

SIP 协议下的对等应用工作模式研究

傅川^{1,2}, 张国清¹

(1. 中国科学院计算技术研究所, 北京 100080; 2. 中国科学院研究生院软件学院, 北京 100049)

摘要: 以 SIP 协议(Session Initiation Protocol)为基础, 重点讨论对等应用终端间通信转移、三方通信等工作模式。利用会话协议来调整对等应用间的数据传递关系, 拓展了应用适用场景, 开阔了应用的设计思路。针对具体应用与 SIP 协议间耦合问题, 给出了一个面向对象的设计和实现方案。

关键词: SIP; 对等应用; 面向对象

Study of Peer Applications Work Pattern Based on SIP

FU Chuan^{1,2}, ZHANG Guoqing¹

(1. Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100080;

2. College of Software, Graduate School of Chinese Academy of Sciences, Beijing 100049)

【Abstract】 Based on session initiation protocol (SIP), this paper discusses work patterns of communication transfer and 3-way conference, among peer application terminals. Using session protocol to adjust data transmission among peer applications opens up scenes and design methods of applications. It also puts forward a method of object-oriented to couple concrete application in SIP stack.

【Key words】 Session initiation protocol (SIP); Peer applications; Object-oriented

SIP (Session Initiation Protocol)^[1]协议为网络应用的发展提供了一条新道路。与传统用客户-服务模式的网络应用相比, 它减弱了应用对环境的假设。在基于SIP协议的系统中, 系统的各个主体相互平等、遵从一定约束, 并保留各自独立的特征。SIP协议是IETF提出的会话初始控制协议, 具有状态初始化、状态变化通知、成员管理及应用定位等特征。SIP协议涵盖了功能实体、协议消息及呼叫流程3部分内容。该协议提倡服务请求和服务处理分离、用服务请求引导服务处理的应用设计思想。

1 SIP 协议分析

1.1 SIP 协议介绍

SIP协议分为功能实体、消息及呼叫3部分。功能实体包括SIP用户终端、SIP用户代理、SIP代理、定向服务器及重定向服务器。SIP消息分为请求和应答两部分。请求消息有: INVITE, ACK, BYE, REGISTER, OPTION, CANCEL, REFER^[2], SUBSCRIBE; 应答消息有: 1XX, 2XX, 3XX, 4XX, 5XX及6XX等6类。

通过呼叫, SIP协议在主叫方和被叫方间建立一个Dialog(对话), 用它来标识正在进行中的通信。应用程序可以通过Dialog对通信进行操作。Dialog由以下3部分组成: Call-ID, Local Tag, Remote Tag。其中Call-ID由主叫方生成; 在主叫方Local Tag表示呼叫的发起方, Remote Tag表示呼叫的接受方; 在被叫方Local Tag表示呼叫的接受方, Remote Tag表示呼叫的发起方。Dialog用于协助通信双方完成会话参数的更新、恢复会话参数、添加会话参与者等任务。从应用角度看, Dialog起基本事务单元的作用。

1.2 SIP 实体运行机制分析

SIP代理在协议中起核心的作用。为了深入理解它的作

用, 我们用面向对象的代理者模式^[4]对SIP协议实体所构成的服务框架进行分析。代理者模式是一种重要的分布式框架, 主要用来构建具有相互独立而又有互操作需求的系统。使用代理者模型可以在以下几点达成平衡: 通过远程, 地点透明的服务调用, 提供组件可以访问其它组件的能力; 组件在运行期间可交换、添加及移动; 向用户隐藏系统的特定实现细节等。代理者框架由6种组件构成: 客户机, 服务器, 代理者, 桥接, 客户机端代理和服务器端代理。

SIP用户终端包括UAC/UAS两部分, 它们分别起客户端代理和服务端代理的作用; SIP代理起代理的作用。服务的访问点由SIP协议中Contract标记进行标识。当用户终端通过UAC发起呼叫后, 接收呼叫的SIP代理将呼叫转发给定向/重定向服务器, 定向/重定向服务器将了解目标的SIP代理地址填入应答消息的Contract标记中, 发回呼叫代理。呼叫代理根据Contract的内容进一步询问目标代理。目标代理将呼叫请求发给目标前, 再次询问定向/重定向服务器, 以确认目标地址没有改变, 并获得用户访问限制。目标在收到呼叫后, 将自己的服务地址填入200消息的Contract标记中作为应答的一部分返回呼叫发起方。通过Contract内容的多次改变的工作方式, 呼叫方在请求服务时不需要知道被叫方的真实地址, 系统负责将双方联系在一起。

1.3 呼叫流程

呼叫是SIP协议实体间利用SIP消息相互应答的过程,

基金项目: 下一代互联网中日IPv6合作项目“IPv6主机和智能终端等接入系统关键技术开发”(20032040)

作者简介: 傅川(1973-), 男, 工程师, 主研方向: 计算机网络管理, 软件工程; 张国清, 副研究员

收稿日期: 2006-05-09 **E-mail:** chuanfu@ict.ac.cn

是该协议的显著特点。如前所述，Dialog 的建立和用户定位都是通过呼叫完成的。SIP 协议将呼叫分为 2 种：3 次呼叫，2 次呼叫。协议将这两种呼叫都定义为事务。典型的 3 次呼叫为 INVITE 引导的呼叫；典型的两次呼叫为 Bye 引导的呼叫。从应用系统的角度看，SIP 协议定义的事务保证了每个呼叫的完整性，防止非事务内信令的干扰。应用系统将每个呼叫作为子事务或检查点处理使用，用于资源的申请或释放。这些子事务受一个事务控制块控制，子事务间存在叠加、嵌套的可能。采用这种先申请再使用的资源分配机制有助于提高应用的可靠性。

2 基于 SIP 的工作模式

使用 SIP 协议为对等应用提供了更灵活的工作模式，如通信转移、三方通信^[3]；减少了应用对底层传输协议和网络环境的依赖，增强了应用的适用面。与电话网络通过局端进行呼叫转移、三方通话的方法相比，基于 SIP 协议的终端系统可以让用户自己控制通信的工作模式。

2.1 通信转移

对等应用间的通信转移指正在通信的双方中的一方将指向自己的数据流转向第三方，或允许由第三方来代替自己继续通信。SIP 协议可以支持 3 种通信转移行为：基本转移，撮合，接替。基本转移是最基本的通信转移(如图 1)。基本转移的典型步骤为：转移的发起方用 REFER 方法向转移的执行方发送转移的目标地址。基本转移不保证转移的成功。转移执行方在转移过程中用 NOTIFY 向转移发起方通告转移的进展情况。撮合是在基本转移的基础上提供通信转移的保障机制。转移发起方在转移发生前，分别同转移执行方和转移目标建立会话。转移发起方向转移执行方发出的 REFER 消息中包含表示转移发起方和转移目标间关系的 Dialog 标识，告知转移执行方代替该 Dialog 中的转移发起方的位置同转移目标进行通信。转移执行方则用带有 Replace 标记的 INVITE 消息向转移目标发起呼叫。接替是在知道一个 Dialog 的前提下，接替者用带有 Replace 标记的 INVITE 消息向会话一方发起通信请求。

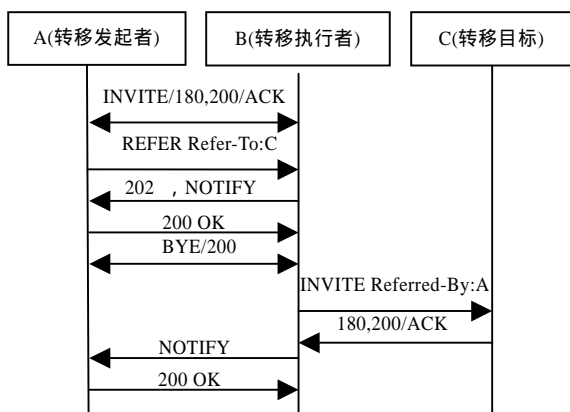


图 1 基本通信转移流程

由用户终端控制通信转移大大丰富了对等应用的工作模式，但这也引起由谁来负责通信转移完整性的问题。由 RFC3515 可以看出，REFER 方法同 RFC3265 中定义的 SUBSCRIBE 方法处理 Dialog 的原理相同。REFER 可以直接产生 Dialog，也可以使用已经存在的 Dialog。当 REFER 在 Dialog 中使用时，该方法创建的子状态集不影响 Dialog 中已存在的其它子状态集。例如，Dialog 由 INVITE 方法生成，

当使用 BYE 方法时，如果 Dialog 中有 REFER 方法生成的子状态集存在，BYE 方法仅释放由 INVITE 方法申请的资源。只有当 Dialog 中最后一个子状态集释放后，Dialog 才被销毁。由此可见，转移发起者与转移执行者间的操作不影响转移执行者同转移目标间的操作，而转移执行者要向转移发起者报告转移执行的进程。

基于以上原则，为了确保转移完整性，在 Dialog 中需要考虑以下 3 种基本情况：由发起者发出 BYE；由执行者发出 BYE；由执行者向目标发出 CANCEL 或目标不接受请求。针对 Dialog 的特性，我们从应用的角度引入了一个事务控制块的概念，要求对一个由 INVITE 创建的通信的转移，只能发生在该 INVITE 方法生成的 Dialog 内。当发起者或执行者发出 BYE 时，释放由 INVITE 申请的资源，但保留 Dialog。当执行者向目标发出 CANCEL 或目标不接受请求时，如果 Dialog 中存在 INVITE 生成的子状态集，那么由转移发起方负责恢复其与执行者间的通信。

2.2 三方通信

3~5 人的小规模会谈是日常工作中常见的情景，如果能不借助多播的方式由应用程序直接组织，无疑增加了应用的适用面。SIP 有两种实现三方通信的方式：邀请，主动加入。邀请行为由两部分组成：会议的主持者用 INVITE 请求邀请第三方参加会议；会议的主持者用 re-INVITE 调整正在参加通信的参与者的会话参数。主动加入是第三方用带有 Join 标记的 INVITE 方法请求参加一个已存在的 Dialog，三方通信的主持者在接受第三方请求后将调整其它参与者的会话参数。

2.3 复杂的工作模式

SIP 终端的灵活性使用户很容易运用基本呼叫组织通信，使应用适合复杂的任务场景。如三方通信的参与者可能希望别人代替自己继续通信。该参与者可以在 REFER 方法中加入 Replace 参数让转移的执行者参与会话。

3 用户终端结构和软件框架

3.1 用户终端结构

SIP 协议的引入改变了终端系统应用层同传输层直接耦合的局面。应用层同传输层直接耦合假定了应用只能发生在由特定 IP 地址和端口定义的关系内。这样，用户被预先绑定在 IP 标识的设备上。当人离开该设备，应用将自动终止。用户终端中使用 SIP 协议使应用有机会根据用户环境进行调整。

用户终端结构如图 2，其中包括：应用逻辑层，SIP 协议栈，数据传输子系统 3 部分。应用逻辑中可以包括多个 SIP 协议栈的 agent。应用逻辑通过 agent 使用 SIP 协议栈，agent 用于将应用逻辑的参数转换为 SIP 消息。SIP 栈分为两层：上层主要用于处理呼叫事务，下层用于接收和发送 SIP 消息。数据传输子系统是具体服务发生时传输数据。

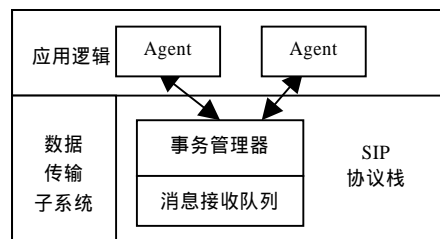


图 2 用户终端结构

