是在前面提到的流媒体服务器上，由于流媒体服务器位于网络架构的中心，相对于以摄像头为中心的视频直播，已经大大提高了大用户量访问下的系统稳定性和可用性。但是如果当用户访问量达到流媒体服务器及其中心网络不能承受的程度，那么系统就不能正常的工作，所以有必要针对这种情况进一步提出解决的方案。

3.1.5 节中提到的流媒体代理服务器能较好的解决上述网络瓶颈问题。代理服务器能代替用户进行流媒体请求并缓存所接收的流媒体数据，而且由于视频直播所有的视频流量都是相同的，代理服务器中缓存的数据在缓存一次之后能够被所有使用该代理服务器的用户所使用，在网络架构中引入流媒体代理服务器能够较好的改善大用户量访问对系统造成的影响。流媒体代理服务器需要根据具体拓扑需要进行部署和手动配置。一般来说需要位于网络区域间的瓶颈地带。图 3-5 是流媒体代理服务器部署的网络架构图：

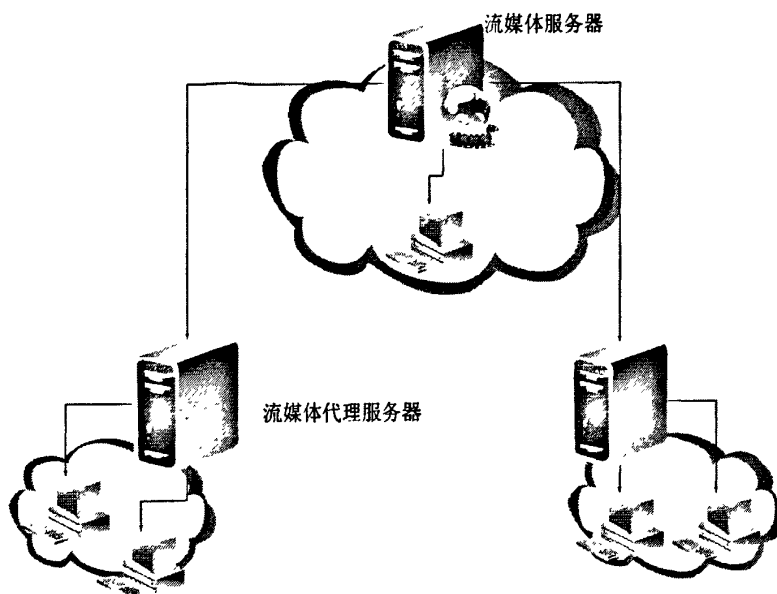


图 3-5 流媒体代理服务器网络架构

3.3 现场直播功能实现

直播采用的流媒体服务器选择了微软的 Windows Media Server，编码方式考虑到硬件支持，损耗和带宽利用率，采用了各方面参数都较折中的 VC-1 编码标准。PC 客户端的视频控制软件基于微软 .Net Framework 4.0 平台开发，底层 API 采用微软 Expression Encoder SDK 开发。

在多媒体教室多干个摄像头通过外接的方式连到教室内的 PC 机上，教师上课的时候通过 PC 端将摄像头的视频流作为输入加入到 PC 端的软件中进行实时管理。对于确定的角度较好的输入视频流可以将其激活，这样这个输入就作为输出流进行编码并推送到远端流媒体服务器上。在任何时候只能有一个输入流处于激活状态，客户端的 PC 软件可以实时查看当前的各个摄像头的情景和推送到远端流媒体的画面，PC 机的软件将输出流进行 VC-1 编码后推送到流媒体服务器，供用户观看。另外，为了使没有观看到直播的人能够在以后能对该视频进行重新观看，PC 机的软件在直播的同时会将视频保存在本地，以供以后进行观看，还可以对该视频重新进行编辑，做成多媒体课件，以提供更真实的观看体验。该部分功能在第四章将进行详细介绍。

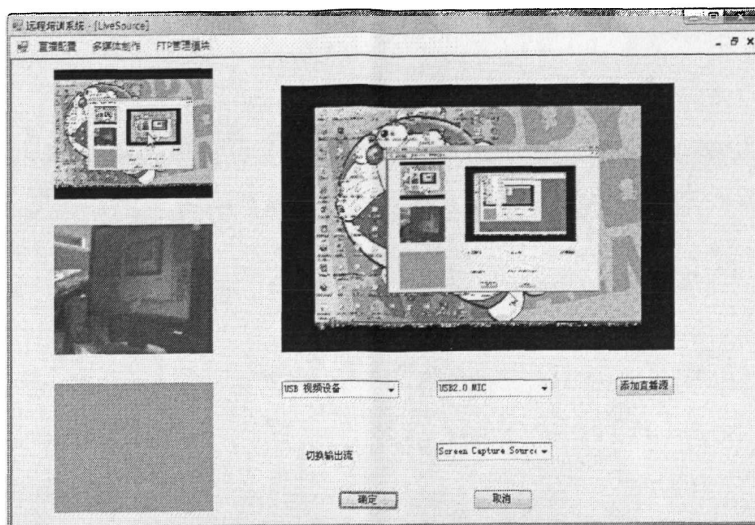


图 3-6 直播客户端程序使用界面

图 3-6 是 PC 端软件使用的实时的界面，首先在左上角直播配置中配置该次直播推送的服务器地址和用户名和密码，然后配置直播选项，左侧是已经添加到软件管理的输入流，可以通过软件实时的查看每个视频流的画面，在这里第一个输入流是电脑屏幕的实时图像，第二个输入流是连接在 PC 机上的摄像头的实时图像，添加输入流完毕后通过右下角的切换输出流的按钮选择激活的输出流，完成直播时的角度切换，在这里软件输出的是 PC 机当前屏幕。

代码示例：

//新建发布点，绑定发布点后数据进行推送

```
Uri pubURL = new Uri(Demo1.Core.Credentials.GetInstance().Connection);
PushBroadcastPublishFormat pubpoint = new PushBroadcastPublishFormat();
pubpoint.PublishingPoint = pubURL;
job.PublishFormats.Add(pubpoint);
job.StartEncoding();
```

//下面是切换输出流的相关代码

```
private void comboSources_SelectedValueChanged(object sender,
EventArgs e)
{
```

```

for (int i = 0; i < lsSources.Count; i++)
{
    if (lsSources[i].VideoDevice.Name == (sender as
ComboBox).SelectedItem)
    {
        //将输入流进行激活
        job.ActivateSource(lsSources[i]);
        break;
    }
}
}

```

访问的用户可以在网页上用内嵌的播放器进行直播观看，也可以使用机器上安装的播放器软件进行直播的观看，比如 Media Player。

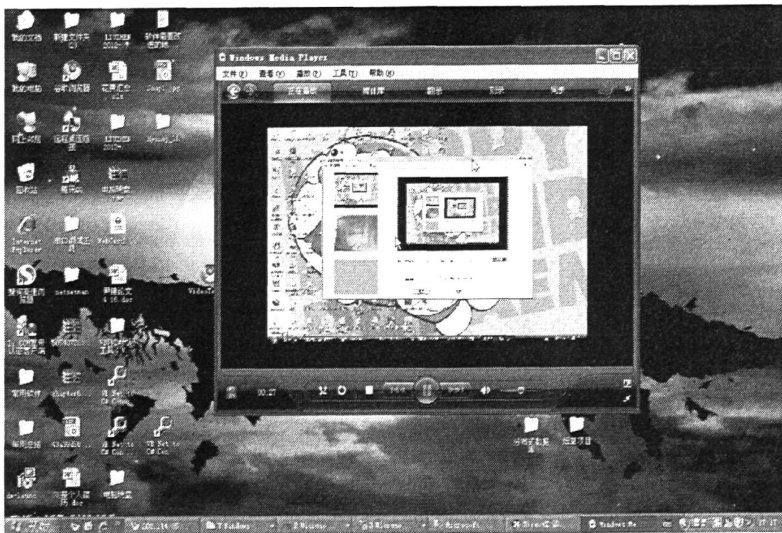


图 3-7 直播播放界面

3.4 本章小结

本章首先对远程培训系统中的视频直播模块所涉及的基本原理进行了分析，并研究了符合视频直播的流媒体网络架构，提出了以流媒体为分发中心的网络架构模型，并讨论了直播分发中推拉两种模式的优劣。具体阐述了直播模

块的整体业务功能和网路架构，并基于 .Net Framework 平台和 Expression Encoder API 对该模块的相关功能进行了实现。

第 4 章 基于时间轴模型和事件驱动机制的多媒体课件制作

4.1 基本原理

前一章节已经详细阐述了远程培训系统的直播模块的网络架构设计, 功能模块结构以及具体实现, 直播模块在培训中属于是一种同步模式的培训模式, 参与培训的员工必须同时接受培训, 而这种方式在有些需要培训时间灵活安排的培训系统中, 就不是很有实用性。

远程系统中的点播模块通过提供随时点播的功能在远程系统中提供了一种异步培训模式, 系统通过将直播的视频与视频相关课程的多媒体资源进行整合, 制作成多媒体课件, 并上传到 Web 服务器上。用户通过随时的点播来灵活地安排自己的培训进程。

4.1.1 多媒体文件的转换

在网络教学中, 传统的多媒体文件能够和流媒体结合通过 Internet 给用户提供最具体和直接的体验, 让用户更好地接收有效的信息, 在这个结合的过程中, 如何进行多媒体课件中元素的同步是决定其是否被用户所接受的关键点。

由于桌面常用的多媒体文件, 比如 PPT, Word 格式的文档并不能在常用的 Web 浏览器中得到解析与展示, 所以在制作过程中必须对其进行转换, 将其转换成能够在浏览器中识别的格式。以 PPT 文件为例, 常用的有以下几种转换方式^[27-33]:

1) PowerPoint 软件本身提供了 PPT 到 HTML 文件的转换, 可以将 PPT 直接转换成可以在 IE 浏览器中进行浏览的 HTML 文件。此种方式操作简单, 并且可用性好。缺点在于和 IE 浏览器绑定紧密, 不支持跨平台跨浏览器。

2) 将 PPT 转换为 JPG 的图片, 然后在浏览器中直接通过图片的方式查看 PPT, 此种方式支持跨浏览器, 不足在于浏览的时候不能对 PPT 中的文字进行复制等操作。

3) 将 PPT 转换为 SWF 的视频格式, 然后在浏览器中通过 Javascript 来进

行类似翻页的操作。不足之处同上。

上述几种方式都可以将类似PPT的多媒体文件转换成可以在浏览器直接观看的格式，只不过使用的技术细节有所不同。具体采用哪种转换模式可以根据具体的格式和具体的需求在系统中进行权衡与选择。

由上述的针对PPT文件转换的讨论可以得出其他类似的多媒体文件格式转换的思路。多媒体文件转换完成后被上传到 Web 服务器上，客户端浏览器通过 HTTP 发起数据请求，然后转换后的文件能直接在浏览器上显示，给予用户良好的用户体验。

4.1.2 同步机制

多媒体课件的元素同步模型采用了事件驱动模型。事件驱动程序模型（Event-driven programming）是一个已经被广泛应用在计算机编程技术中的一个建模技术。与其相对的是批次程序设计模型（Batch programming）。批次程序设计模型是指程序设计者预先设定程序执行的流程，程序执行的时候按照设计者制定的流程进行运行。与批处理程序设计模型不同，事件驱动模型的核心是事件，该模型取代了前者传统上做完一件事情再做另外一件事情的模式，事件由一个不断进行循环的事件收集器进行收集，然后由事件发送器将事件发送到目标对象中，事件处理器做最后的处理工作。事件驱动程序模型理论上可以用任何编程语言进行实现，差别仅仅在于实现的复杂度有所区别。计算机的操作系统就是事件驱动模型实现的一个典型范例，比如中断处理器对各种中断进行处理就是一个事件驱动的例子。

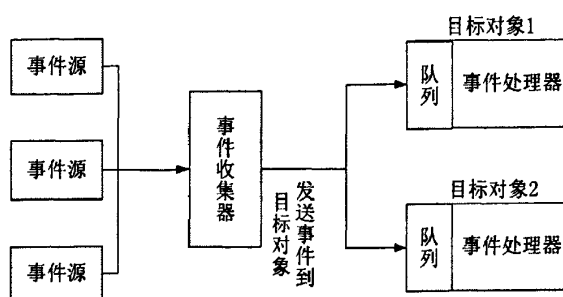


图 4-1 事件驱动模型图

图 4-1 是事件驱动模型的模型图，事件收集器是由一个事件检查的循环实现的，事件源发生事件后，事件收集器中的循环获取事件，并将其发送到对应

接收目标的事件队列中去，然后对应的事件处理器对其进行处理。

根据事件驱动的模式，可以考虑在流媒体播放的时候事件收集器进行收集，当特定的事件发生后用对应目标对象的事件处理器对该事件进行处理，然后在该事件的处理程序中，将对应的资源指向相应的地址，这样就完成了在视频播放的时候多媒体资源的实时转换，用户能够在观看视频的同时获取同步的多媒体信息，比如 PPT 转换而来的相关课件等，更好的理解相关内容。

4.1.3 时间轴模型

为了让流媒体播放的时候系统在特定的时刻触发事件，在多媒体课件制作的过程中必须在待播放的流媒体文件中加入某种特定的信息，而这些信息应该标识着在某时刻的特殊的信息，这样事件源的触发事件才能有所区别，并且最终目标对象才能根据这些信息采取有所区别的处理。这种特殊信息和其时间的关系加入的过程所遵循的模型采用的是一种基于时间轴的模型^[34]。

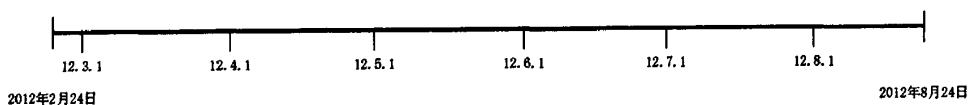


图 4-2 时间轴模型图

图 4-2 是时间轴模型的一个样例图，时间轴模型是把一段时间以一条数轴来进行表示，然后在相应的时间点上进行标注特定的信息，以区别不同时间点所对应的不同的信息。在需要对多媒体文件进行操作的地方，在轴上加入一个标记，当前时间到达上面有标记的地方的时候，就会触发一个事件，此时，事件处理器会对事件进行处理。而此时事件处理器会根据时间轴上的标记的值来进行相应的有所区别的处理。

4.2 多媒体课件制作功能设计

远程培训系统的多媒体课件制作模块主要是将多媒体文件和相关的视频文件进行整合，制作成有着更好用户体验的多媒体课件。

系统首先在 PC 端对多媒体文件资源比如 PPT 文件进行处理，将其转换成浏览器能直接识别并能显示的格式。4.1.1 中已经对常用的多媒体文件转换技术进行了阐述，现在对其中提到的三种转换技术进行比较，第一种技术使用以后

不能做到跨浏览器，而第三种技术需要大量使用 Javascript 进行程序编写，对于整体技术架构不是特别适合，故决定采用第二种方案，将 PPT 文件转换为 JPG 文件，然后上传到 Web 服务器上。

当多媒体资源上传到 Web 服务器上去之后，浏览器就能对其进行请求并显示。多媒体文件的转换完成之后，接着就是对流媒体资源进行处理，主要是针对同步问题，对其根据时间轴模型在特定的时刻加入特定的信息，加入这些信息以后对视频毛坯文件进行编码，在编码完成之后，根据时间轴加入的信息已经内嵌在视频文件中，然后将得到的文件上传到 Web 服务器上。

在用户浏览多媒体课件的过程中，事件驱动机制保证流媒体和多媒体资源的同步。客户端的应用程序内部有循环不停的探寻当前视频播放的进度是否被标记，如果发现被标记，则应用程序内部会触发相应的事件，然后程序的事件处理器部分会对该事件进行处理，处理程序根据标记的信息采取不同的具体处理措施，比如会将页面中多媒体部分的显示资源指向不同的资源地址，通过这种方式完成流媒体和多媒体资源的同步。

4.3 多媒体课件制作功能实现

浏览器端的编程平台主要有两种框架，一种是基于 Javascript 的脚本式开发的解决方案，Javascript 是由著名的 Netscape 开发的一种在浏览器端脚本语言，它是一种动态，弱类型，基于原型的语言，该语言通过 DOM 模型能够对 HTML 元素提供良好的操作。现在，主流的浏览器基本上都提供对 Javascript 的支持。另外一种是基于富客户端的平台方案，由于 Javascript 提供的功能有限，并且在比如视频，图像处理方面的功能并不是十分强大，所以在浏览器和服务器端中间需要有一个能提供强大视频，图像支持的平台。Adobe 公司的 Flash 和微软公司的 Silverlight 都属于这种富客户端开发平台。相比之下，后一种基于富客户端的平台方案在平台上更符合系统的需求，所以，前端实现的平台决定基于富客户端技术。

Flash 是 Macromedia 公司推出的前端富客户端的应用程序解决方案，该公司后来被 Adobe 公司收购。而 Silverlight 则是微软公司所推出的 Web 前端应用程序开发的解决方案，是微软 RIA（丰富型互联网应用程序）主要的开发平台之一，两者都以浏览器组件的形式，提供 Web 应用程序中多媒体与高度交互性

前端应用程序的解决方案。编程语言上，Flash 中使用的编程语言是 ActionScript，其作用是在 Flash 内容和应用程序中实现交互性，数据处理以及其他功能。它也是一种脚本语言，语法上与 Javascript 十分类似。而 Silverlight 的编程语言则有多种选择，C#和 VB.Net 都提供支持。由于系统其他编程框架是基于微软 .Net Framework 技术框架，所以在这个基础上选择 Silverlight 相对于 Flash 而言就能大大减少技术开发和维护的成本，所以在前端浏览器的技术平台上，微软公司的 Silverlight 是最优选择^[51]。

对多媒体文件进行处理采用的是微软的 Expression Encoder API，它和 Silverlight 都能采用 C#语言进行二次开发，有效的减少了技术迁移和学习的成本。同时也有着良好的性能和用户体验。同时 Expression Encoder 相对于微软的 Windows Encoder 提供了更好的性能，功能和更多的编码格式的支持。

多媒体文件的转换，PC 客户端的 Winform 程序通过调用微软 Office 组件的 API 接口，对 PPT 文档进行处理，将其转换成图片，这里采用的是 PPT 转换成图片的方案相对于另外两种方案有着更好的兼容性和实用性。功能截图如图 4-3 所示。

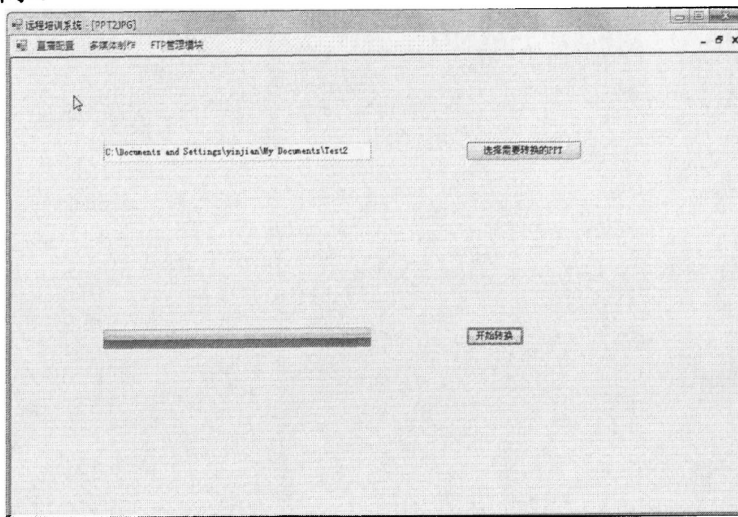


图 4-3 文件转换功能界面

在多媒体文件转换完毕之后，客户端的 Winform 程序对视频进行处理。

由于在 4.1.2 和 4.1.3 中已经讨论了同步实现的思路，在这里我们需要引入一个对象，能够在时间轴的模型上包含特定的信息。在 Silverlight 中 Marker 对象能够满足这个需求。Marker 对象有着两个属性，Time 和 Value,分别对应着时间轴模型上的时间和该时间的信息。可以想象成时间轴上的一个标记就是一

个 Marker 对象。当用户对流媒体文件进行处理的时候，只需要在视频文件中加入类似这样的信息，然后对视频文件进行相应的编码即可。下面是添加 Marker 的代码示例：

```

//添加标记
Marker mrk = new Marker();
//是否生成 KeyFrame
mrk.GenerateKeyFrame = false;
//是否生成缩略图
mrk.GenerateThumbnail = false;
mrk.Time
=
TimeSpan.FromSeconds(this.axWindowsMediaPlayer1.Ctlcontrols.currentPosition);
mrk.Value = this.txtMarker.Text;
mItem.Markers.Add(mrk);

```

下面是客户端 Winform 程序的界面和核心代码示例：

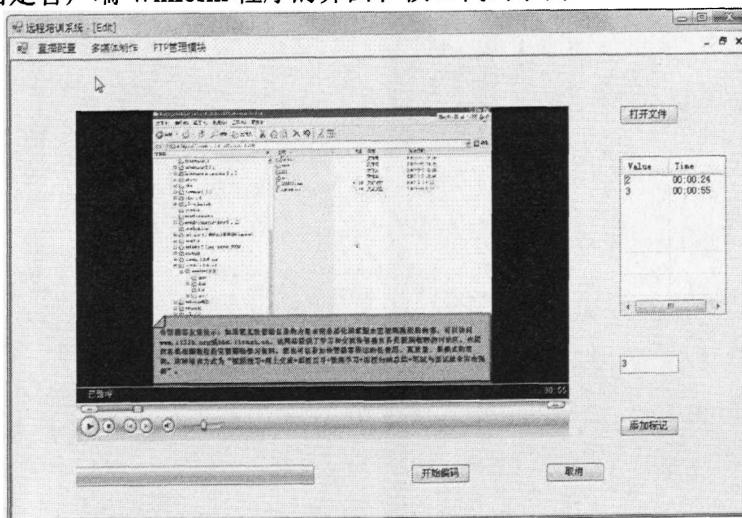


图 4-4 视频编辑界面

代码示例

```

job.MediaItems.Add(mItem);
//确定输出文件夹
job.OutputDirectory = outputPath;
//绑定 Progress 事件
job.EncodeProgress
+=
new

```

```

EventHandler<EncodeProgressEventArgs>(OnProgress);
    //绑定完成事件
    job.EncodeCompleted += new
EventHandler<EncodeCompletedEventArgs>(job_EncodeCompleted);
    job.Encode();
    
```

在这些工作完成之后通过 Web 后台将转换生成的多媒体文件和经过处理的流媒体文件上传到服务器。

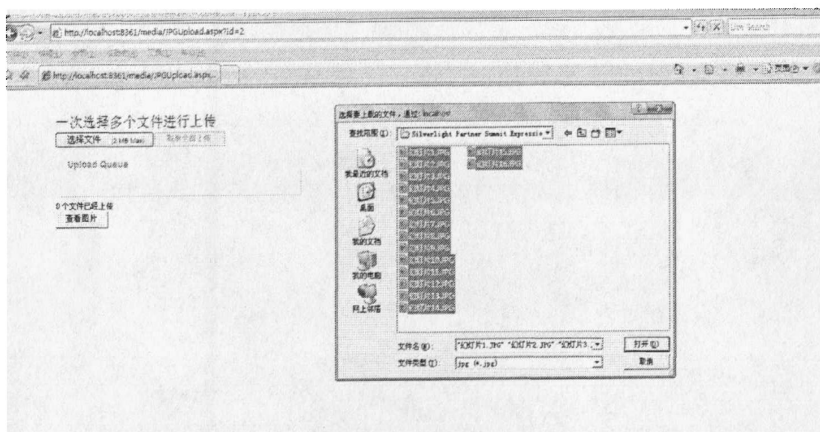


图 4-5 Web 后台上传界面

浏览器浏览时，前端的 Silverlight 应用程序负责流媒体视频的播放，当播放时间到达某个 Marker 标记的时间的时刻，就会触发相应的事件，事件处理器将根据 Marker 的 Value 值来进行处理，在这里 Silverlight 将多媒体资源的路径指向相应的 URL 完成图片资源的获取，在视频播放的过程中将在不同的 Marker 到达的时刻，指向不同的图片，在用户看来就像是在 PPT 和视频两者之间实现了同步。图 6-4 是实际实现的效果图。左侧是负责视频播放的视频播放器，用户在这里对视频进行观看，右侧显示对应的多媒体文件，在这里是对视频相关的 PPT 文件进行实时显示和切换。

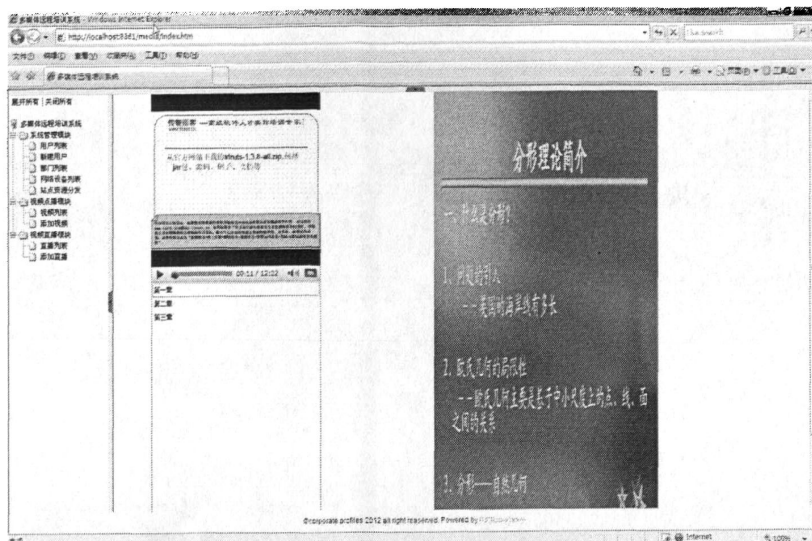


图 4-6 多媒体课件播放界面

下面是 Silverlight 程序中的事件处理器的部分代码，该代码将右侧图片控件的路径进行了重定位。

```
private void me_MarkerReached(object sender,
TimelineMarkerRoutedEventArgs e)
{
    this.imgPPT.Source = new BitmapImage(new
Uri(e.Marker.Text.ToLower(), UriKind.RelativeOrAbsolute));
}
```

4.4 本章小结

本章对远程培训系统中的多媒体课件制作的整体流程进行了分析，并研究了多媒体文件转换的多种思路，多媒体课件元素之间的同步方式，引入了时间轴模型来对时间标记进行存储和读取。在技术实现上，虽然基于不同的技术平台，但是在这些平台上使用了同样的编程语言，大大降低了不同技术的学习和实现带来的技术成本。

第 5 章 基于 CDN 的视频随机点播调度

5.1 基本原理

除了在 3.1.5 中提到的缓存代理服务器以外,在视频点播模块中流媒体调度是缓解带宽瓶颈的另一个重要途径,由于基于流媒体的应用日趋广泛,流媒体调度近几年来也一直是研究的热点问题^[35]。它通过软件实现的特殊算法策略,解决了点播过程中流媒体中心服务器所处中心骨干网的的带宽瓶颈问题,提高了整个系统的工作效率。本章在总结归纳现有流媒体调度解决方案的基础上,提出了基于 CDN 分发策略的视频文件调度方案。该方案吸取了现在流行的 CDN 内容分发网络的核心思想,考虑了实际拓扑结构和流媒体应用的特殊性,在骨干网带宽有限的情况下,减少了整体的服务延迟,提高了整体点播的服务质量和用户体验。

对流媒体调度方案的研究首先从研究现有的调度方案开始,然后引入 CDN 的概念并对其核心思想进行提炼和分析,最后,提出基于 CDN 的流媒体调度方案,该方案包含了静态调度方案和动态调度方案的双重特性。

5.1.1 流媒体调度方案的研究现状

流媒体调度方案按其特性分为两类:静态调度方案和动态调度方案。静态调度方案主要采用中心服务器推送模式,是指中心服务器不考虑用户的请求行为而按已经决定的某种策略对视频流进行广播,对用户提供固定的请求延迟;动态调度算法采用的是客户端拉取的模式,用户首先对流媒体进行请求,中心流媒体服务器根据调度策略响应用户的请求。从上述所描述的特点来看,静态调度算法适用于多数用户感兴趣的视频节目,而动态调度适用于单个用户的请求。下面从静态调度和动态调度方案两个方面对现有的流媒体调度方案进行研究和归纳。

典型的动态调度算法包括 FCFS (first come first serve)算法, Batching 算法, Stream Tapping 算法, SMP 算法。在这些算法中 FCFS 是相对简单的算法。FCFS 按照用户请求的先后顺序来进行视频流的调度,由于没有对调度本身进行优化,

该算法对于带宽本身的瓶颈问题并没有很好的解决，在用户访问量较大的情况下性能表现不是很好。Batching 算法则是将不同的用户的请求绑定在了一个组播流中，以增加用户等待时间的代价来提高系统资源的利用率。为了解决 Batching 算法中等待延迟的问题，Hua 提出了 Patching 算法，该算法允许不同时间请求同一个视频流的用户共享视频流，Patching 算法的性能明显优于其他动态调度算法^[37-40]。

动态调度算法的优点在于：

- 1) 由于系统能实时响应，启动延迟较小。
- 2) 支持 VCR

缺点在于算法本身结构较复杂。

典型的静态调度算法包括周期广播算法，快速广播算法，金字塔算法，摩天大楼算法等。静态调度算法一般利用广播或者组播从服务器端发送视频流。周期广播算法每隔一个周期循环播放视频流，金字塔算法将视频流分成长度递增的片段，然后利用组播发送不同的片段。由于片段长度逐渐递增，流媒体开始的片段长度较小，用户的启动延迟相对较小。摩天大楼算法也是将视频流分成若干片段后进行组播，但是该算法对于片段的长度是按特定数列的比例进行分割^[37]。

静态调度算法的优点在于：

- 1) 策略相对来说简单，并且不受用户多少的影响
- 2) 比较公平，对于不同的用户服务性能相同

缺点在于：

- 1) 启动延时过长
- 2) 不支持 VCR

5.1.2 CDN 内容分发网络

CDN(内容分发网络)是近几年发展起来的一种新型的网络架构，从狭义的角度来说，它是为能在传统的 IP 网络上发布内容而特别优化的网络覆盖层，从广义的角度来说，CDN 是一种基于质量和秩序的网络服务模式的代表。CDN 是一个具有策略性的整体网络系统，它主要包括负载均衡，网络请求重定向，内容管理，和分布式存储这四个组件。

CDN 典型的网络架构主要分为两部分，中心和边缘，中心是指 CDN 网管

中心和 DNS 解析中心，负责全局的负载均衡，边缘主要是指异地的各个 CDN 节点，主要由 Cache 和负载均衡器组成。

图 5-1 展示的是使用传统方式，没有使用 CDN 的 Internet 上的请求过程



图 5-1 传统 Web 请求流程图

- 1) 用户在浏览器中输入要访问的域名。
- 2) 浏览器向计算机中配置的 DNS 服务器发送 DNS 请求，以此得到域名对应的 IP 地址。
- 3) 浏览器根据获得的 IP 地址像服务器发送数据请求。
- 4) 服务器根据请求返回数据，浏览器根据应答显示内容。

CDN 的本质就是在用户和 Web 服务器中间加入一个 Cache 层，将用户请求引导到若干个 Cache 服务器上获得本来存储在源服务器上的数据，请求引导主要是通过 DNS 重定向实现的。

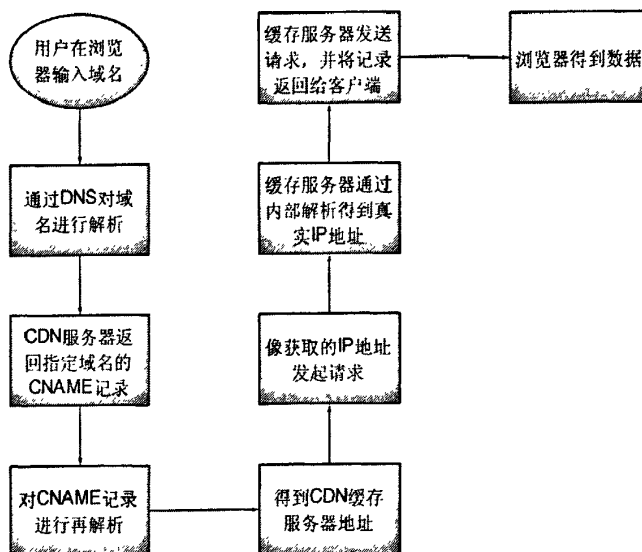


图 5-2 加入 CDN 后的 Web 请求流程图

- 1) 用户在浏览器中输入要访问的域名。
- 2) 浏览器向计算机中配置的 DNS 服务器发送 DNS 请求，由于 CDN 进行了 DNS 重定向，所以用户获得的是地理上靠近用户的缓存服务器的 IP 地址
- 3) 用户向缓存服务器发送请求。
- 4) 缓存服务器根据请求的域名通过内部专用 DNS 解析获得服务器真实的 IP 地

址，然后由缓存服务器向实际 IP 地址提交请求。

5) 缓存服务器获得内容以后一方面将数据返回给用户，一方面将能缓冲的内容进行缓存，这里的过程是一种被动分发的过程，有的 CDN 在事先就将内容分发到边缘服务器上，属于主动分发。

6) 浏览器根据应答显示内容。

缓存服务器在 CDN 下能够处理大部分的静态请求，而动态数据则交由 WWW 服务器来处理，这种方式大大减少了请求时间，并降低了整个骨干网和中心服务器的负载。

CDN 中有着几个关键的技术，现在针对 CDN 中的关键技术来进行阐述，其中的核心思想将被移植到本系统中^[41-45]。

1 服务器负载均衡

CDN 负载均衡系统实现了 CDN 内容路由功能，其核心作用是将用户数据请求导向 CDN 网络中的最佳节点。而最佳节点的选择可以根据多种策略，比如距离最近，节点负载最轻等等。负载均衡系统是整个 CDN 的核心模块之一，其准确性和效率决定了整个 CDN 的效率和性能。

负载均衡通常分为两层：全局负载均衡(GSLB)和服务器负载均衡(SLB)，全局负载均衡的目的是在整个网络范围内将用户的数据请求重定向到最近的节点。所以，就近判断是全局负载均衡的主要功能。服务器负载均衡则局限于一定范围内，其目标是在特定区域范围内寻找一台最合适的节点来提供服务。

负载均衡可以通过多种方法来实现，常用的方法包括 DNS、应用层重定向、传输层重定向等。

为了进行就近判断一般可以采用两种方式，一种是基于静态的配置，服务器根据静态 IP 地址配置表将 IP 地址映射到对应的 CDN 节点上，另一种是基于动态的监测，通过实时探测 CDN 节点到目标 IP 的网络距离，然后基于探测结果进行负载均衡。

2 内容分发

网站的响应速度取决于许多因素，多数情况下，网站整体响应速度和数据请求者与服务器的距离有着密切的联系。内容分发和复制技术就是在请求中占据一定比例的静态网页，图像分发复制到各地的加速节点上。所以内容分发与复制技术是 CDN 所需的另一个主要技术。

内容分发与复制主要分为两种模式：

推模式是一种主动分发技术，由内容管理系统发起，将内容从中心服务器分发到各个边缘服务器，分发的协议采用 HTTP 或者 FTP。通过推模式分发的内容一般都是比较热点的内容，这些内容通过推送分发到边缘 Cache 服务器，可以实现有针对性内容的提供。对于推送分发需要考虑的问题是分发策略，即什么时候分发哪些内容。一般来说，内容是由内容提供商人工管理的，当然也可以通过程序实现智能分发。

拉模式是一种被动分发技术，一般是由用户请求来驱动的。当用户请求边缘缓存服务器时，如果资源不存在，则该服务器从中心服务器上拉取数据。

一般来说在实际的 CDN 实现中，一般都会支持两种分发方式，推模式比较适合内容访问比较集中的情况，而拉模式比较适合内容访问比较分散的情况，主要是根据内容和业务模式采取相应的分发模型。

3 内容存储技术

内容如何在中心服务器上 and 边缘缓存服务器上存储是另一个需要关注的地方。对于中心服务器来说由于内容规模一般比较大，而且吞吐量较大，所以采用 NAS 等海量存储架构。而边缘缓存服务器上的存储则需要考虑到功能和性能等多方面的因素。在功能上需要对各种内容格式的支持以及部分缓存的支持，性能上则包括支持的容量、吞吐率、可靠性以及稳定性。部分缓存能力是指流媒体内容可以以不完整的内容存储和读取。由于用户很多时候并不会完整地收看整个流媒体节目，所以，部分缓存能力能够极大的提高存储空间的利用率，并有效地提高用户请求的响应时间，不过容易出现碎片，需要进行良好的设计。目前多数存储系统都采用了 RAID 技术来保证可靠存储。

4 内容管理

内容管理是指内容进入边缘缓存服务器后的管理。内容管理的主要目标是提高内容服务的效率，提高边缘服务器的存储利用率。通过内容管理，可以在 CDN 节点上实现基于内容感知的调度，通过调度，能够避免将用户重定向到没有内容的边缘缓存服务器上，从而提高负载均衡的效率。通过内容管理还能有效的实现在 CDN 节点内容的存储共享，提高存储空间的整体利用率。

在实现上，内容管理主要包括以下几个方面。

一是内容索引，内容管理首先依赖于对内容的了解。内容索引是实现基于内容感知的调度的关键。

二是内容拷贝。为了提高存储效率，一个内容在一个 CDN 节点中只会保

存一份，如果内容不在该 CDN 节点上，就需要实现内容拷贝。

三是内容访问状态信息收集。

5.2 基于 CDN 的随机点播调度策略

由上述 CDN 的关键技术我们可以得知，服务器负载均衡和内容分发与复制是实现 CDN 的关键，而这些技术的核心思想也能够通过改良应用在流媒体调度中^[46-49]，下面提出一种适用于典型企业网络架构的基于 CDN 的视频文件调度策略。

很多企业和政府机关出于自身系统安全性的考虑并没有将内部系统的访问接口开放给 Internet，在内部系统中并没有传统 Internet 上通过 DNS 把域名解析到 IP 地址类似的机制，访问系统普遍是直接通过 IP 地址加端口的方式直接访问，在这种网路架构下，CDN 中的基于 DNS 重定向的内容路由技术就不是很适用，另外 CDN 租用的价格对于一些企业来说超出了其承受范围，但是其骨干网的带宽瓶颈限制了流媒体点播系统的整体性能，所以有必要针对这种实际网络架构，在吸取 CDN 核心思想的基础上提出一种适合该网络架构的视频文件调度策略。

5.2.1 网络架构

在整体的网络架构上该调度算法借鉴了 CDN 网络的架构，在靠近用户的边缘网络上放置了用于存储分发流媒体的网络硬盘，用网络硬盘来替代 Internet 上的 CDN 节点，典型的网络架构如图 5-3 所示。

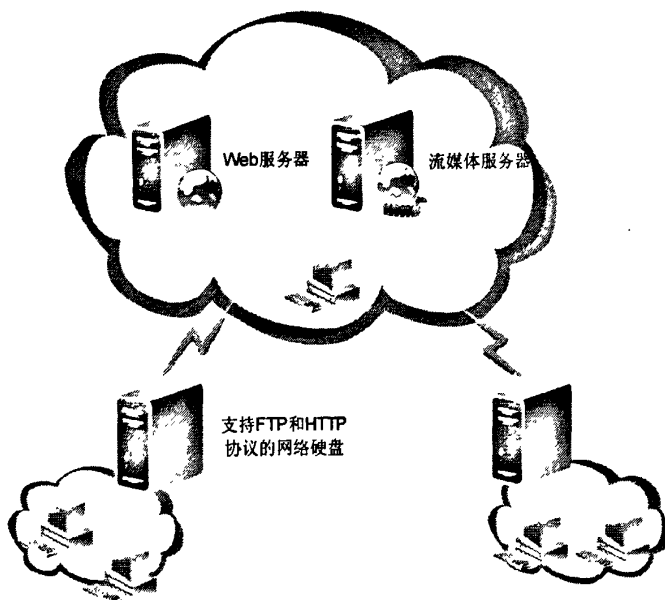


图 5-3 调度策略适用的网络架构图

图 5-3 是该调度算法实现的典型的网络架构，中心区域部署了 Web 服务器和中心流媒体服务器，在靠近用户的边缘部署了网络硬盘，用户到网络硬盘之间的网络由于地理上的优势网络访问的延迟相当小。

5.2.2 重定向

重定向是保证系统负载均衡的核心技术，在这里我们决定采用上述讨论的静态配置的方式，将用户的 IP 地址和其所访问的网络硬盘的 IP 地址进行映射，使用静态配置不用在网络硬盘上进行实时的监测，减少了系统整体的复杂性。当用户对流媒体发起请求的时候，用户的请求被重定向到靠近用户的网络硬盘上。

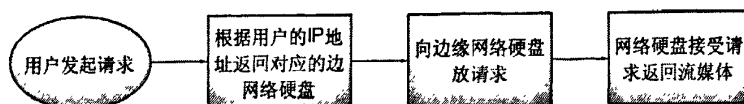


图 5-4 重定向流程图

5.2.3 内容分发与复制策略

5.1.2 中提到 CDN 中的内容与分发有两种模式，推和拉模式，由于拉模式是一种被动分发策略，在这里并不是很适合实际需求，所以在这里采用了推模

式，中心服务器负责根据一定的策略将数据分发复制到网络硬盘上。

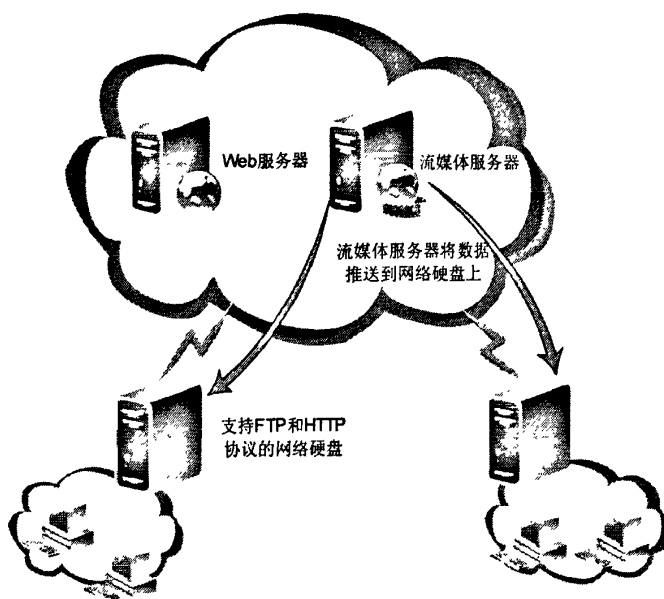


图 5-5 数据分发与复制网络示意图

下面描述分发与复制的具体策略：中心服务器在每经过一定时间的间隔后就查询数据库获取所有的需要分发的边缘网络硬盘的 IP 地址，然后循环分发在其分发文件夹下的所有文件。在中心服务器推送数据到边缘网络硬盘的过程中，由于分发的文件普遍较多，所以需要考虑如何才能减少分发占用的网络流量。现在假设中心服务器上存在一段名字为 001.wmv 的视频文件，边缘网络硬盘现在也存在一个名字为 001.wmv 的视频文件。这两个同名文件有可能内容一样，也可能稍有差别。比如中心服务器上的 001.wmv 文件后来由于业务需求重新用第四章介绍的多媒体课件制作技术重新进行了制作。

策略 1：中心服务器将所有的文件推送分发到边缘网络硬盘上，如果没有重名文件，则直接进行分发与复制，对于重名的文件，不管是不是已经经过修改，都进行覆盖。这种方式极大的浪费了宝贵的骨干网的网络带宽。而且随着系统的使用，文件量的数目越来越大，必然会造成系统的网络拥塞，并不是一个好的策略。

策略 2：中心服务器将其上的文件进行一个签名，这个签名保证内容稍微改变生成的签名就不同。将生成的这个签名和边缘网络硬盘上的同名文件的签名进行对比，如果文件的签名不同则代表文件内容发生了改变，中心服务器对其进行覆盖，如果签名相同则证明是相同的文件则不进行任何操作。

在策略 2 中的文件签名的比对是分发与复制策略的核心，如何准确快捷的将两个文件进行对比，在这里，该系统使用了密码学中 Hash 技术。

5.2.4 哈希算法

Hash 算法又叫做散列算法，其功能是把任意长度的输入通过散列函数，转换成一个固定长度的输出，最后的输出就叫做散列值。Hash 转换是一种压缩转换，它保证了一个基本特性，如果两个散列值是不同的，那么这两个散列值的原始输入也不同，一个具有混淆特性的 Hash 函数会将两个相近的输入产生两个完全不同的 Hash 值。Hash 算法已经广泛的应用到了数字签名，文件校验等方面^[50]。

在这里将流媒体文件看成一个二进制的数据流输入到散列函数中，散列函数根据其算法就能得出该流媒体文件的 Hash 值。通过对流媒体文件 Hash 值的比较，我们来判断两个文件是不是相同。然后根据文件是否相同决定是否将流媒体文件进行分发。

5.3 点播调度功能设计

5.3.1 重定向

随机点播调度模块首先需要根据整体网络拓扑结构将用户 IP 地址和对应的边缘网络硬盘的 IP 地址之间的映射关系录入到 Web 服务器端的数据库中，为了减少数据库中的数据量和提高实际系统运行的效率，该录入机制需要考虑到 IP 网段的操作便利性。比如 192.168.1.1 到 192.168.1.24 这个 IP 段在数据库中就应该只有一条记录，而不是一个 IP 对应一条记录，那样操作会造成数据量过大，影响系统实际运行的性能。由于常用的点分十进制 IP 地址的表示方式在数据库中并不是很适合实际操作，综合上述因素，为了在实际运行的时候便于数据比较等操作，需要将 IP 地址的表达方式进行转换，转换成便于比较和操作的类型，转换原理如下。

IP 地址是长度为 32 位的二进制流，如果直接根据二进制到十进制的转换方法，将其看成一个正整数的话，就能按照十进制的计算方式比较其大小。这样在数据库中直接存储 IP 地址就是其二进制转换到十进制后的值，由转换原理

我们可以知道每个 IP 地址转换后的值是不同的。

再将客户端的 IP 地址转换成这样的十进制数字并存入到数据库，数据库 Sql 语言中直接就提供了对十进制的比较的操作符，所以我们能很方便的通过 Sql 查询语句获取该客户端所需要请求的距离最近的网络硬盘的 IP 地址。并且网段的表示也直接转换为两个十进制数的范围。

比如 192.168.1.200 可以转换为 3232235976，192.168.4.64 可以转换为 3232236608，依次类推，如果要表示网络段的话大小只要位于 3232235976 和 3232236608 这两个数字之间就满足条件了。

当用户进入点播页面的时候，用户首先向 Web 服务器发送一个 Web 动态页面的请求，Web 服务器端的 Web 服务程序通过请求来的报文获取用户的 IP 地址，然后按照上述录入到数据库的信息查询出用户所需要访问的距离最近的网络硬盘的 IP 地址，如果没有找到相应的配置信息则默认返回中央流媒体服务器的 IP 地址，Web 应用程序将查找到的 IP 地址写到返回给用户的服务器应答报文中去，用户的浏览器从获得的应答报文中得到网络硬盘的 IP 地址，然后访问网络硬盘中的流媒体文件。系统通过这种方式完成了从中心流媒体服务器到边缘网络硬盘的重定向功能。由于用户本身并不清楚 Web 页面生成的细节，就像 CDN 中的 DNS 解析过程一样，用户只是发送了一个 Web 页面的请求，然后根据请求流的数据向相应的流媒体网络硬盘发送流媒体请求。中间的重定向功能对用户来说是完全透明的，一切过程都发生在服务器端。

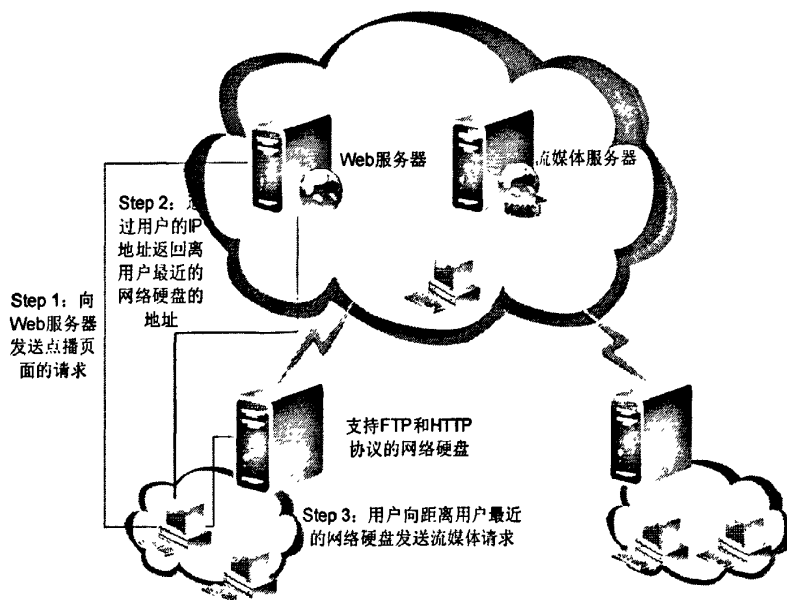


图 5-6 流媒体点播调度网络示意图

5.3.2 视频文件分发与管理

视频文件复制与分发的策略采取的是中心服务器推送的模型，中心服务器上运行着一个应用程序，该程序定时对中心服务器分发文件夹下的所有文件进行遍历操作，对于查找到的文件对其进行分发。服务器上的文件和网络硬盘上的文件的内容比较基于 5.2.3 中阐述的策略。

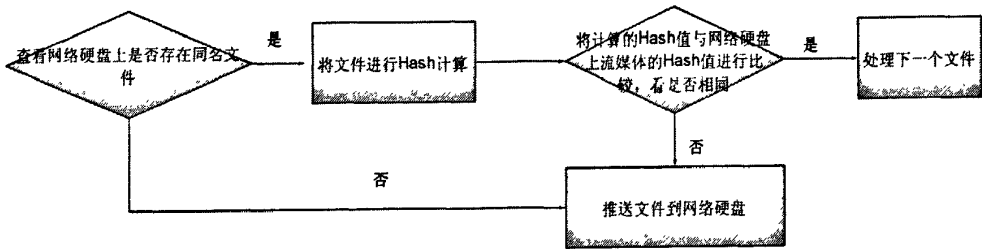


图 5-7 文件比较流程图

这样可以节省网络带宽，同时保证系统更稳定的运行。对于每次实际发生的文件推送过程，服务器上的应用程序的日志模块在数据库中都会记录日志，方便系统管理模块进行查看。

为了方便系统管理员对分发到网络硬盘的视频文件进行管理和特殊情况下的手动分发，中心服务器上的应用程序通过底层网络通信协议对边缘网络硬盘上的资源进行上传，查看，删除等管理操作。

5.3 点播调度功能实现

随机点播调度模块的技术框架采用的是微软的 .Net Framework 4.0 平台，编程语言采用的是 C#，开发环境是 Visual Studio 2010 Ultimate，Web 服务器端的数据库采用的是 Sql server 2005，边缘网络硬盘采用的是支持远程 FTP 和 HTTP 服务的网络硬盘。

数据库主要需要存储用户 IP 和网络硬盘 IP 的映射关系，表设计如下。

表名: IPAddress

序号	列名	数据类型	长度	小数位	标识	主键	允许空	默认值	说明
----	----	------	----	-----	----	----	-----	-----	----

1	ID	uniqueidentifier	16	0		是	否		
2	StartIP	bigint	8	0			否		
3	EndIP	bigint	8	0			否		
4	ServerIP	bigint	8	0			否		

StartIP 和 EndIP 用来存储 IP 网段的两端的 IP 地址，这样能够降低每个 IP 都需要手动录入的工作量。ServerIP 表示该 IP 网段对应的边缘网络硬盘的 IP 地址。

传统的 IP 地址的表达方式是点分十进制，调度模块在用户录入的过程中将其进行转换，转换成便于系统操作的十进制数字，数据库中存储的就是转换后的十进制数字的值，由于 Sql server 2005 的 int 类型的范围有限，故使用了 bigint 型来保存十进制数字。下面对转换的过程进行分析：

IP 地址 127.0.0.1 的二进制表达是 1000000 00000000 00000000 00000000 是一个 32 位的二进制流，如果我们将这个 32 位的二进制流看成一个十进制的数，通过二进制到十进制的转换方法就能得出该十进制的值，我们可以得出该十进制数是 2130706433，并且每个不同的 IP 地址所代表的十进制的数的值一定是唯一的，确认这一点十分重要，因为二进制到十进制值的一一对应性保证了转换过程的唯一性。

在转换的基础上通过十进制的大小判断就能实现 IP 的网段判定的功能。

下面给出在 .Net 中将 IP 地址转换成 long 型的代码

```
public static long IpToLong(IPAddress ip)
{
    int x = 3;
    long o = 0;
    foreach (byte f in ip.GetAddressBytes())
    {
        o += (long)f << 8 * x--;
    }
    return o;
}
```

上面的代码中 ip.GetAddressBytes() 直接获取 IP 地址的二进制表达式，然后第一组 byte 通过位运算朝左移动 24 位，第二组朝左移动 16 位，依次类推，将

位移运算后的值全部相加，最后获得的值就是二进制转十进制的值。

在录入用户 IP 网段和边缘网络硬盘 IP 地址的映射关系以后，当用户对页面发起请求时调度模块从数据库中通过 Sql 语句查询对应的边缘网络硬盘的 IP 地址。

范例 Sql 查询语句如下：SELECT [ServerIP] FROM [IPAddress] WHERE StartIP <= 2130706433 AND EndIP >= 2130706433，如果没查询到结果则默认返回中心流媒体服务器的 IP 地址，然后将得到的 IP 地址写入到服务器响应报文返回给客户端浏览器，浏览器收到响应报文以后通过页面中的视频文件的 URL 路径对网络硬盘发起数据请求。从用户请求点播页面到用户最终获得流媒体文件的数据，中间的重定向过程对用户来说完全透明，但是系统能将用户的请求重定向到不同的网络硬盘中去，有效的提高了系统的效率和网络带宽利用率。

视频文件的的分发与复制，采用的是中心服务器推送的模式。常用的能够在网络上传输数据的常用的两种协议是 HTTP 和 FTP。同 HTTP 协议相比，FTP 提供了更加丰富的目录文件管理功能，以及在文件传输时更好的性能。所以推送的底层通信协议采用 FTP 协议。

中心服务器定时对分发目录下的所有视频文件进行遍历。分发的时候，首先对每一个文件进行 Hash 计算，采用的 Hash 算法是 MD5 算法。MD5 全称 Message-Digest Algorithm 5，是由 Rivest 在 1991 年开发出来的一种已经广泛应用在数字签名中的 Hash 算法。其计算过程简要可以概述为以 512 位的分组来处理输入的信息，且每一个 分组被划分成为 16 个 32 位子分组，然后在经过一系列处理后，算法的输出 4 个 32 位分组组成，最后将这 4 个 32 位分组合级联后生成一个 128 位的散列值。在边缘网络硬盘上的每个媒体文件都有一个同名的并以 txt 结尾的文件，该文件中存储着该文件名的视频文件的 MD5 值。比如 2012.4.1.avi 这个视频文件对应的 txt 文件就是 2012.4.1.avi.txt，该文件存储着 2012.4.1.avi 这个文件的 MD5 值。中心服务器会将计算得出的 MD5 值同边缘网络硬盘上同名文件对应的 MD5 值进行比较，如果不同则将文件进行覆盖并覆盖对应的 txt 文件，确保每个流媒体文件和 txt 的文件是一一对应的，如果两个值相同，则不需要进行任何操作。

内容分发与复制由于耗时普遍比较长，所以程序在中心服务器的后台运行，程序运行过程中会将分发的记录写入数据库中作为日志，Web 管理模块可以对日志进行查看，了解分发与复制的实时运行状态。

下面是部分示例代码：

```

DirectoryInfo rootFolder = new DirectoryInfo(localDirectory);
//先遍历当前目录，在遍历文件夹的每个目录
Demo1.Core.FTPHelper obj = new FTPHelper(remoteFtpUrl,
this.user, this.password);
foreach (FileInfo fi in rootFolder.GetFiles())
{
    try
    {
        obj.CheckAndUpload(fi.DirectoryName, fi.Name);
    }
    catch (Exception e)
    {
        log.Debug("CheckAndUpload      Error"      +
fi.DirectoryName + fi.Name + "--" + this ftpUrl + "--" + this.user + "--" +
this.password);
    }
}
obj.Close();
DirectoryInfo[] subFolders = rootFolder.GetDirectories();
//遍历文件夹
foreach (DirectoryInfo subFolder in subFolders)
{
    try
    {
        obj.MakeDirectory(subFolder.Name);
    }
    catch (Exception e)
    {
    }
}

```

```

        this.Distribute(subFolder.FullName, remoteFtpUrl +
subFolder.Name + "/", this.user, this.password);
    }

```

图 5-8 是分发日志的数据库记录的截图

Date	Thread	Level	Logger	User	Message
2012-03-18 15:32:20.843	11	DEBUG	Demo1.Core.Distribution	(null)	CheckAndUpload Done G:\DistributeTest\YJ 2012-1-10 18.50.51001.wmv-ftp://202.114.85.1...
2012-03-18 15:32:20.860	11	DEBUG	Demo1.Core.Distribution	(null)	CheckAndUpload Done G:\DistributeTest\YJ 2012-1-10 18.50.51Default.html-ftp://202.114.8...
2012-03-18 15:32:20.877	11	DEBUG	Demo1.Core.Distribution	(null)	CheckAndUpload Done G:\DistributeTest\YJ 2012-1-10 18.50.51JobFile.xsj-ftp://292.114.85...
2012-03-18 15:32:20.877	11	DEBUG	Demo1.Core.Distribution	(null)	CheckAndUpload Done G:\DistributeTest\YJ 2012-1-10 18.50.51MediaPlayerTemplate.xap-ftp...
2012-03-18 15:32:20.890	11	DEBUG	Demo1.Core.Distribution	(null)	CheckAndUpload Done G:\DistributeTest\YJ 2012-1-10 18.50.51Preview.png-ftp://202.114.8...
2012-03-18 15:32:21.233	11	DEBUG	Demo1.Core.Distribution	(null)	CheckAndUpload Done G:\DistributeTest\YJ 2012-1-10 18.50.51Settings.dat-ftp://202.114.8...
2012-03-18 15:32:21.250	11	DEBUG	Demo1.Core.Distribution	(null)	CheckAndUpload Done G:\DistributeTest\YJ 2012-1-10 18.50.51SmoothStreaming.xap-ftp://2...
2012-03-18 15:32:21.267	11	DEBUG	Demo1.Core.Distribution	(null)	CheckAndUpload Done G:\DistributeTest\YJ 2012-1-10 18.50.51Thumb.db-ftp://202.114.85...
2012-03-18 15:32:21.280	11	DEBUG	Demo1.Core.Distribution	(null)	CheckAndUpload Done G:\DistributeTest\YJ 2012-1-10 18.50.51TextLibrary.xap-ftp://2...
2012-03-18 15:32:21.953	11	DEBUG	Demo1.Core.Distribution	(null)	CheckAndUpload Done G:\DistributeTest\YJ 2012-1-13 11.18.17001.wmv-ftp://202.114.85.1...
2012-03-18 15:32:21.953	11	DEBUG	Demo1.Core.Distribution	(null)	CheckAndUpload Done G:\DistributeTest\YJ 2012-1-13 11.18.17Default.html-ftp://202.114.8...
2012-03-18 15:32:21.970	11	DEBUG	Demo1.Core.Distribution	(null)	CheckAndUpload Done G:\DistributeTest\YJ 2012-1-13 11.18.17MediaPlayerTemplate.xap-ftp...
2012-03-18 15:32:21.983	11	DEBUG	Demo1.Core.Distribution	(null)	CheckAndUpload Done G:\DistributeTest\YJ 2012-1-13 11.18.17Preview.png-ftp://202.114.8...
2012-03-18 15:32:21.983	11	DEBUG	Demo1.Core.Distribution	(null)	CheckAndUpload Done G:\DistributeTest\YJ 2012-1-13 11.18.17Settings.dat-ftp://202.114.8...
2012-03-18 15:32:22.000	11	DEBUG	Demo1.Core.Distribution	(null)	CheckAndUpload Done G:\DistributeTest\YJ 2012-1-13 11.18.17SmoothStreaming.xap-ftp://2...

图 5-8 数据库日志

当分发完成以后，需要对网络硬盘上的视频文件进行管理，中心服务器上运行的 Winform 程序能够通过 FTP 网络协议对网络硬盘上的文件进行下载，删除，手动分发等操作。

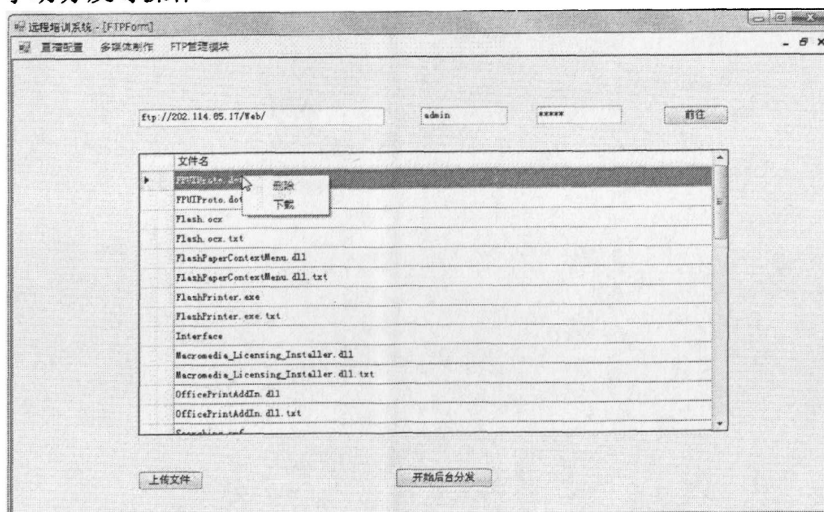


图 5-9 手动管理功能界面

图 5-9 是对网络硬盘进行内容管理的 Winform 管理程序的截图，再将网络硬盘的地址以及用户名和密码写入以后就能对网络硬盘相应文件夹下的资源进行手动管理，并能对文件进行下载和删除操作。同时为了满足某些特殊需要，也能够直接通过管理程序直接向网络硬盘进行文件的分发。

5.4 本章小结

本章对远程培训系统中的随机点播调度策略进行了研究，首先研究了现有的流媒体调度方案，然后对 CDN 内容分发网络的网络架构以及原理进行了阐述，将其核心思想借鉴到视频调度的策略中来。并且根据企业网络自身的特点对调度策略进行了完善，最后，在 .Net Framework 技术平台下用代码对其进行了实现。

第 6 章 系统功能实现

6.1 支撑技术及其技术框架

从前面三章可以看出，远程培训系统的几个功能模块中的支撑技术通过从视频直播到视频点播再到基于 CDN 的视频文件调度这样的流程紧密的联系了起来，每个支撑技术单独负责功能模块内的核心功能实现，而实现的结果是让整个远程培训系统良好地运行。与此同时，模块化的耦合分离使得上述支撑技术也可以很方便地进行具体实现的替换，也可以将具体某个模块的支撑技术进行抽离，并集成到其他类似的系统中去。

比如对于 Web 平台的选择就可以从降低成本考虑选择以开源 Java 为平台的 Web 技术框架，整体系统实现中除了流媒体调度技术的相关实现有所改变以外，其他的模块的支撑技术还是能用到新的多媒体培训系统中去。这种类似插件的可插拔性也是这些支撑技术的价值所在。

这些支撑技术使用的技术框架涉及面较广，从浏览器的富客户端插件到桌面程序再到 Web 服务器端的 Web 应用程序都有涉及，如果只是单单采用一种平台和实现技术很难满足系统的实际需求。

多媒体远程培训系统的实现主要是在三个不同的平台上用代码进行了实现。直播模块，多媒体课件制作模块的部分功能主要是基于 Microsoft .Net Framework 框架的 Winform 技术进行实现，同时底层多媒体相关的 API 采用的 Microsoft Expression Encoder API。多媒体课件播放的功能实现中采用了基于 Microsoft Silverlight 的富客户端技术进行实现。点播调度模块中主要有两个部分，视频文件的分发与复制和访问重定向。前者采用 Microsoft .Net Framework 的 Winform 技术实现后台分发，重定向在 Web 服务器端的程序中实现，采用的是 Microsoft .Net Framework 的 Asp.Net 技术。

可以看出多媒体远程培训系统为了满足系统实际需求，需要在不同的软件平台上进行有针对性的代码实现。在该系统中，虽然采用的技术平台，应用框架不同，但是所有的这些代码实现，采用的是所有平台都支持的 C#（C Sharp）语言。C#是微软为 .Net Framework 开发的一种语言，与 C++和 Java 一样是一种

面向对象的语言。作为 .Net Framework 精简版的 Silverlight Runtime 也支持 C# 语言进行应用程序的开发。系统实现上的统一的代码，相同的类库的实现相对于不同应用框架上的不同语言大大降低了开发的技术迁移成本，也使得代码更容易维护。

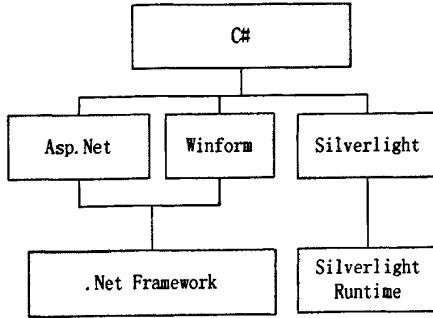


图 6-1 系统应用框架示意图

6.2 系统实现的三层架构

整个系统在代码实现上，基于软件设计高耦合，低内聚的思想，采用了三层架构将软件分为了表现层，业务逻辑层和数据持久层。表现层用来显示数据和提供用户录入数据的接口。业务逻辑层处在表现层和数据持久层的中间，主要处理软件系统的业务逻辑。数据持久层负责对数据的访问。软件的三层架构让开发者更专注的关注结构中的某一层，同时降低了层与层之间的依赖性，增加了软件系统的鲁棒性。

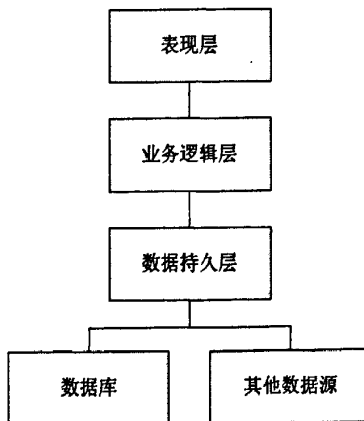


图 6-2 软件三层架构示意图

6.3 系统总体功能实现

多媒体远程培训系统采用模块化的方法对功能进行封装，一共有系统管理模块，直播模块，多媒体课件制作模块，点播调度模块这四个模块。本文的前面三章已经分别对多媒体远程培训系统各个功能模块中的关键性支撑技术的原理进行了分析，实现的过程进行了详细的说明。下面对系统的整体功能和系统的功能模块的实现进行阐述。

多媒体远程培训系统的视频直播模块首先满足了企业机关对于实时视频直播的功能需求，多媒体教室中的若干个摄像头被教室 PC 机上的软件所控制，实时地向中心流媒体服务器推送数据，而且推送出去的数据流能在各个摄像头的数据流中进行实时的切换，来满足教师授课的普遍需求。在直播的同时，直播控制软件会将推送到流媒体服务器的数据保存一份到本地，作为多媒体课件制作的毛坯文件。

在视频直播结束以后，可以用本系统的多媒体课件制作模块中的基于事件触发机制和时间轴模型，将视频直播所产生的毛坯视频文件和本次课程相关的多媒体课件制作成多媒体课件。

在员工进行视频点播的过程中，基于 CDN 的视频文件调度算法按照特定的策略进行请求重定向，同时在后台自动完成视频文件到网络边缘硬盘的分发与复制。通过点播调度的策略，更有效地利用了网络带宽，同时使得多媒体远程培训系统运行地更加稳定。与此同时，系统还能在特殊情况下对网络硬盘上的资源进行手动管理和手动分发。

与此同时，系统管理模块提供了对用户，站点以及网络设备等的基本管理功能。下面是系统用户管理的相关功能界面。

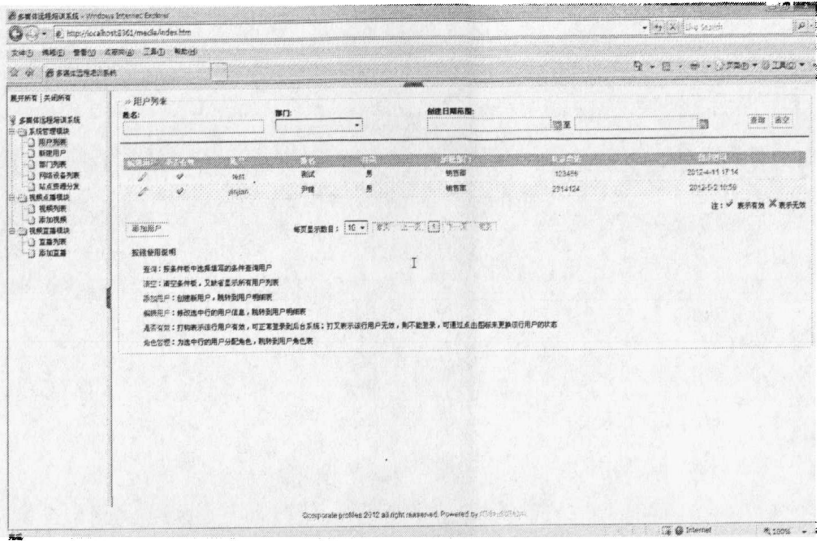


图 6-3 用户管理界面

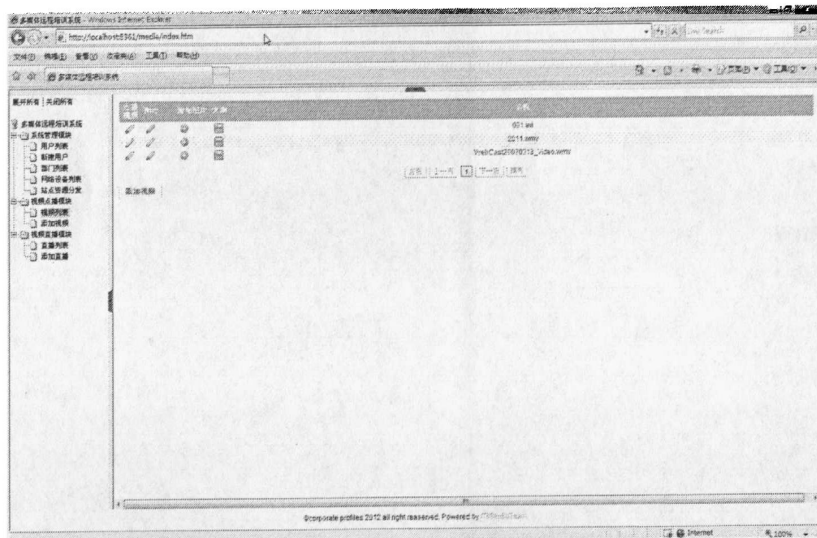


图 6-4 视频管理界面

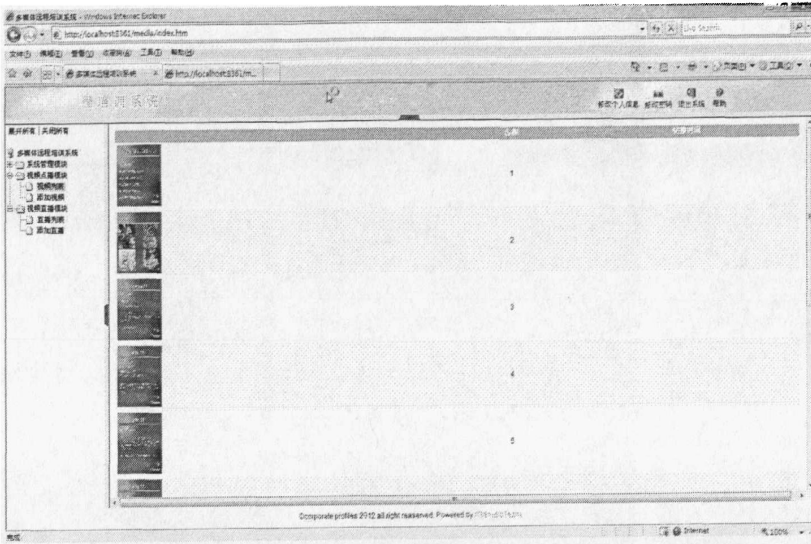


图 6-5 查看视频对应的 JPG 文件

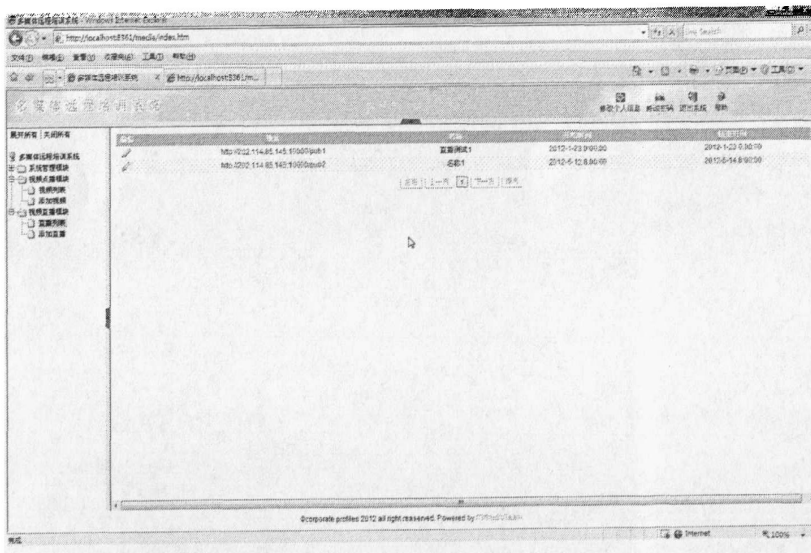


图 6-6 直播发布点的管理界面

整个多媒体远程管理系统实现以后总共有三个应用程序，这三个应用程序分别满足了 2.1 中的四个系统功能模块的需求。具体的应用程序根据 6.1 中介绍的内容是一个用来提供管理功能，和视频点播浏览功能的 Asp.Net 应用程序，一个用来提供视频直播控制，和多媒体课件制作的 Winform 应用程序和一个内嵌到浏览器中用来进行多媒体课件显示的 Silverlight 应用程序。用户和管理员一般使用的是 Web 程序和 Winform 程序。

第 7 章 总结与展望

7.1 总结

本文首先对企业网络架构下的多媒体远程培训系统进行了需求分析，并通过模块化的抽象方法对其进行了功能模块的划分。然后针对不同模块中的关键性的支撑技术的原理，以及具体的实现思想和过程进行了详细的阐述。得到了一下研究成果：

(1)提出了以流媒体服务器为中心的的网络架构以及多路视频推送的现场直播方式；

(2)引入事件驱动机制和时间轴模型提出了一种多媒体课件制作和播放的方法；

(3)在分析 CDN 实现原理的基础上提出了一种适用于典型企业网络架构的基于 CDN 的视频文件调度策略；

(4)在 Microsoft .Net Winform, Microsoft .Net Asp.Net, Microsoft Silverlight 三个平台上用统一的编程语言 C#对不同模块的功能进行了实现，实现了多媒体远程培训系统。

多媒体远程培训系统针对企业的实际培训的需求给出了一个较完整的解决方案，实现了企业网络架构下的网络同步培训和异步培训。从功能模块上划分，该系统一共有系统管理模块、直播模块、多媒体课件制作模块、点播调度模块这四个模块。网络直播，多媒体课件制作以及视频文件调度是该系统的支撑关键技术。

7.2 展望

多媒体远程培训系统中所涉及的几项支撑技术是十分具有应用前景和实用价值的技术。从网络直播，多媒体课件制作到点播的流媒体调度还有值得进一步研究的价值。下一阶段的研究重点主要集中在一下几个方面：

(1)服务器端引入智能流技术通过在传输流媒体数据时实时探测网络带宽

变化无缝切换到不同速率对应的流媒体数据上，提升网络带宽利用率。

(2) 通过 Web 与后台程序的通信完成基于 Web 系统的远程手动分发，更好的满足用户需求。

(3) 对点播调度策略进行进一步研究，尝试引入基于实时网络状态的流媒体调度策略。

致 谢

首先感谢导师陈德军教授，本篇论文的研究和写作工作从始至终都是在陈老师的悉心指导、帮助、督促下完成的。三年来，陈老师不但教我专业知识，同时也教我做人的道理。他科学严谨的治学态度、忘我敬业的工作精神、深厚扎实的理论知识、脚踏实地的工作作风，这一切都给我留下了深刻的印象，都令我今生受益匪浅，激励我在今后的学习和工作中更加努力地进取。

我还要特别感谢我的身边的朋友和实验室所有的兄弟姐妹，他们平时在我的生活和学习中都给过我很多指导与帮助，为我完成这篇论文提供了很大的帮助。和他们相处和生活的这段愉快日子，是一段非常美好的回忆，我永远都不会忘记！

感谢培养我的武汉理工大学信息工程学院和所有教过我的任课老师！能够在这里学习和生活，是我一生的幸运和最为宝贵的财富，感谢武汉理工大学给了我这个机会。

另外我还要感谢我敬爱的父母和亲人，感谢多年来他们对我的养育之恩，他们一直在支持和鼓励我完成学业，在经济上和精神上都给予我最无私的帮助，使我能够全身心的投入到学习和科研工作中去。

此外，我还要衷心感谢各位专家、学者为评阅本文所付出的辛勤劳动，谢谢！

参考文献

- [1] 李云芳. 浅谈多媒体通信技术的应用与未来发展研究[J]. 信息安全与技术, 2011, 8: 23-27.
- [2] 林雄. 农村中小学教师远程培训实施问题研究[D]. 福建: 福建师范大学, 2008
- [3] 冯坚. 基于云计算的现代远程教育展望[J]. 中国电化教育, 2009,10:39-42
- [4] 张秀梅. 我国企业远程培训研究十年述评[J]. 现代远程教育研究, 2010, 3: 64-69.
- [5] Hongyu Zhao. Research on distance education based on cloud computing, Pervasive Computing and Applications (ICPCA), 2011 6th International Conference. 2011,343-348
- [6] Harris, D.A.; Krousgrill, C. Distance Education: New Technologies and New Directions. Proceedings of the IEEE 2008,917-930
- [7] 曾海军. 从国外远程教育看中国高校网络教育的发展[J]. 电化教育研究, 2008, 4:80-83
- [8] 张利飞. 基于传输的网络视频编码技术研究[J]. 计算机系统应用, 2005, 11:59-62
- [9] Acosta, A.; Garcia-Vazquez, M.S.; Colores-Vargas, J. MPEG-4 AVC/H.264 and VC-1 Codecs Comparison Used in IPTV Video Streaming Technology Electronics, Robotics and Automotive Mechanics Conference, 2008. CERMA '08. 2008,122-126
- [10] Rao, K.R.; Do Nyeon Kim Current video coding standards: H.264/AVC, Dirac, AVS China and VC-1. System Theory (SSST), 2010 42nd Southeastern Symposium 2010,1-8
- [11] 冯丽. VC-1 解码算法研究及其 DSP 移植与优化[D]. 北京: 北京交通大学, 2008
- [12] 唐智伟. 基于 H.323 协议的 H.264 视频传输[D]. 湖南: 湘潭大学, 2008
- [13] 刘达. MPEG_4 视频编码核心思想及关键技术研究[J]. 中国数据通信, 2004, 1: 102-108
- [14] 魏于涛. 基于 H.264 和 AVS 的流媒体传输技术研究[D]. 山东: 山东大学, 2008
- [15] 程子傲. 流媒体分类播放技术研究与设计[D]. 辽宁: 辽宁大学, 2011
- [16] Frossard, P.; de Martin, J.C.; Reha Civanlar, M.. Media Streaming With Network Diversity. Proceedings of the IEEE 2008,39-53
- [17] Shi Deng; Di Wu; Zhuying Lin. Streaming media technology and its application in modern network teaching. 2010,477-480
- [18] 方新勇; 马瑞芳; 刘萍芬; 杨文革. 基于 RTP 的流媒体传输系统的设计与实现[J]. 微电子学与计算机, 2007, 24 (11): 183-185
- [19] 李昕. 互联网实时流媒体传输关键技术的研究 [D]. 北京: 北京交通大学, 2010

- [20] 杨新; 徐鑫. 基于 HelixServer 的流媒体服务器平台的搭建[J]. 农业网络信息, 2007, 8,97-101.
- [21] 吴洪祥. 网络实时教学平台的设计与实现[D].南京: 南京师范大学, 2008
- [22] 孔峰. 流媒体代理服务服务器缓存研究与实现[D],北京: 北京邮电大学, 2009
- [23] 刘玮哲. 流媒体代理服务服务器的设计与实现[D].北京: 北京邮电大学, 2008
- [24] 陈亮; 裴海龙. 基于 RTSP 协议的实时视频服务器实现[J]. 微计算机信息, 2009, 25, 5-3: 65-66.
- [25] 王原丽; 刘建伟. 基于 RTP 协议的 MPEG-4 的视频传输系统应用研究[J]. 计算机与现代化, 2007, 12 : 56-58.
- [26] Peng Liang; Yang Shun. Research and implementation of voice transmission based on RTP protocol Computational Problem-Solving (ICCP), 2010 International Conference 2010.416-417
- [27] 崔燕. 浅论多媒体课件制作与使用[D].吉林: 吉林大学, 2010.
- [28] 孙青茹; 赵普军. 基于 Web 的多媒体课件的优势及应注意的问题[J]. 中国校外教育(理论), 2008, 7: 155-156
- [29] Zhongyao Zhang; Karmouch, A Multimedia courseware delivery over the Internet . Electrical and Computer Engineering, 1998. IEEE Canadian Conference 1998. 609 - 612 vol.2
- [30] Wang, R.; Karmouch, A. Multimedia courseware delivery over broadband networks. Communications, 1997. ICC 97 Montreal, 'Towards the Knowledge Millennium'. 1997 IEEE International Conference .1997. 782 - 786 vol.2
- [31] 罗凌. 基于 XML 的多媒体 Web 课件制作系统研究[J]. 计算机工程与应用, 2005,31:229-232.
- [32] 韩凤军. 基于 Web 的多媒体课件的开发原则及架构[J]. 中国电力教育, 2010, 13: 71-72
- [33] 潘陆益. 基于流媒体的 Web 课件设计技术[J]. 教育信息化, 2005,2, 58-59
- [34] 刘芳. 基于时间轴模型的音视频同步的研究与实现[D].广州: 暨南大学, 2008
- [35] 智英建. 基于新型分布式视频点播架构的流媒体调度技术研究[D].郑州: 解放军信息工程大学, 2008
- [36] 彭墨青; 谢建国. 应用层组播节点流媒体调度算法[J]. 计算机工程, 2009, 35 (15): 235-237
- [37] 王海峰. 基于补丁算法的流媒体代理缓存方法研究[D].上海: 华东师范大学, 2008

- [38] 杨戈; 朱晓民; 廖建新; 黄海. 基于缓存窗口和段补丁预取的流媒体动态调度算法[J]. 电子与信息学报, 2007, 29 (5): 1198-1201
- [39] Yunpeng Chai; Zhihui Du; Sanli Li. A New Scheduling Algorithm for Distributed Streaming Media System Based on Multicast Distributed Computing Systems Workshops, 2008. ICDCS '08. 28th International Conference 2008,587-592
- [40] Chao Yu; Ling Xing; Jian-Guo Ma; Rong Fu. Incorporated scheduling strategy for streaming media based on popularity . Apperceiving Computing and Intelligence Analysis, 2009. ICACIA 2009. International Conference 2009,371-374
- [41] 徐悦. 构建大型 CDN 网络的关键技术研究[D].北京: 北京邮电大学, 2008
- [42] 刘瑜; 朱光喜; 尹浩; 刘雪宁. 内容分发网发展综述[J]. 计算机科学 2009,36 (2): 11-14
- [43] 雷孟成. CDN 构架设计及其配置管理系统的设计与实现[D].北京: 北京邮电大学, 2011
- [44] 何锐; 李艳; 刘红. 基于 CDN 内容分发的研究与应用[J]. 通信技术 2011, 44 (03): 111-115
- [45] Moo-Kyung Sung; Chi-Moon Han. A study on architecture of CDN(Content Delivering Network) with content re-distribution function . Advanced Communication Technology, 2009. ICACT 2009. 11th International Conference 2009,772-777
- [46] 吴廷良. P2P CDN 流媒体分发技术的研究与实现[D].北京: 北京邮电大学, 2011
- [47] 王薇薇; 基于 CDN 的流媒体分发技术研究综述[J]. 计算机工程与应用 2004, 8,121-125.
- [48] 张彦明. 基于 CDN 的流媒体代理研究与实现[D].北京: 北京邮电大学, 2010
- [49] Shi Kang; Hao Yin. A Hybrid CDN-P2P System for Video-on-Demand. Future Networks, 2010. ICFN '10. Second International Conference 2010,309-313
- [50] 杨波. 密码学 Hash 函数的设计和应用研究[D].北京: 北京邮电大学, 2008
- [51] 熊云龙. 基于 Silverlight 的 RIA 研究及应用[D].重庆: 重庆大学, 2008

攻读硕士学位期间参加的项目

- [1] 输电线路在线综合监测和决策支持系统
- [2] 北京市朝阳区物联网示范工程
- [3] 职工远程教育培训系统