

计算机会议系统的网络模型及分析_丁伟+潘建平+顾冠群+刘炳华.txt

计算机会议系统的网络模型及分析

丁伟 潘建平 顾冠群 刘炳华
东南大学计算机科学与工程系
南京 210096

摘要

本文首先介绍计算机会议系统的一般结构及与窗口系统的关系，并提出一种层次结构的计算机会议的网络模型，该模型由‘逻辑层’、‘影射层’和‘支撑层’组成。然后对影射层中不同的影射方式对系统性能的影响进行详细的分析和讨论，最后是用‘蒙特-卡罗’方法对各影射方式进行仿真比较的数值结果。

关键词：群通信 计算机会议 计算机会议模型 信息传播模型

一 引言

计算机会议是目前日益活跃的研究课题之一，它是群通信应用的一个具体实例。计算机会议有可分为异步和同步两大类[1]，前者主要提供信息资源的共享服务，对它的研究起步较早；后者是目前人们所理解和熟悉的计算机会议系统，它在功能上类似与‘会议室会议’，其最大特点是‘同等可视’(WYSIWIS-What You See Is What I See)[1, 4]，借助计算机网络、工作站的图形界面和多媒体服务，它可以方便地为医疗(远程会诊)、CAD和教育等行业和领域提供服务。本文的所有讨论均以同步会议系统为对象进行。首先讨论计算机会议的主要功能和以X-WINDOW为代表的窗口系统以及它们之间的相互关系和影响，随后提出了网络环境下的一个计算机会议系统的三层分布式模型，最后是基于该模型的一些分析。

二 计算机会议的主要功能和窗口系统

一般来说，计算机会议系统总是由‘发起人’通过指定‘参与者’和开会时间来发起一个会议。会议系统应至少能提供‘白板’(Whiteboard)和‘指针’(Telepointer)等会议资源，同时可对这些资源及‘主持人’、‘发言者’以及白板和指针的使用权限进行有效的管理和控制。另外，这种会议系统还该能够将某个参与者的应用窗口(包括静态的图纸、照片和动态的图形等)在一定的控制下变为会议的应用窗口，使之能同步地出现在每个与会者的屏幕上。

图1 窗口系统及在计算机会议系统中的变形

另一方面，随着工作站的普及，窗口系统以其良好的性能和界面已广泛为人们接受和认可。目前几乎所有的计算机会议系统均是在窗口系统上实现的。这类以X为代表的窗口系统一般均基于“客户—服务器”模式。服务器维护各类系统资源同时为客户进程提供服务。服务器与客户进程之间的交互一般通过消息传递完成，具体结构如图1a所示[2]。

在用窗口系统实现计算机会议时，需要同一客户向所有会议参与者的服务器提出相同的请求，以便为各与会者显示相同的会议内容，这是‘同等可视’的基本要求。遗憾的是，在这些窗口协议设计的初始，主要考虑的是对普通单个应用的支持，而未考虑单个客户进程对多个服务器的情况。为此，大多数会议系统采用了折衷的方案，即在客户与服务器之间逻辑地插入一个新的‘消息转发进程’，该进程一方面传递客户与服务器之间交互的消息，另一方面通过与其同等转发进程的交互，将客户对本地服务器的请求转发到其它系统的服务器，如图1b所示。从服务器的角度来看，它将原有

的客户和消息转发进程看成一个“虚拟客户”，它们与这些虚拟客户之间的交互与图1a完全相同；为了保证‘同等可视’，‘消息转发进程’之间也需要可靠的通信支持，此外，对于某个确定的‘消息’而言，必须有且仅有一个‘消息转发进程’与有关的‘真实客户’联系，以确保整个会议系统的正常运行[3]。

三 计算机会议的网络模型

计算机会议系统是一个庞大的系统，对于这类系统来说，若采用TCP/IP、OSI等网络协议系统常用的方法，对其整体功能进行纵向分割，则无论是从系统功能分析的角度，还是从系统的实现角度，都会带来很大的方便。

我们认为在网络环境中实现的计算机会议系统从结构和功能来讲可以划分成三个层次，分别是逻辑层，影射层和支撑层。

对逻辑层而言，从功能角度来看，它对应于用户世界中真实的'会议室会议'。一个计算机会议包括"与会者"和"会议资源"等要素，与会者又可分为发起者(一个或多个)、会议资源(比如白板，焦点指针)的拥有

者以及会议的其它参与者等不同身份。发起人负责这个会议的起始、进展、动态扩充和结束等事务；资源拥有者的资格可以在会议参与者之间主动地或通过发起人被动地进行传递；发起人对与会议有关的事宜有决定性的发言权。此外，从逻辑层的功能角度看，计算机会议系统与真实世界中的会议室会议一样，是一个集中式控制系统，同时每个逻辑层中的要素都找到会议室会议中相关的事件与之对应。从实现角度来看'白板'和'指针'的实质就是一个基于窗口系统的应用进程，因其在会议系统中的作用而分别具有一些特殊的性质，如'指针'窗口必须出现在所有窗口的前方，同时只能对其进行'移动' (MOVE) 操作等。显然，在实现中逻辑层要求底层能够提供多址投递功能。

图2 计算机会议的网络模型

对支撑层而言，它是整个计算机会议系统实现的基础，对应着实际的操作平台、网络环境和应用协议。从网络的本质来讲，支撑层是一个分布式结构的系统。尽管目前的许多网络协议在底层支持数据的广播和多址投递，但在高层却只能提供面向连接的服务而无法提供可靠的广播和多址投递服务，这也是影射层必须存在的原因。

影射层帮助会议信息跨越逻辑层而到达支撑层。从功能角度上讲，它能够在不提供广播和多址投递的支撑层之上向逻辑层提供一个'虚拟'的广播网络。在实现上，根据具体情况可以用多种方式实现影射层，图1.b中的消息转发进程就是典型的一种。此外，不同的影射方法(转发规则)会带来不同的系统性能，有关这方面的问题将是本文以下部分所要讨论的问题。

四 转发规则分析

本小节主要讨论'消息转发进程'以各种不同的转发规则转发会议信息时，对会议系统性能的影响。在这里'性能'主要指的是信息扩散时间，即某个会议信息从消息源向第一个与会者发送开始至最后一个与会者收到该信息为止的时间延迟。由于计算机会议系统是一个同步系统，因此该扩散时间应是其最重要的性能之一。

虽然ATM和XTP等都可以支持多址投递，甚至OSI也已开始了这方面的研究，但目前几乎所有的会议系统均是在支持'点到点'的通信协议(如TCP/IP)上实现的，因此本文以下的讨论均以此为背景进行，同时为了分析方便起见还设会议环境为若干(m)个互连局域网构成的封闭环境。互连手段为路由器或其它低速通信设备；封闭环境指的是会议消息所通过的网络设备(如路由器)和子网均属于会议环境。在这样的假设下最好的情况为全连通，即所有子网均可直接相连；最坏的情况为简单

连通(以下简称单连通)，即从消息源到达最远的子网需要经过 $m-1$ 级路由器之间的传递。最后设消息在局域网中的延迟为 d ，通过路由器则为 rd ，显然， $r>>1$ 。与会者人数，包括发起人，发言者，资源拥有者等，为 n 。图3给出了一个 $m=3$ 的示意配置情况。此外，不失一般性地假设消息源M在A网络，所有消息均由消息源负责发送。

图3 集中(a)和层次(b)影射

4.1 集中影射方法

这种影射方法假设M将逐个向位于m个网络中的n-1个其它参与者传递需要扩散的消息。首先假设每次传递最多只需通过一个路由器，同时设参与者均匀分布在各网络，则每个参与者与发起人位于相同网络的可能性为 $1/m$ ，位于不同网络的可能性为 $(m-1)/m$ ，此时扩散时间的数学期望为：

由此可以看出在此假设下扩散时间复杂性与m无关。

对一般的情况，设每次传递所需通过k个路由器的概率为 p_k ， $k=0..m-1$ ，则上述数学期望为：

进一步，若 $p_k=1/m$ ，则

4.2 层次影射方法

在集中影射方式下，大量的重复消息反复地通过路由器，从而造成很大的网络延迟。显然，减少这类延迟最简单的方法是在每个局域网的与会者中选出一个‘消息转发代理’，它负责从真正的消息源处获得消息然后再依次转发给本网中的与会者。在图3b所示的实例中，设网络A，B，C用户数分别为 n_a 、 n_b 和 $n_c=n-n_a-n_b$ ，则扩散时间为：

对一般情况，全连通时：

单连通时：

设 $n=n_i$ 为各子网中的与会者数目，在此条件下求扩散时间的数学期望时，这些 n_i 的获得构成有关自然数n的‘背包问题’，由于该问题目前尚没有公式解，因此在这里无法写出上述扩散时间的数学表达式。但是，通过‘蒙特-卡罗’仿真可以看出层次影射要明显优于集中影射，具体结果在表2中。

4.3 分布影射方法

当 $m=n$ 时，即n个与会者分布在n个子网中时，无论是最好还是最坏情况，层次影射方法与集中影射方法具有相同的扩散时间。此时可以将‘存储-转发’的思想引入消息传播模型，即各子网中的代理不但负责向本网的与会者转发消息，同时也通过路由器替‘消息源’向其它代理转发消息。在这里将这种方法称为分布影射方法，它能够提高网络中消息扩散的并行度，当m较大时具有较好的传播效率。

图4 分布影射方法

图4给出了一个分布影射的实例。图中各父节点由左至右地向子节点扩散消息，各节点的记号是 $i(j)$ ， i 是节点的编号， j 表示相应节点在 $j*rd$ 时刻获得消息，节点0是消息源。由该图可以看出，在这种影射方式下，消息的扩散过程是一棵树，可将其称之为‘(k阶)扩散平衡树’。可以证明：

1 k-阶扩散平衡树的接点个数为 2^k ；

2 若 $(k-1) < \log_2(m) \leq k$ ，则扩散时间为 $j*rd$ 。此外，‘扩散平衡树’的结构具有递归性，可以方便地动态构造。图4中各节点的编号(i)就是该点在树的构造过程中进入扩散结构的次序。这种动态构造规则还具有平衡性、对称性和稳定性。平衡性表现为动态过程的任何时刻进入扩散模型的节点都不会改变在其之前进入模型的节点之间的扩散关系；对称性指的是新节点进入模型的顺序以及与周围节点的扩散关系在任何时刻都可以从模型业已存在的一个对称节点处预先获得；稳定性则表示后进入模型的节点有可能比先进入的节点先获得消息。有关这方面的详细讨论作者将另文给出。

五 转发规则比较

从以上的分析可以看出在一般情况下，层次影射优于集中影射，而分布影射又优于层次影射。为了定量地表明三者之间的差距，下面给出的是用‘蒙特-卡罗’方法仿真6个用户在6个网络中随意配置时不同影射方法的扩散时间比较。表1是用‘皮尔逊相关统计’对用‘随机数发生器’产生的各与会者数据分布之间的独立性的检验结果。可以看出，该表中各与会者的分布除了对角线完全自相关外，其余高度独立。表2中给出三种影射方法的比较结果。

表1 分布数据相关检验

表2 三种影射方法的比较

六 结论

本文首先提出了一个由‘逻辑层’、‘影射层’和‘支撑层’构成的计算机会议的网络模型。随后着重分析和讨论了影射层中的消息传播问题。三种传播方式—‘直接影射’、‘层次影射’和‘分布影射’，从性能角度看是逐个递增的，但同时系统在管理和维护方面所付出的代价也是递增的。此外，文中所有的分析均以各子网间的传递时间相等(等于rd)为基础进行，这是一种比较理想的情况。

然而，无论如何，通过对层次结构的计算机会议网络模型的研究，以及对传播模型的分析和优化，使得‘点-点’通信结构内部的并行性质得到了利用。理论推导和仿真分析表明这种并行特性对系统性能的影响，特别是在用户数目较大的时候，是显著的。

七 参考文献

- [1] Amane Nakajima. Telepointing issues in desktop conferencing system computer communication. vol. 16 no. 9, 1993
- [2] Stephen Gildea. X Window System Version 11 Release 6. Release Note X Consortium, 1994
- [3] Goopeei Chung. Dynamic Participation in a computer- based conferencing system Computer Communications. Vol. 17 no. 1. 1994
- [4] Darren Reed. A Discussion on computer Network Conferencing. Request for Comment RFC 1324 IETF
- [5] Spectragraphics. Team Conference User's Guide. Version 1. 2, Spectragraphics co. 1993
- [6] Ngoh L. H., Miticast Support for Group Communication. Computer Network and ISDN Systems, 1991, (22): 165-178
- [7] J. F. Nunamaker et al. Electronic Meeting System to Support Group Work. COMMUNICATION OF ACM Vol. 34 No. 7, 1991
- [8] T kirsche et al. Communication support for cooperative work. Computer Communication. Vol. 16 N09 1993. pp(594-602)

Networking Model & Analyses for Computer Conference System

Ding Wei, Pang Jian-ping, Gu Guan-qun & Liu Bing-Hua
Computer Department, Southeast University
Nanjing 210096

计算机会议系统的网络模型及分析_丁伟+潘建平+顾冠群+刘炳华.txt

The paper introduces the common architecture of Computer Conference(CC) and its relationship with Window systems firstly. A three layers networking model for CC is given out then which consist of logical layer, mapping layer and infrastructure layer. After that, some mapping ways for mapping layer are proposed, discussed, analysed and simulated by 'Monte-Carlo' method.