

基于 P2P 网络架构的 SIP 视频会议系统设计

唐余亮, 王玮玮, 姚彦

(厦门大学信息科学与技术学院, 福建 厦门 361005)

摘要: 会议初始化协议(SIP)凭借其简单、易于扩展等优点,已成为未来网络发展的重要应用协议。它采用 Client/Server(C/S)的结构体系,当系统业务量增大时,系统容易达到性能瓶颈,成为其技术发展的障碍。基于分布式点对点(P2P)网络的去中心化特征,正好弥补了 SIP 的这一缺陷。因此,首先提出了一种基于 P2P 网络架构的 SIP 视频会议系统模型,将现有 SIP 视频会议系统的 C/S 架构改进成为 P2P 的混合网络体系结构。这既发挥了 P2P 无中心的优越性,又体现了 SIP 的 C/S 模式的优点。其次,阐述了该系统主要功能模块以及相关程序设计,最后介绍了该系统的主要信令工作流程。

关键词: SIP; P2P; DHT; SIP 视频会议

中图分类号: TP 393.02

文献标识码: A

文章编号: 0438-0479(2009)04-0519-05

由 IETF 制定的会议初始化协议(Session initiation protocol, SIP),用于生成、修改和终结一个或多个参与者之间的会话,ITU-T,3GPP 等国际化都将定为下一代网络中多媒体通信系统的核心协议。但由于 SIP 协议的应用依赖于 SIP 代理服务器的 C/S 架构,对于大型的视频会议系统,当业务量增大时,系统容易达到性能瓶颈。Peer to Peer(P2P)网络是一种基于对等的计算模型和基于对等的应用层重叠的网络,其中的参与者既是资源提供者(Server),又是资源获取者(Client)。

与目前网络中占据主导地位的 C/S 体系架构相对比,由于它没有中央服务器并具有自己管理网络的特性,P2P 系统天生就拥有高扩展性、健壮性和高容错性。P2P 技术的应用使得 SIP 视频会议系统控制从集中向分布演化,特别是服务器的分布化,克服了业务节点集中造成的资源瓶颈,大大增强了系统的扩展性和业务容量。

多媒体应用业务正成为未来几年新的业务增长点,其中视频会议系统在许多领域的应用越来越广泛。在 P2P 和 SIP 的领域,目前比较多的研究都集中在 P2P 对数据业务的支持,以及对 SIP 语音业务的应用,对视频会议的研究不多,还处在理论探索阶段,尚未形成完整的体系结构和规范。本设计研究基于 SIP 协议控制的 P2P 视频会议系统模型,设计了相应的控制信

令流程,并进行了实际的程序设计和开发,结果表明,在实际应用上有一定的作用。

1 基于 P2P 的 SIP 视频会议系统

1.1 系统模型

为了支持和控制大型的 SIP 视频会议系统中各终端用户节点 UA,需将其网络划分成多个控制域,每个域内包含一个 SIP 代理服务器(Proxy server, PS)和一个会议服务器(Conference server, CS)逻辑实体,含有 PS 和 CS 的节点称为超级节点 PN^[1]。PN 一般由运营商提供及维护,从而保证会议系统的安全性。采用 P2P 网络架构将所有域中的 PN 节点构成一个 Chord 环,共同维护一个巨大散列表。散列表被分割成不连续的块,每个 PN 节点保存并管理一个属于自己的散列表块。系统使用 Chord 的分布式哈希表^[2](Distributed hash table, DHT)算法来维护该散列表,如图 1 所示。用户节点 UA 平时连接在这些超级节点 PN 上,通过 PN 来实现注册、定位。网络中节点间的信息交互(如节点发现、用户注册等)都通过 SIP 的信令交互来实现。

PN 分为两层,上层是 SIP 层,处理标准的 SIP 信令;下层是 P2P 层,使用特定的机制(本文的设计选用 Chord 协议)互联各个 PN 的 P2P 层并维持它们之间的联系。P2P 层提供给 SIP 层的应用程序编程接口(API)只有函数 find_responsible_pn(user),该函数返回负责管理该 user 的 PN 的 IP 地址和端口,这是 P2P 层和 SIP 层唯一的一个接口。

收稿日期:2008-12-12

Email:tyl@xmu.edu.cn

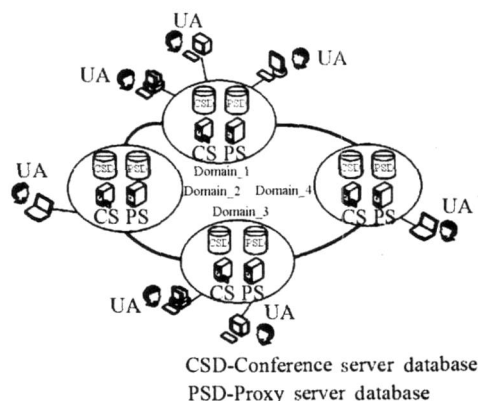


图 1 P2P SIP 视频会议系统结构模型

Fig. 1 P2P SIP video conference system architecture

1.2 功能模块

在 P2P 的 SIP 视频会议系统结构中,代理服务器 PS 主要负责信令操作,用于建立会议并利用 P2P 机制同相邻 PS 实现资源共享. CS 主要负责控制及维护会议状态和处理音视频媒体信息.

1.2.1 PS 结构

PN 中 PS 的结构如图 2 所示,分为以下几部分^[3].

- (1) 用户接口模块:与 UA 直接交互的窗口,保存成员列表,并调用用户定位模块对会议成员进行定位。
- (2) 用户定位模块:PN 内部有一张 DHT 表存储了所有已经成功注册的用户信息.该模块用于调用 DHT 模块定位用户。
- (3) DHT 算法逻辑模块:用来维护 PN 中 DHT 表的用户信息(查找、加入和离开),当用户状态发生变化,即时更新 DHT 表的内容。
- (4) SIP 模块:根据 SIP 协议进行用户的注册,呼叫建立等操作。
- (5) 身份注册模块:处理新节点加入、P2P 网络中已知节点的查找,以及新节点注册信息的生成。

1.2.2 CS 结构

SIP会议框架主要由会议控制中心、会议策略服

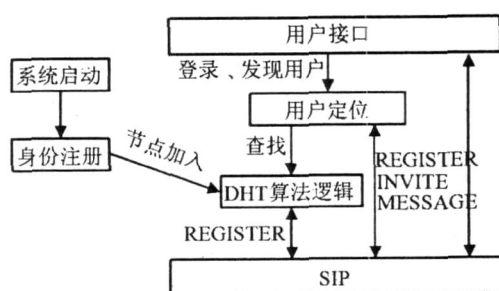


图 2 PS 结构

Fig. 2 PS structure

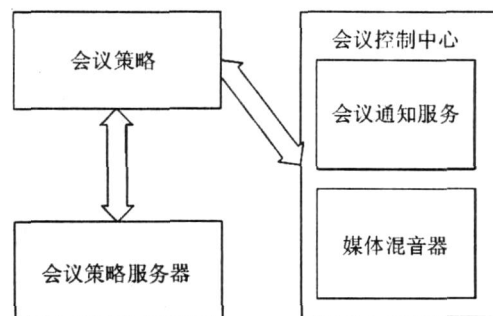


图 3 CS 结构

Fig. 3 CS structure

务器 (Conference policy server)、会议策略 (Conference policy) 组成,各部件均是逻辑部件,并非物理实体.如图3所示.

- (1) 会议控制中心:是整个 SIP 会议的信令控制中心,它和每个与会者之间存在 SIP 会话关系.其中,媒体混音器主要完成会议媒体混合和分发;会议通知服务器主要完成向与会人员通知相应的会议事件和会议状态,通过 SUBSCRIBE 和 NOTIFY 实现.
- (2) 会议策略服务器:主要是用于生成、存储和操作会议策略.
- (3) 会议策略:由 URI 表示,一个会议对应一个会议策略.会议策略包括会议成员策略和会议媒体策略.会议成员策略主要用于会议成员管理,包括用户接入控制、用户权限管理等.会议媒体策略用于会议媒体混合和分发的控制,包括媒体混合策略、同步策略、媒体分发策略等.

2 超级节点的基本功能实现

本系统的所有超级节点构成 Chord 环, 每个节点保存一个 Finger 表, 通过 Finger 表维护 Chord 环上节点的加入、查找和离开, 如图 4 所示, 其中 $N = 64$ 。

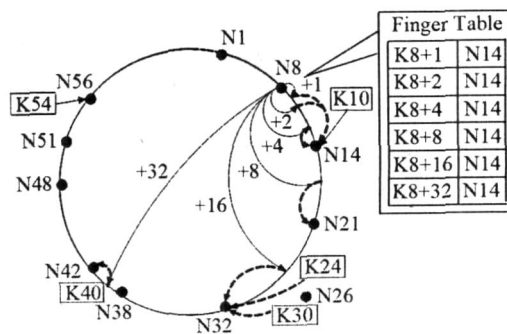


图 4 Chord 环

N-节点, 数字表示该节点的哈希值(由节点的 IP 地址得到); K-资源, 数字表示该资源的哈希值(SIP 的 URI, 如 sip:alice@power.com)

Fig. 4 Chord DHT

(1) 节点加入:若节点 N26 要加入 Chord 环,则先询问自己知道的 Chord 环中任意一个节点(假定为 N8)。N8 通过查询其 finger 表,发现 N26 的后续节点是 N32,便告知 N26。于是 N26 将 N32 设为后续节点,并通告 N32 自己的存在。N32 便把前驱节点设为 N26,并把落在 N26 管理范围内的资源 K24 交由 N26 管理。

(2) 节点查找:假设 N8 要询问资源 K30 的信息, N8 通过查找自己的 finger 表,可以知道 K30 的信息由 N32 管理,所以可以直接访问 N32 来获得所要的信息。

(3) 节点离开:若 N26 要离开,首先它把自己管理的所有节点(K24 等)更新到邻居节点 N32 上,再发送注销消息。

相对其他拓扑结构的 P2P 网络,基于 DHT 的结构化 P2P 网络在扩展性、可靠性、可维护性和节点查找效率等方面性能良好。为此,我们将其应用在 P2P-SIP 多媒体会议系统设计中,并结合 RESIPROCATE 协议栈实现 SIP 的功能。

2.1 系统启动

2.1.1 协议栈的初始化

(1) 在自定义函数 DataaddDomains (Transaction-User & tu, Store & store) 中确定服务器的域。

(2) 在主函数中通过 SipStack stack() 声明一个协议栈实例。

(3) 定义传输层协议,在 IPv4 和 IPv6 之上支持多种传输协议(UDP、TCP、TLS、DTLS),如通过 stack.addTransport(UDP, 5050, V4, StunEnabled) 函数可定义在 IPv4 上采用 UDP 协议,信令端口为 5050。

(4) 声明整个协议栈运行的线程实例: StackThread stackThread(stack)。

2.1.2 注册服务器初始化

```
Registrar registrar; // 创建一个实现注册功能的实例
InMemoryRegistrationDatabase regData; // 声明一个注册信息内存数据库
```

```
SharedPtr < MasterProfile > profile(new MasterProfile);
// 定义 SIP 信令结构
```

在这之后,为了确保注册服务器只能接收到 REGISTER 信令,需要以下步骤:

```
resip::MessageFilterRuleList ruleList;
resip::MessageFilterRule::MethodList methodList;
methodList.push_back(resip::REGISTER);
ruleList.push_back(MessageFilterRule(resip::MessageFilterRule::SchemeList(), resip::MessageFilterRule::Any, methodList));
```

2.1.3 代理服务器初始化

代理服务器中存在一系列处理器模式,在程序中主要有以下 3 种模式: RequestProcessor chain、ResponseProcessor chain、TargetProcessor chain,分别负责处理请求消息、应答消息和发送请求消息。

对其进行初始化设置:

```
ProcessorChain requestProcessors(Processor::REQUEST_CHAIN);
```

```
ProcessorChain responseProcessors(Processor::RESPONSE_CHAIN);
```

```
ProcessorChain targetProcessors(Processor::TARGET_CHAIN);
```

完成对代理服务器的处理器设置后,要声明代理服务器的实例:

```
Proxy proxy(stack,
    resip::Uri("sip:example.com"),
    requestProcessors,
    responseProcessors,
    targetProcessors,
    store.mUserStore,
    180);
```

```
stack.registerTransactionUser(proxy);
```

2.1.4 系统启动完成

在注册服务器和代理服务器实例分别初始化完成后,将两者启动:

```
DialogUsageManager * dum = new DialogUsageManager(stack);
```

```
DumThread * dumThread = new DumThread(*dum);
stackThread.run();
```

```
proxy.run();
```

```
if (dumThread)
```

```
{
```

```
    dumThread->run(); // 所有的进程都是在 dumThread 里头一直循环等待直到事件发生
```

```
}
```

2.2 用户注册

PN 收到注册请求时,首先调用函数 find_responsible_pn(user)。如果返回的地址是 PN 自己,才记录下用户注册信息,否则把注册请求转发给相应的 PN。处理后的请求原路返回给用户^[4]。具体流程如图 5。

(1) IP 地址为 192.168.13.63 的 SIP 终端发 REGISTER 请求给网络中任一 PN (假设 Key 为 69 的 PN)。

(2) Key 为 69 的 PN 收到注册请求,取得请求中 To 头域里的 SIP URI,调用 find_responsible_pn(user),发现不是由自己来处理注册请求,于是转发该 REGISTER 请求给负责处理它的终端(假设是 Key 为 32 的 PN)。

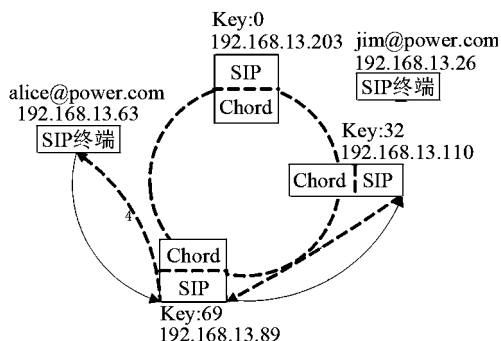


图5 用户注册过程

Fig. 5 Process of register

(3) Key 为 32 的 PN 收到 REGISTER 请求, 进行相同的处理, 发现是由自己处理该请求, 于是接受注册, 其注册消息的具体处理程序由 virtual void dispatch(const SipMessage & msg) 完成. 若注册成功则原路返回应答给用户终端.

(4) SIP 终端接收到应答, 完成注册.

2.3 用户呼叫

在 P2P SIP 网络中, 用户呼叫过程同样需要通过 Chord 环的工作来实现^[5].

(1) 在图 5 中, 若 IP 地址为 192.168.13.26 的 Jim 要发起呼叫, Jim 先将 INVITE 请求发给 P2P SIP 网络中管理它的 PN (这里假设是 Key 为 0 的 PN).

(2) Key 为 0 的 PN 收到 INVITE 请求, 取得请求中 To 头域里的 SIP URI, 调用 find_responsible_pn (user), 发现不是由自己来处理该请求, 于是转发该 INVITE 请求给负责处理它的终端 (假设是 Key 为 32 的 PN).

(3) Key 为 32 的 PN 收到 INVITE 请求, 进行相同的处理, 发现是由自己来处理该请求, 于是把该请求转发给目标用户 Alice.

(4) Alice 若接受呼叫, 则原路返回应答给主叫 Jim.

(5) Jim 接到应答, 返回 ACK 确认信息, 呼叫建立. 此时, ACK 信息不经过 P2P 网络, 而直接发给 Alice 所在的终端节点.

(6) 通话结束时, Jim 和 Alice 直接交换 SIP 信令, 不经过 P2P 网络.

2.4 用户离开

当用户想要离开时, 便向注册服务器发送注销的消息, 进行注销操作. 注销的信令消息传递和处理的过程与注册过程一致, 注销成功之后用户便成功离开. 若用户在异常情况下离开, 则注册检测机制在定时时间到达的时候会检测到该用户已经断开与服务器的连接, 于是得知该用户已经不存在^[6].

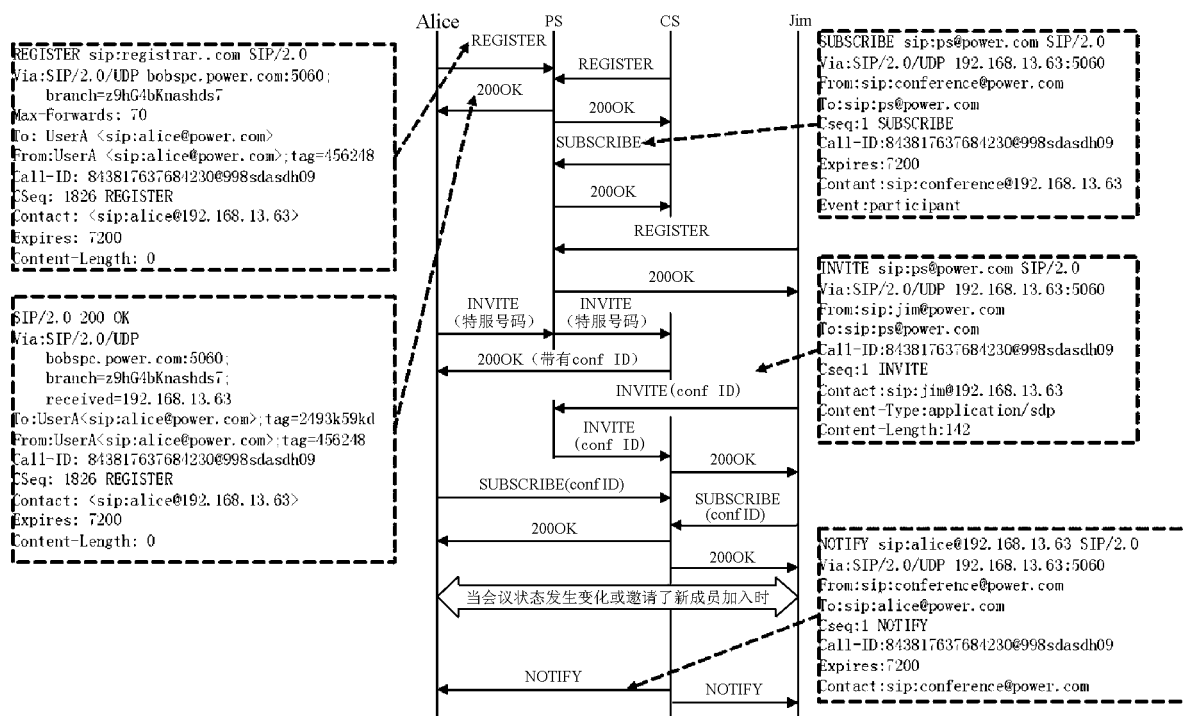


图6 同一域内用户与 CS 和 PS 的信令交互 (省略 ACK)

Fig. 6 Signaling between CS and PS within same domain (without ACK)

3 P2P SIP 视频会议系统主要信令流程

创建会议前,所有用户及会议服务器CS向各自域内的PS进行注册。由于不同域内的CS不能直接交互,必须通过各自域的PS交互,所以注册成功后,CS必须向自己域内的PS发送SUBSCRIBE消息订阅用户信息,以便及时得知用户状态变化情况。图6所示为会议发起过程的信令流程。

(1)在PS注册后,假设用户Alice要创建一个会议,则通过一个特服号码向CS发起呼叫,若CS认证通过则返回一个会议ID(Conf ID)。

(2)此后若有用户Jim要加入会议,则Jim通过PS向CS发送带有该会议ID的INVITE请求,告诉CS自己要加入这个会议。

(3)当CS允许并向Jim返回200 OK后,Jim就成了该会议的成员。

(4)会议成员都须向CS发送SUBSCRIBE消息,订阅会议信息。

(5)当会议成员状态发生变化时,PS通过NOTIFY消息向订阅过的CS发送状态更新消息,CS也会向订阅的用户发送会议更新消息。

4 结束语

结合SIP和P2P两者的技术优点,本文提出了一种新的基于P2P网络架构的SIP视频会议系统,对该

系统的体系结构与工作原理进行了详细分析,并对其重要的程序架构设计进行了介绍。实验结果和分析表明,该系统优化了现有SIP视频会议系统的架构,更能充分提高系统容量,以支持更大规模的视频会议,同时又满足可控制和运营的要求。虽然目前P2P-SIP技术还在发展中,但已显示出以P2P-SIP技术建立多媒体平台的发展趋势。

参考文献:

- [1] Venkatesha Prasad R, Hurni Richard, Jamadagni H S. A scalable distributed VoIP conference using SIP[C]// IEEE Computer Society. Washington DC, USA: IEEE, 2003:608 - 612.
- [2] 叶德谦,张国树. 基于SIP的P2P-SIP框架的研究[J]. 计算机工程, 2006, 32(19): 141 - 161.
- [3] Kundan Singh, Henning Schulzrinne. Peer-to-Peer internet telephone using SIP[C]// Association for Computing Machinery. New York, USA: ACM, 2005: 63 - 68.
- [4] 邢小良. P2P技术及其应用[M]. 北京:人民邮电出版社, 2008.
- [5] Rosenberg J. A framework for conferencing with the session initiation protocol[S/OL]. 2005-05-27. <http://tools.ietf.org/id/draft-ietf-sipping-conferencing-framework-05.txt>.
- [6] Bryan D, Lowekamp B. A P2P approach to SIP registration and resource location[S/OL]. 2006-03-05. <http://tools.ietf.org/id/draft-bryan-sipping-p2p-02.txt>.

SIP Conference System Design Based on P2P Networks

TANG Yu-liang, WANG Wei-wei, YAO Yan

(School of Information Science and Technology, Xiamen University, Xiamen 361005, China)

Abstract: Session Initiation Protocol (SIP) is a key protocol for next multimedia communication networks with its simplicity and extensibility. Due to its Client/Server (C/S) architecture, when the system traffic increases, the system performance will emerge a bottleneck easily, which is a technological obstacle of SIP application and development. The paper presents a novel architecture model of SIP video conference system based on P2P network, which improves present C/S architecture based SIP to mixed network architecture based P2P. It not only has advantages of distributed architecture like P2P system, but also has advantages of C/S architecture in the SIP system. Then the paper discusses the main functional modules and programming design. In the end, the important signaling operation flow is introduced.

Key words: SIP; P2P; DHT; SIP video conference