

# 基于 Asterisk 的软交换运营平台设计

邓国栋, 胡越明, 白英彩

(上海交通大学计算机科学与工程系, 上海 200240)

**摘 要:** 随着下一代网络(NGN)成为 PSTN 和 VoIP 发展的目标, 作为 NGN 的核心 软交换成为了热点。很多运营商都缺乏一个好的软交换运营管理平台来管理软交换业务, 而平台在可扩展性和兼容性等方面的性能更是成为所关心的焦点。该文针对当前一些典型软交换业务的特点, 基于开源 Asterisk 系统结构, 提出了实现诸多新业务的运营平台设计框架, 并在业务扩展以及业务管理等方面做了改进。

**关键词:** 下一代网络; 软交换; VoIP; Asterisk

## Design of Soft-switch Management Platform Based on Asterisk

DENG Guo-dong, HU Yue-ming, BAI Ying-cai

(Department of Computer Science and Engineering, Shanghai Jiaotong University, Shanghai 200240)

**【Abstract】** Since NGN became the future of PSTN and VoIP, soft-switch, as an essential part of NGN, has been more and more popular. The suppliers need a good soft-switch operation management platform, which is good at expansibility as well as compatibility. This paper shows the framework of an operation management platform. The platform, based on the open source Asterisk system architecture, has characteristics of some typical soft-switch operation, and realizes various kinds of new operation. Improvement to the management and the extensibility of operation is made.

**【Key words】** next generation network (NGN); soft-switch; VoIP; Asterisk

软交换运营平台主要是基于软交换和NGN的思想, 充分地结合互联网与企业内部网的特点, 在提供高质廉价的语音通信的同时, 提供和管理各种相关业务。对于软交换运营平台, 一方面要求能够紧跟软交换业务的发展, 不断地提供新的业务支持; 另一方面, 也要求能够对所提供的业务提供完备的管理机制。通过软交换运营平台, 运营商可以方便地管理软交换系统中的各类用户信息、增值业务信息、计费信息, 并可向用户提供各种新型的服务。同时借助系统提供的灵活多样的费用管理机制和完善的运营支持, 吸引更多的用户使用VoIP服务, 提高收益和客户满意度, 从而极大地提升运营商的竞争力<sup>[1]</sup>。

本文结合目前软交换业务特点和发展方向, 参考当前已有的一些软交换系统结构, 提出软交换运营平台 SVP 设计方案, 并对原有体系结构进行了改进。

### 1 SVP 运营平台基本设计模型

基本的 SVP 运营平台模型如图 1 所示。

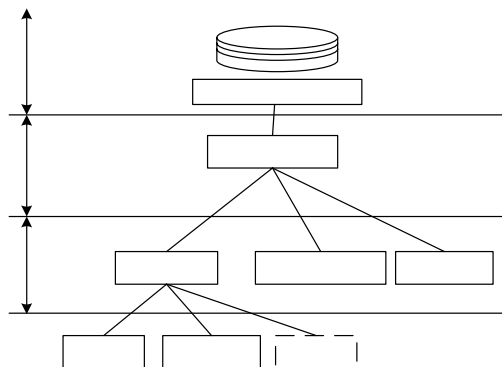


图1 SVP运营平台分层模型

基本的 SVP 运营平台模型其功能主要实现基于 IP 分组网络的呼叫建立、媒体转发和计费功能, 结构分为业务接入层、业务控制层和业务管理层。它是对很多具体平台设计的一个抽象。

业务接入层由一系列 VoIP PROXY、PSTN GATEWAY 和智能外设等设备组成。业务接入层主要实现呼叫处理功能和业务交换功能, 并支持不同协议之间的互连。呼叫处理功能具有接收用户呼叫、执行呼叫建立、结束呼叫等基本业务功能。业务交换功能则能向业务控制层发送报告和接收业务控制命令等。业务接入层需要验证用户合法性, 记录用户注册地址, 并完成 Internet 与 PSTN 之间的信息交互。

业务控制层 SCP 主要包含一个控制模块, 该控制模块要求实现与业务接入层和业务管理层之间的通信, 并完成对具体业务的支持和控制。在很多具体实现中, 都将业务逻辑放在业务控制层来实现, 例如三方通话等。在业务控制层当中, 一般还会有业务状态机的实现。从主叫输入号码开始, 业务控制层根据业务接入层传入的信息, 将业务状态机运行到某个特殊的状态, 并等待相应后继信息, 这个状态用来标识本次通话进行的状态。

业务管理层由业务管理与计费系统和业务生成与计费查询系统组成。业务管理层的功能包括业务数据管理、用户数据管理、业务检测以及业务量管理等, 并要能够提供用户和管理员对于用户数据、业务数据的访问支持。业务管理层需要管理的信息包括路由信息、计费信息以及用户资费信息等。

**作者简介:** 邓国栋(1982 - ), 男, 硕士, 主研方向: 网络通信, VoIP; 胡越明, 副教授; 白英彩, 教授、博士生导师

**收稿日期:** 2006-11-06 **E-mail:** robbyk@sjtu.edu.cn

为了方便用户以及业务控制层获取所需信息,该层后台设计很大程度上需要依赖于数据库,而前台主要依赖于 apache 和 cgi 来实现。

以上提出的模型是根据很多具体设计抽象出的,主要是提出系统框架的概念。在该模型当中,如何实现业务逻辑以及如何实现业务控制层与业务管理层之间的通信是核心问题,在后文中将对模块设计的很多具体问题予以阐述。

## 2 SVP 运营平台设计模型分析

在前文提出的系统结构当中,最核心的部分当属业务控制层。业务控制层位于业务接入层和业务管理层之间,承担业务控制功能,是整个系统的控制枢纽。如何提高业务控制层的性能,进而优化整个系统的体系结构,是整个系统设计的瓶颈。

在很多具体设计当中,业务逻辑都在业务控制层当中实现。这种设计方式需要业务控制层将很多具体业务共享一些执行流程,然后通过 select-case 的模式转入相应业务的执行。这种设计方案使业务执行没有统一的接口,将导致很多业务之间逻辑不清晰。这样的设计将极大地影响到新业务的扩展,在新业务扩展的同时影响到对原有业务的支持。业务控制层的设计,需要关心如何使业务逻辑之间的关系更加清晰。

另外一个问题是业务控制层与业务管理层之间交互的问题。对于数据访问和数据管理,有两种方式:一种是采取直接调用系统库函数的方式。操作系统底层直接支持各种数据库的调用接口,通过调用系统库能实现对于数据库的访问;另一种是利用外部软件调用数据库,如 openradius、freeradius 等,依靠外部软件与数据库联系,并达到连接业务管理层和业务控制层的目的。前一种方式比较直接,需要把整个数据库的管理部分作为一个模块设计,设计起来比较复杂。而后一种方式比较方便,但是业务控制层必须具备与该软件通信的能力。

如上所述,业务的整个实现对业务控制层在数据管理以及业务扩展方面都提出了很高的要求。业务控制层需要提供与业务管理计费系统的通信接口,以保证准确获取和保存用户信息等数据;还要能提供灵活的业务接口,以明晰各种业务之间的关系,实现新业务扩展需求。

综合业务扩展与用户信息管理两个因素考虑,SCP 系统选择采用开源软件 Asterisk 作为业务控制层的核心。

## 3 SCP 的核心 Asterisk

Asterisk 是一个开源的软件,它是一个运行在 Linux 环境下的纯软件 VoIP 解决方案<sup>[2]</sup>。其最大特点是以软件的形式实现了一般 VoIP 业务系统的所有功能,如语音信号的编解码等,并不需要外加 DSP 或者专门的硬件平台。它是一种功能非常齐全的应用程序,提供了许多电信功能。目前业界比较流行的业务,如三方通话、Voicemail 等,它都能予以良好的支持。Asterisk 支持目前比较流行的协议,包括 SIP、H323、IAX 以及 MGCP 等。Asterisk 还具备友好的管理界面,为业务管理层的设计提供了一些参考方案。Asterisk 提供了一种廉价的 VoIP 解决方案,可以完美地应用于中小企业的设计。

在业务控制层的设计中以 Asterisk 为核心,与其以下特点有很大关系:

(1) Asterisk 采用模块化设计方案,多数功能模块都是由动态模块加载器根据编译结果在运行时动态加载。这一特点使笔者能够很方便地根据系统功能需要,提供相关模块支持。

(2) Asterisk 为业务控制逻辑提供了相应的接口支持。对

于 callingcard 等增值功能,只需要以脚本的形式完整地实现其控制流程,并在配置文件中实现配置之后,就能够正常运行业务。当然,很多新的增值业务的实现需要增加新的功能模块。但是由于 Asterisk 动态加载模块的机制,使得新模块的增加不影响原有系统功能的运行。

(3) 因为以上两个特点,所以 Asterisk 能通过脚本形式实现与 radius 软件的通信。Radius 软件在很多设计中,被用来实现 AAA 功能和数据库访问功能。通过 perl 脚本,以及现有功能函数的支持, Asterisk 能很方便地实现与 radius 软件的交互。这样,系统设计时将很自然地在业务管理层引入 Freeradius 和开源数据库 MySQL 来实现。可以说采用 Asterisk 之后,业务控制层与业务管理层之间已经实现了无缝衔接,系统设计将极大完善。

(4) 对于业务接入层而言, Asterisk 既可以做 VoIP 服务器,也可以做 VoIP 客户端。也就是说它既可以向其他客户端提供账号注册,也可以向其他外部服务器注册,这将给系统级联提供很好的思路。

由于 Asterisk 良好的模块扩展功能、业务扩展功能以及在级联方面表现出的特性,因此在本系统中采用 Asterisk 将极大地提高系统性能,完善设计。

## 4 基于 Asterisk 的软交换平台 SVP 模型改进

采用 Asterisk 作为业务控制层的核心之后,系统结构将出现两点变化:(1)在系统中引入了业务扩展模块作为业务扩展接口,这将极大方便对新业务的支持,是改进后的系统具备的一大特性;(2)在业务接入层支持与其他外部系统的级联,这部分的工作将提高系统的级联特性。如图 2 所示。

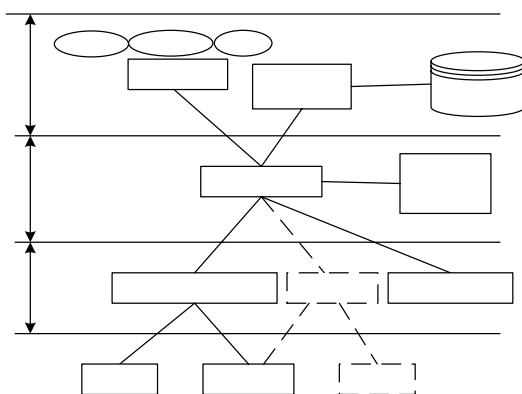


图 2 基于 Asterisk 的运营平台模型

改进后的业务控制层主要是根据业务接入层发送的业务报告,向业务扩展模块索取业务逻辑,然后根据回复启动不同的业务逻辑,根据业务逻辑向相应的业务接入层设备发送呼叫控制指令。业务控制层主要完成对于业务逻辑的控制实现,完成各种不同通信协议之间的互连转换等。而业务管理层的功能将包括业务逻辑管理、业务数据管理、用户数据管理、业务检测以及业务量管理等,并要能够实现对新业务的支持。以一次具体的语音通信业务流程为例,当用户拨号之后,业务控制层首先需要从业务接入层接收业务启动信息,完成外部 VoIP 消息到系统内部消息之间的转换,然后向业务管理层发送业务查询信息,并从回复中得到相关业务的实现逻辑以及所需数据。在得到业务逻辑及所需数据之后,业务控制层将根据业务逻辑实现对于具体业务的控制。在业务执行当中,业务控制层仍需不断和业务接入层以及业务管理层通信。与业务接入层的通信主要是完成对于整个语音通话流

程的控制,如振铃控制、摘机控制以及挂机控制等;而与业务管理层之间的通信主要是完成对于用户数据信息的管理,如用户鉴权控制、通话时间控制以及通话结束之后的余额控制等,这些工作更多是为了实现对于卡号业务以及AAA(authorize, authenticate, accounting)业务的支持<sup>[3]</sup>。

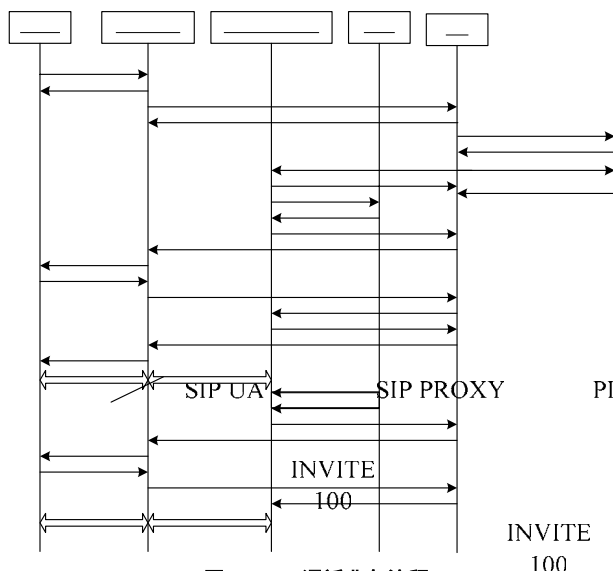


图3 SIP 通话业务流程

以 SIP 协议的实现为例,业务控制层与业务管理层之间的通信将通过 perl 脚本 b2bua 来实现,b2bua 当中将定义具体的通信流程以及相应的消息内容,Asterisk 得到其流程和消息内容之后则具体完成与 radius 的通信。而业务接入层则需引入 ser 来实现,sip 软件 ser 可以用来充当系统结构中的 SIP REGISTER/PROXY 的角色,用于验证账户的合法性,保留账户注册地址等信息。接收 SIP UA 会话请求,并向 SIP

REGISTER 查询被叫用户地址,然后将会话请求转发给同域的被叫用户 UA,或其他域的 SIP PROXY。图3描述了用 SIP 账号拨打 PSTN 电话的具体实现流程。这里为了使通信流程清晰,省略了与业务扩展模块之间的通信过程。

其中 ACCESS REQ 消息将向业务管理层请求获取主叫的呼叫权限、被叫的地址等消息,而 ACCOUNTING START 消息将请求开始计费。从图3中可以看出,该系统能很好地支持计费、AAA 等功能,并能实现不同协议 SIP 与 PSTN 之间的呼叫连接。而图中隐藏的业务扩展接口将使业务扩展变得很方便,这是新一代软交换运营平台设计的必然趋势。

## 5 结束语

如今软交换业务迅猛发展,软交换运营平台对于提高运营商的竞争力,提高业务运营水平,使用户方便快捷地享受到廉价的软交换业务至关重要。

本设计主要是基于目前流行的运营平台设计模式,在业务扩展、用户数据管理以及系统级联上做了一些改进。本文提出了统一的业务扩展接口,为系统级联提供了支持,提高了平台对于新业务的扩展能力。当然,也不可避免地带来了一些问题,如 Asterisk 的纯软件设计,导致其在执行业务尤其是编解码业务时将消耗较多的系统资源。如何合理地解决这个矛盾,也是将来系统改进的一个重要方面。本设计已经在相关平台上进行了调试,具备了所提业务功能,系统运转良好。

- 参考文献
- 1 Van Meggelen J, Smith J, Madsen J. Asterisk: The Open Source Telephony [M]. [S. l.]: O'Reilly, 2005.
  - 2 汤非凡, 朱杰. 基于 Asterisk 的 VoIP 解决方案[J]. 计算机系统应用, 2005, (7): 79-81.
  - 3 顾茜. 平滑构建软交换业务平台[J]. 通信世界, 2005, (21): 38-38.

(上接第 139 页)

二次(或多次)DHCP 过程的发生,DHCP 服务器可以通过得到的这个非 DHCP 客户机的 IP 地址来检查自己的配置文件/etc/dhcpd.conf(这里假设 DHCP 服务器是一台装有 Linux 操作系统的机器),验证该地址是否包含于它的空闲地址池内。

如同非 DHCP 客户机一样,此解决方案要求修改 DHCP 服务器端的 DHCP 守护进程,在这个进程中,加入如下内容:如果接收到来自非 DHCP 客户机的“伪 DHCP Discover”,就创建一个“分裂子进程”。分裂子进程的职责就是从“伪 DHCP Discover 包”中提取出非 DHCP 客户机的 IP 地址,并根据得到的 IP 地址来对本机的 DHCP 配置文件/etc/dhcpd.conf 进行相应的处理。

分裂子进程对 DHCP 配置文件的处理过程是这样的:假设得到的非 DHCP 客户机的地址为 192.168.1.9,如果 dhcpd.conf 文件中的地址池范围为 192.168.1.1~192.168.1.250,说明非 DHCP 客户机的 IP 包括在该地址池内,那么 DHCP 服务器就在此时“自动分裂”自己的地址池,把地址池分为两部分:192.168.1.1~192.168.1.8 和 192.168.1.10~192.168.1.250。这样,就把 192.168.1.9 这个非 DHCP 客户机的 IP 地址排除在地址池之外了。以此类推,在多个非 DHCP 客户机的 IP 都在地址池的范围内的情况下,经过多次地址池“自动分裂”过程,最终可以从地址池中排除掉所有的非 DHCP

客户机的 IP 地址。

## 4 结束语

该解决方案改善了 DHCP 协议,DHCP 服务器可以由此机制获得非 DHCP 客户机的地址,从而避免了二次甚至多次 DHCP 过程的发生,减少了大量冗余数据包的传送,降低了发生网络堵塞现象的可能性,进而提升了网络的性能。这个改进机制实施起来无需对现有的网络结构进行更改,只需更改非 DHCP 客户机和 DHCP 服务器机器上运行的程序,便可以达到提高网络工作效率的目的。网络升级所需的成本很低,大大加强了此方案的可行性。另一方面,正是因为这个方案需要在非 DHCP 客户机和 DHCP 服务器两端各自运行新的进程,所以必然会带来机器本身资源的消耗。从用户的角度来说,用户必须面临安装补丁软件的问题。

## 参考文献

- 1 Dynamic Host Configuration Protocol[S]. RFC 1541, 1993-10.
- 2 DHCP Options and BOOTP Vendor Extensions[S]. RFC 1533, 1993-10.
- 3 Forouzan B A, Fegan S C. TCP/IP 协议族[M]. 2 版. 谢希仁,译. 北京:清华大学出版社,2003.
- 4 李方敏,卢锡城,汪钟鸣. 主机动态配置协议(DHCP)原理及其实现[J]. 湘潭矿业学院学报,1997,12(3): 17-24.

