

应用层语义可靠的自适应多播*

谭焜, 史元春, 廖春元, 徐光祐

(清华大学 计算机科学与技术系, 北京 100084)

E-mail: cohen@media.cs.tsinghua.edu.cn

http://www.tsinghua.edu.cn

摘要: Internet 上的可靠多播具有很大的应用前景,但是同时也面临着挑战,主要是因为 Internet 的异构性以及 IP Multicast 仍不能广域内得以实现.提出了一种应用层语义可靠的自适应多播体系结构(application semantics reliable multicast,简称 ASRM).ASRM 不以 IP Multicast 为基础,而是采用一种混合 IP 单播和多播的方法来实现多点数据通信.ASRM 采用一种简单、自然,且具有更好的伸缩性的方法来命名多播会话,并可以在转发过程中,根据变换模型、变换规则 and 用户意愿进行自适应的数据传输,从而解决异构性问题.ASRM 适用于广域范围内的小规模可靠多播应用.

关键词: 语义可靠;自适应;异构性;多播

中图法分类号: TP393 **文献标识码:** A

可靠的多点数据通信是许多新颖网络应用必须的基础通信服务.最近几年提出的许多可靠多播方案^[1-3],都是基于 IP Multicast 技术^[4]的,并在其上增加一定的功能来达到可靠的传输.但在现有的 Internet 中,这些方法都面临着同样的两个基本问题:(1) Internet 固有的异构性;(2) 现有的 Internet 还远不能支持 IP Multicast.

异构性是 Internet 所固有的,其表现在 Internet 中的网络连接和终端设备都有着很大的区别.因此,当所有这些不同的用户,使用不同的终端设备,以不同的接入方式加入到同一个多播应用中时,如何决定数据的传输策略,就成为一个严重的问题.在许多应用中,例如视频会议,并不要求所有的用户接收的所有数据都严格一致,因此就有可能对数据进行自适应的传输,高端用户和低端用户获取不同质量的数据,但在语义上保持一致.

另一方面,IP Multicast 在广域范围内的实用在短期内仍然是无法实现的.这主要有两个问题:首先,仍然缺乏高效、强壮、适用于大规模广域的路由协议;其次,缺乏在广域中 IP Multicast 地址分配、会话建立和发现的有效方案.因此,在大规模 Internet 上支持 IP Multicast 仍然有很长的路要走.

本文提出了一种新颖的多播体系结构,称为应用层语义可靠的自适应多播(application-level semantics reliable multicast,简称 ASRM).它试图在现有的 Internet(IP)基础上建立起一层覆盖网络,从而实现在广域环境下的数据多点自适应传输.

ASRM 有两个关键点:(1) ASRM 不以全局 IP Multicast 为基础,而仅仅把 IP Multicast 作为一种可以在局域范围(或多播域)内使用的网络技术,例如 LAN 等.ASRM 将尽可能地使用 IP Multicast 技术来提高效率,但是它不是必须的.我们通过建立起一层覆盖网络来实现多点数据通信.(2) ASRM 可以在数据传播的过程中,对数据进行处理和变换,从而能够根据不同网络和终端的情况,自适应地传送相应的数据.在变换的过程中,我们将保持数据在应用层语义上的可靠性.

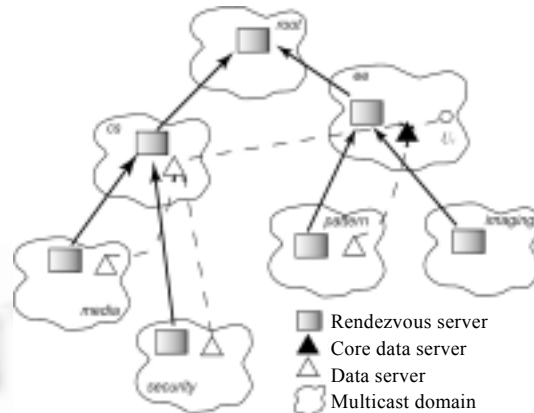
* 收稿日期: 2000-09-14; 修改日期: 2001-02-16

基金项目: 985 清华大学学科规划重点项目(985TH-22-06)

作者简介: 谭焜(1973 -),男,江苏南京人,博士生,主要研究领域为多媒体网络;史元春(1967 -),女,河南汤阴人,博士,副教授,主要研究领域为多媒体,人机交互技术;廖春元(1976 -),男,云南人,硕士生,主要研究领域为多媒体技术;徐光祐(1940 -),男,上海人,教授,博士生导师,主要研究领域为计算机视觉,人机交互技术.

1 应用层语义可靠的自适应多播

我们将能够使用 IP Multicast 的局部范围称为多播域.我们的任务是在 Internet 上建立起一层覆盖网络,利用成熟的单播协议(如 TCP)将这些多播域连接起来,从而使得多播可以在全局范围内实现.为此我们引入了两类服务器:聚集服务器(rendezvous server)和数据服务器(data server).在每一多播域中,都存在着一个指定的聚集服务器.聚集服务器是多播会话的管理者,用以创建、加入、查找多播会话.数据服务器是覆盖网络中的中继节点,在 ASRM 中,这些数据服务器被组织成为树状的覆盖网络.多播会话的数据包通过它们从一个多播域传递到另一个多播域中.图 1 显示了 ASRM 体系结构的概况.



聚集服务器, 核心数据服务器, 数据服务器, 多播

Fig.1 ASRM architecture

图 1 ASRM 体系结构

1.1 创建会话

在 ASRM 中,任何一个会话都有一个创始者 U_c ,即首先注册多播会话的应用.当一个应用需要创建一个会话 G 时,它将首先与其所在多播域(设为 A_c)中的聚集服务器 RS_c 联系. U_c 向 RS_c 发送一个 Request_create(Name(G)). RS_c 将在 A_c 内检查 Name(G)是否冲突,如果没有,将为 G 在 A_c 中分配一个局部的 IP Multicast 地址 $mip_addr_A_c(G)$ 和端口号 $mip_port_A_c(G)$.同时, RS_c 将创建一个数据服务器 DS_c .因为在多播会话中 DS_c 有着重要的作用,我们称它为核心数据服务器.此后, RS_c 发送 Reply_create_succ($mip_addr_A_c(G)$, $mip_port_A_c(G)$)给 U_c .这样 U_c 和 DS_c 之间就可以通过分配的多播地址和端口来通信.在 RS_c 中为 G 注册的 Entry(G)内容见表 1.

Table 1 The registration of a session in rendezvous server

表 1 在聚集服务器中,一个会话注册的信息项

Entry(G)	$ID(G)=\{Name(A_c):Name(G)\}$ $Core_Addr(G)=\{IP_Addr(DS_c),IP_Port(DS_c)\}$ $(Creator_Addr(G)=\{IP_Addr(U_c),IP_Port(U_c)\})$ $Description(G)=\text{"The description of the session"}$
--------------	--

1.2 加入会话

每一个客户都是通过本地的数据服务器加入到多播会话之中,而所有的数据服务器都通过 TCP 连接组织成为一棵树(Tree(G)).尽管 Tree(G)的构造和多播路由的构造有着许多相似之处,但是它应该具有以下 3 个本质特征:(1) G 的成员是动态的.这就是说允许任意的 DS ,在任意的时候加入或离开 G ,从而 Tree(G)也是动态的.(2) Tree(G)的构造是分布的.由于我们的目标是在广域中实现多播,所以就不能有理由认为存在某个节点可以对全局的情况都能够了解,因此节点加入 Tree(G)的行为只能通过本地的测量信息所决定.(3) 我们坚持 ASRM 应处于应用层,这样做的好处在于,使得 ASRM 具有最大的灵活性和适用性,因此,所有的测量都是端对

端的。

$Tree(G)$ 的构造是不断将新成员加入到树中。加入树算法的基本思路就是通过端对端测量找出与加入成员最近的树上成员,然后在两者之间建立一条连接。设在 A_1 中的应用 U_1 要加入到 G 中,则它首先与本地的聚集服务器 RS_1 联系,向 RS_1 发送一个 $Request_join(ID(G))$ 的消息。在接到这个消息之后,如果本地没有为 G 服务的数据服务器,则 RS_1 就根据 $ID(G)$ 查找出 G 的注册信息*, 然后再创建出一个本地的数据服务器 DS_1 。 DS_1 再向核心服务器 DS_c 发送一个 $Request_join_tree(Addr(RS_1))$ 请求。如果这个请求被允许,则 DS_c 会返回一个 $Reply_join_tree_succ(List(DS_all))$ 。 $List(DS_all)$ 是一个包括 $Tree(G)$ 上所有数据服务器的列表。 DS_1 就可以找出与它最近的一个数据服务器,设为 DS_2 , 然后再向 DS_c 发送一个 $Join_tree(Addr(DS_2))$, 接着再与 DS_2 之间建立一个 TCP 连接。这样, DS_1 就加入到 $Tree(G)$ 中形成新的 $Tree(G)$ 。

1.3 数据传输和差错恢复

当数据服务器之间的传播树生成之后,ASRM 的数据传输模式是比较直接的。在一个多播域中,我们可以通过现有的可靠多播协议来传送数据,例如 SRM(*scalable reliable multicast*)^[2]等。数据服务器从多播域中接收数据源发送的数据,然后再通过 TCP 连接向其他的数据服务器转发。而当数据服务器从一个 TCP 连接上获取数据后,则会向其所在的多播域中转发该数据,同时,还将向其余的 TCP 连接传送。

在 ASRM 中,我们仍然采用端对端可靠的概念,因为尽管 TCP 连接提供了可靠的数据传输,但是我们不能假设数据服务器具有无限多的缓冲区,在一些情况下,数据服务器必须丢弃数据。另一方面,即使不发生丢失数据,后加入者也可能需要请求过期的数据。因此,必须具备端对端数据恢复的机制。当接收方发现一个数据丢失的时候,将发送一个数据恢复请求。本地的数据服务器接收到这个请求后,将首先在自己的缓存中寻找。如果存在相应的数据,则进行本地重传,否则,这个请求就沿着数据源的方向向上一数据服务器转发。依此类推,直到数据被修复为止。

1.4 退出树和关闭会话

在一个多播域中,用户退出的方法是与所使用的可靠多播协议相关的。如果使用 SRM,则用户可以直接关闭应用程序。本地的数据服务器会定时地向多播域发送一个 $request_alive$ 消息,收到这个消息的应用则会应答一个 $Keep_alive$ 消息。当数据服务器在相当长的时间内未获得 $Keep_alive$ 消息,则在本地多播域中已经没有用户了。

如果数据服务器检测到在多播域中已经没有用户,并且为传播树中叶子节点时(度为 1),则该数据服务器可以退出传播树。设 DS_i 要退出传播树,则 DS_i 将向核心数据服务器 DS_c 发送一个 $Request_quit$ 消息,然后再退出。核心数据服务器收到 $Request_quit$ 消息后,将把 DS_i 从列表中删除。

当一个多播会话的核心数据服务器关闭的时候,则该多播会话被终止。当出现以下情况之一时,则核心数据服务器被关闭:(1) 收到了一个授权的 $Request_close_session$ 消息,通常由 U_c 发出;(2) 如果核心数据服务器是“孤独”的,也即没有任何数据服务器或是终端用户与其相连;(3) 如果在指定的时间 T 内在多播树上没有数据被传送。

1.5 应用层分帧和数据变换

ASRM 是基于应用层分帧原则(application level framing,简称 ALF)^[5]的,因而在传播树上传送的数据是应用层的数据报,或称应用数据单元(application data unit,简称 ADU)。对于媒体对象的变换可以从两个方向进行:一是质的方向,另一是量的方向。对于不同类型的媒体进行的变换操作在文献[6]中进行了总结。数据服务器将根据用户的意愿和系统的实际条件自主地决定采用的变换方法和过程。在 ASRM 中,我们主要考虑用户的等待时间要求和质量要求。由于一个数据服务器所服务的用户往往有许多个,因此,影响数据变换的不是单个用户意

* 我们假设任一多播域的聚集服务器都可以通过某种方式获得会话 G 的注册信息,这可以通过聚集服务器之间的协议实现,就像 DNS 或 Mbone 中的 sdr 服务一样。

愿,而是所有受影响用户(所有多播域中的用户)的群体意愿.我们认为局域范围内网络的状况远远好于广域范围,即 $BW_{wide} \ll BW_{local}$, 可以忽略局部网络中的延时,故对于传播网络中的每一数据服务器有

$$\tilde{T}_{ds} = \sum W_i T_{expi} / \sum W_i \quad (1)$$

其中 \tilde{T}_{ds} 为该数据服务器的群体等待时间,而 T_{expi} 为该多播域中第 i 个客户的希望等待时间.而 W_i 为第 i 个客户的权重.类似地,有:

$$\tilde{Q}_{ds} = \sum W_i Q_{expi} / \sum W_i \quad (2)$$

其中 \tilde{Q}_{ds} 为数据服务器的群体期望质量. $\tilde{Q}_{ds} \leq 1$.

设有传播子树如图 2 所示.我们希望决策出在中间数据服务器 i 上应采用的数据变换策略.设 D 为 i 下行树上的任一数据服务器,其群体等待时间为 \tilde{T}_d .考察从数据源到 D 的传播路径 $P(v_0, v_1, \dots, v_n)$, 则其群体期望等待时间在路径 P 上按照加权平均的方式分配:

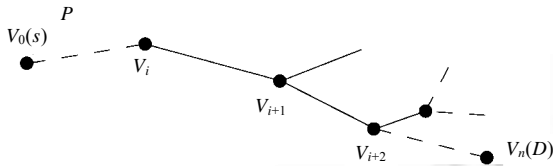


Fig.2 A distribution path in ASRM
图 2 ASRM 上的一段传播路径

$$\tilde{T}_d(v_i, v_{i+1}) = \rho_i \frac{1}{B_{v_i, v_{i+1}} + 1} \tilde{T}_d / \sum_k (\rho_k \frac{1}{B_{v_k, v_{k+1}} + 1}) \quad (3)$$

其中 $\tilde{T}_d(v_i, v_{i+1})$ 为 D 在路径 (v_i, v_{i+1}) 上期望耗费的时间,而 $B_{v_i, v_{i+1}}$ 为路径 (v_i, v_{i+1}) 上的带宽. ρ_i 称为负载因子, $\rho_i \geq 1$, 它表示了 v_i 的负载情况,当 v_i 处于轻载情况时, $\rho_i = 1$.我们用式(4)来确定在 (v_i, v_{i+1}) 上的期望时间:

$$\tilde{T}(v_i, v_{i+1}) = \sum_d \tilde{T}_d(v_i, v_{i+1}) W_d / \sum_d W_d \quad (4)$$

此外,我们取路径 (i, j) 上的期望质量为 $\tilde{Q}(v_i, v_{i+1}) = \text{Max}(\tilde{Q}_d)$.

设 V_i 的进入媒体对象为 m , 大小为 $S(m)$, 经过变换 α 后转发给 V_{i+1} . 变换后的对象为 $\alpha(m)$, 大小为 $S(\alpha(m))$. 变换延时为 $D(\alpha)$. 则应该有

$$D(\alpha) + \frac{S(\alpha(m))}{B_{v_i, v_{i+1}}} < \min(\tilde{T}(v_i, v_{i+1}), \frac{S(m)}{B_{v_i, v_{i+1}}}) \quad (6)$$

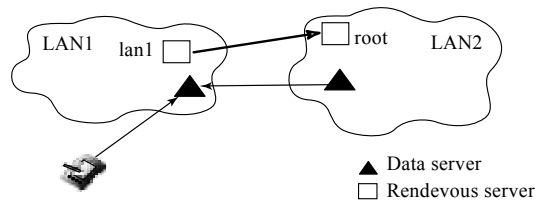
式(6)表示了数据服务器进行数据变换所必须遵循的条件,称为决策不等式.根据决策不等式,我们还需要进一步确定需要进行的具体变换方案.这可以通过对于变换空间的搜索来实现.搜索的目的是获得一个变换的操作序列,使得能够满足式(6),并使得变换后的数据质量 $Q(\alpha(m))$ 与 $\tilde{Q}(v_i, v_{i+1})$ 尽可能接近.由于篇幅的原因,搜索算法就不再赘述,可参见文献[7].

在实际实现时,我们可以采用基于局部信息的贪心算法.如式(7),设 m 到达 V_i 的累计延时为 d_0 , 且数据服务器均为轻载,即 $\rho_i = 1$, 我们令 $W_d = 0, \forall d, \text{Hops}(d, V_{i+1}) > 1$, 即忽略掉远处的数据服务器对数据变换的影响.

$$\tilde{T}(v_i, v_{i+1}) = ((\tilde{T}_{v_{i+1}} - d_0) W_{v_{i+1}} + \sum_d W_d \frac{B_{d, v_{i+1}} (\tilde{T}_d - d_0)}{B_{v_i, v_{i+1}} + B_{d, v_{i+1}}}) / (W_{v_{i+1}} + \sum_d W_d), \forall d, \text{Hops}(d, v_{i+1}) = 1 \quad (7)$$

2 评 测

我们完成了一个 ASRM 应用: Artemis 远程授课系统^[8]. Artemi 实现了一个共享的多媒体板. 在这个多媒体板上可以显示 HTML 文档, 并且可以动态增加文字、图形以及图像对象. 目前的演示环境建构于两个局域网(每一个都是一个多播域)上. 为了演示自适应的传输, 我们也将一个 HP 公司的掌上电脑通过拨号网络连入多播会话, 如图 3 所示.



数据服务器, 聚集服务器.

Fig.3 Test setup

图 3 ASRM 测试环境

演示环境中的两个局域网络都直接连接到清华大

学校园网上, 两者之间的实测传输率为 1.6Mbps. 我们比较采用 ASRM 和基于 IP Multicast 方式的 SRM 传输数

据时,在不同发送率的情况下的丢包率和端到端延时¹.测试数据包的典型大小为 1 024 个字节,突发传送时间为 20 秒.为了增加显示度,我们在每一个局域网中都使用了一个干扰源,干扰源总是向另一局域网中发送固定数率的比特流(1.28Mbps).

图 4 显示了丢失率随发送率变化的情况.可以看到,随着发送率的增加,使用 SRM 的丢包率显著地增加了,而使用 ASRM 传送数据则丢包率几乎保持为 0.其中的原因是十分明显的,这是因为 SRM 中没有进行流量控制,因此当发送率显著增加时,就会发生拥塞,从而导致丢包率的提高.而在 ASRM 中,由于利用了 TCP 的流量控制,因而就可以有效地降低丢包率.图 5 中显示了端到端延时的变化情况.可以看到,当发送数据率较低时,利用 SRM 和 ASRM 具有相同的延时,而当发送率逐渐提高,超过了网络实际带宽时,SRM 就会发生拥塞,这时接收方就必须发送 NACK 消息来请求重传丢失的数据包,从而增加了端到端的延时.

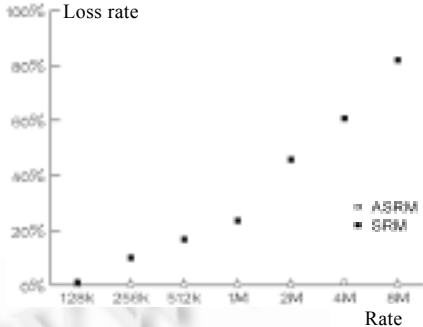


Fig.4 Comparison of loss rate using global SRM vs. ASRM

图 4 ASRM 和 SRM 随发送率增加丢包率变化情况

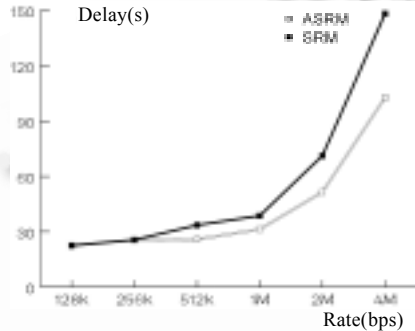


Fig.5 Comparison of end-to-end delay using global SRM vs. ASRM

图 5 发送接收特定数量的测试数据包总的端到端延时

图 3 中掌上电脑通过拨号网络接入.拨号线路的带宽为 28.8kbps,实际有效带宽为 24.8kbps.测试图像为 JPEG 格式,大小为 114K 字节.采用的变换方法为对原测试图像进行缩放.我们采用 Han^[9]的方法对缩放操作的时间和结果大小进行线性预测(如图 6 所示).图 6(a)中黑线是结果大小的预测曲线,而浅色线是实际变换后图像的大小.图 6(b)中黑线给出了 JPEG 图缩放操作所需时间的预测曲线.

而浅色线是实际所需的时间在测试中,我们将用户期望质量设置为 1,也就是说希望系统在满足时间要求时给出最佳的质量.

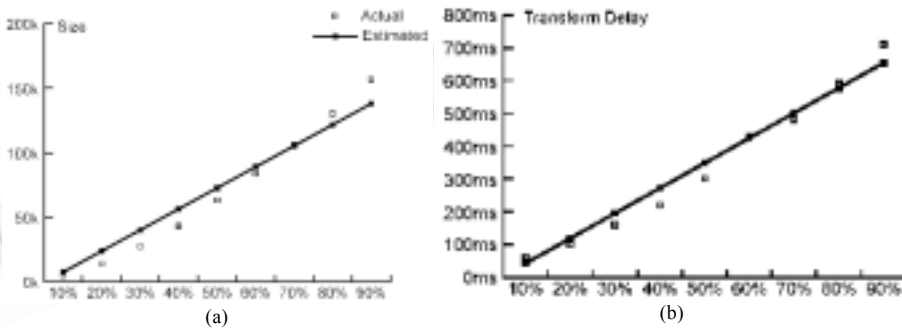


Fig.6 Time and size prediction of JPEG image scaling operation
图 6 对 JPEG 图像进行缩放的操作时间和结果大小的线性预测

图 7 给出了随着用户期望等待时间的变化,采用不同尺度进行图像缩放时的端到端延时和图像质量.从图 7 中可以看出,系统可以根据用户要求的期望等待时间决定相应的变换策略,从而满足用户的要求(如图 7(a)所

¹ 在这个测试中的 SRM 是我们的一个简单实现,没有流量控制和局部数据恢复.为公平起见,ASRM 没有采用数据变换.

示),这是通过牺牲一些质量而获得的(如图 7(b)所示).图 7(a)中的黑色线给出的是预测的端对端延时,而浅色线则是实际的端对端延时.图 7(b)为图像质量的变化曲线.图像的质量是这样确定的:图像质量 = $\frac{\text{缩放后的图像面积}}{\text{原图的面积}}$, 图像质量 < 1. 我们可以看到,当用户的期望等待时间大于 40s 时,系统就不再进行数据变换

了.因为这时用户的期望等待时间已经大于传输原图的时间(约 36s),因此就将原图像直接传输.

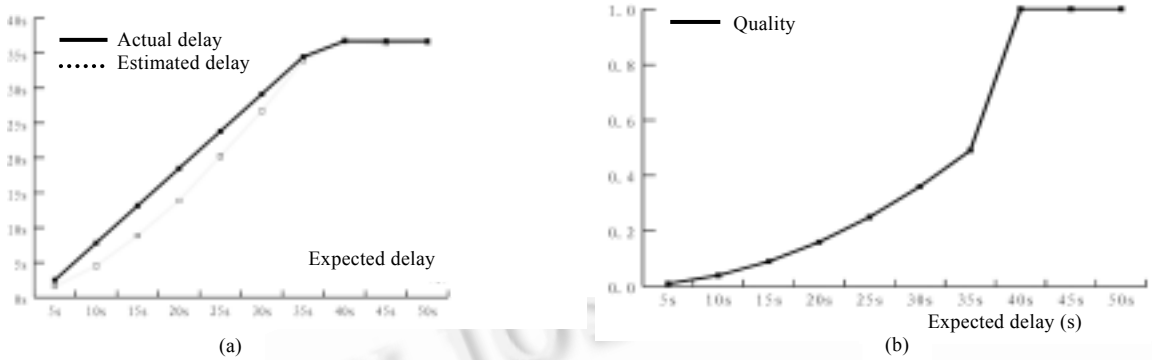


Fig.7 With different expected delay, transcoder outputs images

图 7 随着用户期望等待时间的变化,而采用不同尺度进行图像缩放时的端对端延时和图像质量

3 相关工作

尽管一些端系统多播协议,例如 Yallcast^[10],也是利用覆盖网络实现多播,但是 ASRM 与它们有两点重要的区别.首先在端系统多播中,传播树上的每个节点都是一个客户终端,以此客户终端设备既是数据的接收者,又是数据的转发者.与此不同,在 ASRM 中,数据服务器将作为一种特殊的应用层服务,因而可以要求数据服务器具有更好的性能和可靠性.另一方面,ASRM 的数据服务器对于数据的应用层语义有所了解,因而可以根据用户的意愿进行数据变换,从而解决可靠多播中的异构性问题.RMX^[11]是一个在可靠多播环境下进行自适应的应用层网关,并在最近扩展成为一个在 Internet 上的传播结构^[12].与 ASRM 一样,RMX 也试图解决可靠多播中的异构性问题,但是它仍然试图延续 IP Multicast 的所有概念和模式,因此在多播会话的管理、传播树的生成以及自适应策略上仍然是很不完善的.

4 总结

本文提出了一个在现有的 Internet 基础上的语义可靠的自适应多播的体系结构(ASRM).ASRM 的目标是提供应用层语义的可靠的、同步的多播数据传输.ASRM 采用了一种混合 IP 单播和多播的方法来实现多点数据通信.这种做法的好处在于:首先,使得多点通信能够在现有的 Internet 上实现;其次,IP Multicast 可以在局域网范围内使用以提供效率;再次,这种“分而制之”的方法使得在广域范围内更加健壮,伸缩性更好.

在每一个多播域中设有聚集服务器,它是多播域名称和多播会话的管理者.我们采用了一种简单、自然的方法来命名多播会话.这种命名机制在广域中具有更好的伸缩性.

ASRM 建立在 ALF 上,因而可以利用应用层语义进行数据变换.数据变换的目的在于通过质量而换取效率的提高.数据服务器可以在转发的过程中,根据变换模型、变换规则 and 用户意愿自主地选择变换方法和过程,为不同的用户群体提供适合他们需要的数据表现.

References:

- [1] John, C., Lin, S.P. RMTP: a reliable multicast transport protocol. In: Proceedings of the IEEE Infocom'96. San Francisco, CA: IEEE, 1996. 1414~1424.

- [2] Sally, F., Van Jacobson, McCanne, S., *et al.* A reliable multicast framework for light-weight sessions and application level framing. In: Proceedings of the ACM SIGCOMM'95. Boston, MA: ACM, 1995. 342~356.
- [3] Speakman, T., Farinacci, D., Lin, S., *et al.*, PGM reliable transport protocol specification, 1998. <http://www.globe.com.net/ietf/draft/draft-speakman-pgm-spec-02.html>.
- [4] Stephen, E.D. Multicast routing in a datagram internetwork [Ph.D. Thesis]. Stanford, CA: Stanford University, 1991.
- [5] Clark, D.D., Tennenhouse, D.L. Architectural considerations for a new generation of protocols. In: ACM SIGCOMM'90. Philadelphia, PA: ACM, 1990. 200~208.
- [6] Smith, J.R., Mohan, R., Li, Chung-sheng. Transcoding Internet content for heterogeneous client devices. In: Proceedings of the IEEE ISCAS. Monterey, CA: IEEE, 1998. 599~602.
- [7] Liao, Chun-yuan, Shi, Yuan-chun, Xu, Guang-you. AMTM: an adaptive multimedia transport model. In: Proceedings of the SPIE International Symposia on Voice, Video, and Data Communication 2000. Boston, MA: SPIE, 2000. 141~149.
- [8] Tan, Kun, Liou, Shih-ping, Shi, Yuan-chun, *et al.* Artemis: providing live course for distant learning. In: Proceedings of the 4th International Workshop on CSCW in Design'99. Compiègne, 1999. 253~256.
- [9] Richard, H., Pravin, B., *et al.* Dynamic adaptation in an image transcoding proxy for mobile Web browsing. IEEE Personal Communications Magazine, 1998,56:8~17.
- [10] Francis, P. Yallcast: extending the Internet multicast architecture. 1999. <http://www.yallcast.com/docs/ycHtml/htmlRoot.html>.
- [11] Chawathe, Y., Fink, S.A., McCanne, S., *et al.* Brewer. a proxy architecture for reliable multicast in heterogeneous environments. In: Proceedings of the ACM Multimedia. Bristol: ACM. 1998. 151~159..
- [12] Chawathe, Y., McCanne, S., Brewer, E.A.. RMX: reliable multicast for heterogeneous networks. In: Proceedings of the IEEE Infocom 2000. Tel Aviv: IEEE, 2000. 795~804.

An Application-Level Semantic Reliable Multicast Architecture for the Internet*

TAN Kun, SHI Yuan-chun, LIAO Chun-yuan, XU Guang-you

(Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China)

E-mail: cohen@media.cs.tsinghua.edu.cn

<http://www.tsinghua.edu.cn>

Abstract: Reliable multicast is a useful network service but challenged by the Internet heterogeneity and the nonsupport of IP Multicast in wide-area Internet. In this paper, an application-level semantic reliable multicast (ASRM) is presented, which realizes application-level semantic reliable multicast data delivery. ASRM uses a hybrid way that combines unicast delivery with multicast delivery instead of using global IP multicast as the basis. ASRM provides a simpler and more natural way to name and locate the sessions. During the data forwarding, ASRM can use application-defined semantics to transcode content according to the transcoding models, rules and user' preferences. Thus, it can cope with the heterogeneity problem very well. ASRM is suitable for wide-area, limited scale, reliable multicast applications.

Key words: semantics reliability; transcoding; heterogeneity; multicast

* Received September 14, 2000; accepted February 16, 2001

Supported by the 985 Key Projects of Tsinghua University of China under Grant No.985TH-22-06