

# 分布式多媒体系统服务质量管理实现支持机制\*

王兴伟 张应辉 刘积仁 李华天

(东北大学软件中心研究部 沈阳 110006)

**摘要** 近年来,分布式多媒体系统中的服务质量 QoS(quality of service)管理及其实现支持机制取得了很大进展.本文从接纳控制与资源预约、基于速率的流控、差错控制、调度与同步几个方面出发,讨论 QoS 管理实现支持机制,并展望今后的研究工作.

**关键词** 分布式多媒体,服务质量,服务质量管理,实现支持机制,多媒体通信.

**中图分类号** TP393

近年来,分布式多媒体系统得到了很大发展,出现了很多分布式多媒体应用,如远程学习、计算机会议、远程医疗诊断、分布式多媒体信息点播、多媒体电子邮件等.用户对这些应用一般都有一定程度的服务质量 QoS(quality of service)要求.因此,分布式多媒体系统需要强有力的 QoS 管理及其实现支持机制,以满足用户的 QoS 要求.

在分布式多媒体系统中,由于静态媒体和连续媒体同时存在,因而 QoS 管理面临很多新的挑战.尽管目前还没有一种完整、通用、实用的解决方案,但是人们已经开始在很多方面达成共识:QoS 管理面临的挑战主要来自连续媒体;QoS 管理应该是可配置的、完全端到端的、层次化的、动态的,应该提供面向不同应用的确定型、统计型和尽全力型 QoS 承诺,应该包括 QoS 协商、监控、适配、再协商以及映射等功能;对各层的最优 QoS 参数集虽未完全达成共识,但已获得了一定的实际经验.这一切都为 QoS 管理的设计、实现乃至标准化奠定了良好的基础.本文从接纳控制与资源预约、基于速率的流控、差错控制、调度和同步几个方面讨论了 QoS 管理实现支持机制.

## 1 接纳控制和资源预约

在提供端到端 QoS 时,接纳控制与资源预约是基本手段之一.<sup>[1~3]</sup>对于确定型和统计型 QoS 承诺,接纳控制和资源预约是必需的.随着用户数量的增加,分布式多媒体系统的负载将不断升高,其性能最终将达到一个临界点,在该临界点处,如果再接纳一个用户请求,就会导致系统过载,致使系统不能继续满足现有用户和新到用户的 QoS 要求.因此,任何提供 QoS 管理机制的系统都应该在完全端到端基础之上提供接纳控制功能.

接纳控制通常分两个阶段进行.第 1 阶段判断在完全端到端路径上的端系统和沿途各中间节点能否获得所需的资源.如果该阶段成功,则在第 2 阶段为用户请求预约所需的资源.第 1 阶段从源地向目的地地前向传播,第 2 阶段从目的地向源地地向后传播.如果系统不能按用户所申请的 QoS 接纳用户请求,那么用户可以选择再协商较低的 QoS.

接纳控制通过接纳测试来确定资源的可用性.系统通常至少要进行 3 类接纳测试:端系统和网络中间节点的 CPU 可调度性测试;缓冲空间分配测试;系统总线和网络链路带宽测试.当某个用户请求在端系统和沿途各中间节点都通过接纳测试之后,分布式多媒体系统就应该向该用户提供先验性能保证,同时信守对其他用户已经作出的 QoS 承诺. Tenet 方法通过对接纳测试的结果进行数学证明做到了这一点.<sup>[3]</sup>进行资源预约时,多数系统是沿着从源地到目的地的方向在由接纳测试确定的路径上预约资源.预约资源可以采用两种不同的策略.悲观策略按照最坏情况预约资源,即预约最长的 CPU 处理时间、最高带宽等.该策略的资源利用率较低,但可以避免冲突、保证质量,适合于确定型 QoS 承诺.乐观策略基于平均工作负载预约资源,因而资源利用率较高,但系统可能出现过载,导致 QoS 降级.该策略比较适合于统计型 QoS 承诺.对于多数应用来说,系统还应该允许在某用户不使用为它预约的那些资源的时候,其

\* 本文研究得到国家“九五”科技攻关项目基金资助.作者王兴伟,1968 年生,在职博士生,讲师,主要研究领域为分布式多媒体信息处理技术,计算机网络.张应辉,1972 年生,博士生,主要研究领域为分布式多媒体信息处理技术.刘积仁,1955 年生,博士,教授,博士生导师,主要研究领域为分布式多媒体信息处理技术与方法学,CSCW,组件技术.李华天,1922 年生,教授,博士生导师,主要研究领域为分布式多媒体信息处理技术与方法学.

本文通讯联系人:王兴伟,沈阳 110006,东北大学信息学院计算机系 129#

本文 1996-10-15 收到原稿,1997-05-19 收到修改稿

他用户(特别是任务紧急的高优先级用户)可以使用这些资源。当然,这会提高系统资源管理的难度。用户应用结束后,系统回收相应的资源。

尽力型 QoS 承诺不需要接纳控制与资源预约,因为系统不对其提供 QoS 保证。

## 2 基于速率的流控

在分布式多媒体系统中,由于连续媒体数据的生成、传送与播放具有严格的实时性和等时性,需要维持媒体内时间连续性,因此,源地应该以目的地的播放速率发送媒体量子。即使收发双方的速率不能完全吻合,也应该相差甚微。这样才能保证发送方既不会淹没接收方,也不会使接收方产生饥饿。例如,在 NTSC 制式下视频流的接收方应该每秒获得 30 个视频帧,而且应该是每隔 1/30 秒获得一个视频帧。如果视频帧到达过早,接收方就需要缓冲它,直到适合该帧的播放时刻到来为止。如果这种局面持续的时间太长,则由于每个视频帧需要的缓冲空间较大,连续缓冲早到的视频帧就会使系统很快用完缓冲区,造成后续视频帧的丢失。如果帧到达得太迟,图象运动就不平稳。因此,为了提供 QoS 保证、有效地克服抖动现象的发生,分布式多媒体系统通常采用基于速率的流控机制,而不是采用传统的滑动窗口机制;这样做不仅可以建立连续媒体数据流与速率受控传送之间的自然对应关系,使发送方的通信量平稳地进入网络以便与接收方的处理能力相匹配,而且可以将流控和差错控制机制解耦,漏篮(Leaky Bucket)方法<sup>[4]</sup>是比较常用的基于速率的流控方法。使用该方法时,发送方将数据放到篮中。随着按一定速率发送数据,数据逐渐排出篮底。数据实际离开主机进入网络的速率,由位于篮底的调节器控制。篮的尺寸限制因等待而累积的数据量。

## 3 差错控制

在传统的通信服务(如文件传送)中,高可靠性至关重要,系统必须保证数据提交的值域正确性。在分布式多媒体系统中,多媒体数据在目的地的正确提交通常有两方面含义:值域正确性和时域正确性。不同的分布式多媒体应用对两者的相对重要性的要求往往有所不同,通常需要在两者之间进行适当的权衡。此外,传统通信系统通常只是在低层(如数据链路层、网络层、传送层)对差错进行检测、指示与恢复;在应该对差错控制策略与行为作出决策的高层(如应用层),差错控制机制往往却被忽略了。因此,分布式多媒体系统不能简单地照搬传统通信系统中的差错控制机制,不仅需要增加高层的面向应用的差错控制手段,而且应该提供可配置能力,允许端用户对差错检测、差错指示和差错纠正进行适当的组合,以便剪裁出最适合于应用要求的差错控制策略。象音频和视频之类的连续媒体具有一定的内在容错能力,可以容忍一定比例的比特错和帧丢失。对于一般的视频应用来说,如果某帧丢失,系统可以通过简单地重播前一帧来维持播放的连续性,提供差错检测与指示机制即可。对于某些应用来说,即使不提供差错控制机制,用户也可能接受。但是,对于实时差错敏感型数据(如远程医疗诊断中患者 X 射线影像)的传送,则需要支持高可靠性、实时与等时传送的差错控制机制。

连续媒体及其压缩技术的引入,向分布式多媒体系统中的差错检测机制提出了新的挑战。例如,在采用预测编码的视频数据(如 MPEG 流)中,如果帧边界被破坏,则差错无法恢复。这就要求系统应该保护数据流中的结构信息,需要对现有的差错检测机制(如检查和、定序)进行适当的扩充,以携带必要的结构信息。

超时重传机制不适应连续媒体通信的实时与等时要求,因为连续媒体播放不能容忍由于超时重传而带来的高延迟。在分布式多媒体系统中,迟到的视频帧和音频帧通常只能被丢掉,其效果等同于帧丢失。预防型差错纠正模式<sup>[5]</sup>,如前向纠错 FEC(forward error correction)和优先级信道编码模式,比较适合连续媒体通信。采用 FEC 模式时,发送方可以在原始数据流中增加必要的冗余信息,如对关键帧进行复制,以便接收方在检测到差错时立即通过冗余信息进行恢复,这对于端到端延迟保证的实现非常有利。FEC 通常也不需要从接收方到发送方的控制连接。但是,增加的冗余数据会相应地提高对系统吞吐量的要求,因此,系统需要知道发送方与接收方之间的连接出错概率以及应用所需的可靠性,以便合理地确定数据冗余量,减轻对网络带宽的压力。此外,FEC 不能确保从出错数据中恢复过来(如果出错数据恰好没有进行复制或者源与副本同时出错,就会出现这种情况),因此,应该对冗余数据的安排进行优化,尽可能提高系统的容错能力。在优先级信道编码方法中,系统将媒体分成多个优先级不同的数据流,为那些对于重建原始媒体来说比较重要的数据(如 MPEG 中的 I 帧)赋予较高优先级,为其他数据(如 MPEG 中的 P 帧和 B 帧)赋予较低优先级。在拥挤期间,系统首先丢掉低优先级数据。

还应指出,压缩技术虽然大大降低了多媒体数据的带宽要求,但同时也大大降低了多媒体数据的内在冗余性,因此,相应地增加了差错控制实现的难度。

## 4 调度

调度机制是分布式多媒体系统向用户提供并维持所需 QoS 水平的又一基本手段。分布式多媒体系统使用的调度算法很多,如最早最后期限优先 EDF(earliest deadline first)算法、虚拟时钟算法、加权公平排队算法等。其中 EDF 算法是非常著名的实时调度算法之一。在每一个新的就绪状态,调度器都是从那些已就绪,但还没有完全处理完毕的任务中选择最后期限最早的任务,并将执行该任务所需的资源分配给它。在有新任务到来时,调度器必须立即计算 EDF,排出新的定序,即正在运行的任务被剥夺,并且根据新任务的最后期限决定是否调度它。如果新任务的最后期限早于被中断的当前任务,就立即处理新任务,被中断任务的处理将在稍后继续进行。EDF 算法适合于周期性实时任务的调度,而且可以对任务的处理提供相应的保证。由于连续媒体的生成、传送与播放具有连续性、实时性和等时性,因此,媒体量子的生成与处理具有明显的周期性,连续媒体流中的每个媒体量子都有一个隐含的最后期限,基于最后期限来调度进程去处理这些媒体量子非常自然。这样,EDF 算法就成为比较常用的调度算法。

延迟 EDD(delay earliest due date)<sup>[6]</sup>算法是对 EDF 算法的一种扩充。服务提供者按照同每个源协商好的服务合同工作。合同申明,如果源遵守峰值和平均发送速率,服务提供方将保证提供有界延迟。服务提供者根据合同将分组的最后期限设置为应该发送该分组的时间。该时间实际上就是该分组的预计到达时间与服务提供者延迟界限两者之和。通过预留峰值速率带宽,延迟 EDD 可以确保每条连接的延迟界限。抖动 EDD(Jitter EDD)<sup>[6]</sup>算法对延迟 EDD 做了进一步的扩充,以便提供延迟抖动界限。在为某个分组提供服务之后,交换机就为该分组打上一个“戳”,该戳用来指示该分组的最后期限和实际结束时间之差。位于下一个交换机入口处的调节器在将该分组标记为符合调度条件之前,要将该分组保留戳所指示的那段时间。这样做就提供了最小和最大延迟保证,从而有效地控制延迟抖动。

延迟 EDD 属于工作保养型(Work-conserving)调度算法。该类算法总是以较高的速率为分组提供服务,只要这不影响对其他连接的性能保证即可。只要服务用户有分组要发送,服务提供者就不会空闲。抖动 EDD 属于非工作保养型(Non-work-conserving)调度算法,该类算法不是在任何情况下都以较高速率为分组提供服务。使用该类算法时,每个分组都被显式或隐式分配一个合格时间。在服务提供者空闲时,即使有分组但其合格时间未到,服务提供者也不传送分组。工作保养型和非工作保养型调度算法都可以提供吞吐量保证。

## 5 同步

分布式多媒体应用需要实时同步机制,以便满足用户对媒体间同步的要求。在分布式多媒体应用中,需要两种风格的实时同步<sup>[7]</sup>:事件驱动同步和连续同步。事件驱动同步是一种通知某事件或事件集已经发生,并且随后导致有关操作或操作集发生的同步操作。与事件驱动同步相关的所有操作应该及时完成。例如,如果用户用鼠标点一下与停止视频播放对应的停止按钮,视频播放就应该立即停止。连续同步是在需要将媒体播放设备“捆绑”在一起,以便按固定速率播放媒体数据时产生的同步。视频和与之相伴的音频之间的“对口型”关系是该类同步的典型例子。

实现同步的方法可以分成两类:① 静态方法。<sup>[8]</sup>在该类方法中,每个媒体量子都被分配一个时间戳。在播放场地有一个同步实体,该实体的任务就是在指定的时间播放相应的媒体量子。该方法比较适合于预先定义好的多媒体播放,因为整个播放期间的同步需求都预先知道。但是,该方法不适合于有动态用户交互作用来改变播放过程的实时交互式应用。② 动态方法。在播放期间,每当有媒体量子或者有来自用户的交互作用到达时,与之相关的事件就被通知给应用,但是处理这些事件的责任留给了应用程序员。<sup>[9]</sup>显然,用来实现媒体间同步的机制本身应该是实时的。

## 6 今后的研究工作

应该指出的是,已有的研究工作大多只是针对某种实现支持机制,缺乏对各种实现支持机制的有机集成,因此,对 QoS 管理实现的支持往往有限。另一方面,分布式多媒体应用种类繁多,而且新的应用不断出现,不同应用的 QoS 需求通常各异,因此,需要组合不同的实现支持机制以提供满足应用实际需要的 QoS 管理机制。目前,我们正在基于“构件—框架”思想设计一种通用性与可扩充性好的 QoS 管理模型。在该模型中,各种实现支持机制作为标准构件被放到可扩充的构件库中,组合机制根据应用特征挑选所需构件,装配成满足应用实际需要的 QoS 管理机制。此外,在分布式多媒体组应用(如计算机会议)中,由于各成员使用的端系统的能力、所上网络的能力以及所愿支付的费用不同,因此,在不同成员处实际达到的 QoS 水平应该有所不同。提供适宜的实现支持机制以满足不同组成员的异构 QoS 需求也是我们目前正在深入研究的课题。

分布式多媒体系统中的 QoS 管理非常复杂,不仅要支持连续媒体数据生成、传送、访问的连续性、实时性和等时

性,还要支持各种媒体之间的同步与集成,因而需要有强有力的支持机制实现 QoS 管理,需要我们进一步研究与探索。

### 参考文献

- 1 Ferrari D *et al.* Network support for multimedia: a discussion of the Tenet approach. *Computer Networks and ISDN Systems*, 1994,26(10):1267~1280
- 2 Campbell A *et al.* A quality of service architecture. *ACM Computer Communication Review*, 1994,24(2):6~27
- 3 Blair G S *et al.* System support for multimedia applications: an assessment of the state of the art. *Information and Software Technology*, 1994,36(2):203~212
- 4 Turner J S. New directions in communications. *IEEE Communications*, 1986,24(10):8~15
- 5 Biersack E W. Performance evaluation of forward error correction in an ATM environment. *IEEE Journal on Selected Areas in Communications*, 1993,11(4):631~640
- 6 Zhang Hui. Service disciplines for guaranteed performance service in packet-switching networks. *Proceedings of the IEEE*, 1995,83(10):1374~1396
- 7 Coulson G *et al.* Supporting the real-time requirements of continuous media in open distributed processing. *Computer Networks and ISDN Systems*, 1995,27(8):1231~1246
- 8 Srinivas Ramanathan *et al.* Towards personalized multimedia dial-up services. *Computer Networks and ISDN Systems*, 1994,26(10):1305~1322
- 9 Nicolaou C. An architecture for real-time multimedia communication systems. *IEEE Journal on Selected Areas in Communications*, 1990,8(3):391~400

## Quality of Service Management Implementation Supporting Mechanisms in Distributed Multimedia Systems

WANG Xing-wei ZHANG Ying-hui LIU Ji-ren LI Hua-tian

(Research Division Software Center Northeastern University Shenyang 110006)

**Abstract** In recent years, great advances have been made in the QoS(quality of service) management and its implementation supporting mechanisms in distributed multimedia systems. In this paper, the QoS management implementation supporting mechanisms are discussed from following aspects: admission control and resource reservation, rate-based flow control, error control, scheduling and synchronization. Finally, future research works are proposed.

**Key words** Distributed multimedia, quality of service, QoS management, implementation supporting mechanisms, multimedia communication.