

# **ESCUELA POLITÉCNICA DEL EJÉRCITO**

**FACULTAD DE INGENIERÍA ELECTRÓNICA**

**PROYECTO DE GRADO PARA LA OBTENCIÓN DEL  
TÍTULO EN INGENIERÍA ELECTRÓNICA**

**“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA EMULADOR  
DE UNA CENTRAL TELEFÓNICA VoIP PARA EL  
LABORATORIO DE COMUNICACIONES DE LA FACULTAD DE  
INGENIERÍA ELECTRÓNICA FIE”**

**LUIS GEOVANNI ESTÉVEZ ORQUERA**

**SANGOLQUÍ – ECUADOR**

**SEPTIEMBRE 2005**

# INDICE

<b>1. INTRODUCCIÓN .....</b>	<b>1</b>
1.1. OBJETIVOS.....	1
1.1.1. <i>Objetivo general</i> .....	2
1.1.2. <i>Objetivos específicos</i> .....	2
1.2. IMPORTANCIA .....	2
1.3. ALCANCE.....	3
1.4. CONTENIDO POR CAPÍTULOS .....	3
<b>2. TEORIA DE VOZ SOBRE IP VOIP .....</b>	<b>5</b>
2.1. INTRODUCCIÓN .....	5
2.2. TENDENCIAS DE LA TELEFONÍA IP .....	6
2.3. ¿CÓMO FUNCIONA LA TELEFONÍA IP? .....	8
2.4. ¿PERO QUE ES EL INTERNET PROTOCOL O IP? .....	8
2.4.1. <i>Características Fundamentales</i> .....	9
2.4.2. <i>IP en el Modelo OSI</i> .....	9
2.4.3. <i>Formato de Datagramas IP</i> .....	10
2.4.3.1. Cabeceras .....	11
2.4.3.2. Tipos de servicio .....	16
2.4.4. <i>Direcciones IP</i> .....	18
2.5. PROTOCOLOS DE VOZ SOBRE IP O VOIP.....	25
2.5.1. <i>Protocolo H.323</i> .....	25
2.5.2. <i>Protocolo SIP</i> .....	28
2.5.2.1 Aspectos generales.....	29
2.5.2.2 Elementos funcionales.....	32
2.5.2.3 Mensajes.....	35
2.5.2.4 Solicitudes (Métodos) SIP:.....	36
2.5.2.5 Cabeceras .....	43
2.5.3. <i>Protocolo H.248/MEGACO</i> .....	56
2.5.3.1 Componentes que intervienen .....	57
<b>3. SISTEMA EMULADOR DE LA CENTRAL TELEFONICA IP .....</b>	<b>59</b>
3.1. INTRODUCCIÓN.....	59
3.2. ANTECEDENTES DE VOCAL.....	59
3.3. QUE ES VOCAL? .....	60
3.4. ARQUITECTURA DE VOCAL.....	61

3.4.1.	<i>Servidores de vocal</i> .....	62
3.4.1.1.	Servidores Marshal.....	62
3.4.1.2.	Servidores Feature.....	62
3.4.1.3.	Servidores Redirect.....	63
3.4.2.	<i>Interfaz grafico de vocal</i> .....	63
3.4.2.1.	Nivel de Acceso de Administrador.....	64
3.4.2.2.	Nivel de Acceso de Técnico.....	64
3.5.	FUNCIONAMIENTO DE VOCAL.....	65
3.6.	COMPONENETES DE VOCAL.....	66
3.6.1.	<i>User agent</i> .....	67
3.6.2.	<i>Servidor marshal</i> .....	68
3.6.2.1.	Tipos de Servidor Marshal.....	68
3.6.3.	<i>Translators</i> .....	71
3.6.4.	<i>Servidor provisioning</i> .....	72
3.6.5.	<i>Servidor redirect</i> .....	73
3.6.5.1.	Lista de Suscriptores o Abonados.....	74
3.6.5.3.	Lista de Contactos.....	79
3.6.6.	<i>Servidor feature</i> .....	83
3.6.7.	<i>Servidor voice mail</i> .....	83
3.6.7.1.	Servidor Voice Mail Feature.....	84
3.6.8.	<i>Servidor call detail record o cdr</i> .....	86
3.6.9.	<i>Servidor policy</i> .....	87
3.6.10.	<i>Servidor heartbeat</i> .....	88
3.6.10.1.	Gestor de Red.....	89
3.7.	IMPLEMENTACIÓN DEL SISTEMA VOCAL.....	91
3.7.1.	<i>Instalación del sistema operativo</i> .....	92
3.7.1.1.	Verificaciones de la instalación del Sistema Operativo.....	92
3.7.2.	<i>Instalación de java runtime environment JRE</i> .....	94
3.7.3.	<i>Instalación del software de vocal</i> .....	95
3.7.3.1.	Verificación de la instalación de VOCAL.....	95
3.8.	CONFIGURACION DE VOCAL.....	96
3.8.1.	<i>Mejorando el desempeño de vocal</i> .....	98
3.8.2.	<i>Configuracion de vocal para obtener un detalle de llamadas</i> .....	100
3.8.3.	<i>Configurando usuarios o extensiones de vocal</i> .....	105
3.9.	CONFIGURACIÓN DE TELEFONOS IP.....	108
<b>4.</b>	<b>EVALUACIÓN DEL SISTEMA</b> .....	<b>116</b>
4.1.	INTRODUCCIÓN.....	116
4.2.	PRUEBAS A NIVEL DE LAN.....	116
4.3.	PRUEBAS A NIVEL DE WAN.....	119
4.4.	ANALISIS DE TRAFICO GENERADO POR UNA LLAMADA A TRAVES DEL SITEMA VOCAL.....	123

<b>ANÁLISIS ECONÓMICO .....</b>	<b>127</b>
5.1. INTRODUCCION.....	127
5.2. SOLUCIÓN DE TELEFONÍA IP PARA LA FIE UTILIZANDO VOCAL .....	127
5.3. SOLUCIÓN DE TELEFONÍA IP PARA LA FIE UTILIZANDO TECNOLOGÍA CISCO.....	128
5.3.1. <i>Media convergence server o mcs</i> .....	130
5.3.2. <i>Teléfonos ip</i> .....	131
5.3.2.1. Cisco IP Phone 7960 .....	131
5.3.2.2. Cisco IP Phone 7940 .....	132
5.3.2.3. Cisco IP Phone 7910 y 7910+Sw .....	132
5.3.2.4. Softphone IP.....	133
5.3.3. <i>Conmutadores Catalyst</i> .....	134
5.3.4. <i>Gateways de voz</i> .....	134
5.3.4.1. Cisco VG248.....	135
5.3.5. <i>Cisco 2691</i> .....	136
5.3.6. <i>Costos de equipos cisco</i> .....	137
<b>6. CONCLUSIONES Y RECOMENDACIONES .....</b>	<b>138</b>

## **CAPITULO I**

### **1. INTRODUCCIÓN**

En este capítulo se resumirá los objetivos, importancia, alcance y contenido por capítulos del presente proyecto.

#### **1.1. OBJETIVOS**

La necesidad inherente de los seres humanos por comunicarse hace que cada vez el aporte del Ingeniero Electrónico en Telecomunicaciones sea más tangible, y es este aporte el que ayuda a que los pueblos logren el objetivo de comunicarse y por lo tanto estén más unidos.

El uso de tecnología digital para lograr esta comunicación esta en auge, y la continua necesidad de comunicarse por medio de la voz, hace indispensable la búsqueda de nuevas alternativas que lleven a conseguir este fin.

La forma actual en la que se establece una llamada telefónica dentro de una organización como es el caso de la ESPE y por ende de la FIE, es a través de una central telefónica convencional o PBX, la cual cumple principalmente con la función de comunicar a esta organización tanto interna como externamente.

Sin embargo las necesidades de los usuarios son crecientes ya no solo buscan el establecimiento de una llamada telefónica, sino además acceder a aplicaciones multimedia tales como voice mail y demás.

### 1.1.1. OBJETIVO GENERAL

Realizar el Diseño e Implementación de una Sistema Emulador de una Central Telefónica VoIP para la Facultad de Ingeniería Electrónica FIE.

### 1.1.2. OBJETIVOS ESPECÍFICOS

- Estudiar los estándares y protocolos que permiten implementar voz sobre IP (VoIP).
- Analizar el funcionamiento del software libre especializado de nombre vocal (vovida open communication application library) sobre plataforma gnu/linux, para ser utilizado como un sistema emulador de una central telefónica basada en VoIP.
- Diseñar e implementar el sistema emulador sobre la infraestructura de red LAN existente la facultad de ingeniería electrónica de la ESPE, con la configuración del sistema VOCAL.

## 1.2. IMPORTANCIA

La palabra clave en las telecomunicaciones de presente y futuro es la *convergencia de servicios*, estos se van adaptando tanto a las necesidades de los usuarios como a la capacidad tecnológica con la que se dispone en ese momento; el mencionado usuario desea tener dentro de un mismo servicio múltiples aplicaciones como parte de un valor agregado, lo que haría de su contacto diario con la tecnología más transparente.

Si a esto agregamos que en un futuro se prevé que todo tipo de información como voz, video, datos, fluirá sobre IP; es ese el camino a tomar.

### **1.3. ALCANCE**

El proyecto que propone este documento es una interesante alternativa para cubrir múltiples necesidades al establecer una comunicación telefónica, no solo las tradicionales como transferencia de llamadas, identificador de llamadas, sino además servicios como voice mail y en general multimedia.

Dicha alternativa es una aplicación sobre redes IP “Internet Protocol”, como es la de transmisión de voz, que en la actualidad se conoce como Voz sobre IP VoIP, la cual brinda un nuevo enfoque a lo que conocemos como telefonía, ya que desde un solo dispositivo como es el teléfono IP o un software que haga las veces de este, podremos tener concentrados múltiples servicios, esto gracias a una central telefónica VoIP o un sistema emulador que haga las veces de esta.

Si a esto se agrega que la mayoría de organizaciones tanto públicas como privadas tienen ya como parte de su infraestructura un backbone que además de ser utilizado para transportar datos serviría también para transportar voz. Por lo tanto, haremos una menor inversión en lo que es la capa física, y esto se traduce en un ahorro de dinero.

### **1.4. CONTENIDO POR CAPÍTULOS**

#### **CAPITULO I: Introducción**

En este capítulo se resumirá los objetivos, importancia, alcance y contenido por capítulos del presente trabajo.

#### **CAPITULO II: Teoría de Voz sobre IP o VoIP**

En este capítulo se presentará un análisis detallado de la tecnología de transmisión VoIP.

### **CAPITULO III: Sistema Emulador de la Central Telefónica IP**

En este capítulo se presentará la propuesta técnica para el diseño e implementación del Sistema Emulador de la Central Telefónica IP.

### **CAPITULO IV: Evaluación del sistema**

En este capítulo se presentarán las demostraciones de funcionamiento de los servicios que proporciona el Sistema Emulador y el correspondiente análisis de evaluación de desempeño del mismo.

### **CAPITULO V: Análisis Económico**

En este capítulo se hará un análisis económico del proyecto.

### **CAPITULO VI: Conclusiones y Recomendaciones**

En este capítulo se indicarán las conclusiones y recomendaciones recopiladas a lo largo del proyecto de Tesis.



## **CAPITULO II**

### **2. TEORIA DE VOZ SOBRE IP VOIP**

En este capítulo se realizará un estudio detallado de la Teoría de Voz sobre IP VoIP.

#### **2.1. INTRODUCCIÓN**

En los últimos años el interés por las redes de paquetes IP (Internet Protocol) como soporte de tráfico multimedia ha experimentado notable crecimiento.

En tal sentido, tanto ITU (Internacional Telecommunications Union) como el IETF (Internet Engineering Task Force) han estado desarrollando arquitecturas y protocolos para sistemas multimedia sobre IP.

La idea de utilizar la infraestructura de una Red de Datos, para llevar tráfico de voz utilizando esa misma red, ha sido por mucho tiempo una ambición de los responsables del área de comunicaciones dentro de una organización.

Una aplicación de transmisión de voz a través de una red IP es la llamada telefonía IP, la cual brinda un nuevo enfoque a lo que conocemos como telefonía, ya que desde un solo dispositivo como es el teléfono IP o un software que haga las veces de este, podremos tener concentrados múltiples servicios no solo los tradicionales como transferencia de

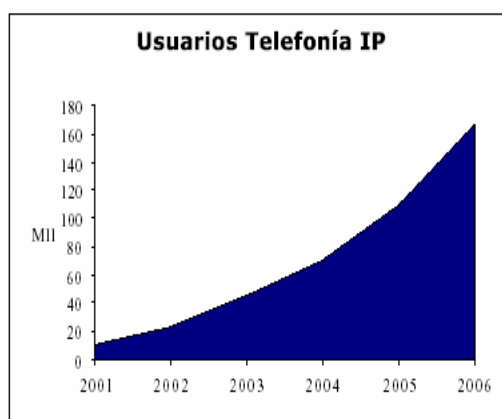
llamadas, identificador de llamadas, sino además dentro de un solo buzón tendremos voice mail y en general multimedia.

Si a esto se agrega que la mayoría de organizaciones tanto públicas como privadas tienen ya como parte de su infraestructura un backbone que además de ser utilizado para transportar datos serviría también para transportar voz.

## 2.2. TENDENCIAS DE LA TELEFONÍA IP

La telefonía IP ha experimentado un crecimiento muy rápido y es por esto que muchos están apostando por esta solución para el futuro.

*Los analistas manejan una previsión de que el número de usuarios de Telefonía IP va a crecer en los próximos años desde los 11 millones que había en el año 2001 (a nivel mundial) hasta los 166 millones previstos para 2005.*



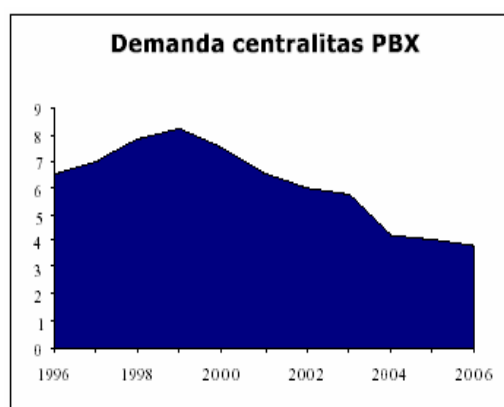
*Fuente: Getronics*

*El tráfico telefónico de llamadas se incrementará desde los 3,4 billones de minutos en 2000 hasta 251 billones 2005.*



*Fuente: Getronics*

*Al contrario que la demanda de soluciones de Telefonía IP, la demanda de centralitas PBX, que presentaba una tendencia creciente hasta el año 1999, presenta una tendencia negativa (hasta un 25%) motivada por el aplazamiento de las inversiones en comunicaciones que adoptaron muchas compañías (el 62% según el estudio de Phillips InfoTech) a la espera de implantar soluciones de Telefonía IP.*



*Fuente: Phillips Infotech, 2002*

### **2.3. ¿CÓMO FUNCIONA LA TELEFONÍA IP?**

Para poder entender como funciona la telefonía IP, se debe entender en que difiere esta, de la telefonía convencional.

La telefonía convencional utiliza una Red Conmutada de Circuitos, cuya principal característica es la de ser un servicio Orientado a la Conexión, es decir que establece una conexión permanente durante la transmisión de voz, a diferencia de la telefonía IP que utiliza una Red Conmutada de Paquetes, cuya principal característica es la de ser un servicio Orientado a la No Conexión, es decir que no necesita una conexión permanente para la transmisión de voz.

*Cuando se realiza una llamada telefónica IP, la voz se digitaliza, se comprime y se envía en paquetes de datos IP, llamados datagramas IP. Los mencionados datagramas viajan hasta la persona con la que se entabla una conversación, cuando estos alcanzan su destino, son ensamblados de nuevo, descomprimidos y convertidos en la señal de voz original.*

*<http://www.recursosvoip.com/intro/index.php>*

### **2.4. ¿PERO QUE ES EL INTERNET PROTOCOL O IP?**

A continuación se da una visión general de Internet Protocol o IP.

### 2.4.1. CARACTERÍSTICAS FUNDAMENTALES

El protocolo IP es el más utilizado para la interconexión entre redes. Su trabajo es proporcionar un medio para el transporte de datagramas del origen al destino, sin importar si estas máquinas están en la misma red, o si hay otras redes entre ellas. IP está implementado en todos los computadores y dispositivos de encaminamiento. Se preocupa de la retransmisión de los datos de un ordenador a otro ordenador, pasando por uno o varios dispositivos de encaminamiento nodo a nodo. No sabe de que aplicación son los paquetes, únicamente sabe de máquina son.

### 2.4.2. IP EN EL MODELO OSI



Figura 2.1. Modelo OSI

### 2.4.3. FORMATO DE DATAGRAMAS IP

De acuerdo con el Modelo OSI, los datos proporcionados por la capa de transporte son divididos en datagramas y transmitidos a través de la capa de red (capa Internet). Durante el camino puede ser fragmentado en unidades más pequeñas, si deben atravesar una red o subred cuyo tamaño de paquete sea más pequeño. En la máquina destino, estas unidades son reensambladas para volver a tener el datagrama original que es entregado a la capa de transporte.

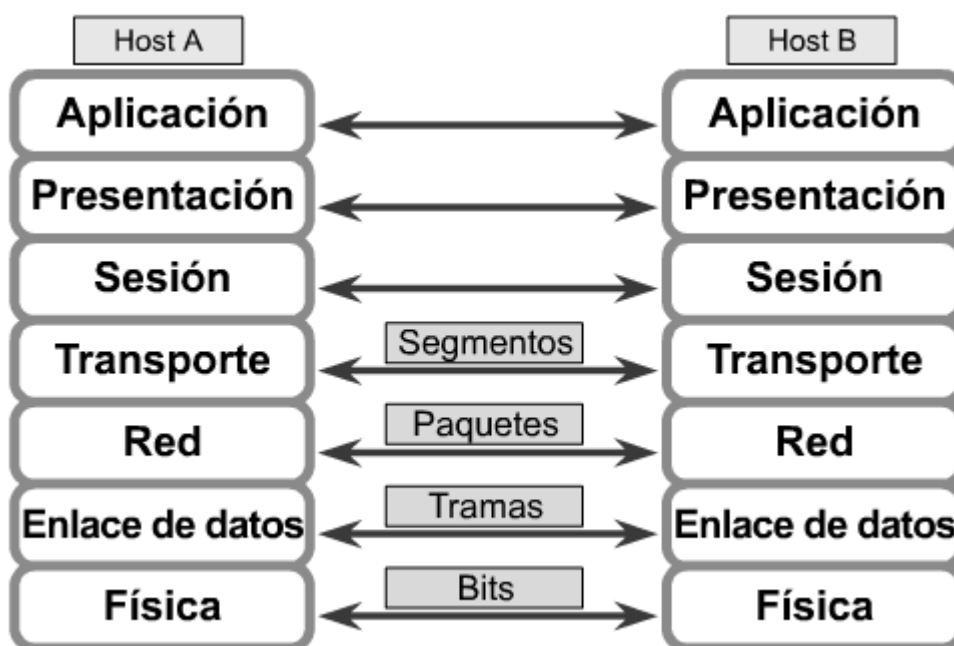


Figura 2.2. Formato de Datagramas IP

El formato de los datagramas IP, consiste en una parte de cabecera y en una parte de datos cuyo tamaño es variable.

### 2.4.3.1. Cabeceras

En la cabecera hay una parte fija de 20 bytes y una parte opcional de longitud variable.

En la siguiente figura se puede ver el formato de la cabecera IP.

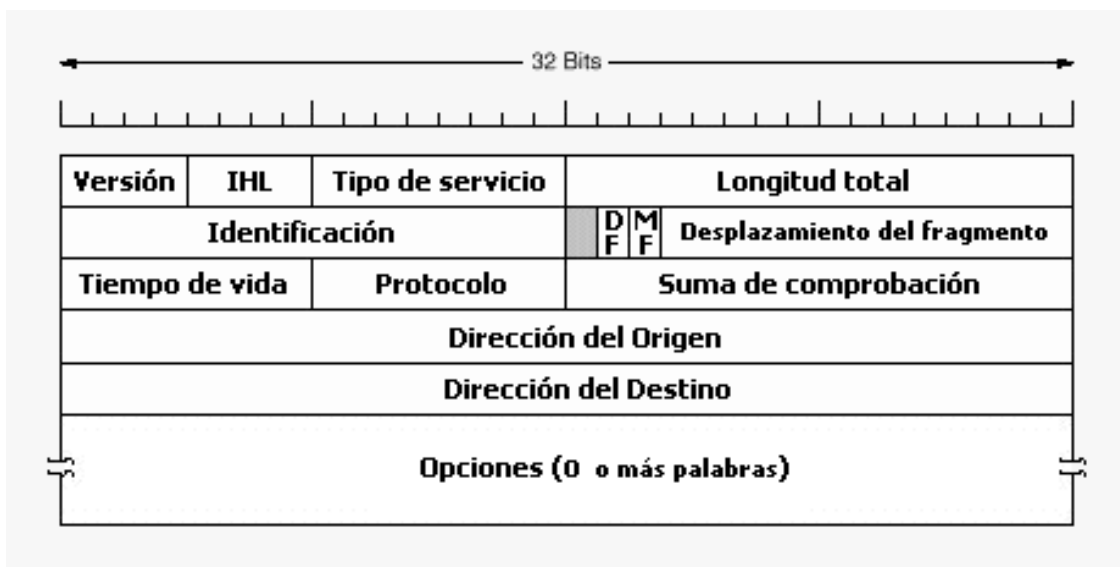


Figura 2.3. Formato de la cabecera IP

A continuación hay una descripción de cada uno de los campos que forman la cabecera:

**Versión (4 bits):** Indica el número de versión del protocolo al que pertenece el datagrama, lo que permitirá la evolución futura del protocolo y que la transición entre las versiones se pueda hacer ejecutándose en unas máquinas la versión antigua y en otras la versión nueva.

**IHL (Internet Header length) (4 bits):** Indica la longitud de la cabecera en palabras de 32 bits (4 bytes). El valor mínimo es cinco ( $20/4=5$ ). Este campo es necesario por no ser constante el tamaño de la cabecera como se comentó anteriormente. El valor máximo

puede ser 15 (1111) lo que limita la cabecera a 60 bytes ( $15 \times 4$ ) y en consecuencia el campo de opciones a 40 (60-20). En el caso de que, por ejemplo, se quiera registrar la ruta de un paquete este valor puede ser insuficiente y ser totalmente inútil esta opción.

**Tipo de servicio (8 bits):** Permite que el host especifique que clase de servicio quiere, pudiéndose combinar confiabilidad y velocidad. Para la voz digitalizada es más importante realizar la entrega de forma rápida que precisa, mientras que para la transferencia de ficheros no importa a que velocidad se realiza la transferencia pero sí que esté libre de errores. De los 8 bits, 3 son para el campo de precedencia que en realidad es una prioridad de 0 (normal) a 7 (para los paquetes de control de red). A continuación aparecen los bits de seguridad (alta o baja), retardo (alto o bajo cuando se intenta minimizar el retardo) y rendimiento (normal o alto cuando se intenta maximizar el rendimiento durante la transmisión del datagrama).

**Longitud total (16 bits):** en bytes que tendrá todo el datagrama, considerando tanto la cabecera como los datos. Hay que tener en cuenta que el tamaño máximo de un datagrama es de 65535 bytes lo que puede ser insuficiente en las redes de alta velocidad.

**Identificador (16 bits):** es un número de secuencia que junto a la dirección origen, la dirección destino y el protocolo de usuario, sirven para que la máquina destino determine a que datagrama pertenece el fragmento que ha recibido. Todos los fragmentos de un datagrama contienen el mismo valor en el campo identificador y este número debe ser único para la dirección origen, la dirección destino y el protocolo de usuario durante el tiempo en el que el datagrama permanece en el conjunto de redes.



**Indicadores (3 bits):** El primer bit no se utiliza actualmente. El indicador de mas fragmentos (MF) cuando vale 1 indica que este datagrama tiene mas fragmentos y toma el valor 0 en el último fragmento. El indicador de no fragmentar (DF) prohíbe la fragmentación cuando vale 1. Es una orden que se le da a los encaminadores de que no fragmenten el datagrama cuando el destino es incapaz de reensamblarlo. Si este bit vale 1, el datagrama se descartará si se excede el tamaño máximo en una subred de la ruta. Por lo tanto, cuando este bit vale 1, es aconsejable usar encaminamiento por la fuente para evitar subredes cuyo tamaño máximo de paquete sea menor que el tamaño del datagrama.

**Desplazamiento del fragmento (13 bits):** Indica en que posición del datagrama original, medido en unidades de 8 bytes (64 bits), va el fragmento actual. Debido a esto, todos los fragmentos excepto el último contienen un campo de datos con una longitud múltiplo de 8 bytes. Como se proporcionan 13 bits, puede haber un máximo de 8912 ( $2^{13}$ ) fragmentos por datagrama, y por lo tanto el tamaño máximo de un datagrama es de 65536 bytes, uno mas que el campo de longitud total.

**Tiempo de vida (8 bits):** Es un contador que sirve para limitar la vida de un paquete. Aunque lo lógico sería pensar que cuenta el tiempo en segundos, en realidad lo que cuenta es el número de saltos de dispositivo de encaminamiento que realiza. Cuando el contador llega a cero, el paquete se descarta y se envía de un paquete al computador origen avisándole. Con este mecanismo se consigue que los datagramas no permanezcan indefinidamente en la red si, por ejemplo, se dañan las tablas de encaminamiento.

**Protocolo (8 bits):** Se utiliza por la capa de red para saber a que protocolo de la capa de transporte le tiene que enviar el datagrama una vez que lo ha reensamblado. Existen diferentes protocolos de transporte, entre ellos TCP y UDP.

**Suma de comprobación (16 bits):** Sirve para verificar el contenido de la cabecera y es útil para la detección de errores generados durante la transmisión del datagrama. Como algunos de los campos de la cabecera pueden cambiar en alguno de los dispositivos de encaminamiento (por ejemplo, el tiempo de vida y algunos campos relacionados con la segmentación), este valor es verificado y recalculado en cada uno de los dispositivos de encaminamiento. El algoritmo empleado consiste en sumar todas las medias palabras de 16 bits a medida que van llegando, usando la aritmética de complemento a 1, y luego obtener el complemento a 1 del resultado. Se supone que la suma de comprobación de la cabecera es cero cuando llega. Este algoritmo es algo más robusto que una suma normal. Existen algunas técnicas para acelerar el cálculo.

**Dirección origen (32 bits):** Indica el número de red y el número del ordenador que envía el datagrama.

**Dirección destino (32 bits):** Indica el número de red y el número del ordenador al que se envía el datagrama.

**Opciones (variable):** Contiene las opciones solicitadas por el usuario que envía los datos y se diseñó para que las versiones posteriores del protocolo pudieran incluir información no considerada originalmente, para que los investigadores pudieran probar cosas nuevas y para que aquella información que es utilizada pocas veces no tuviera asignada unos bits determinados en la cabecera. Cada una de las opciones empieza en 1 byte que identifica la opción. Algunas de las opciones vienen seguidas de un campo de 1 byte para indicar la longitud de la opción y a continuación uno o más bytes de datos. Hay seis opciones (*Seguridad, Encaminamiento estricto desde el origen, Encaminamiento libre*

*desde el origen, Registrar la ruta, Identificación de secuencia, Marca de tiempo*) definidas actualmente pero no todas son reconocidas por todos los dispositivos de encaminamiento:

***Seguridad:*** Permite añadir una etiqueta para indicar lo secreta que es la información que contiene el datagrama. Por ejemplo, se podría utilizar para que los dispositivos de encaminamiento no consideren redes en concreto. Pero en realidad esta etiqueta es ignorada y realmente para lo único que sirve es para ayudar a los espías a encontrar con mayor facilidad la información importante.

***Encaminamiento estricto desde el origen:*** Es una secuencia de direcciones IP que sirve para indicar la trayectoria completa que debe seguir el datagrama desde el origen hasta el destino. Esta opción es usada sobre todo cuando los administradores de sistemas envían paquetes de emergencia porque las tablas de encaminamiento se han corrompido o para hacer mediciones de tiempo.

***Encaminamiento libre desde el origen:*** Es una secuencia de direcciones IP que sirve para indicar que el datagrama debe pasar obligatoriamente por esos dispositivos de encaminamiento y en ese orden, pero también puede pasar por otros dispositivos de encaminamiento. Esta opción es útil cuando por diversas consideraciones se deben pasar por algunos dispositivos de encaminamiento en concreto.

***Registrar la ruta:*** Sirve para indicar que los dispositivos de encaminamiento agreguen su dirección IP al campo de opción y de esta manera tener conocimiento de la ruta seguida por el datagrama. Se utiliza, por ejemplo, para poder determinar si los algoritmos de encaminamiento están funcionando correctamente. Los 40 bytes de tamaño máximo que

puede tener el campo de opciones sólo permiten registrar 9 saltos, lo que puede ser en las redes actuales en muchos casos insuficiente.

**Identificación de secuencia:** Se utiliza cuando hay recursos reservados para un servicio, por ejemplo voz.

**Marca de tiempo:** En este caso, además de registrar las direcciones de los dispositivos de encaminamiento como se hacia en la opción registrar la ruta, se utilizan 32 bits para guardar una marca de tiempo expresada en milisegundos. Esta marca es usada principalmente para buscar fallos en los algoritmos de encaminamiento.

**Relleno (variable):** El campo de opciones se rellena para que su tamaño sea múltiplo de 32 bits (4 bytes).

#### 2.4.3.2. Tipos de servicio

La siguiente tabla muestra los valores del campo tipo de servicio recomendados por defecto.

PROTOCOLO	Tipo de servicio
TELNET	Minimizar retardo
FTP: control	Minimizar retardo
Datos	Maximizar throughput

TFTP	Minimizar retardo
SMTP: fase de comandos	Minimizar retardo
fase de datos	Maximizar throughput
DNS: pregunta UDP	Minimizar retardo
pregunta TCP	Servicio normal
zona de transferencia	Maximizar throughput
NNTP	Minimizar coste monetario
ICMP: errores	Servicio normal
preguntas	Servicio normal
respuesta	El mismo que tenga la pregunta
Cualquier IGP	Maximizar seguridad
EGP	Servicio normal

SNMP	Maximizar seguridad
BOOTP	Servicio normal

**Tabla 2.1. Tipos de Servicios**

#### **2.4.4. DIRECCIONES IP**

Cada computador y cada dispositivo de encaminamiento tendrán una dirección única cuya longitud será de 32 bits, que será utilizada en los campos dirección origen y dirección destino de la cabecera. Esta dirección consta de un identificador de red y de un identificador de computador. La dirección, como puede verse en la siguiente figura, está codificada para permitir una asignación variable de los bits utilizados al especificar la red y el computador. Este formato de direcciones permite mezclar las tres clases de direcciones en el mismo conjunto de redes. La dirección IP más pequeña es la 0.0.0.0 y la mayor es 255.255.255.255.

Existen tres clases de redes que se pueden clasificar teniendo en cuenta la longitud del campo de red y del campo ordenador. La clase a la que pertenece una dirección puede ser determinada por la posición del primer 0 en los cuatro primeros bits. Las direcciones están codificadas para permitir una asignación variable de bits para especificar la red y el ordenador.

**Clase A:** Pocas redes, cada una con muchos ordenadores. 7 y 24 bits (+1). Por ejemplo ARPANET.

**Clase B:** Un número medio de redes, cada una con un número medio de ordenadores. 14 y 16 bits (+2)

**Clase C:** Muchas redes, cada una con pocos ordenadores. 21 y 8 bits (+3). Por ejemplo un red de área local.

**Clase D:** Permite hacer multitransmisión (o multicasting) en la cual el datagrama se dirige a múltiples ordenadores. Podemos enviar un paquete IP a un grupo de máquinas que por ejemplo pueden estar cooperando de alguna manera mediante la utilización de una dirección de grupo.

**Clase E:** Reservado para el futuro.

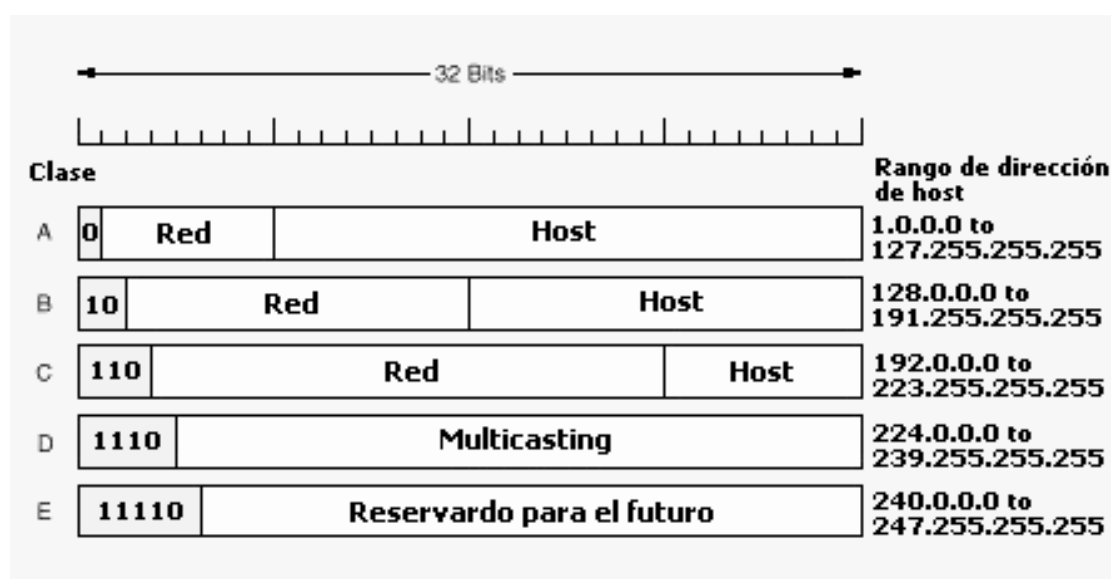


Figura 2.4. Clases de Direcciones IP

La siguiente tabla muestra el número de redes y de ordenadores por red en cada una de las tres clases primarias de direcciones IP:

CLASE	BITS EN EL PREFIJO	MAXIMO N° DE REDES	BITS EN EL SUFIJO	MAXIMO N° DE ORDENADORES POR RED
A	7	128	24	16777216
B	14	16384	16	65536
C	21	2097152	8	256

**Tabla 2.2. Número de redes y de ordenadores por red**

Normalmente las direcciones se suelen escribir en notación decimal con puntos. Por ejemplo, la dirección 82CE7C0D (1000 0010 1100 1110 0111 1100 0000 1101 que es de clase B) se escribe como 130.206.124.13.

$$82 = 8 * 16 + 2 = 128 + 2 = 130$$

$$CE = C * 16 + E = 12 * 16 + 14 = 192 + 14 = 206$$

$$7C = 7 * 16 + C = 112 + 12 = 124$$

$$0D = D = 13$$



Observando la figura anterior puede verse que no todas las direcciones han sido asignadas a una clase en concreto. Algunas de estas direcciones se utilizan como direcciones especiales:

***Este ordenador:*** La dirección 0.0.0.0 significa esta red o este ordenador y únicamente es usada por los ordenadores cuando son arrancados, sin que se vuelva a utilizar posteriormente. De esta forma las máquinas se pueden referir a su propia red sin saber su número, pero tiene que saber su clase para saber cuantos ceros debe incluir.

***Un ordenador de esta red:*** Poniendo el campo red todo a ceros (es necesario saber la clase de la red para decidir cuantos ceros se deben poner).

**Difusión de red local o limitada:** La dirección 255.255.255.255 (todos 1s) se usa como dirección para indicar todos los ordenadores de la red indicada y es utilizada para hacer difusión.

***Difusión de una red distante o dirigida:*** También se puede hacer difusión a una red distante poniendo la dirección de la red y rellenando el campo ordenador con 1's.

***Retrociclo:*** Las direcciones 127.xx.yy.zz se reservan para pruebas de realimentación. Los paquetes que tienen esta dirección no son enviados por la red sino que son procesados localmente y se tratan como si fueran paquetes de entrada. Esto permite que los paquetes se envíen a la red local sin que el transmisor conozca su número. Esta característica también se usa para la detección de fallos en el software de red.

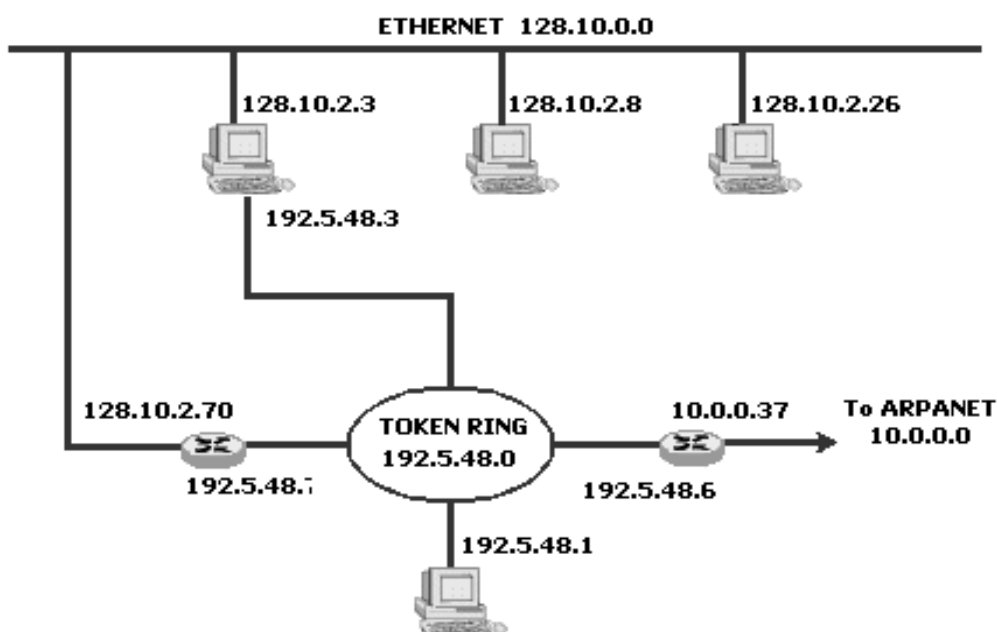
Este ordenador	<b>Todos 0's</b>	
Ordenador de esta red	<b>Todos 0's</b>	<b>Ordenador</b>
Difusión limitada	<b>Todos 1's</b>	
Difusión dirigida	<b>Red</b>	<b>Todos 1's</b>
Retroalimentación	<b>127</b>	<b>Cualquier cosa</b>

**Figura 2.5. Direcciones Especiales**

Para estar seguros de que la dirección Internet es única, todas las direcciones de Internet son asignadas por una autoridad central. El Internet Assigned Number Authority (IANA) tiene el control sobre los números asignados. Sin embargo, cuando una organización quiere una dirección debe obtenerla de INTERNIC (Internet Network Information Center). La autoridad central solo es necesaria para asignar la porción de la dirección correspondiente a la red, cuando una organización ya tiene su prefijo, puede asignar un único sufijo a cada ordenador sin contactar con la autoridad central.

Una máquina puede estar conectada a varias redes y tener una dirección IP diferente en cada red. En este caso recibe el nombre de “multihomed”. Esto se utiliza para aumentar la seguridad pues si una red falla el ordenador aún esta conectado a Internet utilizando la otra red. Por otra parte, también es usado para aumentar el rendimiento de la red pues permite enviar directamente el tráfico a una red en concreto sin tener que pasar por los dispositivos de encaminamiento.

En la siguiente figura podemos ver las direcciones asignadas a diferentes ordenadores y dispositivos de encaminamiento que están conectados en tres redes diferentes: a una Token Ring de clase C, a una Ethernet de clase B y a una red de clase A, en concreto Arpanet. Para la interconexión de las tres redes se están utilizando 2 dispositivos de encaminamiento (routers). Además la máquina A es una máquina multihomed pues está conectada directamente a dos redes, a la Token Ring y a la Ethernet.



**Figura 2.6. Direcciones asignadas a diferentes ordenadores y dispositivos de encaminamiento**

El que la dirección de la red esté guardada en la dirección Internet tiene algunos inconvenientes:

Si la dirección IP identifica la red a la que se conecta el ordenador, no al ordenador que tenemos conectado, no es posible asignarle a un ordenador una dirección IP permanente. Por lo tanto, si movemos un ordenador de una red a otra su dirección IP debe cambiar. Este

problema se da cuando por ejemplo cuando nos llevamos un ordenador portátil de un sitio a otro y queremos conectarlo a la red.

Como el número de ordenadores asignados a la clase C (255) puede resultar insuficiente en muchos casos y que la transición a la clase B no es fácil debido a que muchos programas no permiten que una red física tenga múltiples direcciones, no se pueden introducir nuevas direcciones poco a poco y es necesario reconfigurar toda la red para la nueva clase.

Como existe la facilidad de que una máquina pueda estar conectada a dos redes y por lo tanto tenga dos direcciones diferentes, y que el encaminamiento se hace teniendo en cuenta la dirección IP, el comportamiento de los paquetes puede ser totalmente diferente dependiendo de la dirección que estemos utilizando. Esto puede resultar sorprendente para los usuarios.

[http://eia.udg.es/~atm/tcp-ip/tema\\_4\\_5.htm](http://eia.udg.es/~atm/tcp-ip/tema_4_5.htm)

En vista que se ha hecho una revisión de Protocolo de Internet o IP, se verá a continuación cuales son los protocolos utilizados por Voz sobre IP o VoIP.

## **2.5. PROTOCOLOS DE VOZ SOBRE IP O VOIP**

Los principales protocolos utilizados para la Voz sobre IP son:

- Protocolo H.323
- Protocolo SIP
- Protocolo H.248 (Megaco)

### **2.5.1. PROTOCOLO H.323**

A finales de 1997 el VoIP forum del IMTC (Internacional Multimedia Telecommunications Consortium) llego a un acuerdo que permite la interoperabilidad de los distintos elementos que pueden integrarse en una red VoIP. Debido a la ya existencia del estándar H.323 del ITU-T (for Telecommunication Standardization Sector of the International Telecommunications Union), que cubría la mayor parte de las necesidades para la integración de la voz, se decidió que el H.323 fuera la base del VoIP. De este modo, el VoIP debe considerarse como una clarificación del H.323, de tal forma que en caso de conflicto, y a fin de evitar divergencias entre los estándares, se decidió que H.323 tendría prioridad sobre el VoIP. El VoIP tiene como principal objetivo asegurar la interoperabilidad entre equipos de diferentes fabricantes, fijando aspectos tales como la supresión de silencios, codificación de la voz y direccionamiento, y estableciendo nuevos elementos para permitir la conectividad con la infraestructura telefónica tradicional. Estos elementos se refieren básicamente a los servicios de directorio y a la transmisión de señalización por tonos multifrecuencia (DTMF).

El VoIP/H.323 comprende a su vez una serie de estándares y se apoya en una serie de protocolos que cubren los distintos aspectos de la comunicación:

***Direccionamiento:***

*RAS (Registration, Admission and Status)*. Protocolo de comunicaciones que permite a una estación H.323 localizar otra estación H.323 a través de el Gatekeeper.

*DNS (Domain Name Service)*. Servicio de resolución de nombres en direcciones IP con el mismo fin que el protocolo RAS pero a través de un servidor DNS

***Señalización:***

*Q.931* Señalización inicial de llamada

*H.225* Control de llamada: señalización, registro y admisión, y paquetización / sincronización del stream (flujo) de voz

*H.245* Protocolo de control para especificar mensajes de apertura y cierre de canales para streams de voz

***Compresión de voz:***

*Requeridos:* G.711 y G.723

*Opcionales:* G.728, G.729 y G.722

**Transmisión de voz:**

*UDP.* La transmisión se realiza sobre paquetes UDP, pues aunque UDP no ofrece integridad en los datos, el aprovechamiento del ancho de banda es mayor que con TCP.

*RTP (Real Time Protocol).* Maneja los aspectos relativos a la temporización, marcando los paquetes UDP con la información necesaria para la correcta entrega de los mismos en recepción.

**Control de la transmisión:**

*RTCP (Real Time Control Protocol).* Se utiliza principalmente para detectar situaciones de congestión de la red y tomar, en su caso, acciones correctoras.



**Figura 2.7. Pila de protocolos en VoIP**

Hasta ahora sólo hemos visto la posibilidad de utilizar nuestra red IP para conectar las centralitas a la misma, pero el hecho de que VoIP se apoye en un protocolo de nivel 3, como es IP, nos permite una flexibilidad en las configuraciones que en muchos casos está todavía por descubrir. Una idea que parece inmediata es que el papel tradicional de la

centralita telefónica quedaría distribuido entre los distintos elementos de la red VoIP. En este escenario, tecnologías como CTI (computer-telephony integration) tendrán una implantación mucho más simple. Será el paso del tiempo y la imaginación de las personas involucradas en estos entornos, los que irán definiendo aplicaciones y servicios basados en VoIP.

Unitronics Comunicaciones

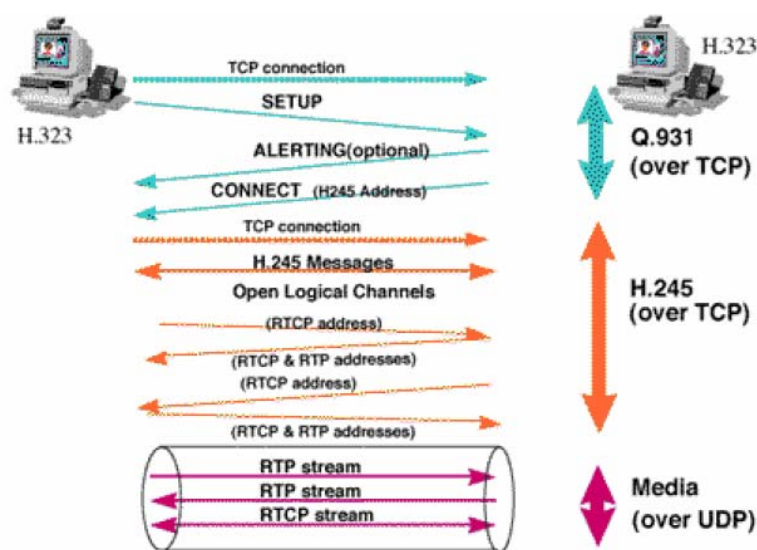


Figura 2.8. Protocolo H.323

**2.5.2. PROTOCOLO SIP**

El protocolo de señalización SIP (Session Initiation Protocol) del IETF, es mucho más sencillo en su concepción y funcionamiento que la solución ITU equivalente, pero también con menor grado de normalización y penetración en el mercado que ésta.

Con SIP es posible implementar, sobre redes IP, servicios telefónicos básicos y avanzados.



Su sencillez resulta muy atractiva, y con un futuro prometedor, en tanto que se vislumbra su utilización para el soporte de otros tipos de servicios y también en los sistemas de Tercera Generación de Comunicaciones Móviles.

### 2.5.2.1. Aspectos generales

SIP (RFC 2543) [1] es un protocolo de control del nivel de aplicación, concebido para la señalización y el control de llamadas, esto es, para el control del establecimiento, modificación y terminación de sesiones o llamadas multimedia. En la figura 1 se muestra su ubicación en la pila de protocolos de telefonía IP.

SIP forma parte de las especificaciones del IETF para comunicaciones multimedia, conjuntamente con otros protocolos como RSVP (Resource reReservation Protocol), RTP (Real Time Protocol), SDP (Session Description Protocol), etc., pero su funcionalidad no depende de ninguno de éstos. Soporta comunicaciones entre usuarios de redes IP, y también, con el empleo de pasarelas, con usuarios de otras redes, por ejemplo, con terminales de las redes telefónicas convencionales.

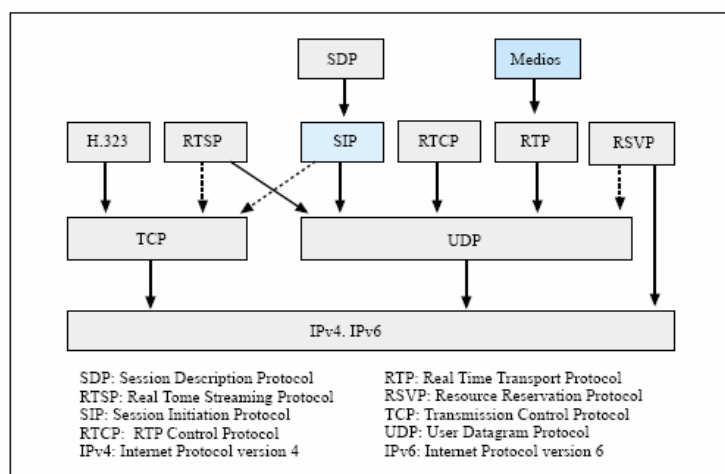


Figura 2.9. Arquitectura de protocolos de Telefonía IP

SIP es neutral en relación con los protocolos de las capas inferiores, por lo que puede soportarse sobre TCP (Transmisión Control Protocol) o UDP (User Datagram Protocol), igualmente sobre IP, ATM (Asynchronous Transfer Mode), F-R (Frame Relay) o X.25. Recientemente se ha presentado una propuesta para valorar la utilización de SCTP (Stream Control Transmission Protocol) como protocolo de transporte para SIP.

Múltiples transacciones SIP pueden ser soportadas en una simple conexión TCP, o en una asociación SCTP posiblemente.

En cambio, un simple datagrama UDP solo porta un mensaje SIP. Los datagramas UDP, incluyendo todas las cabeceras, no deben ser mayores que la MTU (Media Transmisión Unit) del trayecto, si es que ésta se conoce, o no mayores de 1500 bytes si no se conoce.

SIP es un protocolo basado en texto, utiliza el juego de caracteres ISO 10646 con codificación UTF-8 (RFC 2279), lo que posibilita una fácil implementación y depuración, lo hace flexible y extensible. El sobre encabezamiento que implica usar un protocolo basado en texto no tiene aquí mayor trascendencia, pues SIP es un protocolo de señalización, no es un protocolo para el trasiego de datos de usuario.

SIP soporta, entre otras cosas:

- Correspondencia (“mapping”) de nombres y servicios de re direccionamiento,
- Movilidad de usuarios y de terminales,
- Llamadas con múltiples interlocutores a través de MCU’s (Multipoint Control

Unit),

- Servicios suplementarios y de red inteligente,
- Identificación de usuarios con URI's (Uniform Resource Identifier), posibilitando a éstos iniciar una llamada haciendo un simple "click" sobre un enlace web,

- Operación stateless o stateful,

- Cualquier sintaxis en el cuerpo de sus mensajes,

- Ser utilizado conjuntamente con otros protocolos de señalización,

- Sus propios mecanismos de seguridad.

Genéricamente, las comunicaciones multimedia con el protocolo SIP comprenden:

- Localización del usuario,
- Determinación de la disponibilidad del usuario para la comunicación,
- Determinación de los medios para la comunicación, esto es, las capacidades de usuario,
- Establecimiento de los parámetros de la comunicación entre las partes involucradas,

- Manipulación de las llamadas, es decir, establecimiento, modificación y terminación de llamadas.

### 2.5.2.2. Elementos funcionales

Los elementos funcionales en la arquitectura SIP son:

- Agentes de Usuario (User Agent, UA), y
- Servidores de red

Los Agentes de Usuario son aplicaciones que residen en las estaciones terminales SIP, y contienen dos componentes: Agentes de Usuario Clientes (User Agent Client, UAC) y Agentes de Usuario Servidores (User Agent Server, UAS).

Los UAC originan las solicitudes SIP (asociados al extremo que origina la llamada), y los UAS responden a estas solicitudes, es decir, originan respuestas SIP (asociados al extremo que recibe la llamada). Los UA's deben implementar el transporte tanto sobre TCP como sobre UDP, y quizás también sobre SCTP [4].

Los UAC's y UAS's pueden, por si solos y sin los servidores de red, ser capaces de soportar una comunicación básica (modelo de llamada básico, directamente entre endpoints). No obstante, la potencialidad de SIP se aprovecha con el empleo de los servidores de red.

Los servidores de red se clasifican, desde un punto de vista lógico, de la manera siguiente:

- Servidores de redirección,
- Servidores proxy, y
- Servidores de registro

### ***Servidores de redirección***

*(Redirect Server)*: procesan mensajes INVITE, que son solicitudes SIP, y retornan la dirección (o direcciones) de la parte llamada, esto es, el SIP – URL (Uniform Resource Locator) de la parte llamada, o cómo contactar con ella (respuesta 3xx). De lo contrario rechaza la llamada, enviando una respuesta de error (error de cliente 4xx o error de servidor 5xx). Desarrollan una funcionalidad similar al Gatekeeper H.323 cuando en la solución ITU se emplea el modelo de llamada directo.

### ***Servidores proxy***

*(Proxy Server)*: corren un programa “intermediario” que actúa como servidor y como cliente, respecto al llamante se comporta como servidor, y respecto al llamado como cliente. Un servidor Proxy puede re-enviar solicitudes hasta el destino final sin efectuar cambio alguno en ellas, o cambiar alguno de sus parámetros si se requiere, por ejemplo, en el caso de las cabeceras “Via” “Record Route”. Los servidores proxy desarrollan el “routing” de los mensajes de solicitudes y respuestas SIP. Pueden ser “stateful” o “stateless”.

Los servidores proxy stateful retienen información de la llamada durante el tiempo que dure el establecimiento de ésta, no así los servidores proxy stateless, los que procesan un mensaje SIP y entonces “olvidan” todo lo referente a la llamada hasta que vuelven a recibir otro mensaje SIP asociado a la misma. Esto se refiere al “estado” de la llamada, sin embargo, pueden mantener un “estado” para una simple transacción SIP, lo que es denominado “minimal state”. La implementación stateless provee buena escalabilidad, pues los servidores no requieren mantener información referente al estado de la llamada una vez que la transacción ha sido procesada. Además, esta solución es muy robusta dado que el servidor no necesita “recordar” nada en relación con una llamada.

Sin embargo, no todas las funcionalidades pueden ser implementadas en un servidor proxy stateless, por ejemplo, las funcionalidades relativas a la contabilización y facturación de las llamadas puede requerir funcionalidades proxy stateful, de manera que se le pueda “seguir el rastro” a todos los mensajes y estados de una comunicación.

### ***Servidores de registro***

(*Registrar Server*): registran las direcciones SIP (SIP – URL) y sus direcciones IP asociadas, es decir, garantizan el “mapping” entre direcciones SIP y direcciones IP. Típicamente están co localizados con servidores Proxy o servidores de redirección.

Es un servidor que acepta solo mensajes de solicitud REGISTER, posibilitando el registro correspondiente a la localización actual de los usuarios, esto es, “seguir el rastro” de los usuarios, pues por diferentes razones (conexión vía ISP, usuarios móviles, conexión vía LAN con DHCP) las direcciones IP de éstos puede cambiar.

También se les denomina servidores de localización (Location Server), pues son utilizados por los servidores proxy y de redirección para obtener información respecto a la localización o localizaciones posibles de la parte llamada.

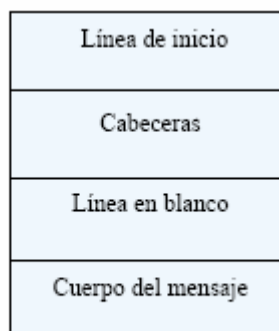
Ahora bien, en rigor, los Location Server (LS) no son servidores SIP, ni entidades SIP, si no bases de datos, que pueden formar parte de arquitecturas de comunicaciones que utilicen SIP.

Entre un LS y un servidor SIP no se utiliza el protocolo SIP, por ejemplo, en ocasiones se emplea entre éstos el protocolo LDAP (Lightweight Directory Access Protocol). La información registrada en los servidores de registro, esto es, el registro del mapping de direcciones SIP correspondiente a un usuario, no es permanente, requiere ser “refrescado” periódicamente, de lo contrario, vencido un “time out” (por defecto, una hora), el registro correspondiente será borrado. Este valor por defecto del “time out” puede ser modificado según valor que se especifique en la cabecera “Expires” de un mensaje de solicitud REGISTER. En consecuencia, para mantener la información de registro, el terminal (o el usuario) necesita refrescarlo periódicamente. Igualmente, un registro vigente puede ser cancelado y/o renovado por el usuario. Usualmente, un servidor de red SIP implementa una combinación de los diferentes tipos de servidores SIP ya comentados: servidor proxy + servidor de registro y/o servidor de redirección + servidor de registro. En cualquier caso deben implementar el transporte sobre TCP y UDP.

### **2.5.2.3. Mensajes**

Los mensajes SIP, solicitudes (métodos) y respuestas (códigos de estado), emplean el formato de mensaje genérico establecido en la RFC 822, esto es:

- Una línea de inicio
- Uno o más campos de cabeceras (header)
- Una línea vacía (indica final del campo de cabeceras)
- Cuerpo del mensaje (opcional)



Mensaje genérico (RFC 822)

**Figura 2.10. Mensaje Genérico RFC 822**

#### 2.5.2.4. Solicitudes (Métodos) SIP:

El formato de los mensajes de solicitud es como sigue:

Request-Line

\*( general-header| request-header| entity-header)

CRLF

[ message-body ]



La línea “*Request-Line*” tiene la estructura siguiente:

***Method SP Request-URI SP SIP-Version*** CRLF donde “SP” es el carácter “espacio” y “CRLF” es la secuencia “retorno del carro” y “nueva línea”.

Veamos cada uno de estos tres componentes de la línea “Request-Line”.

***Método (Method)***: son seis los métodos básicos definidos en SIP, en su núcleo (RFC 2543), a saber: INVITE, ACK, OPTIONS, BYE, CANCEL, REGISTER.

Otros métodos, ya normalizados o en fase draft, están más allá de la RFC 2543, es decir, son extensiones al núcleo básico de SIP, y no se presentan aquí.

***INVITE***: invita a un usuario, o servicio, a participar en una sesión. El cuerpo del mensaje contiene, generalmente, una descripción de la sesión.

***ACK***: confirma que el cliente solicitante ha recibido una respuesta final desde un servidor a una solicitud INVITE, reconociendo la respuesta como adecuada.

Solo para reconocer solicitudes INVITE, y no otros mensajes de solicitud.

***OPTIONS***: posibilita “descubrir” las capacidades del receptor.

***BYE***: finaliza una llamada, o una solicitud de llamada. Puede ser enviado por el agente llamante o por el agente llamado.

*CANCEL*: cancela una solicitud pendiente, pero no afecta una solicitud ya completada. Este método finaliza una solicitud de llamada incompleta.

*REGISTER*: se utiliza este método como un servicio de localización que registra la localización actual de un usuario.

Los métodos que no sean soportados por servidores, Proxy o de redirección, son tratados por éstos como si se tratase de un método *OPTION*, y en consecuencia reenviados

Los métodos que no sean soportados por los servidores UAS o Registrar, provocan el mensaje de respuesta 501, “no implementado”.

*Request-URI*: es un SIP-URL o un URI (Uniform Resource Identifier). Indica el usuario o servicio al que la solicitud está siendo dirigida. A diferencia de la cabecera “To”, el “Request-URI” puede ser re-escrito por servidores proxy.

Típicamente, el UAC establece el “Request-URI” y la cabecera “To” al mismo SIP-URL, pero puede no ser así.

Los servidores proxy o de redirección pueden usar la información en los campos *Request-URI* y cabeceras de solicitud para manejar la solicitud, y posiblemente re-escribir el *Request-URI*. Los servidores proxy son responsables del procesamiento del *Request-URI*.

*SIP-Version*: da cuenta de la versión del protocolo SIP en uso, y se incluye tanto en mensajes de solicitud (métodos) como en mensajes de respuesta (códigos de estado).

En la figura se muestra el formato general de los mensajes de solicitud o métodos SIP.

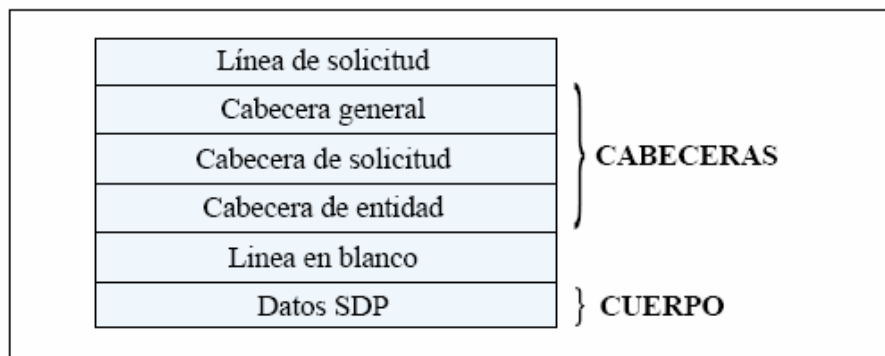


Figura 2.11. Formato general de los mensajes de solicitudes SIP.

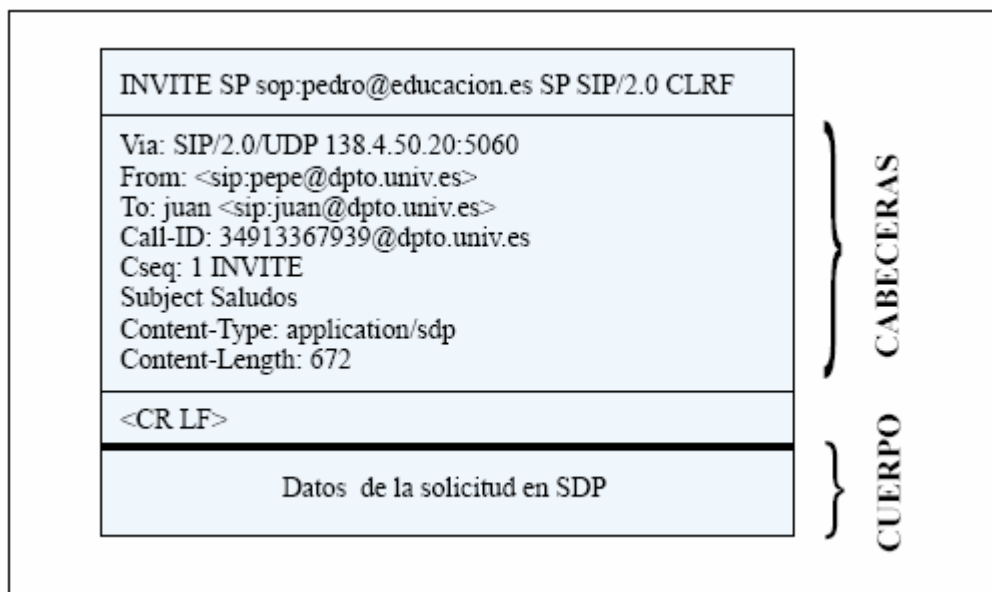


Figura 2.12. Ejemplo de un mensaje de solicitud SIP

Obsérvese los tres campos de cabeceras. La figura 2.12 es un ejemplo de una solicitud INVITE.

*Respuestas (Códigos de estado) SIP:* Después que se recibe e interpreta un mensaje de solicitud SIP, el receptor del mismo (servidor SIP) responde con un mensaje (o varios) de respuesta (código de estado) El formato de los mensajes de respuesta es como sigue:

Status-Line

\*( general-header| response-header| entity-header )

CRLF

[ message-body ]

La línea “*Status-Line*” comprende:

*SIP-Version* SP *Status-Code* SP *Reason-Phrase* CRLF donde “SP” es el carácter “espacio” y “CRLF” es la secuencia CRLF.

*SIP-Version:* versión del protocolo SIP

*Status-Code:* código de tres enteros para ser interpretado por máquinas. Indica el resultado de comprender y satisfacer o no una solicitud. Hay seis tipos diferentes de *Status-Code*:

*1xx:* Informativo. Solicitud recibida, se continua para procesar la solicitud. Por ejemplo, 180, RINGING.

2xx: Solicitud exitosa. La solicitud (acción) fue recibida de forma adecuada, comprendida y aceptada. Por ejemplo, 200, OK

3xx: Re direccionado. Más acciones deben ser consideradas para completar la solicitud. Por ejemplo, 302, MOVED TEMPORARILY.

4xx: Error de cliente. La solicitud contiene mal la sintaxis o no puede ser resuelta en este servidor. Por ejemplo, 404, NOT FOUND.

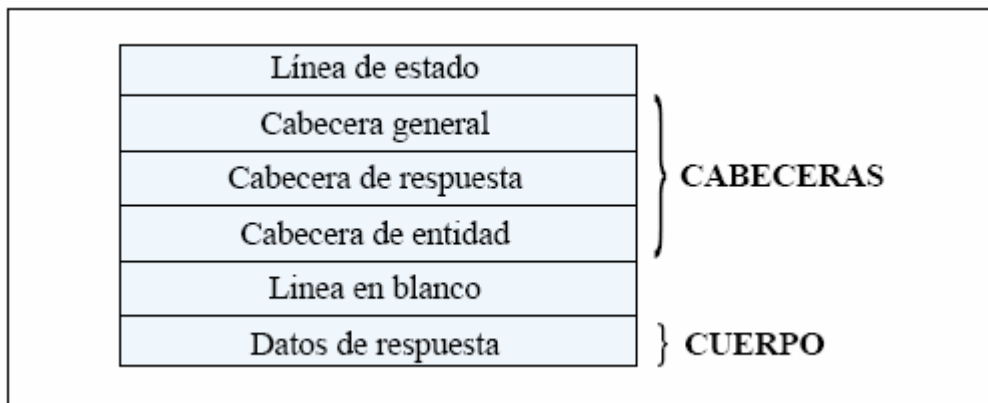
5xx: Error de servidor. El servidor ha errado en la resolución de una solicitud aparentemente válida. Por ejemplo, 501, NOT IMPLEMENTED.

6xx: Fallo global. La solicitud no puede ser resuelta en servidor alguno. Por ejemplo, 600, BUSY EVERYWHERE.

Los mensajes respuestas 2xx, 3xx, 4xx, 5xx y 6xx son “respuestas finales”, y terminan la transacción SIP. En cambio, los mensajes de respuestas 1xx’s son “respuestas provisionales”, y no terminan la transacción SIP.

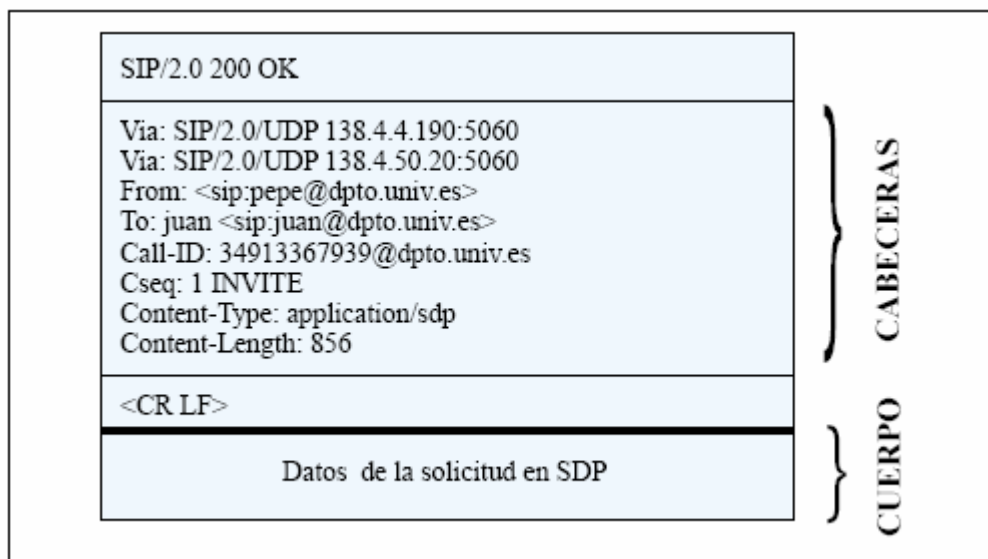
*Reason-Phrase*: explicación textual muy breve del Status- Code, para ser interpretada por humanos.

En la figura 2.13 se muestra el formato general de los mensajes de respuestas SIP. Obsérvese la similitud en cuanto a los campos de cabeceras en relación con los métodos SIP.



**Figura 2.13. Formato general de los mensajes de respuesta SIP.**

Por otra parte, la figura 2.14 muestra un ejemplo de mensaje de respuesta SIP correspondiente al código de estado 200, OK.



**Figura 2.14. Ejemplo de un mensaje de respuesta SIP.**

### 2.5.2.5. Cabeceras

Las cabeceras SIP son similares a las cabeceras utilizadas en el protocolo HTTP (Hyper Text Transfer Protocol), tanto en la sintaxis como en la semántica. Especifican aspectos referentes a los participantes, trayectos, etc.

Generalmente, el orden en que aparecen las cabeceras no tiene mayor importancia, siempre que se cumpla que las cabeceras del tipo “salto a salto” (hop-by-hop) deben aparecer antes que cualquier cabecera del tipo “extremo a extremo” (end-to-end). Las primeras pueden ser modificadas o añadidas por los servidores proxy, en cambio las segundas deben ser transmitidas por éstos sin modificación alguna. Los servidores proxy no deben re-ordenar las cabeceras, pero si pueden adicionar ciertos tipos de cabeceras, como por ejemplo la cabecera “Via”, así como otras del tipo “salto a salto” (hop-by-hop). Si bien pueden modificar ciertas cabeceras, no deben alterar los campos que son autenticados. Determinadas cabeceras están presentes en todos los mensajes, otras no, solo en algunos. Igualmente, una aplicación que contenga el protocolo SIP no requiere necesariamente tener que comprender todas las cabeceras, aunque si es deseable.

En el mismo sentido, si un participante SIP no entiende una cabecera, la ignora. Las cabeceras no especificadas deben ser ignoradas por los servidores. El número total de cabeceras SIP definidas hasta ahora es 46, que rebasa el número de 37 cabeceras de la especificación inicial de SIP (RFC 2543) [1], pudiendo ser agrupadas de la manera siguiente:

***Cabeceras generales (general headers):*** se utilizan tanto en los mensajes de solicitud como en los mensajes de respuesta. Estas son: *Call-ID, Contact, CSeq, Date, Encryption, From, Organization, Retry-After, Sub-ject, Supported, Timestamp, To, User Agent y Via.*

**Cabeceras de “entidad” (entity headers):** dan información adicional referente al cuerpo del mensaje, o si éste no está presente, entonces a cerca del recurso identificado por la solicitud.

Son las siguientes: *Allow, Content-Encoding, Content- Length, Content-Type, Content-Disposition, Expires* y *MIME-Version*. Cabeceras de solicitud (request headers): posibilitan que el cliente pase información adicional al servidor referente a la solicitud, y también a cerca del propio cliente. Actúan como un modificador de solicitud. Son las siguientes: *Accept, Accept- Encoding, Accept-Language, Accept-Contact, Authorization, Hide, In-Reply-To, Max-Forwards,Priority, Proxy-Authorization, Proxy-Require, Record- Route, Reject-Contact, Request- Disposition, Require, Response- Key, Route, Rack* y *Session-Expires*.

**Cabeceras de respuesta (response headers):** posibilitan al servidor pasar información adicional en relación a la respuesta, información que no puede situar en el campo “Status-Line”. Da información del servidor y también referente al recurso identificado por “Request-URI”. Son éstas: *Proxy-Authenticate, Server,Unsupported, Warning, WWW-Authenticate* y *Rseq*.

A continuación se comentan las cinco cabeceras generales más significativas, que no deben faltar en un mensaje SIP: Call-ID: identifica, unívocamente, una llamada. Se utiliza para diferentes propósitos, por ejemplo:

- para hacer corresponder respuestas con las correspondientes solicitudes,
- para detectar duplicados de solicitudes INVITE,



- para cambiar dinámicamente parámetros de una sesión

*Cseq*: identifica cada solicitud. Está compuesta por un número decimal y el nombre del método. En una sesión SIP *Cseq* se incrementa por cada nueva solicitud, excepto en solicitudes ACK y CANCEL. La respuesta a una determinada solicitud lleva el mismo *Cseq* que dicha solicitud, esto es, el UAS copia el valor del *Cseq* de la solicitud en la correspondiente respuesta, por lo que la *pareja relacionada solicitud- respuesta* tiene el mismo valor de cabecera *CSeq*.

*From*: identifica el origen de la solicitud en la pareja relacionada solicitud-respuesta, por tanto siempre se refiere al origen de la solicitud. Contiene la dirección del origen y, opcionalmente, el nombre de éste. Está presente en todas las solicitudes y respuestas.

*To*: identifica el destino de una solicitud. Debe estar presente en todos los mensajes de solicitudes y respuestas.

*Via*: se emplea para registrar la ruta de una solicitud, en aras de posibilitar a los servidores

SIP que intervienen re-enviar las respuestas por la misma ruta seguida por la solicitud. En consecuencia, cada proxy adiciona una nueva cabecera “Via”. Esta cabecera es una de las potencialidades más importantes del protocolo SIP, mostrando que éste ha sido diseñado teniendo muy en cuenta la interconexión IP.

### 2.5.2.6. Cuerpo de mensajes

Los mensajes SIP, solicitudes y respuestas, opcionalmente pueden contener un cuerpo de mensaje, como ya sido indicado. Generalmente éste es una descripción de sesión con SDP, pero puede ser cualquier otro contenido, en forma “clara” o cifrado. El contenido del cuerpo de mensaje solo es de interés para los UA’s, no para los servidores de red, si de funcionalidades SIP se trata, pues éstos para encaminar los mensajes SIP solo necesitan conocer los contenidos de la *línea de solicitud* o de la *línea de estado*, según el caso, y de las cabeceras.

### 2.5.2.7. Modo de operación

Ya antes se planteó que los UAs pueden, por si solos y sin los servidores de red, ser capaces de soportar una comunicación básica, esto es, directamente entre endpoints. Pero también se dejó establecido que la potencialidad de SIP se aprovecha con el empleo de los servidores de red.

Entonces, la señalización SIP puede desarrollarse por intermedio de servidores proxy, o servidores de redirección, a través de los cuales se llevan a efecto las transacciones SIP. De manera muy resumida, la operatividad SIP es la siguiente:

- un UAC emite una solicitud,
  
- un servidor proxy interviene en la localización de la parte llamada,
  
- un UAS acepta (o rechaza) la llamada, o
  
- un UAC emite una solicitud,

- un servidor de redirección notifica la dirección de la parte llamada,
- un UAS acepta (o rechaza) la llamada, Por lo que se requiere la localización de servidores y usuarios SIP, invitaciones SIP, servicios de registro SIP, transacciones SIP, direcciones SIP, etc. Direcciones SIP: los “objetos” direccionados a través del protocolo SIP son usuarios en hosts, que se identifican mediante SIP

– URL’s del tipo user@host, donde:

- user: nombre de usuario o número telefónico;
- host: nombre de dominio o dirección numérica de red.

Los SIP- URL’s son utilizados en los mensajes SIP para indicar, en relación con una solicitud: origen (from), destino actual (request- URI) y destino final (to). También para especificar direcciones de “redirección”.

Por otra parte, los campos de cabeceras SIP pueden no contener SIP-URL’s, por ejemplo, en el caso que una llamada desde un teléfono convencional se trasvase por Internet con el empleo del protocolo SIP, entonces la cabecera “from” puede contener un URL telefónico.

**Localización de un servidor SIP:** cuando un cliente SIP desea enviar una solicitud, debe:

- enviar ésta a un servidor proxy configurado localmente, o

- enviar dicha solicitud a la dirección IP y puerto que corresponda, por lo que es evidente que el software cliente debe disponer de una, o de un conjunto de direcciones SIP relativas a servidores SIP.

**Transacciones SIP:** una vez efectuada la localización de un servidor SIP, tienen lugar las transacciones, entendiéndose como tal el conjunto de solicitudes (o solicitud) enviadas por el cliente al servidor, y las respuestas (o respuesta) retornadas por éste al cliente. Se trata de solicitudes y respuestas *relacionadas*, o lo que es lo mismo, con determinados parámetros idénticos (por ejemplo, call-ID, Cseq, to, from). Si se emplea TCP como protocolo de transporte, la solicitud (o solicitudes) y la respuesta (o respuestas) de una transacción usan la misma conexión TCP. Varias solicitudes desde el mismo cliente y al mismo servidor pueden emplear la misma conexión TCP, o para cada transacción una conexión TCP diferente. Esto es, una conexión TCP puede soportar una o varias transacciones entre un cliente y un servidor.

**Invitaciones SIP:** una invitación SIP consta de dos solicitudes, por tanto corre a cargo de un cliente. Estas parejas de solicitudes son INVITE y ACK, o INVITE y BYE, según el caso. La solicitud INVITE contiene, generalmente, la descripción de la sesión, descrita con formato SDP. Esta descripción informa a la parte llamada qué tipo de medios el solicitante puede aceptar, y dónde éste desea le sean enviados los datos.

**Localización de usuarios SIP:** dado que el usuario llamado puede desplazarse entre diferentes terminales, con un Servidor de Localización estas localizaciones pueden registrarse dinámicamente.

Para esto puede usarse uno de diferentes protocolos, por ejemplo, *finger*, LDAP, etc. Un Servidor de Localización puede retornar varias localizaciones, bien porque el usuario esté “loggeado” en diferentes hosts al mismo tiempo, o porque el Servidor de Localización tenga temporalmente información no actualizada.

Con esta información, y según el tipo de servidor SIP, proxy o de redirección, se procede así:

- Servidor de redirección: retorna dicha información de dirección al cliente solicitante, y éste procede a direccionar a la parte llamada.
- Servidor proxy: secuencialmente, o en paralelo, intenta éste direccionar a la parte llamada.

**Servicio de registro SIP:** posibilita que el cliente, mediante una solicitud REGISTER, notifique a un servidor, proxy o de redirección, su ubicación. Ya se ha comentado antes que usualmente en la práctica un servidor de localización o registro acompaña a los servidores proxy y redirección. Nótese que, tanto uno como otro, debe consultar a un servidor de localización para poder desarrollar su tarea.

#### **2.5.2.8. Implementación mínima**

Una implementación mínima de SIP debe cumplir, en relación con los elementos funcionales clientes y servidores, lo siguiente:

**CLIENTES:** deben ser capaces de generar las solicitudes INVITE y ACK, así como las cabeceras Call-Id, Content-Length, Content-Type, Cseq, Require,

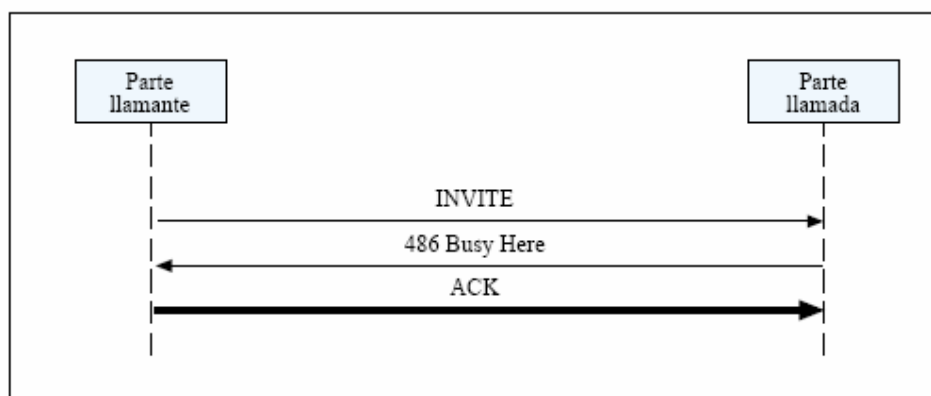
From y To. También deben “entender” el protocolo SDP y ser capaces de reconocer las clases 1 hasta la 6 de los *status code*.

SERVIDORES: deben “entender” las solicitudes INVITE, ACK, OPTIONS y BYE. De tratarse de servidores proxy, también la solicitud CANCEL..

También deben ser capaces de generar de manera apropiada las cabeceras Call-Id, Content- Length, Content-Type, CSeq, Expires, From, Max-Forwards, Require, To y Via.

### 2.5.2.9. Ejemplos de señalización SIP

En las figuras 2.11, 2.12, 2.13 y 2.14 se muestran ejemplos genéricos simplificados de transacciones SIP, directamente entre UA's (figuras 2.15 y 2.16), y a través de servidores SIP (figuras 2.17 y 2.18).



**Figura 2.15. Llamada SIP directamente entre terminales (UA's). Llamada no aceptada, parte solicitada ocupada**

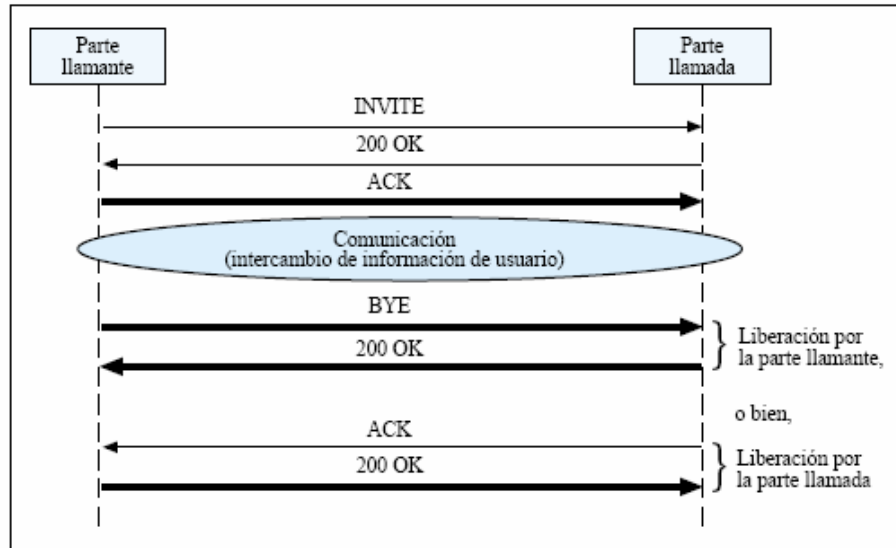


Figura 2.16. Establecimiento y liberación de llamada con SIP directamente entre terminales (UA's)

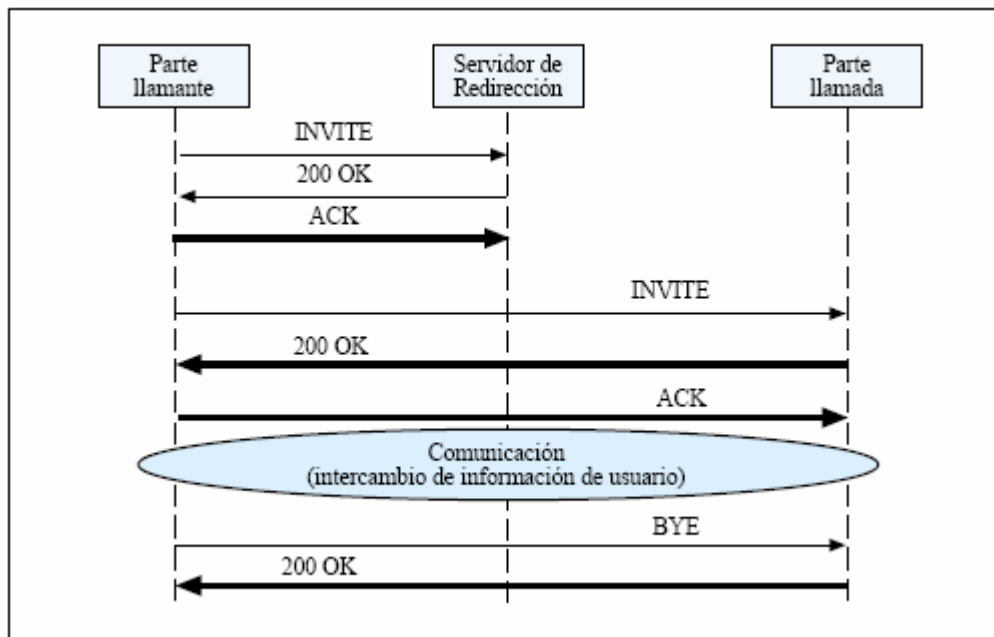
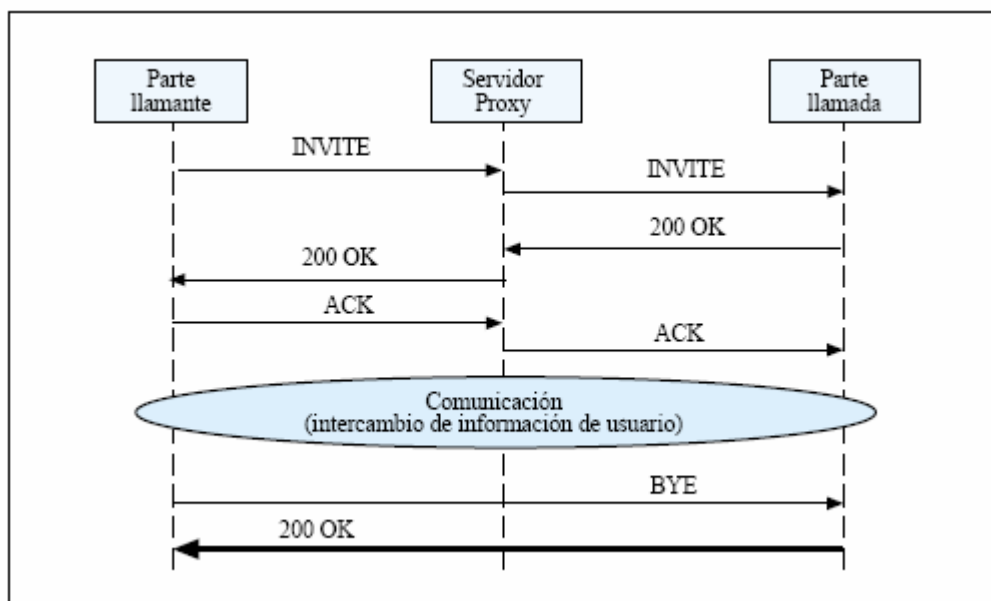


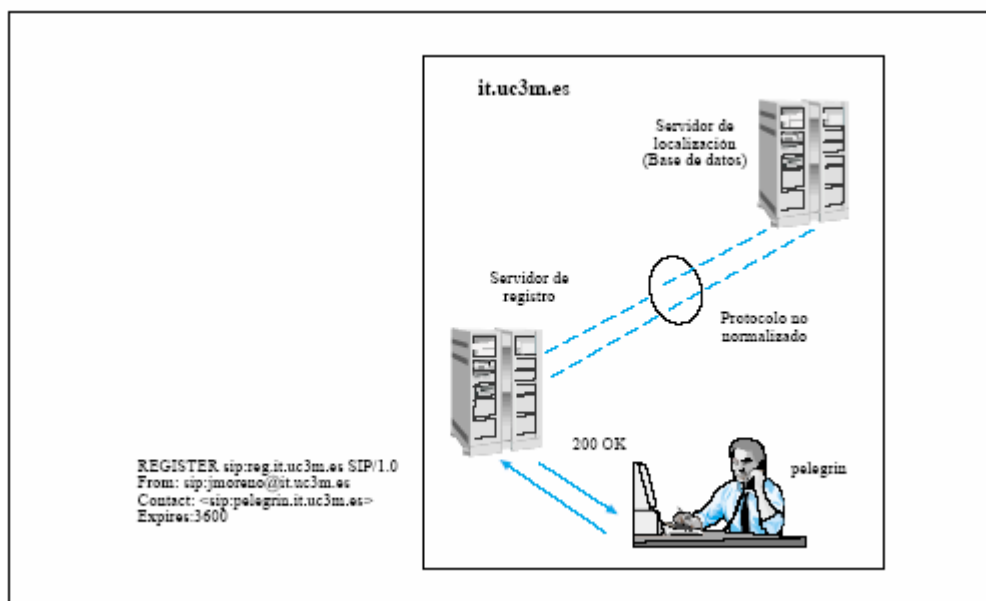
Figura 2.17. Llamada SIP a través de Servidor de Redirección. Llamada terminada por la parte solicitante



**Figura 2.18. Llamada SIP a través de Servidor Proxy. Llamada terminada por la parte solicitante**

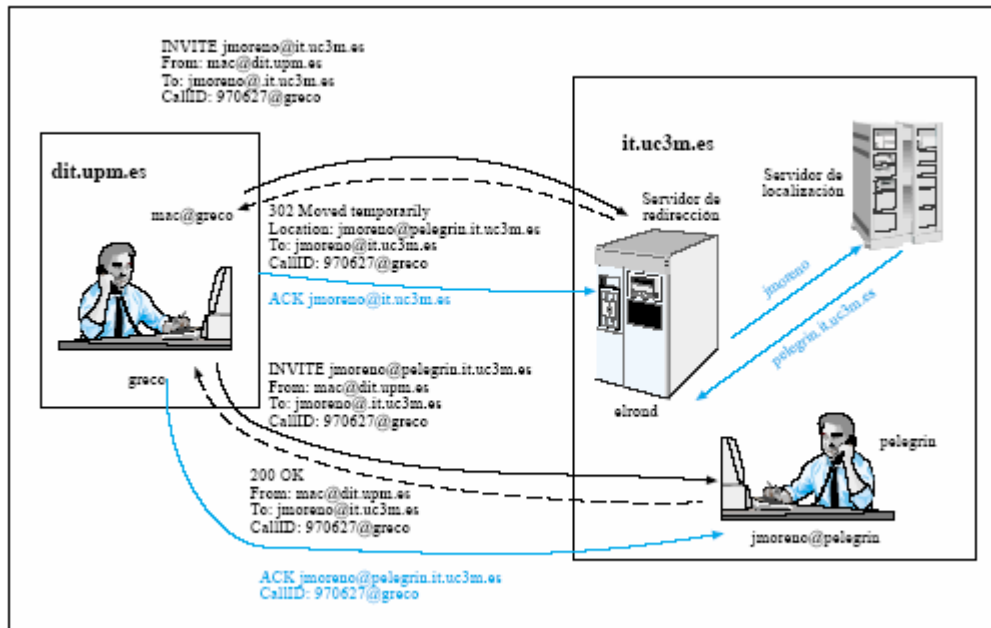
Por otra parte, en las figuras 2.19, 2.20 y 2.21 se muestran gráficamente tres ejemplos específicos de sesiones de señalización SIP. En cada caso se indican también parte de los contenidos de algunas cabeceras de los mensajes SIP intercambiados. La figura 2.19 muestra el proceso de registro a través de un servidor de registro. El usuario *jmoreno@it.uc3m.es* registra su nueva ubicación *pelegrin.it. uc3m.es*. Obsérvese que el protocolo entre este servidor y el servidor de localización es un protocolo no normalizado, no es el protocolo SIP.





**Figura 2.19. Proceso de registro a través de un servidor de registro**

En la figura 2.20 se ejemplifica el establecimiento de una sesión SIP por intermedio de un servidor de redirección. El usuario *mac@dit.upm.es* desea comunicar con el usuario cuyo SIP URL es *jmoreno@it.uc3m.es*, pero que de acuerdo a la figura 2.19 éste ahora está localizable a través del SIP URL *jmoreno@pelegrin.it.uc3m.es*, como lo indica la respuesta del servidor de redirección. En consecuencia, la solicitud se re envía a la dirección IP donde está ubicado ahora dicho usuario, y entre ambos usuarios se desarrolla la señalización SIP para establecer la sesión.



**Figura 2.20. Establecimiento de una sesión SIP por intermedio de un servidor de redirección**

La figura 2.21 corresponde al caso para llamada SIP a través de servidor proxy. Éste, después de consultar el servidor de localización, re envía la solicitud a la ubicación actual del usuario llamado. A través del servidor proxy se desarrolla todo el proceso de señalización SIP para establecer la sesión.

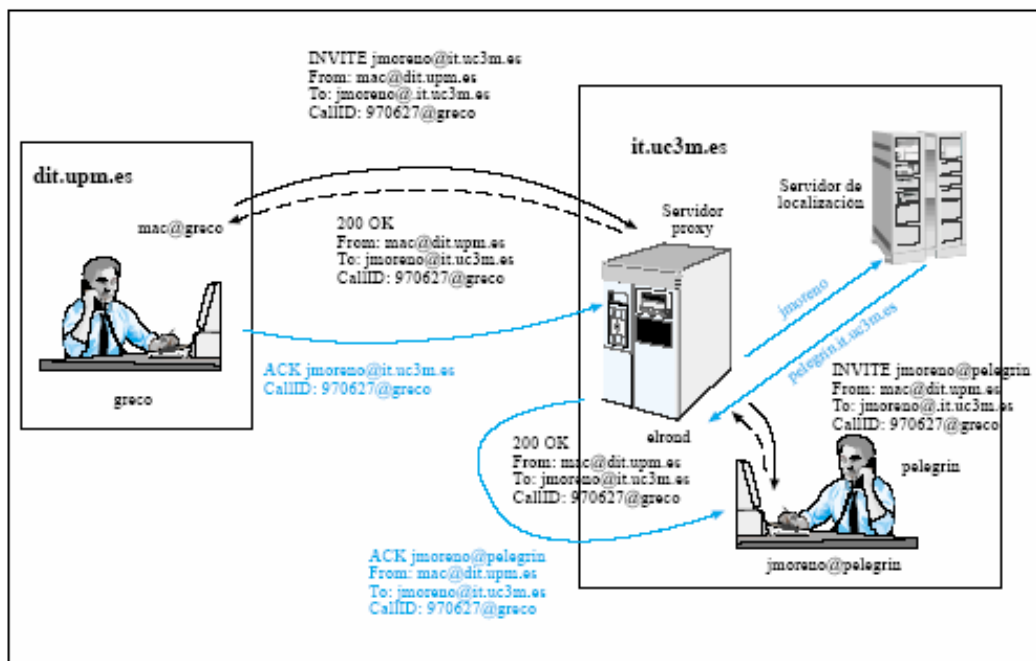


Figura 2.21. Llamada SIP a través de un servidor Proxy

[www.ahciet.net/revista/91/009.pdf](http://www.ahciet.net/revista/91/009.pdf)

Autores: Manuel Moreno Martín, Manuel Álvarez-Campana Fernández-Corredor, Joan Vinyes Sanz

La siguiente tabla hace una comparación entre H.323 y SIP

H.323	SIP
Protocolo Complejo	Protocolo mas Simple
Representación Binaria	Representación Textual
No demasiado modular	Muy Modular
No demasiado escalable	Muy Escalable
Muchos elementos	solo 37 cabeceras
Difficil detección de bucles	Fácil detección de bucles
Muy Extendido	Poco extendido

Tabla 2.3. Tabla comparativa entre H.323 y SIP

### 2.5.3. PROTOCOLO H.248/MEGACO

H.248/MEGACO. Un nuevo Standard H.248, también conocido como MEGACO es el resultado de la cooperación entre la UIT (Unión Internacional de Telecomunicaciones) y la IETF (Internet Engineering Task Force) y se podría contemplar como un protocolo complementario a los protocolos SIP y H.323.

Si bien en sí es neutral con respecto a los protocolos de señalización de llamada utilizados para el control del establecimiento de la llamada, la norma H.248 forma parte de los esfuerzos desplegados por la UIT para completar la familia de normas H.323, ahora bien reconocida, con nuevos conjuntos de funciones que tienen por finalidad responder a las necesidades del mercado, comprendidos el fax en tiempo real, la movilidad, la seguridad o la comunicación de textos, así como con mejores medios para realizar operaciones que ya admitía la norma H.323.

Las implementaciones de pasarelas H.323 tendrán ahora una mayor escalabilidad porque la norma H.248 descompone la función pasarela H.323 en subcomponentes funcionales y especifica los protocolos utilizados por estos componentes para comunicarse. Además de su alto grado de flexibilidad, la norma H.248 también se usa para interconexión con las redes de conmutación de circuitos existentes y hace que su aplicación sea rápida, fácil y rentable para los operadores de redes. Y como ahora, la norma H.248 permite que las pasarelas H.323 incluyan componentes suministrados por múltiples vendedores, distribuidos a través de múltiples plataformas físicas, se espera que los consumidores se beneficien de mejores ofertas de servicios y menores costos. MGCP y H.248/MEGACO fueron diseñados para proveer una arquitectura donde el control de la llamada y los servicios pudieran estar centralmente añadidos a una red VoIP. En ese sentido, una

arquitectura usando estos protocolos se asemeja a la existente PSTN y sus servicios. En una arquitectura centralizada, MGCP y H.248/MEGACO permite a las compañías construir redes a grandes escalas, escalables, flexibles y redundantes. Esta provee mecanismos para interconectarse con otras redes VoIP, para añadir inteligencia y características al “Call Agent” o “Softswitch”.

### 2.5.3.1. Componentes que intervienen

\*Media Gateways (Pasarela de medios) (MG): Son los elementos funcionales que median entre los puntos finales, es decir, los clientes.

\*Media Gateway Controller (Controlador de la pasarela de medios) (MGC): Controlaran a los Media Gateways. El MGC también se suele llamar Call Agent o SoftSwitch.

Una de las características fundamentales de este modelo, es que las Pasarelas de Media son capaces (en teoría) de mantener comunicaciones tanto con el H.323 como con el SIP, algo fundamental para la óptima implantación del sistema VoIP.

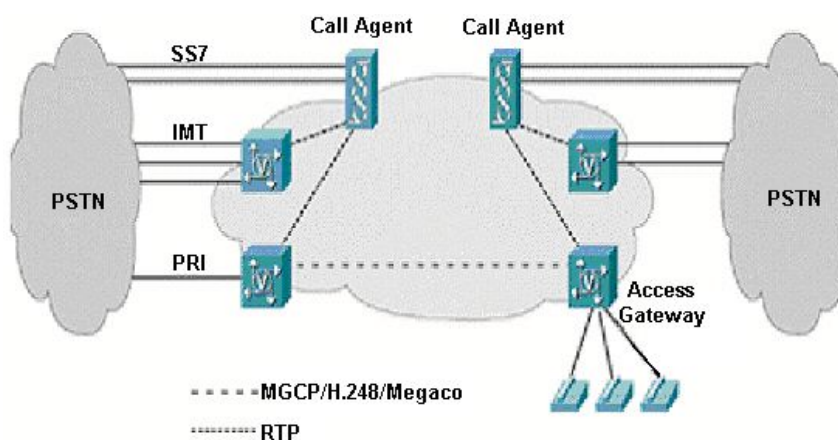


Figura 2.22. Red H.248

[www.cujae.edu.cu/revistas/telematica/Articulos/100.htm](http://www.cujae.edu.cu/revistas/telematica/Articulos/100.htm)

Autores: Marco Fong Lucero, MSc. Maite Martín López

Si un usuario desea realizar una llamada, mediante este sistema, los pasos que se realizan son los siguientes:

1. El usuario descuelga el teléfono y marca el número de teléfono del destinatario. Esta llamada, le llega al Media Gateway.
2. El Media Gateway, notifica al Media Gateway Controller de que una llamada está en camino.
3. El Media Gateway Controller busca en su base de datos, el número del teléfono del destinatario para saber su IP y su número de puerto. Entonces, busca el Media Gateway del destinatario, y le envía un mensaje para indicarle que le esta llegando una llamada.
4. El Media Gateway del destinatario, abre una RTP (Protocolo en tiempo real) cuando el usuario descuelga. El Media Gateway, permite tener múltiples teléfonos conectados (algo muy útil para las empresas).

## **CAPITULO III**

### **3. SISTEMA EMULADOR DE LA CENTRAL TELEFONICA IP**

En este capítulo se presentará la propuesta técnica para el diseño e implementación del Sistema Emulador de la Central Telefónica IP.

#### **3.1. INTRODUCCIÓN**

El Sistema Emulador de la Central Telefónica IP, será implementado mediante el software especializado de nombre VOCAL (Vovida Open Communication Application Library) de distribución libre es decir Open Source, que funciona sobre la plataforma GNU/Linux.

#### **3.2. ANTECEDENTES DE VOCAL**

VOCAL o Vovida Open Communication Application Library fue diseñado por Vovida Networks ([www.vovida.org](http://www.vovida.org)), que era una compañía independiente hasta 1999, y luego formaría parte en el año 2000 de Cisco Systems.

Vovida debe su nombre a la fusión de la primera sílaba de tres palabras Voz, Video y Datos, y es un grupo internacional de talentosos programadores que desarrollan software Open Source para aplicaciones de Voz sobre IP VoIP.

### 3.3. QUE ES VOCAL?

El sistema VOCAL es una red distribuida de servidores que proveen Voz sobre IP VoIP, para servicios telefónicos; este sistema soporta dispositivos que se comunican usando Session Initiation protocol o SIP, además de dispositivos que utilizan otros protocolos de VoIP como son MGCP y H.323 a través de translators e incluso soporta teléfonos análogos, a través de gateways residenciales.

Como se muestra en la figura, VOCAL ofrece las siguientes características:

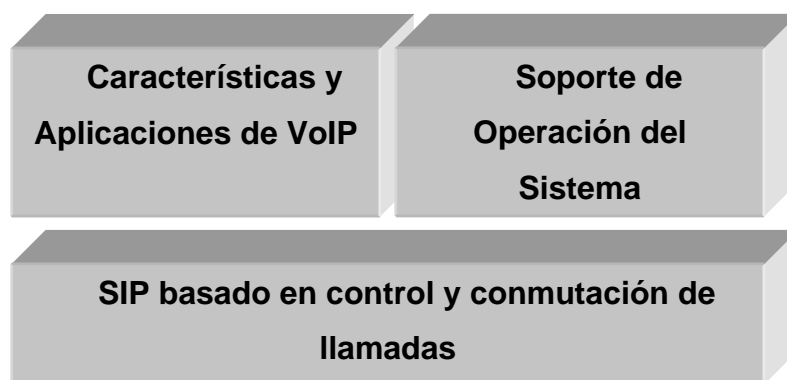


Figura 3.1. Características Generales de VOCAL

#### *Características y Aplicaciones de VoIP*

- Características básicas como: reenvío de llamadas, bloqueo de llamadas, transferencia de llamadas y llamada en espera.
- Características de VoIP como: voice mail.
- Una librería de software para nuevas característica y aplicaciones creadas en: C++, Call processing language (CPL), Java telephony API (JAPI).

#### *Soporte de Operación del Sistema*



- La configuración del sistema VOCAL se realiza desde un navegador de Internet a través de web GUI (Graphical User Interface).
- Monitoreo de los elementos de red, utilizando un gestor de red con SNMP (Simple Network Management Protocol).
- Adicción y gestión de abonados, con sus correspondientes características.
- Autenticación de abonados.

#### *SIP basado en control y conmutación de llamadas*

- Registro de usuarios.
- Iniciación de llamada.
- Modificación de llamada.
- Terminación de llamada.

### **3.4. ARQUITECTURA DE VOCAL**

El sistema VOCAL contiene dos tipos de servidores: *Proxy* y *Redirect*.

El *Servidor Proxy* contiene dos tipos de servidores, el *Servidor Marshal* y *Servidor Feature*.

El *Servidor Redirect* contiene dos tipos de servidores, el *SIP Redirect* y *Servidor Location*.

VOCAL también posee una interfaz gráfico o GUI Graphical User Interface, desarrollado en Java; que permite la configuración y administración del sistema.

### **3.4.1. SERVIDORES DE VOCAL**

Los servidores principales que contiene el sistema VOCAL son los que se detalla a continuación:

#### **3.4.1.1. Servidores Marshal**

Los servidores Marshal son los primeros dispositivos que reciben todo ingreso de señales, autenticación de usuarios y reenvío de señales autenticadas dirigidas para el servidor Redirect; además reciben y reenvían señales de otros servidores dentro del sistema VOCAL.

En VOCAL tenemos cuatro tipos de servidores Marshal, estos son:

##### **Servidor Gateway Marshal**

Trabaja con señales que vienen desde y van hacia el gateway de la red PSTN.

##### **Servidor Intenetwork Marshal**

Trabaja con señales que vienen desde y van hacia un servidor SIP Proxy conocido en otra red IP.

##### **Servidor User Agent Marshal**

Trabaja con ambos tipos de señales, las que vienen desde y van hacia teléfonos IP que están conectados a la red.

#### **3.4.1.2. Servidores Feature**

Provee importantes características telefónicas como son reenvío y bloqueo de llamadas.

### 3.4.1.3. Servidores Redirect

Este servidor guarda la información de los usuarios quienes son registrados en la red y proveen la información de ruteo, para ayudar al ingreso y salida de llamadas que arriban de destinos fijos.

La siguiente figura muestra una versión simplificada del sistema VOCAL:

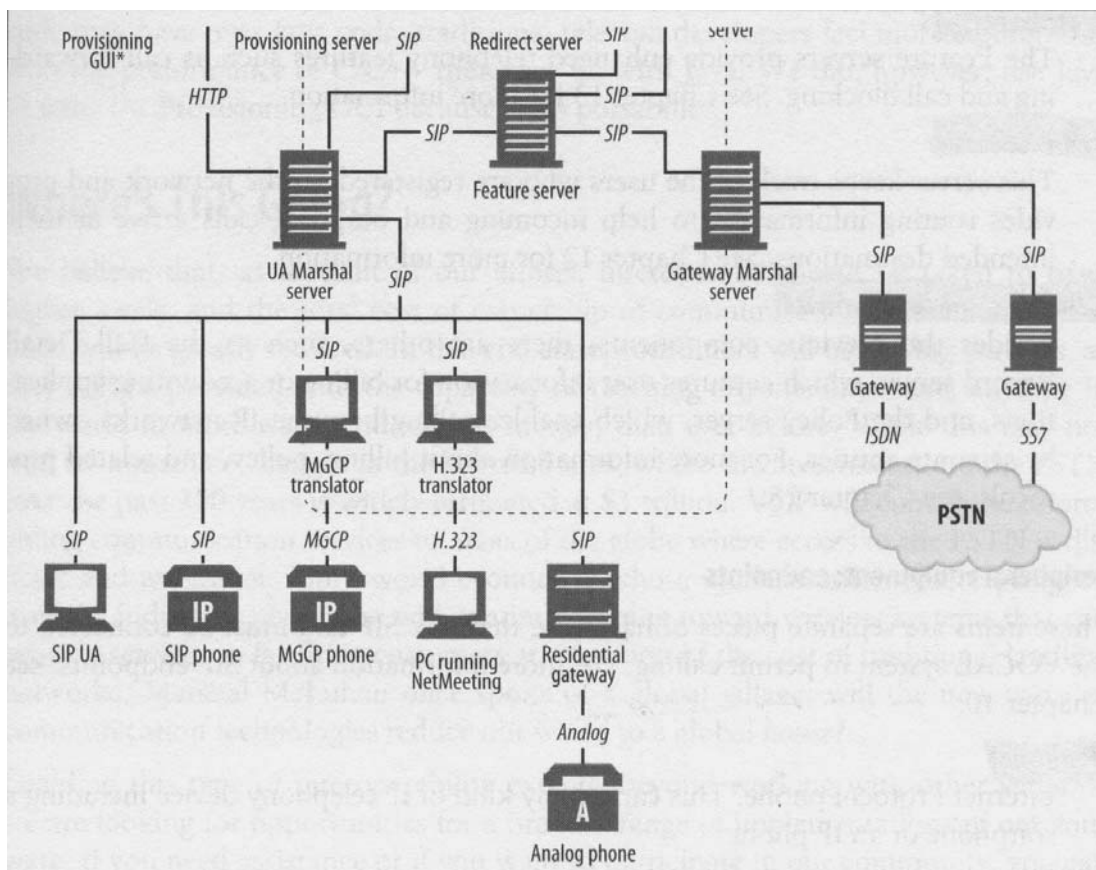


Figura 3.2. Versión Simplificada del Sistema VOCAL

### 3.4.2. INTERFAZ GRAFICO DE VOCAL

El sistema VOCAL posee un interfaz gráfico para su configuración; GUI o Graphical User Interface, fue desarrollado con Java, y consta de dos niveles de acceso: *Administrador* y *Técnico*, como muestra la siguiente figura:

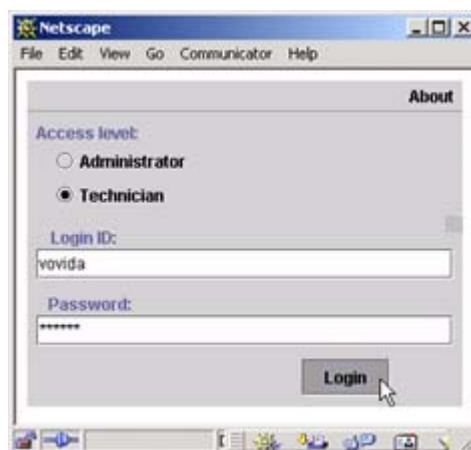


Figura 3.3. Interfaz gráfico con sus niveles de acceso

### 3.4.2.1. Nivel de Acceso de Administrador

El nivel acceso Administrador permite principalmente realizar la configuración de usuarios, es decir: añadir, borrar y modificar los mismos; como se muestra en la siguiente figura:

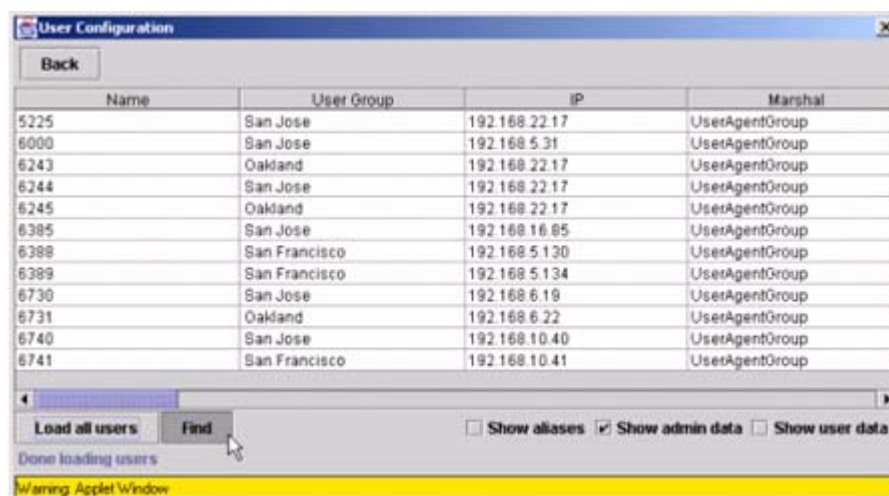


Figura 3.4. Nivel de Acceso de Administrador

### 3.4.2.2. Nivel de Acceso de Técnico

El nivel acceso Técnico permite principalmente realizar la configuración del sistema y de los servidores de VOCAL, es decir: añadir, borrar y modificar tanto características del sistema, así como los servidores del mismo; como se muestra en la siguiente figura:

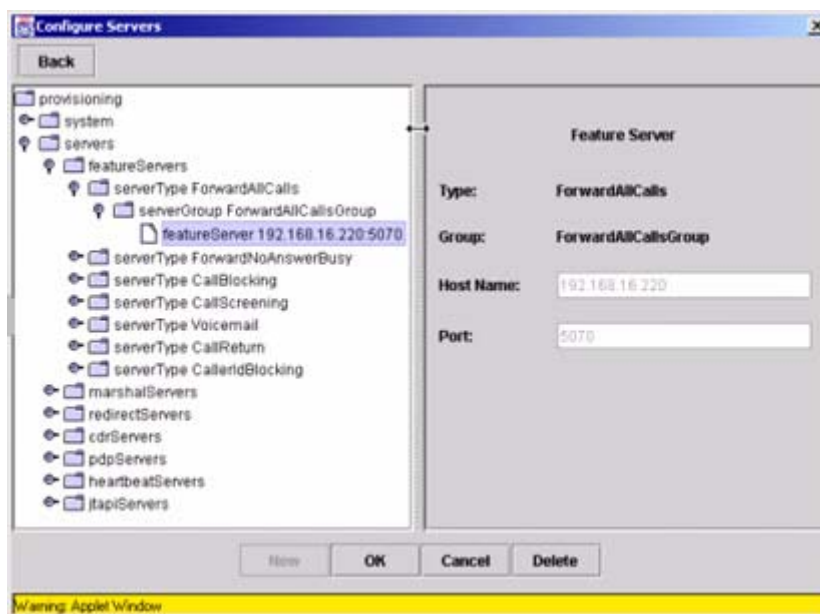


Figura 3.5. Nivel de Acceso de Técnico

### 3.5. FUNCIONAMIENTO DE VOCAL

A continuación se indica paso por paso una llamada básica SIP, utilizando el sistema VOCAL:

- a) El usuario A marca el número del usuario B, el teléfono SIP A envía un mensaje INVITE para el servidor Marshal.
- b) El servidor Marshal A reenvía el mensaje INVITE para el servidor Redirect.
- c) El servidor Redirect responde con un mensaje 302 que contiene información del servidor Marshal A para contactar al servidor Marshal B.
- d) El servidor Marshal A reenvía un mensaje INVITE para el servidor Marshal B.
- e) El servidor Marshal B reenvía el mensaje INVITE para el servidor Redirect.
- f) El servidor Marshal B envía el mensaje INVITE para el usuario B.

- g) Timbra el teléfono SIP del usuario B, el mensaje 180 es enviado de vuelta hacia el teléfono SIP del usuario A.
- h) Cuando el usuario B levanta el teléfono SIP, un mensaje 200 OK es enviado.
- i) El usuario del teléfono SIP A responde con un mensaje ACK.
- j) La ruta RTP es ahora establecida.

Este proceso se representa en siguiente figura:

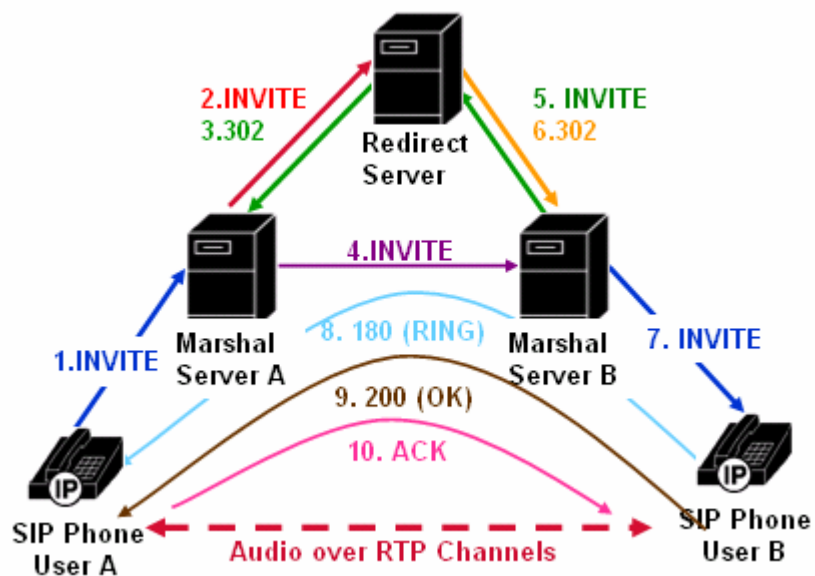


Figura 3.6. Proceso de una llamada básica SIP

### 3.6. COMPONENTES DE VOCAL

Los principales componentes de VOCAL se muestran en la siguiente figura:

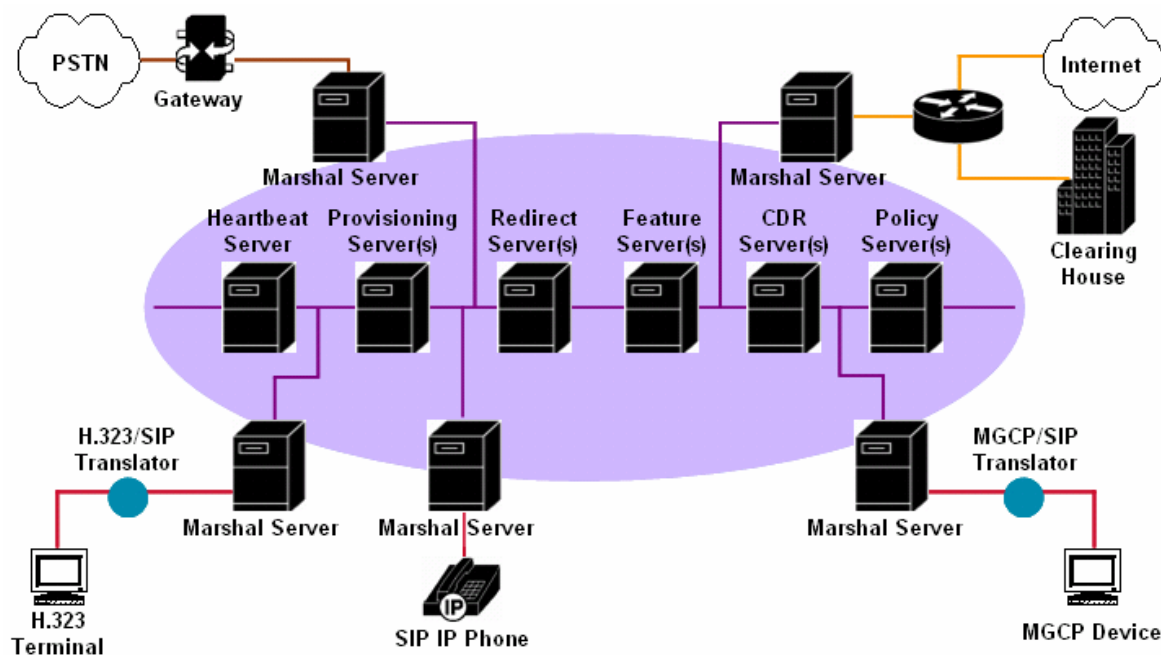


Figura 3.7. Componentes de VOCAL

### 3.6.1. USER AGENT

El sistema VOCAL a través de SIP User Agent o Agentes de Usuario SIP, soporta servicios como: establecimiento de llamada, llamada en espera, transferencia de llamada y registro en un servidor Marshal o un servidor SIP Proxy.

Ejemplos de agentes de usuario son: teléfonos SIP IP, softphones (software que hace las veces de un teléfono), gateways de VoIP; algunos de estos se muestran en la siguiente figura:



**Figura 3.8. Ejemplos de User Agent o Agentes de Usuario**

### **3.6.2. SERVIDOR MARSHAL**

El servidor Marshal es el único punto de contacto de todo dispositivo externo con el sistema VOCAL, provee la función lógica de un servidor SIP Proxy y de un servidor de SIP Registration; además realiza funciones como: traslación de mensajes SIP, autenticación y seguridad.

#### **3.6.2.1. Tipos de Servidor Marshal**

Los tipos de servidor Marshal en el sistema VOCAL son: User Agent Marshal, Gateway Marshal, Conference Bridge Marshal y Intenetwork Marshal.

##### **Servidor User Agent Marshal**

Las funciones de este servidor son:

- Interactuar con los User Agents o Agentes de Usuario.
- Recibir los mensajes INVITE de parte de los User Agents.



- Autenticar a un usuario comparándolo con los perfiles de usuarios almacenados en el servidor Redirect.
- Pedir la información de ruteo al servidor Redirect.

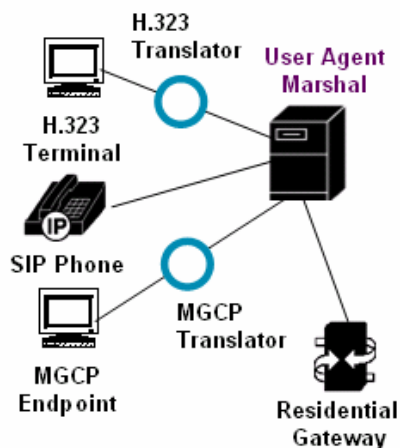


Figura 3.9. Servidor User Agent Marshal

### Servidor Gateway Marshal

Las funciones de este servidor son:

- Interactuar con los Gateways SIP o con los servidores SIP Proxy.
- Los gateways proveen traslación o interconexión entre IP y la red PSTN.

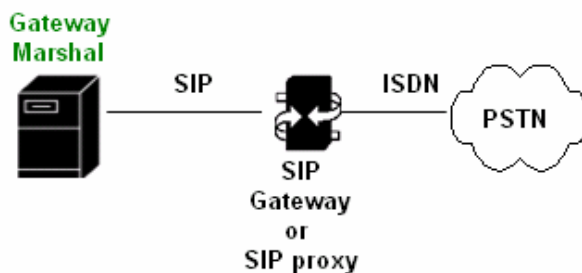


Figura 3.10. Servidor Gateway Marshal

## Servidor Conference Bridge Marshal

El sistema VOCAL soporta dos tipos de conferencias:

- *Meet – Me*: usuarios que llaman a un número y durante un tiempo, predefinidos.
- *Ad-Hoc*: usuarios sumándose a múltiples usuarios en una llamada. Es en este tipo de conferencia donde se utiliza el servidor Conference Bridge Marshal.

Las funciones de este servidor son:

- Insertar un identificador de llamadas común en los mensajes SIP salientes del SIP gateway.
- Remueve el identificador de llamadas común e inserta un único identificador de llamadas, por cada mensaje SIP entrante desde el gateway SIP.

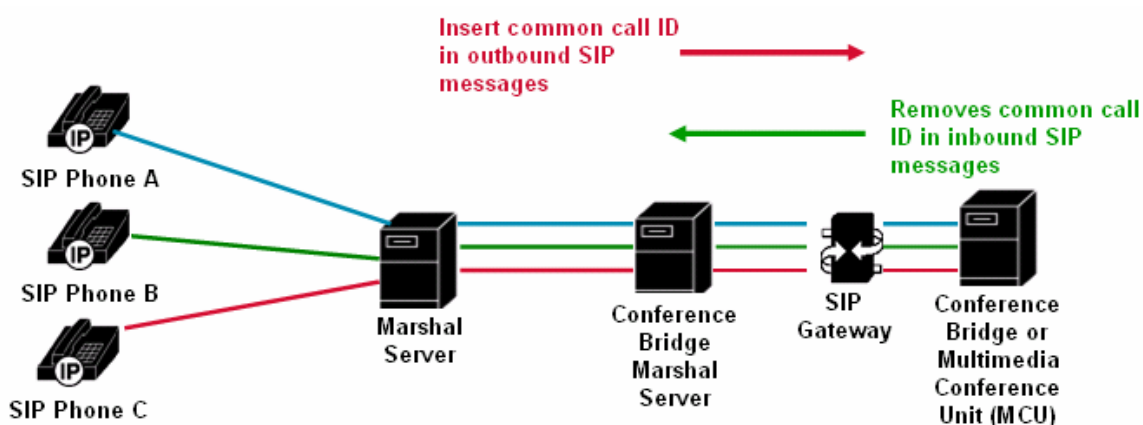


Figura 3.11. Servidor Conference Bridge Marshal

## Servidor Internetwork Marshal

Este servidor es usado para la interconexión con otros sistemas SIP que usan el protocolo OSP (Open Settlement Protocol).

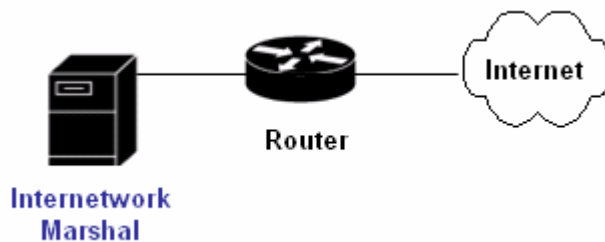


Figura 3.12. Servidor Internetwork Marshal

### 3.6.3. TRANSLATORS

El sistema VOCAL soporta terminales que al no utilizar SIP como protocolo de VoIP utilizan protocolos como H.323 y MGCP, a través de los llamados translators.

#### H.323 Translator

Las funciones de este translator son:

- Proveer la traslación de señalización de llamada entre terminales H.323 y el servidor SIP.
- Los terminales H.323 aparecerán como User Agents SIP, en el sistema VOCAL, el cual trabaja con Microsoft NetMeeting como terminal H.323.

#### MGCP Translator

Las funciones de este translator son:

- Proveer la traslación de señalización de llamada entre terminales MGCP y el servidor SIP.

- El MGCP translator puede actuar igual que un agente de llamada MGCP que controla los MGCP gateways.

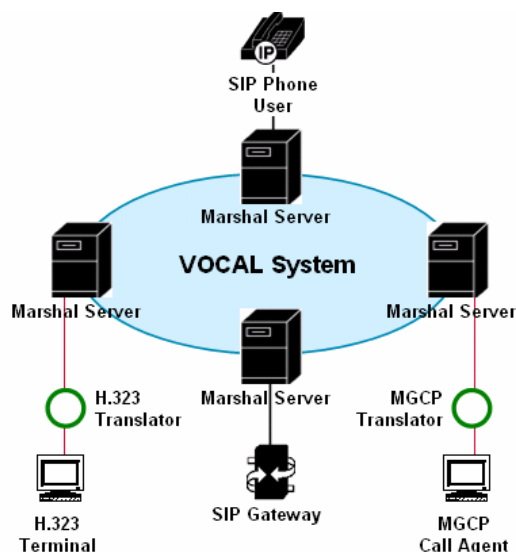


Figura 3.13. Translators

#### 3.6.4. SERVIDOR PROVISIONING

El servidor Provisioning cumple con la función de almacenar datos de los usuarios y servidores dentro del sistema VOCAL. Se puede acceder a este servidor vía un buscador de Internet, a través de GUI.

Este servidor es usado para:

- Configurar el sistema VOCAL.
- Administrar, habilitar, suscribir o eliminar características de usuario.

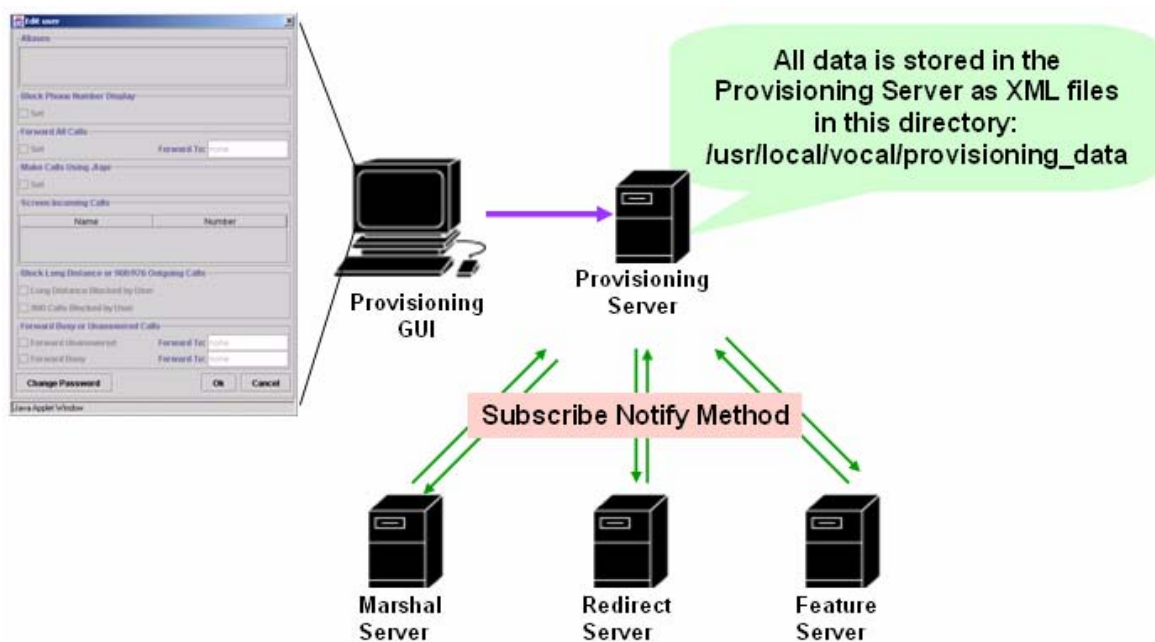


Figura 3.14. Servidor Provisioning

### 3.6.5. SERVIDOR REDIRECT

El servidor Redirect provee de los servicios y funciones SIP como: Registro, Redirección y Localización de los usuarios dentro de la red; además provee la información de ruteo para los servidores Feature y Marshal para escoger la ruta de cada llamada, y lo hace de la siguiente manera:

- Busca en la lista de suscriptores (abonados) o en del Dial Plan (Plan de Marcado), ambos previamente elaborados.
- Con la información de la lista de contactos, elabora el mensaje INVITE.
- Genera el mensaje 302 con la información de ruteo.

### Cuando es elaborada la lista de suscriptores y el Dial Plan?

La lista de suscriptores es elaborada solo en el arranque del servidor Redirect y el registro de los usuarios del sistema; a diferencia del Dial Plan que es ingresado usando el interfaz GUI.

### Cuando es elaborada la lista de contactos?

La lista de contactos es elaborada cuando se realiza un par de llamadas básicas, es decir cuando el servidor Redirect recibe un mensaje INVITE.

#### 3.6.5.1. Lista de Suscriptores o Abonados

Para la elaboración de la lista de suscriptores se toman en cuenta tres pasos:

- a) En el arranque del servidor Redirect se recogen los nombres de los usuarios para enviarlos al servidor Provisioning.

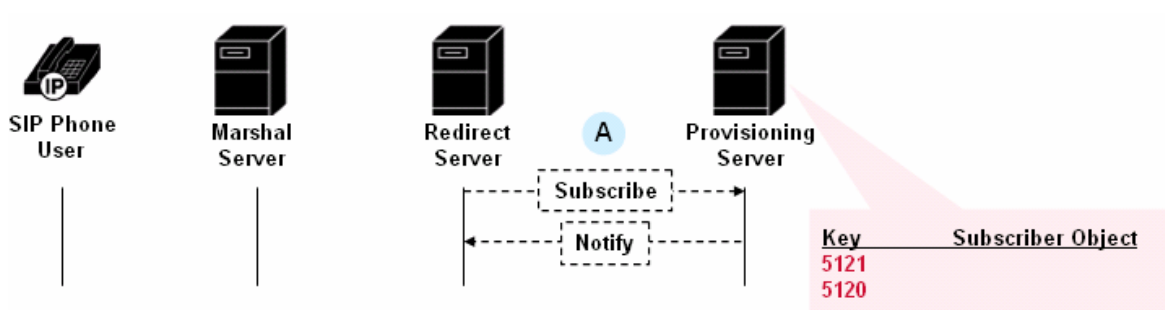


Figura 3.15. Paso A en la elaboración de la lista de suscriptores o abonados

- b) Se busca información de dichos nombres en el mensaje REGISTER.

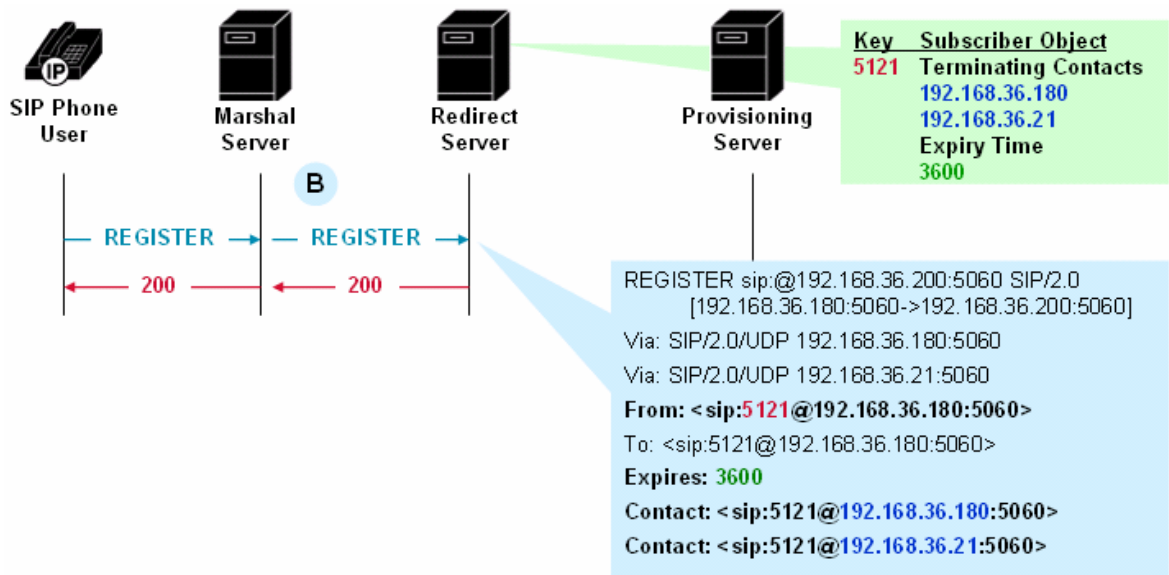


Figura 3.16. Paso B en la elaboración de la lista de suscriptores o abonados

c) Recoge características y datos de los usuarios para enviarlos al servidor Provisioning.

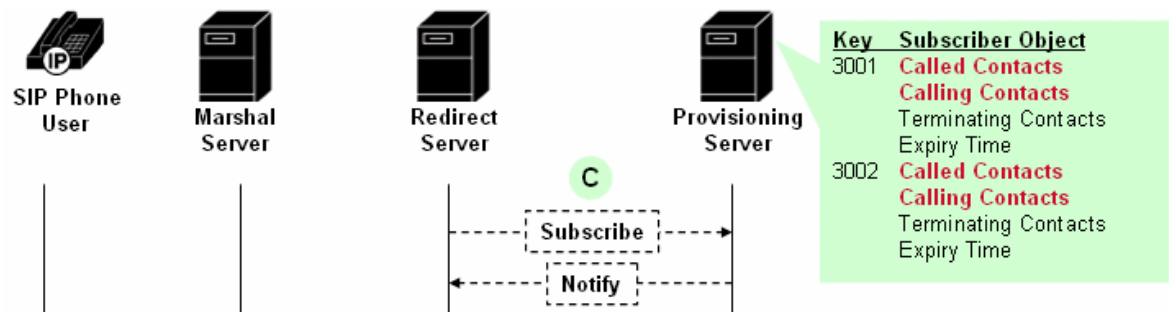


Figura 3.17. Paso C en la elaboración de la lista de suscriptores o abonados

La siguiente figura muestra todos los pasos de la elaboración de la lista de suscriptores o abonados:

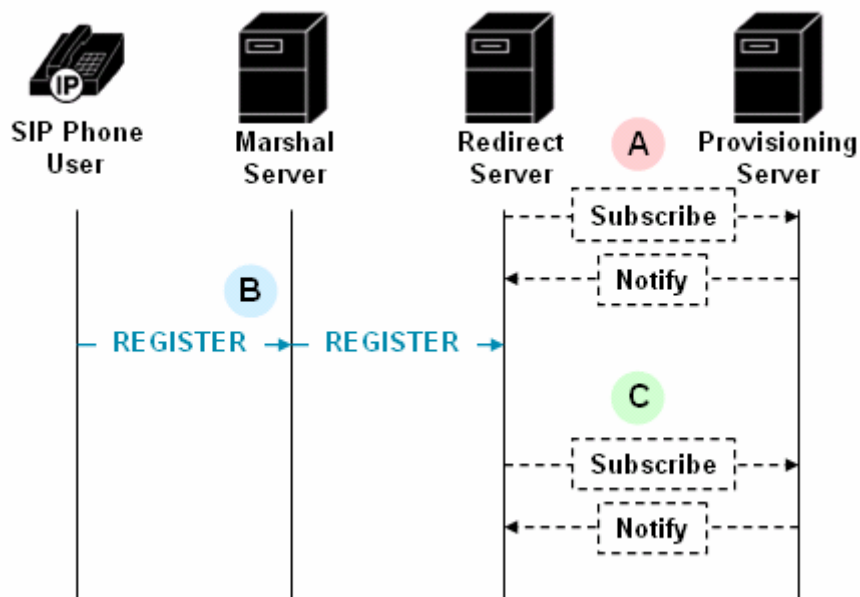


Figura 3.18. Elaboración de la lista de suscriptores o abonados

El siguiente es un ejemplo de una lista de suscriptores o abonados:

<u>Key</u>	<u>Subscriber Object</u>	
5121	Caller ID Blocking	← Called Contacts
	Call Forward No Answer	← Calling Contacts
	192.168.36.180	← Terminating Contacts
	192.168.36.21	← Terminating Contacts
	3600 milliseconds	← Expiry Time
5120	Call Blocking	← Called Contacts
	192.168.36.181	← Terminating Contacts
	192.168.36.20	← Terminating Contacts
	3600 milliseconds	← Expiry Time

### 3.6.5.2. Dial Plan o Plan de Marcado

El Dial Plan o Plan de Marcado consiste en una lista de *indexes* (*indicadores*), *keys* (*llaves*) y *contacts* (*contactos*); todos organizados en una tabla. Este sistema es utilizado si los usuarios que hacen la llamada no se encuentran en la lista de suscriptores o abonados.

Cada uno de los elementos de esta tabla cumple con las siguientes funciones:



## Index o Indicador

El Index o Indicador es usado para atribuir un orden de prioridades a los keys y contacts, dentro de la tabla del plan de marcado.

## Keys o Llaves

Los Keys o Llaves indican el tipo de marcación que le corresponde a cada contacto, dentro de la tabla del plan de marcado.

## Contacts o Contactos

Los Contacts o Contactos proveen la información de posibles rutas que puede tomar la llamada.

La siguiente tabla muestra algunos ejemplos de Dial Plan o Plan de Marcado:

Key	Contact	Descripción
<b>^sip: .{7}@</b>	sip:\$USER@192.168.116.110:5060; user=phone	Cuando el usuario marque 7 dígitos de un numero telefónico, reenvía esta llamada al Servidor Gateway Marshal.
<b>^sip: 0@</b>	sip:\$USER@192.168.116.110:5060; user=phone	Cuando el usuario marque 0 el sistema reenvía esta llamada al Servidor Gateway Marshal.

<b>^sip: 9.*@</b>	sip:\$USER@192.168.116.110:5060; user=phone	Cuando el usuario marque 9 saldrá del sistema local y entrara a la red PSTN, a través del Servidor Gateway Marshal.
<b>^sip: 7000@</b>	sip:\$7000@192.168.116.110:5060; user=phone	Cuando el usuario marque 7000 el sistema reenvía esta llamada para los voicemail user agents.
<b>^sip: [0-9]{4}@</b>	sip:\$7000@192.168.116.110:5060; user=phone	Con esta regla de marcado cualquier llamada de 4 dígitos que corresponda a un UA puede ser enrutada por el servidor UA Marshal.

**Tabla 3.1. Ejemplos de Dial Plan o Plan de Marcado**

### Como se crea el Dial Plan o Plan de Marcado?

El plan de marcado se crea desde el interfaz gráfico GUI, y puede ser de dos tipos:

- *Digital Plan*: elaborado en base a números telefónicos (user = phone).
- *IP Plan*: elaborado en base a SIP URI (user = IP).

La siguiente figura muestra el interfaz gráfico GUI, configurando el Dial Plan o Plan de Marcado:

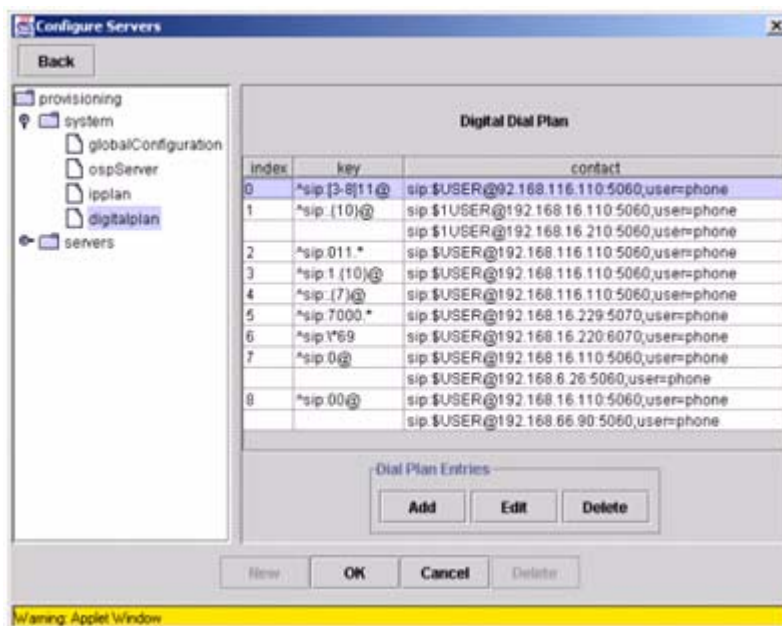


Figura 3.19. Dial Plan o Plan de Marcado

### 3.6.5.3. Lista de Contactos

La lista de contactos es generada por el Servidor Redirect, cuando este recibe un mensaje INVITE de parte del Servidor Marshal o el Servidor Feature.

Esta lista tiene como contactos a los que llaman, los que reciben la llamada y los que finalizan la llamada. Esta información es usada para generar el mensaje 302 en respuesta al mensaje INVITE.

Dicha lista no es guardada y es valida durante el tiempo que el servidor Redirect necesita para generar la información de ruteo.

La siguiente figura muestra el proceso en el que interactúan el servidor Marshal y el servidor Redirect, para la creación de la lista de contactos:

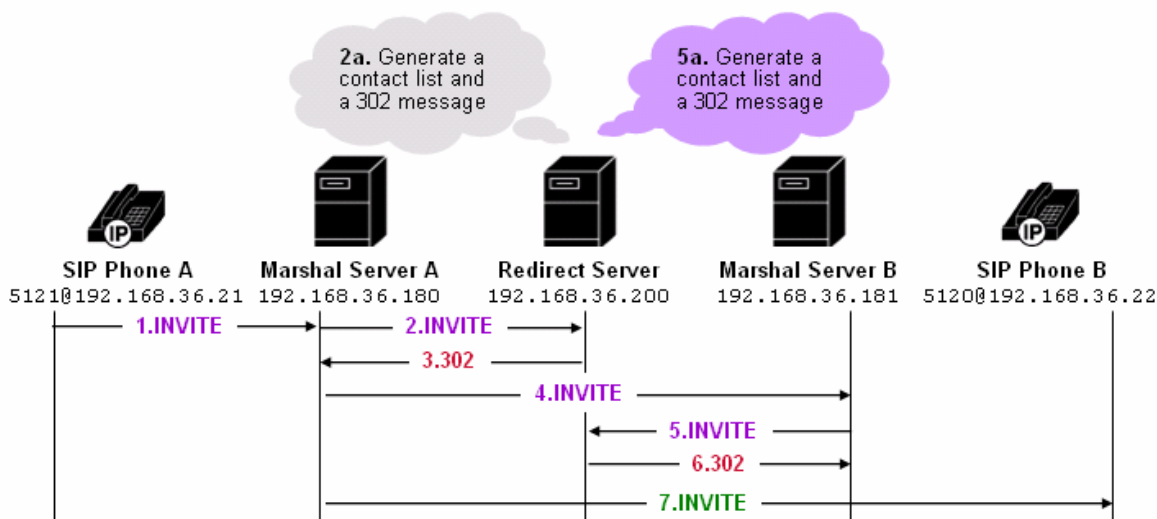


Figura 3.20. Interacción entre el Servidor Marshal y el Servidor Redirect

Cuando el servidor Redirect recibe un mensaje INVITE, este extrae información del: campo REQUEST URI y del campo FROM; usando esta información dicho servidor busca en la lista de suscriptores o abonados y elabora la lista de contactos, los campos antes mencionados se muestran en la siguiente figura:

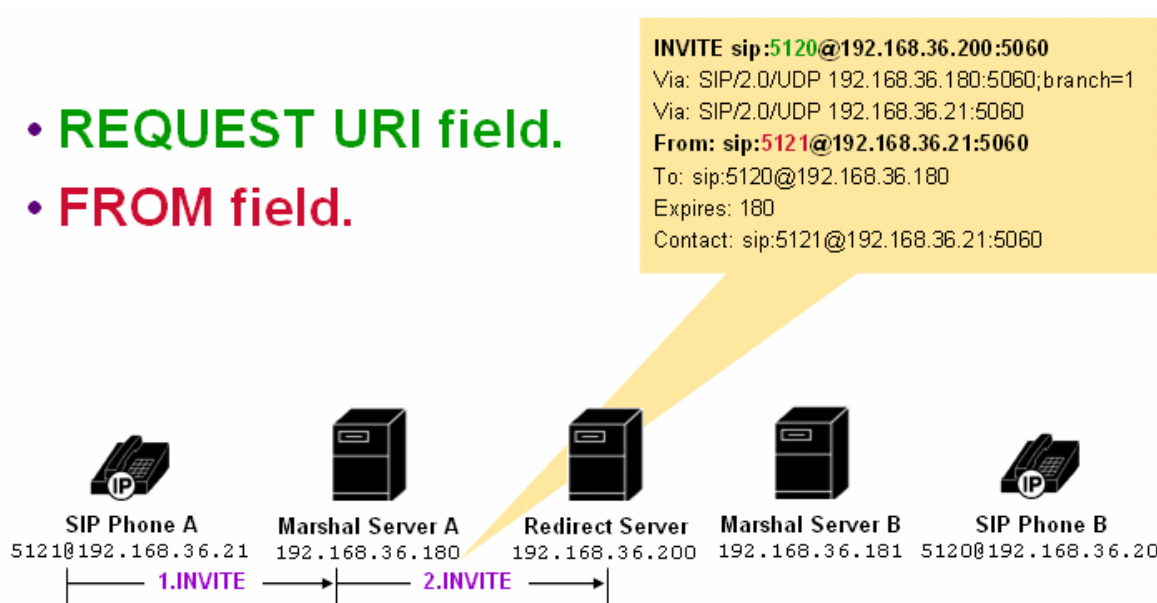


Figura 3.21. Campos REQUEST URI y FROM

La elaboración de la lista de contactos consta de los siguientes pasos:

- **FROM:** los que llaman
- **FROM:** los que llaman
- **REQ URI:** los que reciben la llamada
- **REQ URI:** los que constan en el Dial Plan o Plan de Marcado
- **REQ URI:** los que finalizan la llamada

De esta lista de contactos el servidor Redirect determina el contacto que incluirá en el mensaje 302.

El servidor Redirect busca en el campo VIA para determinar por donde se realizará la llamada; así también este servidor tiene un algoritmo que determina cuál de los contactos en la lista está siendo usado. Este campo se muestra en la siguiente figura:

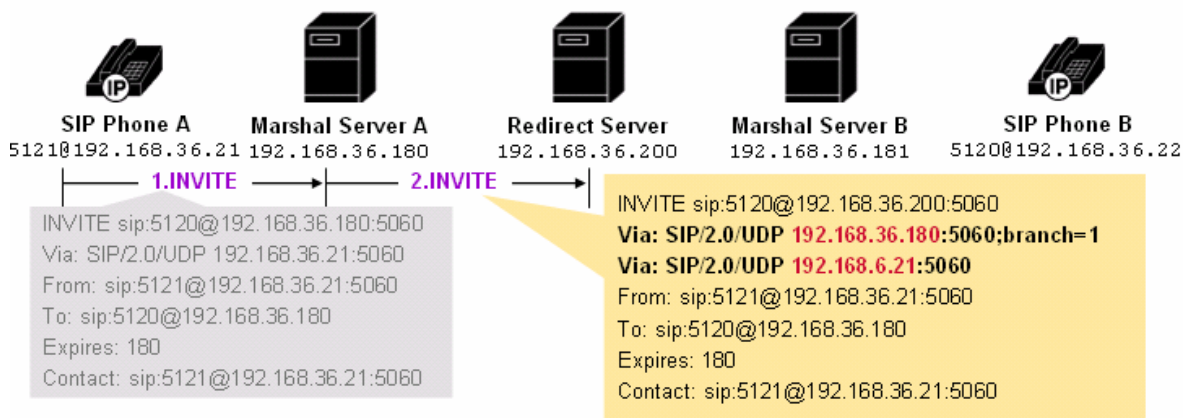


Figura 3.22. Campo VIA

La siguiente figura muestra la conformación del mensaje 302:

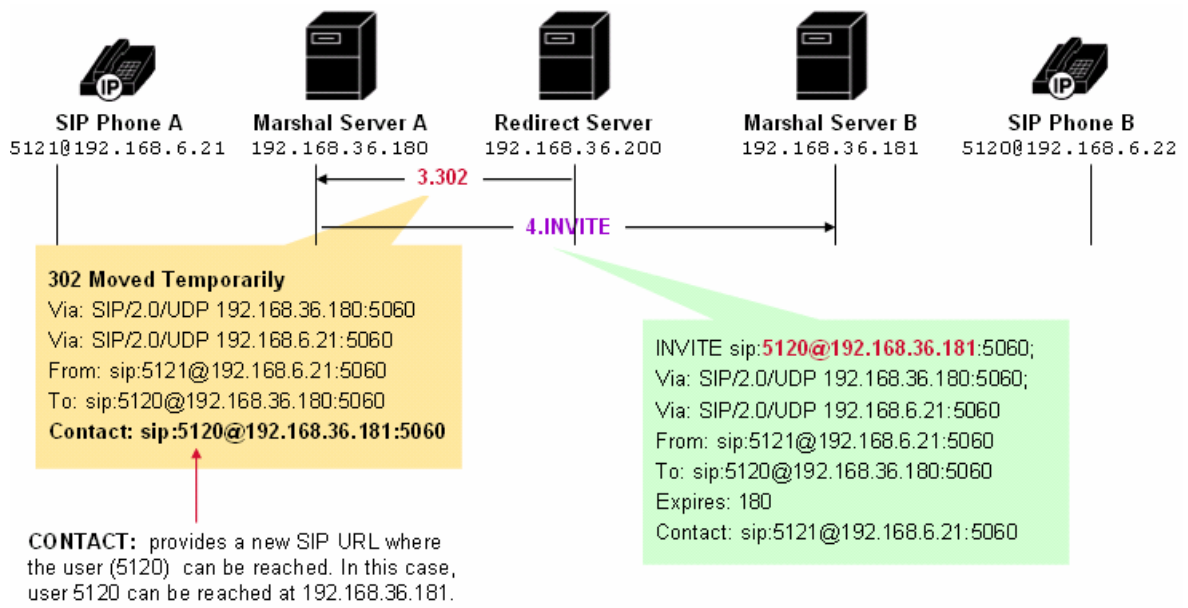


Figura 3.23. Mensaje 302

La siguiente figura muestra la conformación de una llamada completa a través del sistema VOCAL:

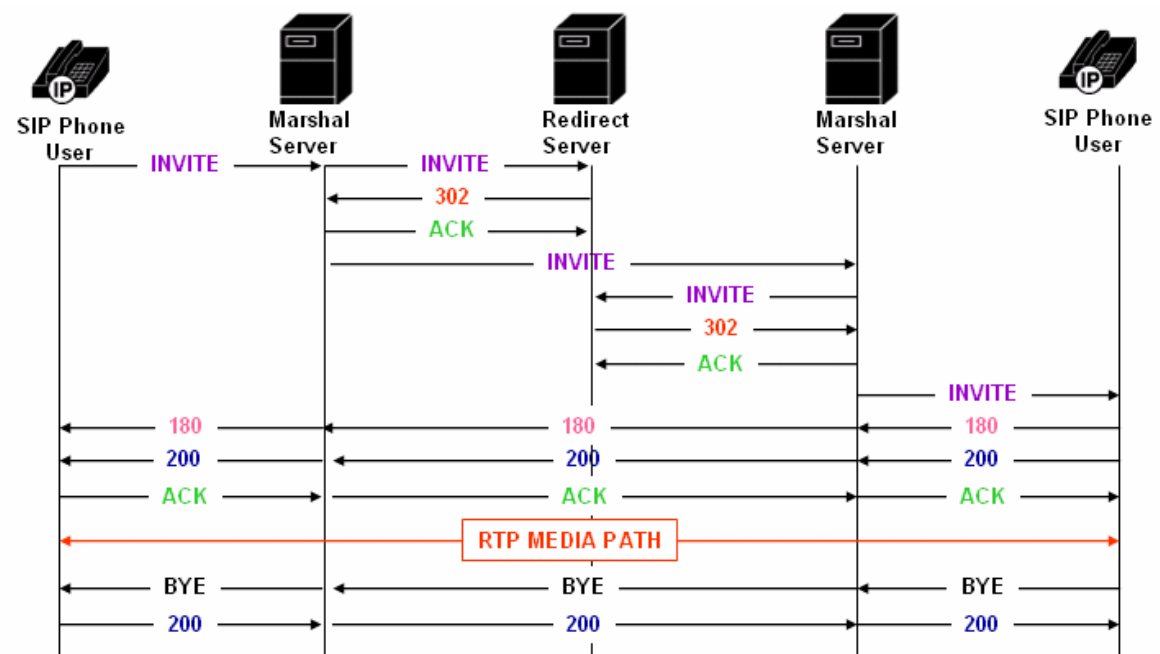


Figura 3.24. Llamada Completa a través del sistema VOCAL

### 3.6.6. SERVIDOR FEATURE

La función del Servidor Feature es la de proveer de características tales como: reenvío de llamadas, bloqueo de llamadas, etc. al sistema VOCAL; así como soportar algunas otras características propias de los equipos terminales tales como: transferencia de llamadas, llamadas en espera, conferencias, etc.

Las características de cada usuario son generadas por un script (pequeño programa) CPL (Call Processing Language), gracias al servidor Provisioning, estas son habilitadas y configuradas, usando el Provisioning GUI.

A continuación la figura muestra como ejemplo la característica de bloqueo de llamada:

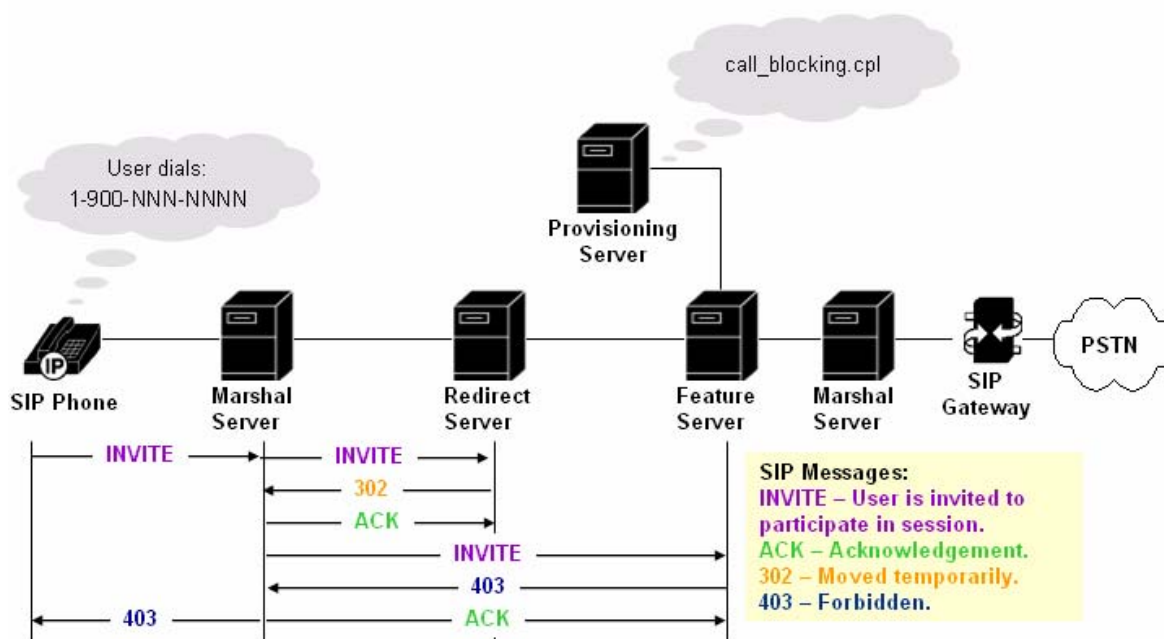


Figura 3.25. Bloqueo de una llamada

### 3.6.7. SERVIDOR VOICE MAIL

El Voice Mail o Correo de Voz del sistema VOCAL funciona gracias a tres componentes básicos: el Servidor Voice Mail Feature, al Voice Mail User Agent y al Servidor Voice Mail.

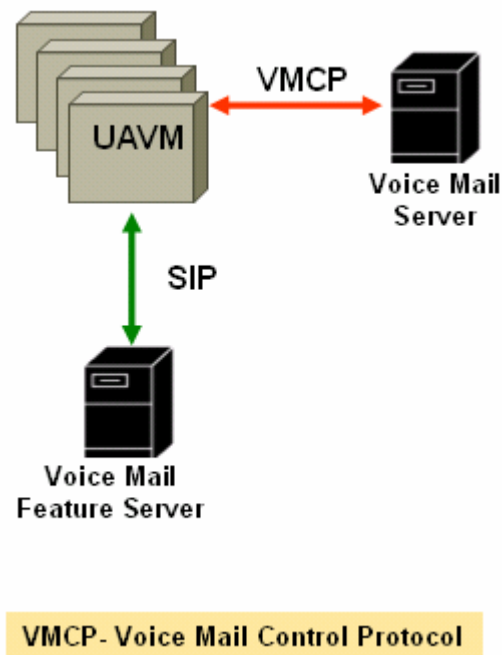


Figura 3.26. Componentes del Voice Mail o Correo de Voz

### 3.6.7.1. Servidor Voice Mail Feature

El Servidor Voice Mail Feature cumple con las siguientes funciones:

- Distribuye las llamadas disponibles de los Voice Mail User Agents (UAVM).
- Verifica cual UAVM esta activo y disponible.
- Reenvía un mensaje INVITE al primer UAVM disponible.



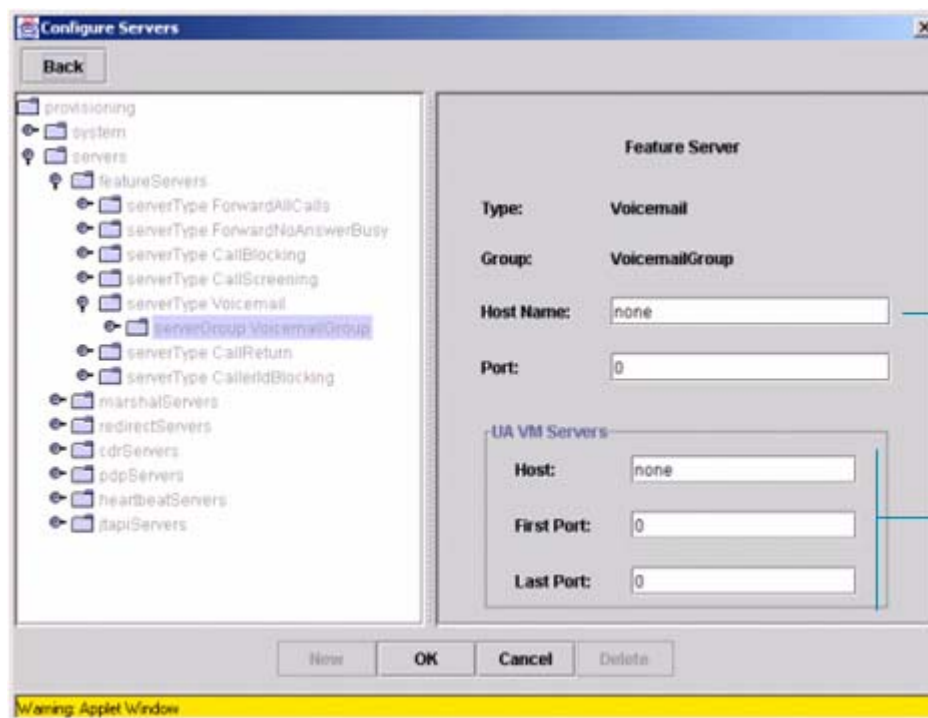


Figura 3.27. Servidor Voice Mail Feature

### 3.6.7.2. Voice Mail User Agent o Agente de Usuario de Correo de Voz

El Voice Mail User Agent o Agente de Usuario de Correo de Voz cumple con las siguientes características:

- Actúa como un gateway, trasladando mensajes SIP a mensajes VMCP (Voice Mail Control Protocol o Protocolo de Control de Correo de Voz).
- Se comunica con el servidor Voice Mail gracias a VMCP.
- El número de UAVM's puede ser configurado usando el Provisioning GUI.
- Cada UAVM soporta una llamada a la vez.

### 3.6.7.3. Servidor Voice Mail

El Servidor Voice Mail es usado para:

- Reproducir los mensajes grabados
- Guardar los mensajes de voz como archivos .wav , dentro de un directorio temporal.
- Envía el archivo .wav como archivo adjunto en un correo, cuyo destinatario tiene una dirección de correo pre-configurada. Esta dirección de correo es especificada en el archivo de configuración de cada usuario.

### 3.6.8. SERVIDOR CALL DETAIL RECORD O CDR

El Servidor Call Detail Record o CDR realiza las siguientes funciones:

1. Recibe la indicación de inicio y final de llamada por parte del servidor Marshal.
2. Por cada llamada formatea los datos dentro del CDR.
3. Reenvía los datos CDR hacia un tercer sistema que es el de cuentas, este utiliza como protocolo de cuentas a RADIUS.

La siguiente figura muestra los tres pasos antes mencionados:

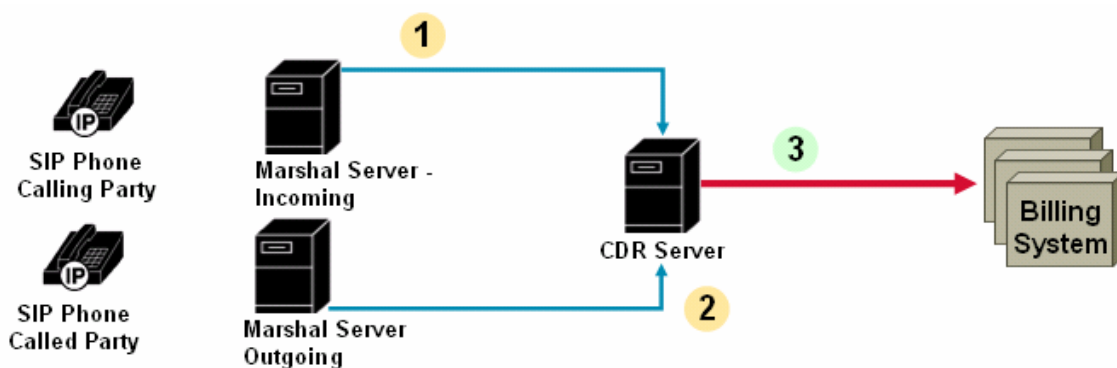


Figura 3.28. Funciones del Servidor CDR



Figura 3.29. Servidor Call Detail Record o CDR

### 3.6.9. SERVIDOR POLICY

El Servidor Policy cumple con las siguientes funciones:

- Administra la admisión de peticiones de ancho de banda o de calidad de servicio QoS.
- Interactúa con el servidor Internetwork Marshal para realizar el mejor esfuerzo y brindar QoS.
- Provee de interfaces libres para autorizar el uso de la red y el uso entre redes, y así realizar llamadas.

### 3.6.10. SERVIDOR HEARTBEAT

Los servidores del sistema VOCAL (pero no todos), envían y escuchan paquetes de nombre heartbeat, en un puerto multicast; si algún servidor no envían estos paquetes después de cierto tiempo, el servidor es considerado caído. El Servidor Heartbeat cumple la función de monitorear el intercambio de dichos paquetes, además envía información del estado de los servidores del sistema a través de un Gestor de Red SNMP.

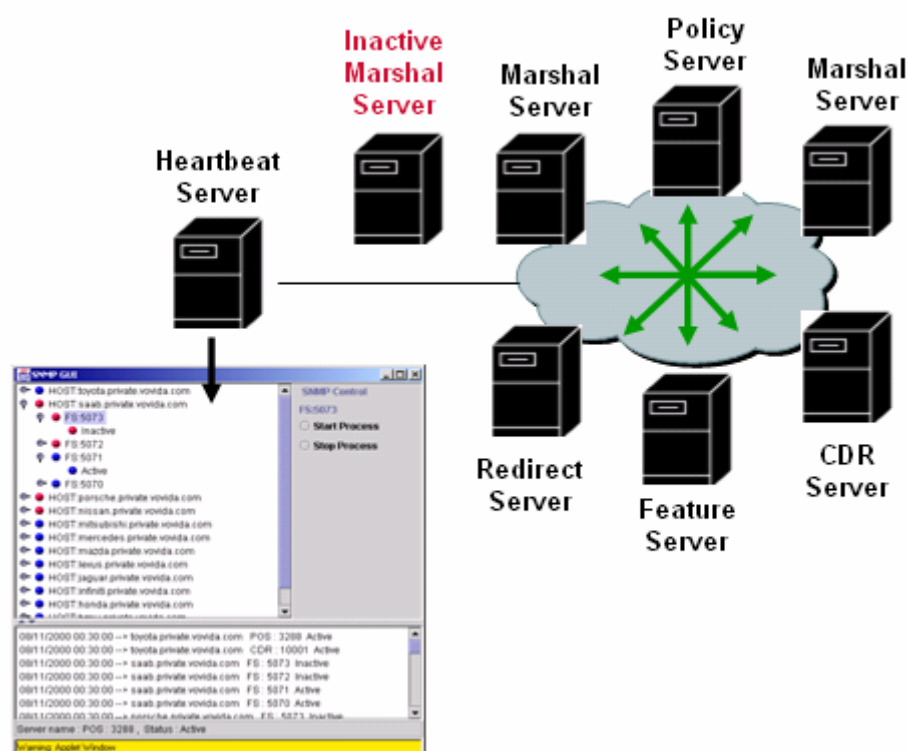


Figura 3.30. Servidor Heartbeat

Entre los parámetros de configuración de este servidor tenemos:

#### Host y Puerto de Multicast

Este es usado para enviar mensajes de heartbeat broadcast.

### Intervalo Heartbeat

Es el tiempo en milisegundos entre los mensajes heartbeat

### Máximos Mensajes Heartbeat Omitidos

Es el máximo número de mensajes heartbeat, que pueden ser omitidos antes de que cambie de status, a inactivo cualquier servidor.

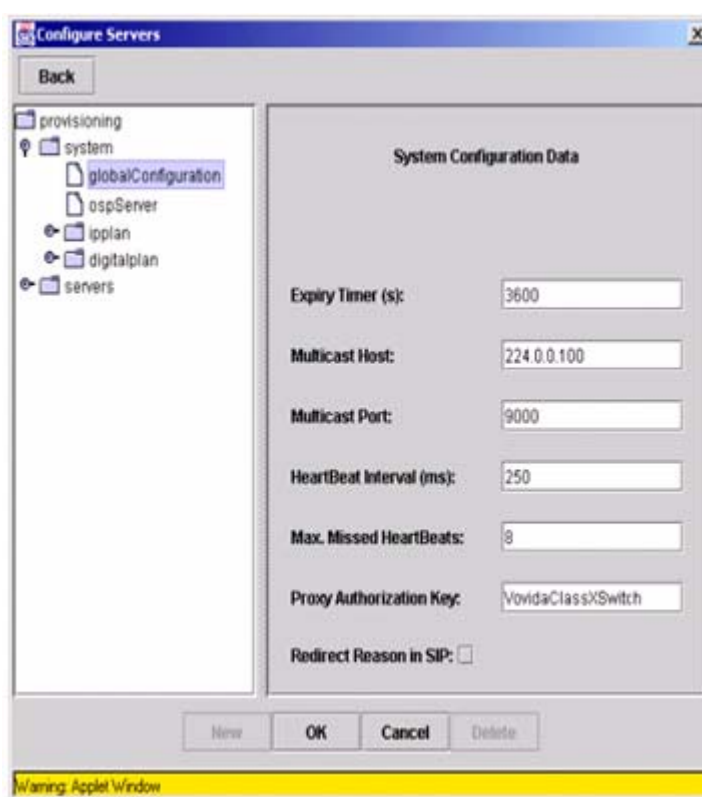


Figura 3.31. Características del Servidor Heartbeat

#### 3.6.10.1. Gestor de Red

El sistema VOCAL para su monitoreo y gestión de red, utiliza SNMP (Simple Network Management Protocol). A través de SNMP GUI puede monitorear es estado de los servidores VOCAL, como se muestra en la siguiente figura:

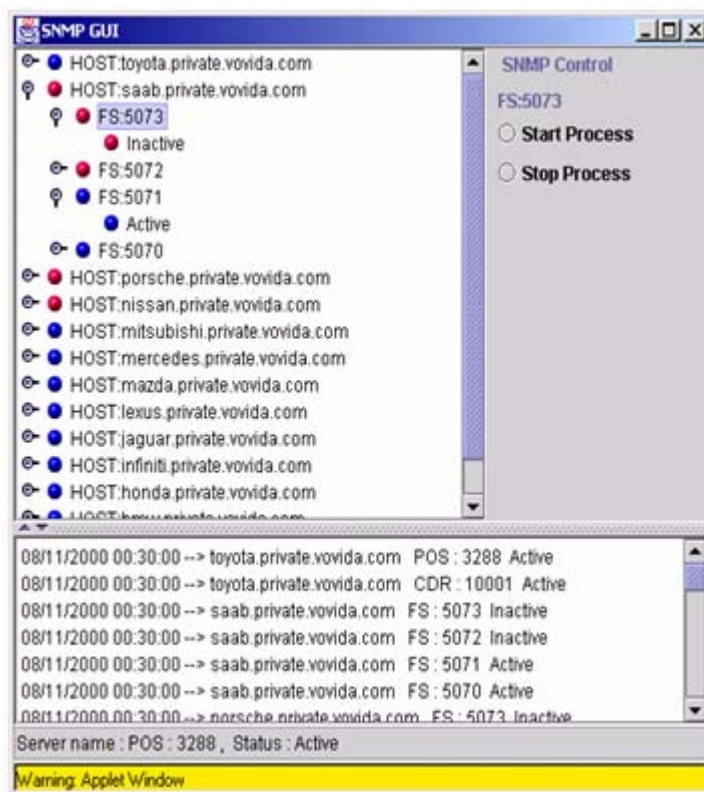


Figura 3.32. SNMP GUI

La siguiente figura muestra en resumen el funcionamiento del gestor de red dentro del sistema VOCAL:

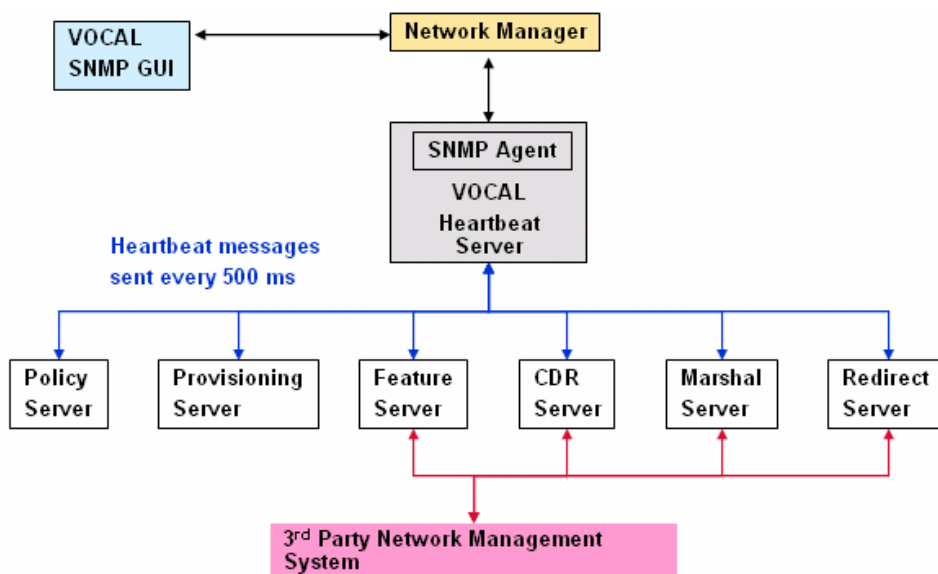


Figura 3.33. Gestor de Red

Una vez descritos todos los componentes del sistema VOCAL, se muestra a continuación la implementación del sistema.

### **3.7. IMPLEMENTACIÓN DEL SISTEMA VOCAL**

Para la implementación del sistema VOCAL utilizamos un PC de marca PREMIO con las siguientes características generales:

- Procesador: Pentium III de 866 MHz.
- Memoria RAM: 128 MB.
- Disco Duro: 30 GB.

Por demás esta mencionar que estas características son de sobrado mejores a las mínimas que exige el sistema VOCAL, que son:

- Procesador: Pentium II de 400 MHz.
- Memoria RAM: 128 MB.
- Disco Duro: 1 GB.

El sistema VOCAL fue probado con los siguientes sistemas operativos, en el mismo orden: Red Hat Linux 9, Red Hat Enterprise y Red Hat 7.3; siendo este último el que mostró mayor facilidad de configuración y además mayor estabilidad.

Las versiones de Java Runtime Environment que se probaron en el proyecto fueron: Versión 1.4.1 y Versión 1.3.1; siendo esta última la que permitió la configuración del sistema VOCAL a través de GUI.

Las versiones del software de VOCAL que se utilizó en el proyecto fueron: Versión 1.5 y Versión 1.4; siendo esta última la que mostró mejores resultados tanto en configuración como en estabilidad.

### **3.7.1. INSTALACIÓN DEL SISTEMA OPERATIVO**

El sistema operativo que se instaló fue Red Hat Linux 7.3, y durante la instalación se escogieron las siguientes opciones:

- Idioma: Español.
- Particionamiento: Automático.
- Tipo de Instalación: Personalizada.
- Paquetes a instalar: Todos los paquetes.
- Seguridad: No Firewall.

#### **3.7.1.1. Verificaciones de la instalación del Sistema Operativo**

Para comprobar que se instaló correctamente el sistema operativo se puede realizar las siguientes verificaciones:

##### **Verificación de la dirección IP y multicast**

Desde una ventana de tipo Terminal se digita lo siguiente: `ifconfig eth0`

Aparecerá lo siguiente:



```
[root@vocal root]# ifconfig eth0
eth0      Link encap:Ethernet  HWaddr 00:0C:76:5C:61:9D
          inet addr:10.0.0.158  Bcast:10.0.0.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:11110 errors:0 dropped:0 overruns:0 frame:0
          TX packets:920 errors:0 dropped:0 overruns:0 carrier:0
          collisions:109 txqueuelen:100
          RX bytes:1727733 (1,6 Mb)  TX bytes:205941 (201,1 Kb)
          Interrupt:10 Base address:0xd400
```

### Verificación DNS

Desde una ventana de tipo Terminal se digita lo siguiente: `cat /etc/resolv.conf`

Aparecerá lo siguiente: `search <nombre del dominio>`

`nameserver <ip del servidor de DNS>`

Desde una ventana de tipo Terminal se digita lo siguiente: `ping www.yahoo.com`

Aparecerá lo siguiente:

### Verificación de tu hostname

Desde una ventana de tipo Terminal se digita lo siguiente: `hostname`

Aparecerá lo siguiente: `<nombre de tu host>`

### Verificación de tu archivo hosts

Desde una ventana de tipo Terminal se digita lo siguiente: `cat /etc/hosts`

Aparecerá lo siguiente: `127.0.0.1 localhost.localdomain localhost`

`<tu dirección ip> <nombre de tu host>`

### 3.7.2. INSTALACIÓN DE JAVA RUNTIME ENVOIRONMENT JRE

La versión de Java que se instaló fue la versión 1.3.1, cuyo archivo instalador es: `j2re-1.3.1_13-linux-i586.rpm.bin` y puede ser descargada de la pagina web <http://java.sun.com/>, a un directorio específico de libre elección.

La instalación de java consta de los siguientes pasos:

1. Desde una ventana tipo Terminal y ubicado en el directorio donde se descargo el archivo instalador se hace ejecutable el mismo, digitando lo siguiente:

```
chmod a+x j2re-1.3.1_13-linux-i586.rpm.bin
```

2. Se ejecuta el archivo instalador:

```
./j2re-1.3.1_13-linux-i586.rpm.bin
```

Seguido de este paso, se mostrará la información acerca de la licencia de Java y si acepta las condiciones de dicha licencia, a lo que se deberá responder con la palabra *yes*, de esta forma se creará un paquete RPM de nombre `jre-1.3.1_13-linux-i586.rpm`

3. Se instala el paquete RPM:

```
rpm -Uvh jre-1.3.1_13-linux-i586.rpm
```

De esta forma JRE se instalará en el directorio `/usr/java/j2re-1.3.1_13`

4. Como último paso se realiza un enlace simbólico entre Java y el navegador Mozilla, que es el que se utilizó en el proyecto; digitando los siguientes comandos:

```
ln -s /usr/lib/mozilla/plugins  
/usr/java/jre1.3.1_13/plugin/i386/ns600/libjavaplugin_oji.so
```

5. Para comprobar que se instaló correctamente java, así como que se realizó correctamente el enlace simbólico, se accede al navegador Mozilla y en el espacio donde se ingresa la dirección de la página web a la que se quiere acceder se deberá escribir lo siguiente: `about:plugins`, como resultado en el navegador se desplegará la información de los plugins que están disponibles en dicho navegador.

### 3.7.3. INSTALACIÓN DEL SOFTWARE DE VOCAL

La versión del software de VOCAL que se instaló fue la versión 1.4, que consta de los siguientes instaladores: `vocalbin-1.4.0-17.i386.rpm` y `vocalbin-tools-1.4.0-17.i386.rpm`, ambos se pueden descargar de la página web [www.vovida.org](http://www.vovida.org).

Para instalar el software de VOCAL se digitan los siguientes comandos, desde una ventana tipo Terminal:

```
rpm -Uvh vocalbin-1.4.0-17.i386.rpm y
```

```
rpm -Uvh vocalbin-tools-1.4.0-17.i386.rpm
```

#### 3.7.3.1. Verificación de la instalación de VOCAL

Para comprobar la correcta instalación de VOCAL desde una ventana tipo Terminal se digita lo siguiente: `/usr/local/vocal/bin/allinoneconfigure/verifyinstall`

Aparecerá lo siguiente: `Basic call succeeded. Installation appears to be OK.`

De no aparecer este mensaje se recomienda volver a instalar VOCAL.

### 3.8. CONFIGURACION DE VOCAL

La configuración inicial de VOCAL se realiza a través de un script, al cual se accede digitando desde una ventana tipo Terminal, lo siguiente:

```
/usr/local/vocal/bin/allinoneconfigure/allinoneconfigure
```

A continuación se desplegarán los siguientes avisos, a los que se debe responder dando un ENTER:

WARNING WARNING WARNING WARNING WARNING WARNING

The following may destroy any configuration that you currently have on your system. If you would like to exit, press Control-C now.

WARNING WARNING WARNING WARNING WARNING WARNING

Welcome to the VOCAL all-in-one configuration system. This program is intended to configure a small example system which has all of the servers running on one box, known as the "all-in-one" system.

This all-in-one system is NOT intended as a production system, but as a simple example to get users started using VOCAL.

This configuration WILL destroy any currently configured system on this machine. If this is not acceptable, please quit by pressing Control-C now.

Host IP Address [<mi dirección IP>]:

Remote Contact hostname or address (this should NOT be loopback or .127.0.0.1) [<mi dirección IP >]:

Multicast Heartbeat IP Address [224.0.0.100]:

Multicast Heartbeat Port [9000]:

Log Level [LOG\_ERR]:

User to run as [nobody]:

HTML directory to install .jar and .html files into [/usr/local/vocal/html]:

Provisioning your VOCAL system requires the ability to view the contents of /usr/local/vocal/html from the web. There are two ways to do this, review both options before answering the next prompt.

Option 1:

Step 1: Answer y to the next prompt. This will let this script attempt to add the following to your Apache httpd.conf file:

```
Alias /vocal/ "/usr/local/vocal/html/"
<Directory "/usr/local/vocal/html">
    AllowOverride None
    Order allow,deny
    Allow from all
</Directory>
Adding this script creates an alias from the
following URL:
```

```
http://<hostname>/vocal/
which points to /usr/local/vocal/html,
```

Step 2; After this script has completed running, restart your copy of Apache (httpd) for the change to take effect.

Option 2: Answer n to the next prompt. Then, manually copy the directory /usr/local/vocal/html to your web server's HTML directory. You should not need to restart your copy of Apache after the script has completed running.

Would you like this script to attempt Option 1, Step 1 (y), or would you like to perform Option 2 manually (n)? (If y, you must restart Apache after this script has completed running.) [y]:

Directory where Apache's httpd.conf is located [/etc/httpd/conf]:

\*\*\*\*\*

Configuration:

```
Host IP Address:          <mi dirección IP>
Remote Contact Address:  <mi dirección IP>
Multicast Heartbeat IP Address:  224.0.0.100
Multicast Heartbeat Port:    9000
Log Level:                LOG_ERR
User to run as:           nobody
HTML directory:          /usr/local/vocal/html
Add alias to:            /etc/httpd/conf/httpd.conf
```

\*\*\*\*\*

Continue [n]:

\*\*\*\*\*

Configuration complete!

To configure your VOCAL system

\* Go to

```
http://<hostname>/vocal/
and select Provisioning'.
```

```
[root@<hostname> vocal]#
```

Después de realizado este proceso la información de la IP del sistema VOCAL se guardará en el archivo `/usr/local/vocal/etc/vocal.conf`. Si se desea cambiar el IP del sistema se debe cambiar primero la IP de Linux, verificar que la nueva IP esta dada de alta en el archivo `/etc/hosts`, borrar el archivo `vocal.conf` y correr el comando `/usr/local/vocal/bin/vocalctl start`, posterior a este paso el sistema por si solo correrá el script `allinoneconfigure` .

### 3.8.1. MEJORANDO EL DESEMPEÑO DE VOCAL

Adicionalmente para mejor desempeño de VOCAL se debe desactivar el sistema de ventanas de Linux, además de actualizar el servidor Apache y SSH (por motivos de seguridad para evitar posibles intrusiones), estos servidores permiten administrar y configurar el sistema VOCAL desde un computador remoto, a través de un navegador web (con soporte de java) con la ayuda del servidor Apache y a través de un cliente SSH como Putty con la ayuda del servidor SSH; todo lo anterior se detalla a continuación:

1. El sistema de ventanas de Linux se desactiva editando el archivo `/etc/inittab` en su última línea, con la ayuda de cualquier editor de Linux, reemplazando el comando `respawn` por el comando `off`, y guardando los cambios realizados en este archivo:

```
x:s:respawn:/etc/x11/prefdm – nodaemon (antes)
```

```
x:s:off:/etc/x11/prefdm – nodaemon (después)
```

2. Posterior al paso antes mencionado y para que los cambios tengan efecto, desde una ventana tipo Terminal se digita el comando:

```
init q
```

3. Para actualizar el servidor Apache y SSH, se debe bajar de Internet las últimas actualizaciones de estos servidores en archivos de extensión rpm (pueden ser varios archivos) ; e instalarlos desde una ventana tipo Terminal de la siguiente manera:

```
rpm -ivh apache* (para el caso del servidor Apache)
```

```
rpm -ivh open* (para el caso del servicio SSH)
```

4. Como último paso se reinicia los servidores Apache y SSH:

```
/etc/rc.d/init.d/httpd restart (para el caso del servidor Apache)
```

```
/etc/rc.d/init.d/sshd restart (para el caso del servidor SSH)
```

Una vez completados estos pasos, ya se puede abrir el navegador e ingresar la dirección `http://<hostname>o<IP de VOCAL>/vocal/`, para poder configurar el sistema VOCAL de forma gráfica utilizando GUI, como se muestra en la figura:

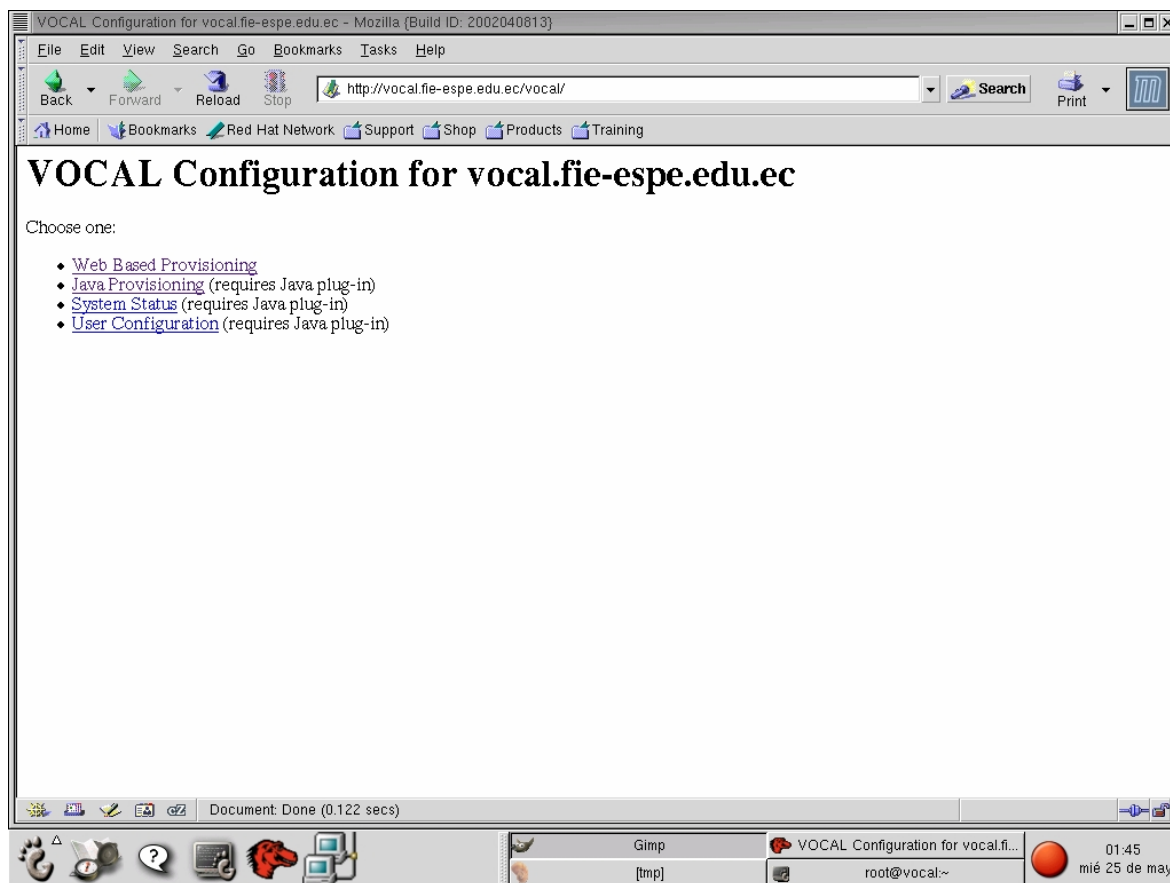


Figura 3.34. Configuración de VOCAL vía web

### 3.8.2. CONFIGURACION DE VOCAL PARA OBTENER UN DETALLE DE LLAMADAS

Para tener un reporte de todas las llamadas realizadas a través del sistema VOCAL, se deben realizar algunos cambios detallados a continuación:

1. Se debe ingresar como *root* al servidor Linux y desde una ventana tipo Terminal ingresar al directorio */usr/local/vocal* :

```
cd /usr/local/vocal
```

2. Crear un directorio llamado billing:



`mkdir billing`

3. Poner como propietario de este directorio al usuario de VOCAL (normalmente apache o nobody)

`chown apache billing`

4. Dar permisos de acceso 755 al archivo billing

`chmod 755 billing`

5. Luego de realizados estos cambios al digitar el comando `ls -l`, debe desplegarse una información como la siguiente:

```
drwxr-xr-x  4 apache  root   8192 May 11 14:25 billing
```

6. Desde el navegador ingresar a la dirección `http://<hostname>o<IP de VOCAL>/vocal/provisioning.html` , el navegador mostrara la ventana de niveles de acceso (Figura 3.3).
7. Dentro de esa ventana se escoge la opción *Technician* con su respectivo usuario así como su contraseña, y se desplegara una ventana como la que se muestra a continuación:

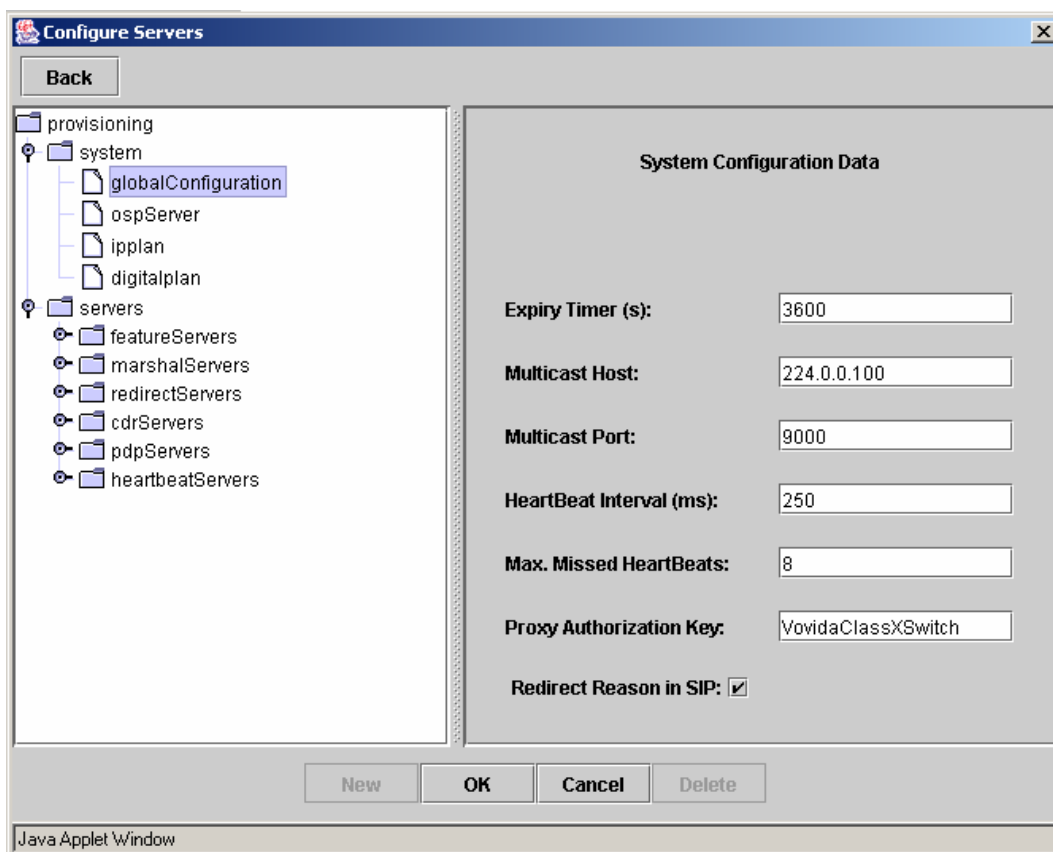
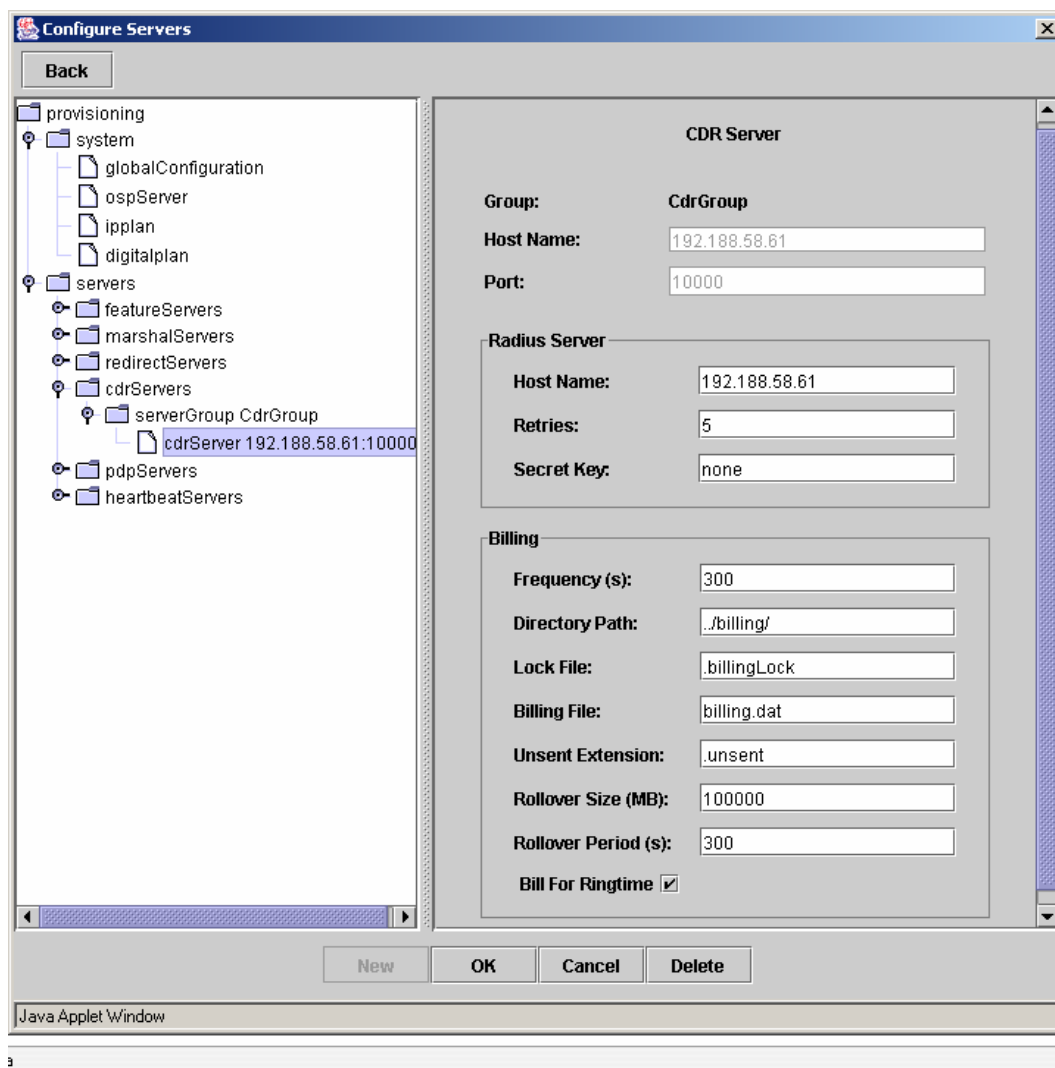


Figura 3.35. Configuración Global

8. Dentro de esta ventana en el directorio *system*, en *global configuration*, deben estar los parámetros tal como se muestran en la figura 3.35, el único parámetro que puede estar distinto a como se muestra en la figura es el de *Proxy Authorization Key*; de no estar de esta forma se deberá realizar los cambios en los mencionados parámetros, para que queden tal como en la figura.
  
9. Dentro de la misma ventana en el directorio *servers*, en *cdrServers*, se deberá crear un CDR Server, con los parámetros tal como se muestran en la siguiente figura:



**Figura 3.36. Creación del CDR Server**

10. Como paso final se debe resetear VOCAL, con la siguiente línea de comandos desde una ventana tipo Terminal:

```
/usr/local/vocal/bin/vocalctl stop (parar VOCAL)
```

```
/usr/local/vocal/bin/vocalctl start (iniciar VOCAL)
```

El CDR Server generará archivos cada 10 minutos en los que constan los detalles de las llamadas realizadas a través de VOCAL, los mencionados archivos se alojarán en el directorio `/usr/local/vocal/billing` y tienen la siguiente forma:

**billing.dat.061105-115119**

en donde el primer número de la extensión del archivo que para el ejemplo anterior es el *061105* , es la fecha en la que se generó dicho archivo, en el siguiente formato *mes día año*.

Cada llamada a través de VOCAL genera varias entradas dentro de los archivos *billing.dat.xxxxx*, de estas entradas las que son de interés son aquellas que empiecen con la bandera CALL\_BILL, dentro de este campo hay 27 identificadores separados con una coma y son los siguientes:

```
01. call_event,  
02. call_id,  
03. user_id,  
04. ani,  
05. dtmf_called_number,  
06. e164_called_number,  
07. formatted_start_date,  
08. formatted_start_time,  
09. start_date_time_secs,  
10. start_time_msecs,  
11. formatted_end_date,  
12. formatted_end_time,  
13. end_date_time_secs,  
14. end_time_msecs,  
15. start_ring_time_secs,  
16. start_ring_time_msecs,  
17. formatted_duration,  
18. duration_secs,  
19. duration_msecs,  
20. originator_ip,  
21. originator_line,  
22. terminator_ip,  
23. terminator_line,  
24. call_type,  
25. call_parties,  
26. call_disconnect,  
27. call_direction
```

**Figura 3.37. Identificadores CALL\_BILL**

En el siguiente ejemplo se visualiza el campo CALL\_BILL con sus 27 identificadores:

```
billing.dat.081905-133352.unsent:CALL_BILL,01241607-43e8-42a5-b835-  
45a249fed4a3@10.0.0.228,2000,,2005,,08/19/2005,18:25:33,1124475933,391,08/19/2005,18:26:0  
8,1124475968,762,1124475933,391,000:35:371,35,371,192.188.58.61,0,192.188.58.61,0,V,1,N,I
```

### 3.8.3. CONFIGURANDO USUARIOS O EXTENSIONES DE VOCAL

Desde el navegador ingresar a la dirección `http://<hostname>o<IP de VOCAL>/vocalconf/webconf.cgi`, en donde se debe hacer click en el ítem *Web Based Provisioning*, para añadir usuarios o extensiones dentro del sistema VOCAL y aparecerá la siguiente figura:

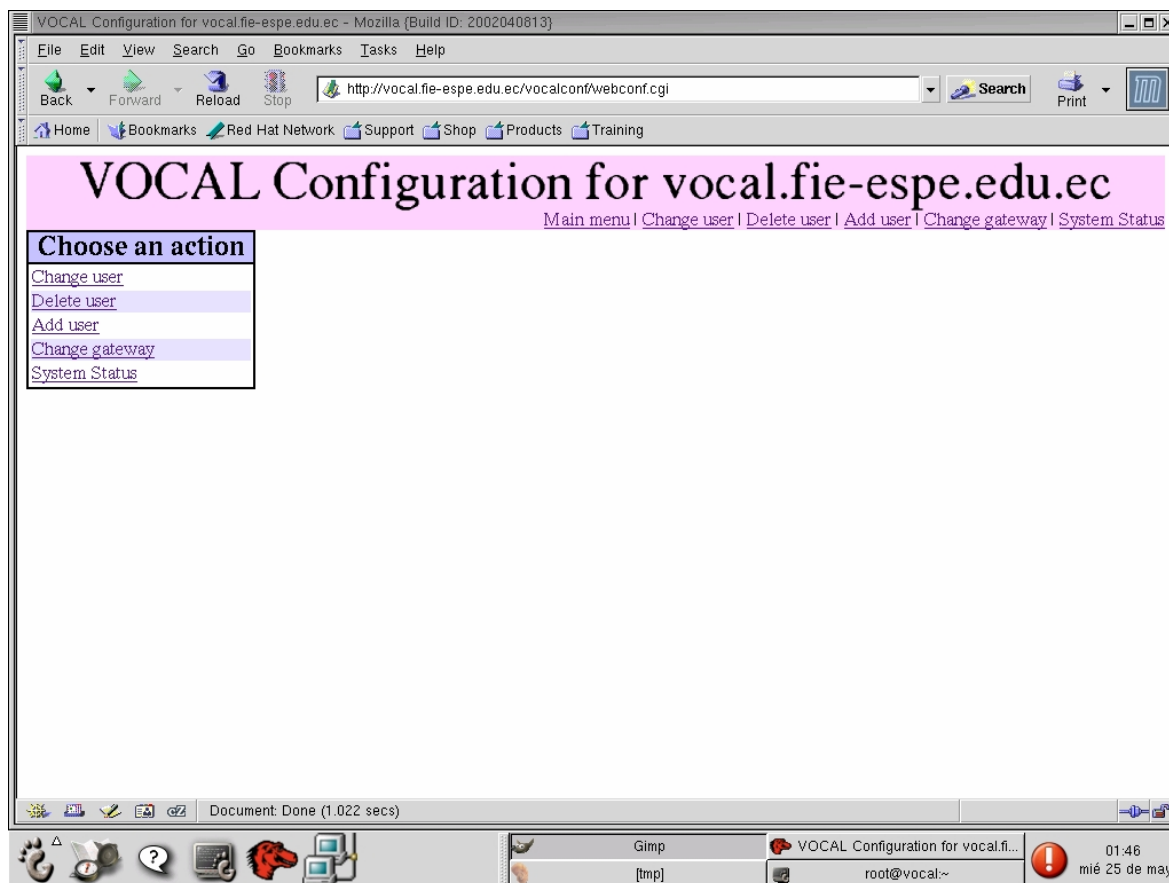
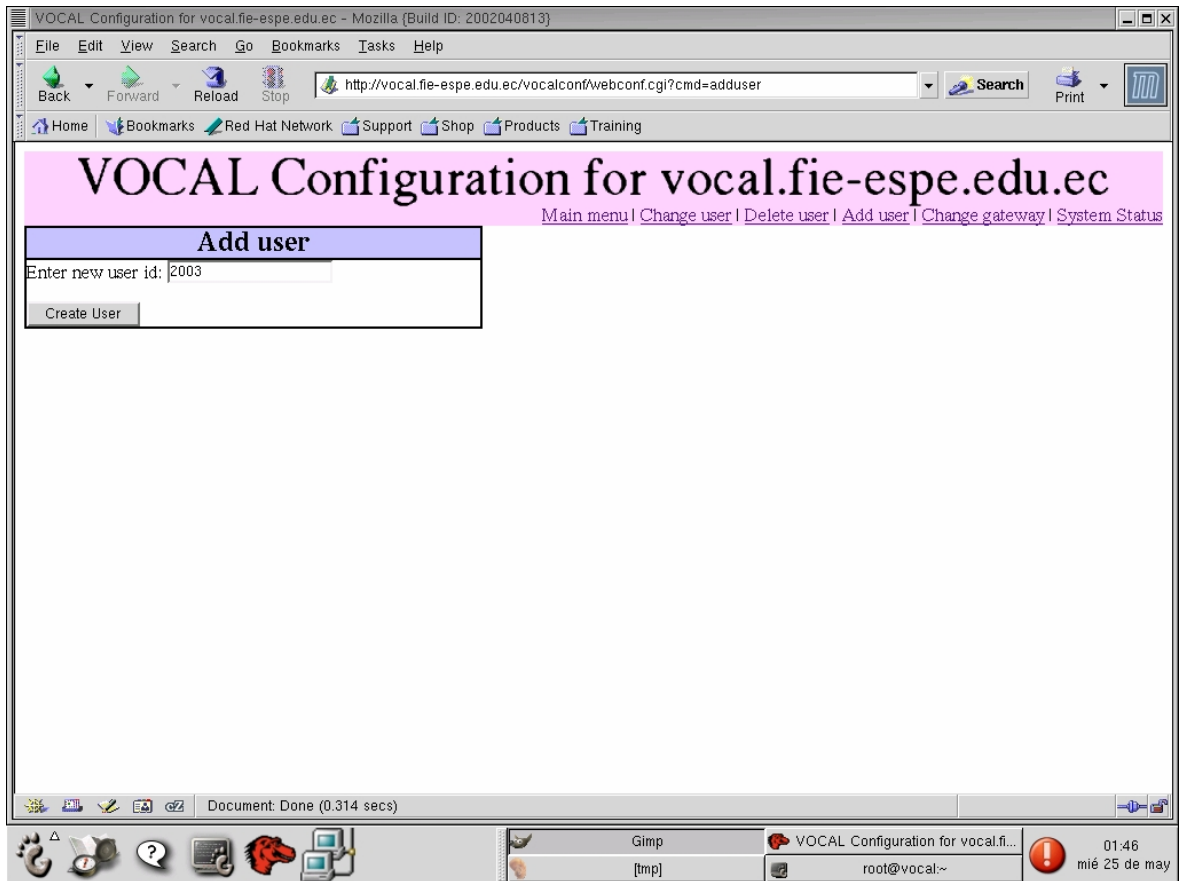


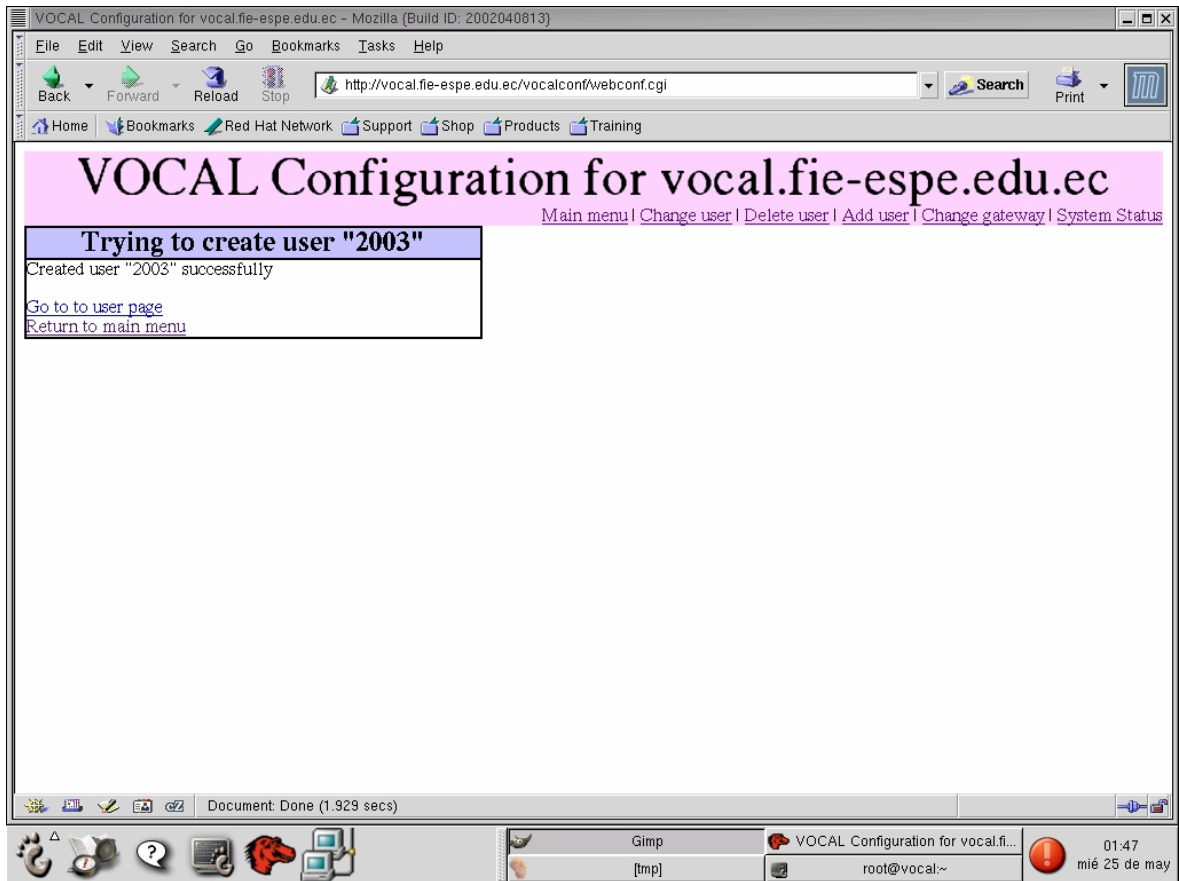
Figura 3.38. Menú principal de Web Based Provisioning del sistema VOCAL

Dentro de esta pantalla se hará click en el ítem *Add user*, para poder ingresar los usuarios o extensiones del sistema VOCAL, que deberán ser cuatro dígitos; posterior a ingresar estos dígitos, se hará click en el botón *Create user* ; como se muestra en la siguiente figura:



**Figura 3.39. Añadir usuarios en el sistema VOCAL**

Si este procedimiento se realizó de forma adecuada, se muestra el siguiente mensaje en la pantalla:



**Figura 3.40. Creación de un usuario o extensión del sistema VOCAL en forma exitosa**

Dentro de esta página se hará click en el enlace *Go to to user page* y se llenarán los siguientes parámetros: *User*, *User password*, *Password*, *Allow local calls* y *Allow Gateway calls*; como se muestra en la siguiente figura:

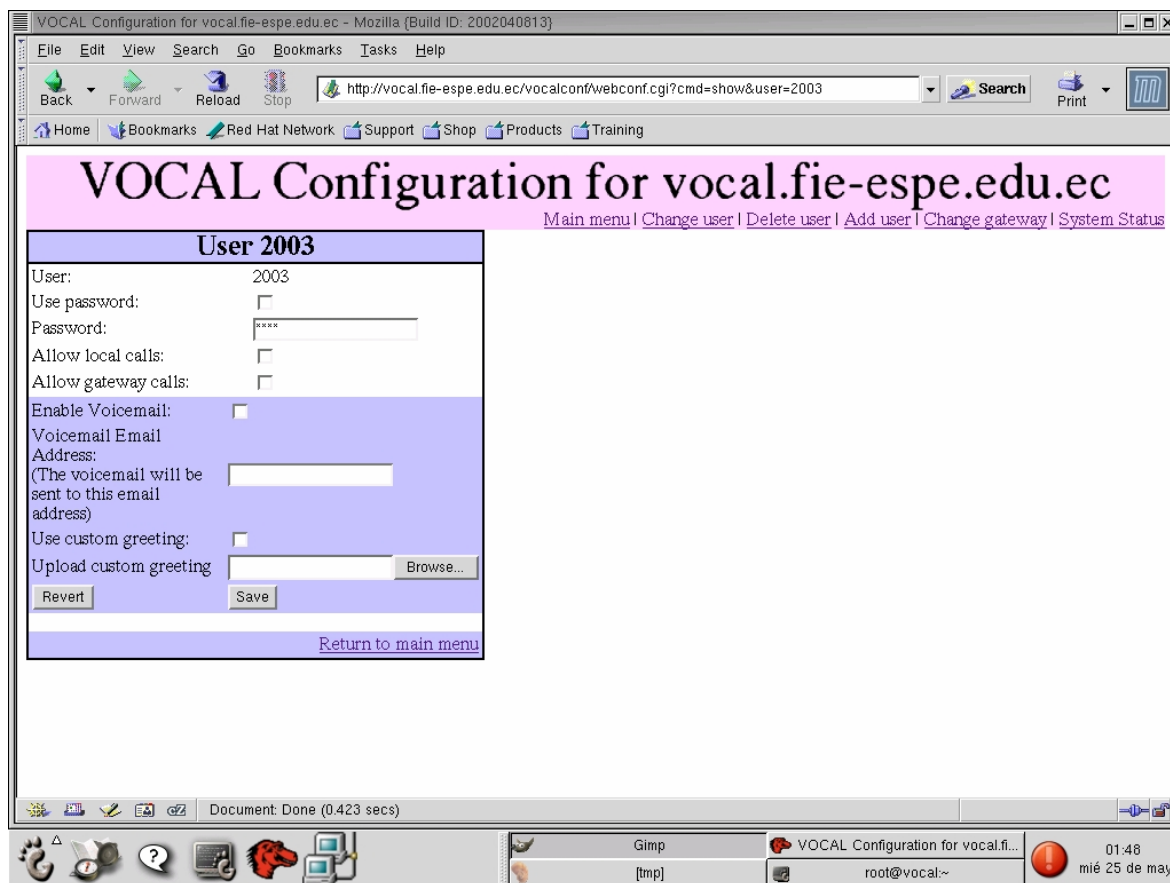


Figura 3.41. Parámetros de configuración de los usuarios o extensiones del sistema VOCAL

Todo el proceso antes mencionado se repetirá tantas veces como usuarios se desee ingresar.

### 3.9. CONFIGURACIÓN DE TELEFONOS IP

Los teléfonos IP que se utilizaron son softphones, es decir un software que hace las veces de un teléfono y para este proyecto en particular fue el softphone de nombre *eStara* que puede ser descargado gratuitamente desde Internet de la página <http://www.estara.com>

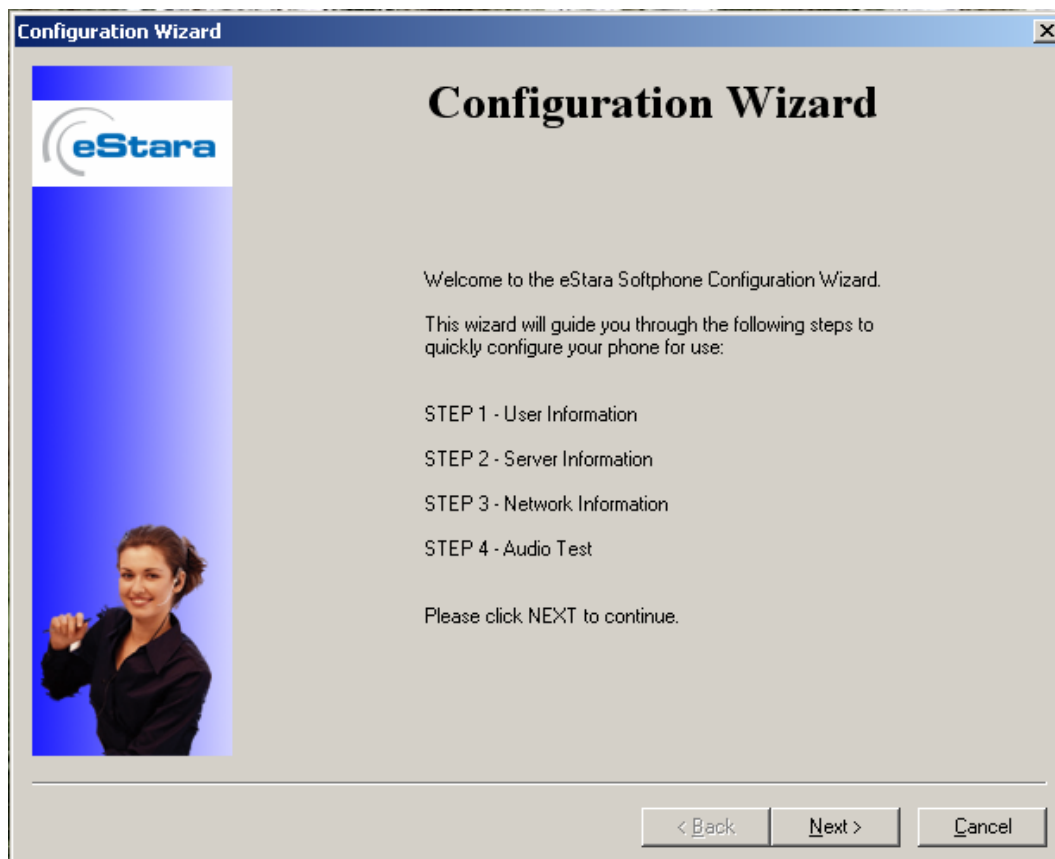
Este teléfono tiene como principal característica la de utilizar SIP como protocolo de Voz sobre IP o VoIP, además de otras facilidades como poder realizar o atender tres llamadas al mismo tiempo como si se tuviera tres líneas a la vez, numeración para marcado telefónico, identificador del teléfono, identificador de llamadas, tiempo de duración de la llamada, codificación de voz con algoritmos como G.723, G.711, libreta de direcciones,



controles de volumen tanto del micrófono como de los parlantes, etc. La instalación de este software es muy sencilla y no reviste complicaciones.

La configuración de estos teléfonos se detalla a continuación:

#### 11. Inicio del Configuration Wizard del softphone eStara:



**Figura 3.42. Configuration Wizard del softphone eStara**

12. Dentro de la información del usuario consta el *Friendly Name*, que es el nombre que tendrá el teléfono, además de la *IP Address* de la PC en la cual se instala dicho softphone:

**Configuration Wizard**

**eStara**

## User Information

Enter a friendly name.  
This name will be displayed to people who you call so they can identify you.

**Friendly Name**

Select the IP Address of your computer.  
If you are unsure of which IP Address to select, ask  
your Network Administrator for advice.

**IP Address**

< Back    Next >    Cancel

**Figura 3.43. Información del usuario**

13. Información del servidor SIP, en donde se detalla campos como: *SIP Server Address* que es la dirección IP del sistema VOCAL, *SIP Server Port Number* que es la 5060, *SIP Server Username* que es el usuario o extensión que se ingreso en el sistema VOCAL y *SIP Server Password* que es la contraseña que se ingreso junto con el usuario dentro de VOCAL.

**Configuration Wizard**

**eStara**

## Server Information

eStara's Softphone may be configured to connect to a SIP Server. Doing so will allow you to place calls to and receive calls from regular telephones. If you would like to signup for an eStara SIP Account, please visit <http://www.estara.com/softphone/signup.php>.

**Would like to connect to a SIP Server?**

No - I only need to connect with other softphones

Yes - I need to connect with regular telephones

**Enter your SIP Server's address and port number.**

SIP Server Address: 192.188.58.61

SIP Server Port Number: 5060

SIP Server Username: 2000

SIP Server Password: \*\*\*\*

< Back    Next >    Cancel

**Figura 3.44. Información del Servidor SIP**

14. Información de la Red en la cual esta el softphone; en donde se detalla campos como si el teléfono esta conectado a una dispositivo que hace NAT (Network Address Translation), que en caso de escoger la opción *Static NAT routing*, se deberá especificar la *NAT IP Address* que es la dirección IP del dispositivo que esta haciendo NAT en la red y *NAT Port Number* que es la 5060; o en caso de escoger la opción *STUN*, se deberá especificar *STUN Server Domain Name* que es *stun.estara.com* y *STUN ServerPort Number* que es el 8000.

**Configuration Wizard**

**eStara**

## Network Information

eStara's Softphone may be configured to operate through a Network Address Translator (NAT). You may configure the softphone to use Static NAT routing or to use a STUN Server.

**Is your computer connected to a NAT device?**

No - My computer has been assigned a publicly routable IP Address

No - I have a NAT aware SIP proxy that supports COMEDIA

Yes - I want to use Static NAT routing

Yes - I want to use STUN

**Enter your NAT's IP Address and a port number to use.**  
**You must configure your NAT to forward all inbound UDP traffic to this computer.**

NAT IP Address

NAT Port Number

**Enter your STUN Server's domain name (or IP Address) and port number.**

STUN Server Domain Name

STUN Server Port Number

< Back    Next >    Cancel

Figura 3.45. Información de la Red

15. El audio test que permite configurar los niveles de volumen tanto en el micrófono como en los parlantes:



Figura 3.46. Audio Test

16. Finalización de la configuración del softphone eStara:

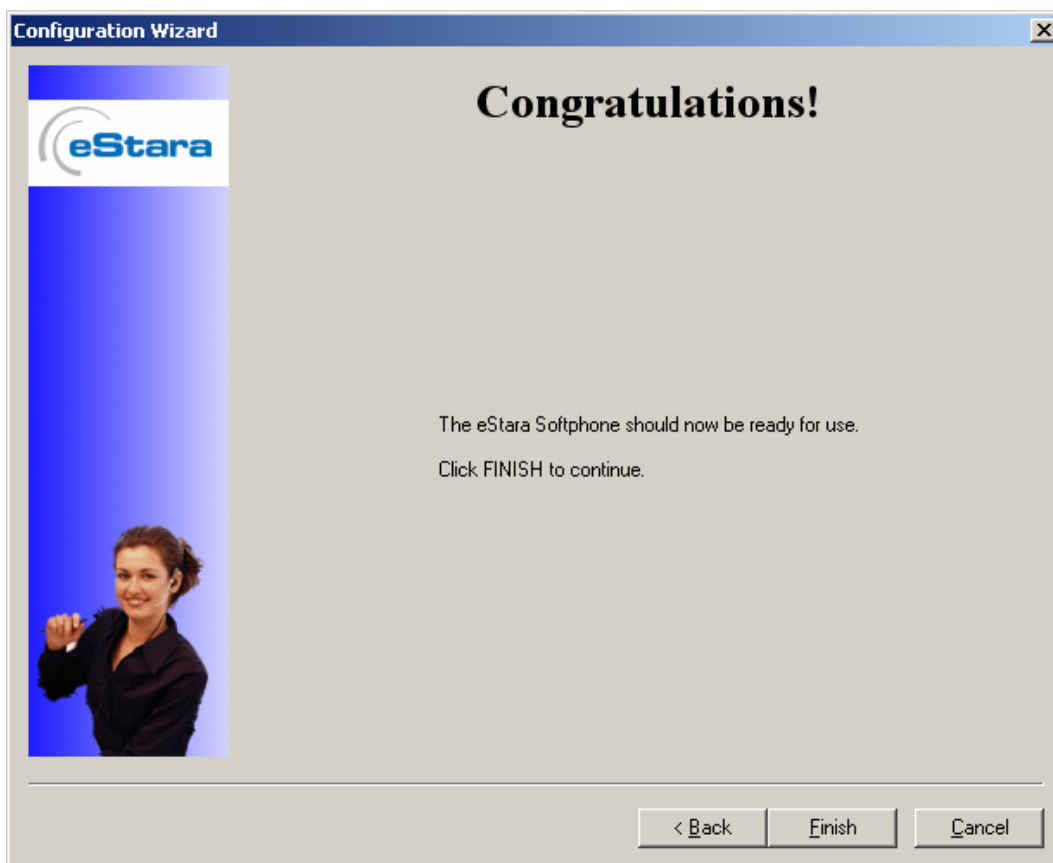


Figura 3.47. Finalización de la configuración del softphone eStara

17. Posterior a la configuración del softphone aparece un mensaje de *Login Progress*:

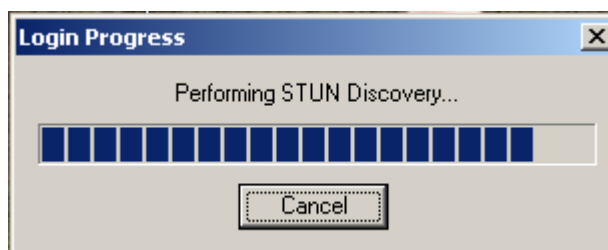


Figura 3.48. Login Progress

18. Posterior al Login Progress aparecerá el softphone, cuyo interfaz gráfico es muy amigable para el usuario final:



Figura 3.49. eStara softphone

## **CAPITULO IV**

### **4. EVALUACIÓN DEL SISTEMA**

En este capítulo se presentarán las demostraciones de funcionamiento de los servicios que proporciona el Sistema Emulador VOCAL y el correspondiente análisis de evaluación de desempeño del mismo.

#### **4.1. INTRODUCCIÓN**

Una vez configurado tanto el sistema emulador así como los teléfonos IP, se procedió a realizar pruebas de desempeño del sistema en conjunto, que en principio fueron realizadas a nivel de LAN, de la red FIE. Posterior a las pruebas a nivel de la LAN se realizaron las pruebas a nivel de WAN, a través de llamadas de larga distancia, gracias a la colaboración de una empresa especializada en soluciones empresariales de VoIP bajo Linux. Además se realizó el análisis del tráfico generado por la llamada, es decir que ancho de banda necesita el usuario para realizar una llamada a través del sistema VOCAL.

#### **4.2. PRUEBAS A NIVEL DE LAN**

Como se mencionó anteriormente, en principio el sistema en conjunto fue probado a nivel de LAN, con los softphones instalados en las PCs, el sistema VOCAL y la red FIE, como se muestra en la siguiente figura:



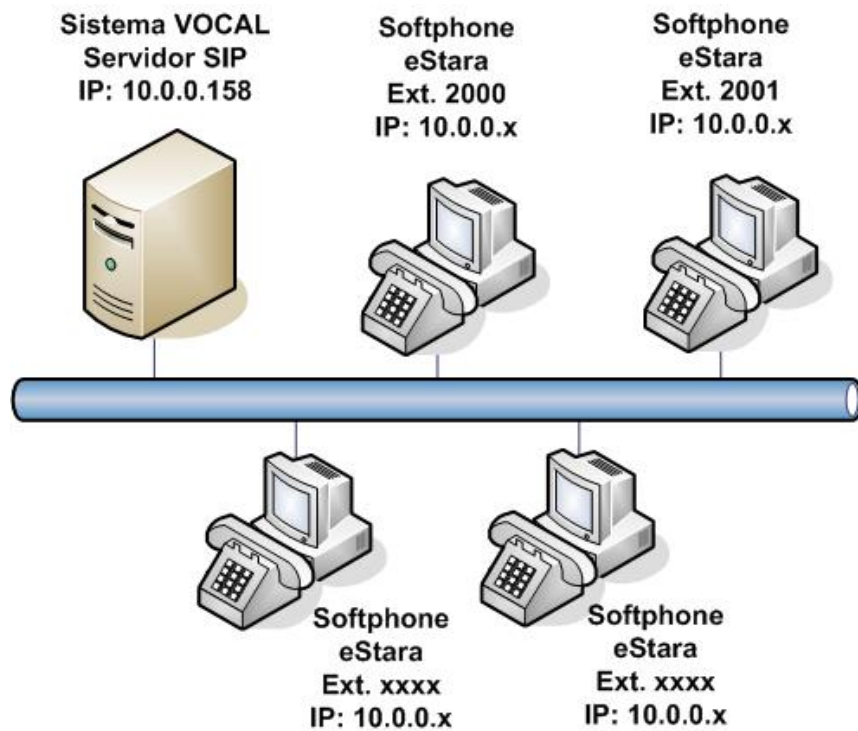


Figura 4.1. Sistema emulador VOCAL a nivel de LAN

Como se representa en la figura anterior tenemos al sistema VOCAL con una dirección IP: 10.0.0.158, y algunas PCs que tienen instalados softphones que harán las veces de teléfonos para las pruebas de desempeño, los archivos generados por el servidor CDR (Call Detail Record) servirán de base para saber como se desarrollo la llamada, en la figura siguiente se muestra, la salida de los archivos `billing.dat.mmddaa-xxxxxx` , con el identificador `CALL_BILL`, mostradas en una ventana de sesión SSH a través del cliente SSH de nombre Putty:

```

root@vocal:usr/local/vocal/billing
[root@vocal root]# cd /usr/local/vocal/billing/
[root@vocal billing]# grep CALL_BILL *
billing.dat.063005-175001.unsent:CALL_BILL,3f8944e6-4067-47a4-8a8f-7ff5223119fd@10.0.0.239,2001,,2002,,06/30/2005,22:41:55,112
0171315,235,06/30/2005,22:42:31,1120171315,306,1120171315,235,000:36:71,36,71,10.0.0.158,0,10.0.0.158,0,V,1,N,I
billing.dat.063005-175001.unsent:CALL_BILL,a6cf1ee5-ab56-422d-9b9b-5ea1c85807eb@10.0.0.228,2002,,2001,,06/30/2005,22:46:40,112
0171600,156,06/30/2005,22:47:16,1120171636,443,1120171600,156,000:36:287,36,287,10.0.0.158,0,10.0.0.158,0,V,1,N,I
billing.dat.081805-143628.unsent:CALL_BILL,34148e96-1eb3-4e8e-a85d-1alc4d4aba347@10.0.0.228,2000,,3000,,08/18/2005,19:28:04,112
4393284,401,08/18/2005,19:29:22,1124393362,591,1124393284,401,001:18:190,78,190,192.188.58.61,0,192.188.58.61,0,V,1,N,I
billing.dat.081805-143628.unsent:CALL_BILL,ec4ae121-8ec2-4597-a7c7-ef7a9c1f2824@10.0.0.238,2001,,2000,,08/18/2005,19:35:07,112
4393707,92,08/18/2005,19:35:59,1124393759,253,1124393707,92,000:52:161,52,161,192.188.58.61,0,192.188.58.61,0,V,1,N,I
billing.dat.081805-144628.unsent:CALL_BILL,63936eb0-771e-4539-bec4-5e9e0cc2d348@10.0.0.238,2001,,2000,,08/18/2005,19:38:33,112
4399913,502,08/18/2005,19:39:19,1124393959,502,1124393913,502,000:46:00,46,0,192.188.58.61,0,192.188.58.61,0,V,1,N,I
billing.dat.081805-144628.unsent:CALL_BILL,b2115175-cda5-4816-bae7-e3c69ba62241@10.0.0.238,2001,,2000,,08/18/2005,19:43:19,112
4394199,534,08/18/2005,19:43:39,1124394219,390,1124394199,534,000:19:00,19,0,192.188.58.61,0,192.188.58.61,0,V,1,N,I
billing.dat.081805-145628.unsent:CALL_BILL,81535f4e-5fec-44a1-8545-b4ba10c33932@10.0.0.238,2001,,2000,,08/18/2005,19:50:13,112
4394613,973,08/18/2005,19:51:55,1124394715,649,1124394613,973,001:41:00,101,0,192.188.58.61,0,192.188.58.61,0,V,1,N,I
billing.dat.081805-150628.unsent:CALL_BILL,245b9a41-a3e0-4db8-a51a-78149ef2c42f@10.0.0.228,2000,,2001,,08/18/2005,19:58:52,112
4395132,256,08/18/2005,19:59:53,1124395193,196,1124395132,256,001:00:00,60,0,192.188.58.61,0,192.188.58.61,0,V,1,N,I
billing.dat.081805-150628.unsent:CALL_BILL,1bfb7beb-97d0-4290-981c-8b5bb1dacd53@10.0.0.228,2000,,2001,,08/18/2005,20:01:42,112
4395302,101,08/18/2005,20:02:23,1124395343,32,1124395302,101,000:40:00,40,0,192.188.58.61,0,192.188.58.61,0,V,1,N,I
billing.dat.081805-151628.unsent:CALL_BILL,0007eb20-ad000045-689d7a8e-2d2c1a76@170.76.33.196,3000,,2001,,08/18/2005,20:03:36,1
124395416,50,08/18/2005,20:07:59,1124395679,623,1124395416,50,004:23:573,263,573,192.188.58.61,0,192.188.58.61,0,V,1,N,I
billing.dat.081805-151628.unsent:CALL_BILL,0007eb20-ad000046-109e819b-6fb778e3@170.76.33.196,3000,,2001,,08/18/2005,20:13:25,1
124396005,549,08/18/2005,20:13:59,1124396039,32,1124396005,549,000:33:00,33,0,192.188.58.61,0,192.188.58.61,0,V,1,N,I
billing.dat.081805-152628.unsent:CALL_BILL,0007eb20-ad000047-5c468b01-1fcbcb9b@170.76.33.196,3000,,2000,,08/18/2005,20:14:49,1
124396089,957,08/18/2005,20:21:01,1124396461,209,1124396089,957,006:11:00,371,0,192.188.58.61,0,192.188.58.61,0,V,1,N,I
billing.dat.081805-153628.unsent:CALL_BILL,58f1afcc-6246-4571-a762-bb784e3511e2@10.0.0.228,2000,,2001,,08/18/2005,20:26:03,112
4396763,143,08/18/2005,20:26:35,1124396795,265,1124396763,143,000:32:122,32,122,192.188.58.61,0,192.188.58.61,0,V,1,N,I
billing.dat.081805-153628.unsent:CALL_BILL,5f1dd923-068a-4ee3-9fcf-b5329417c55b@10.0.0.228,2000,,2001,,08/18/2005,20:31:12,112
4397072,582,08/18/2005,20:31:42,1124397102,212,1124397072,582,000:29:00,29,0,192.188.58.61,0,192.188.58.61,0,V,1,N,I
billing.dat.081805-153628.unsent:CALL_BILL,03015c5e-0d01-4b36-891f-79c9cf8ce45fb@10.0.0.238,2001,,3000,,08/18/2005,20:33:58,112
4397238,914,08/18/2005,20:34:41,1124397281,397,1124397238,914,000:42:00,42,0,192.188.58.61,0,192.188.58.61,0,V,1,N,I
billing.dat.081905-100352.unsent:CALL_BILL,8c5b2e6c-8465-4cda-87ea-19e0dc779eb1@10.0.0.238,2001,,2000,,08/19/2005,15:01:02,112
4463662,595,08/19/2005,15:01:22,1124463682,70,1124463662,595,000:19:00,19,0,192.188.58.61,0,192.188.58.61,0,V,1,N,I
billing.dat.081905-102352.unsent:CALL_BILL,08c3fd22-ba4c-4aa5-a685-0c5d22b134f4@10.0.0.228,2000,,2005,,08/19/2005,15:15:02,112
4464502,968,08/19/2005,15:15:37,1124464537,826,1124464502,968,000:34:00,34,0,192.188.58.61,0,192.188.58.61,0,V,1,N,I
billing.dat.081905-112352.unsent:CALL_BILL,130ded65-87f5-46cc-82c8-ff6a9b510cb6@10.0.0.228,2000,,2005,,08/19/2005,16:15:36,112
4468136,933,08/19/2005,16:16:08,1124468168,635,1124468136,933,000:31:00,31,0,192.188.58.61,0,192.188.58.61,0,V,1,N,I
billing.dat.081905-112352.unsent:CALL_BILL,d9326434-f30e-4d07-8b3f-7f2a6c23f087@10.0.0.228,2000,,2005,,08/19/2005,16:18:14,112
4468294,22,08/19/2005,16:18:34,1124468314,353,1124468294,22,000:20:331,20,331,192.188.58.61,0,192.188.58.61,0,V,1,N,I
billing.dat.081905-123352.unsent:CALL_BILL,ce478c48-77f9-4ddb-8c5b-3f7de8ec8c12@10.0.0.228,2000,,6665939568212263,,08/19/2005,
17:29:47,1124472587,42,08/19/2005,17:33:37,1124472817,374,1124472587,42,003:49:00,229,0,192.188.58.61,0,192.188.58.61,0,V,1,
N,I
billing.dat.081905-130352.unsent:CALL_BILL,3a63e711-f357-4803-bb1c-e3f47a9c1d71@10.0.0.228,2000,,6665939568212263,,08/19/2005,

```

Figura 4.2. Archivos billing.dat en las pruebas a nivel de LAN

Tomando uno de los archivos billing.dat, como por ejemplo el de la segunda línea tenemos lo siguiente:

```

billing.dat.063005-175001.unsent:CALL_BILL,a6cf1ee5-ab56-422d-9b9b-
5ea1c85807eb@10.0.0.228,2002,,2001,,06/30/2005,22:46:40,1120171600,156,06/30/2005,22:47:
16,1120171636,443,1120171600,156,000:36:287,36,287,10.0.0.158,0,10.0.0.158,0,V,1,N,I

```

De acuerdo con cada uno de los campos de los archivos billing.dat, se tiene:

01. call\_event, CALL\_BILL
02. call\_id, a6cf1ee5-ab56-422d-9b9b-5ea1c85807eb@10.0.0.228
03. user\_id, 2002
04. ani,
05. dtmf\_called\_number, 2001
06. el64\_called\_number,
07. formatted\_start\_date, 06/30/2005
08. formatted\_start\_time, 22:46:40

09. start\_date\_time\_secs, 1120171600
10. start\_time\_msecs, 156
11. formatted\_end\_date, 06/30/2005
12. formatted\_end\_time, 22:47:16
13. end\_date\_time\_secs, 1120171636
14. end\_time\_msecs, 443
15. start\_ring\_time\_secs, 1120171600
16. start\_ring\_time\_msecs, 156
17. formatted\_duration, 000:36:287
18. duration\_secs, 36
19. duration\_msecs, 287
20. originator\_ip, 10.0.0.158
21. originator\_line, 0
22. terminator\_ip, 10.0.0.158
23. terminator\_line, 0
24. call\_type, V
25. call\_parties, 1
26. call\_disconnect, N
27. call direction, I

De esta información se desprende que el usuario 2002 con IP 10.0.0.228, llamo al usuario 2001, el 30 de Junio del 2005, a las 22:46:40 de Greenwich, termino la llamada el mismo 30 de Junio del 2005, a las 22:47:16, la duración de la llamada fue de 36 segundos y 287 milisegundos, la llamada se origino a través de la IP 10.0.0.158 y se termino en la misma IP 10.0.0.158

### 4.3. PRUEBAS A NIVEL DE WAN

Las pruebas a nivel de WAN, se realizo a través de llamadas de larga distancia, gracias a la colaboración de una empresa especializada en soluciones empresariales de VoIP bajo Linux, como se muestra en la siguiente figura:

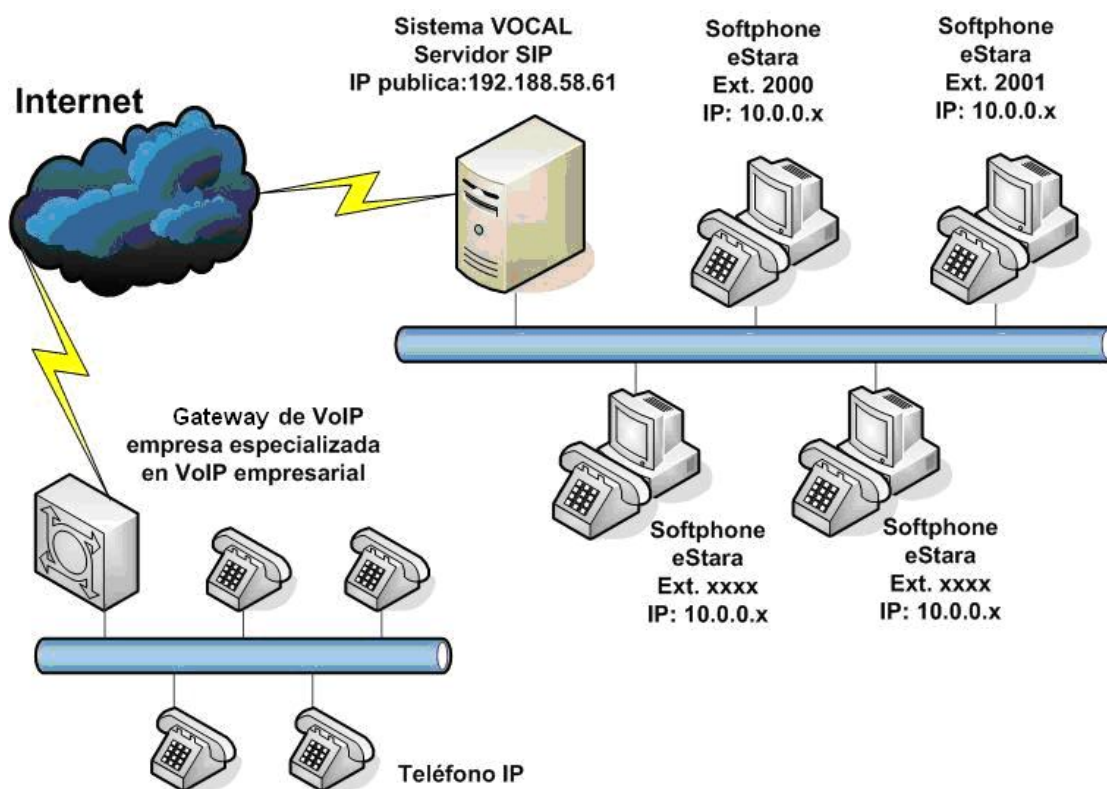
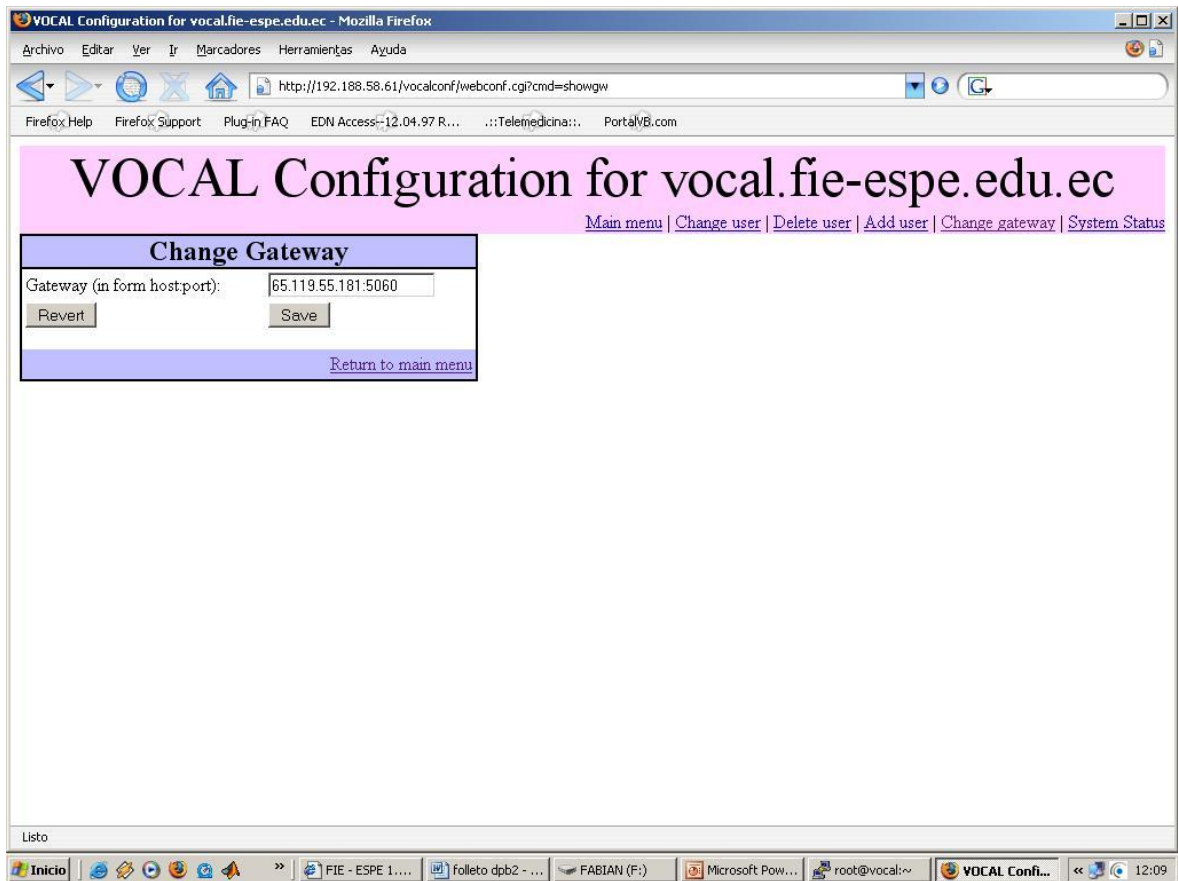


Figura 4.3. Sistema emulador VOCAL a nivel de WAN

Como se representa en la figura anterior tenemos al sistema VOCAL con una dirección IP válida: 192.188.58.61 y algunas PCs que tienen instalados softphones que harán las veces de teléfonos para las pruebas de desempeño. Además en la configuración de VOCAL se debe especificar el gateway, que es el que permite realizar llamadas fuera de la red interna, ya que si el sistema detecta que el usuario al que se está llamando no está dentro de la misma red, utiliza este gateway para encontrar dicho usuario; en la siguiente figura se muestra la pantalla de configuración del gateway vía web:



**Figura 4.4. Configuración del Gateway del sistema VOCAL**

Los archivos generados por el servidor CDR (Call Detail Record) servirán de base para saber como se desarrollo la llamada, en la figura siguiente se muestra, la salida de los archivos billing.dat.mmddaa-xxxxxx , con el identificador CALL\_BILL, mostradas en una ventana de sesión SSH a través del cliente SSH de nombre Putty:

```

root@vocal:usr/local/vocal/billing
17:57:03,1124474223,179,08/19/2005,17:57:30,1124474250,398,1124474223,179,000:27:219,27,219,192.188.58.61,0,192.188.58.61,0,V,1,N,I
billing.dat.081905-130352.unsent:CALL_BILL,8c1d74e7-7ced-4d04-a9bd-254cce29d300@10.0.0.228,2000,,6665939568212263,,08/19/2005,17:57:04,1124474224,620,08/19/2005,17:58:55,1124474335,709,1124474224,620,001:51:89,111,89,192.188.58.61,0,192.188.58.61,0,V,1,N,I
billing.dat.081905-130352.unsent:CALL_BILL,37e045e6-cbf5-444f-ada1-7594a96134ef@10.0.0.228,2000,,6665939568212263,,08/19/2005,17:57:53,1124474273,597,08/19/2005,18:00:26,1124474426,275,1124474273,597,002:32:00,152,0,192.188.58.61,0,192.188.58.61,0,V,1,N,I
billing.dat.081905-130352.unsent:CALL_BILL,263a361b-0a4e-4357-96d8-f5b0549918a1@10.0.0.228,2000,,6665932102993298,,08/19/2005,18:00:52,1124474452,305,08/19/2005,18:01:37,1124474497,180,1124474452,305,000:44:00,44,0,192.188.58.61,0,192.188.58.61,0,V,1,N,I
billing.dat.081905-131352.unsent:CALL_BILL,b473fa6e-405a-4ece-a090-fe709ffa233b@10.0.0.228,2000,,6665932507480818,,08/19/2005,18:08:17,1124474897,22,08/19/2005,18:08:54,1124474934,346,1124474897,22,000:37:324,37,324,192.188.58.61,0,192.188.58.61,0,V,1,N,I
billing.dat.081905-132352.unsent:CALL_BILL,1f187dbe-2f59-4068-aba8-8fb0050d79fe@10.0.0.238,2005,,2000,,08/19/2005,18:21:36,1124475696,145,08/19/2005,18:22:50,1124475770,574,1124475696,145,001:14:429,74,429,192.188.58.61,0,192.188.58.61,0,V,1,N,I
billing.dat.081905-133352.unsent:CALL_BILL,fe224c86-4374-427e-8e6f-5ce3a43281d8@10.0.0.238,2005,,6665939545366599,,08/19/2005,18:22:55,1124475775,950,08/19/2005,18:24:14,1124475854,5,1124475775,950,001:18:00,78,0,192.188.58.61,0,192.188.58.61,0,V,1,N,I
billing.dat.081905-133352.unsent:CALL_BILL,36252090-5f7f-40f2-a81b-0e8bef056151@10.0.0.238,2005,,6665939545366599,,08/19/2005,18:13:51,1124475231,274,08/19/2005,18:24:15,1124475855,367,1124475231,274,010:24:93,624,93,192.188.58.61,0,192.188.58.61,0,V,1,N,I
billing.dat.081905-133352.unsent:CALL_BILL,01241607-43e8-42a5-b835-45a249fed4a3@10.0.0.228,2000,,2005,,08/19/2005,18:25:33,1124475933,391,08/19/2005,18:26:08,1124475968,762,1124475933,391,000:35:371,35,371,192.188.58.61,0,192.188.58.61,0,V,1,N,I
billing.dat.081905-133352.unsent:CALL_BILL,024a0b5e-06df-4196-a365-a00a456caa56@10.0.0.238,2005,,2000,,08/19/2005,18:26:17,1124475977,727,08/19/2005,18:26:40,1124476000,910,1124475977,727,000:23:183,23,183,192.188.58.61,0,192.188.58.61,0,V,1,N,I
billing.dat.081905-133352.unsent:CALL_BILL,2e15aeae-6cf2-44e6-a152-8e3d5715775f@10.0.0.238,2005,,6665933052574713,,08/19/2005,18:32:01,1124476321,445,08/19/2005,18:32:32,1124476352,926,1124476321,445,000:31:481,31,481,192.188.58.61,0,192.188.58.61,0,V,1,N,I
billing.dat.081905-134352.unsent:CALL_BILL,bfc5feb4-4ce8-4362-b13c-ae1ea90da342@10.0.0.238,2005,,6665937864731116,,08/19/2005,18:34:52,1124476492,335,08/19/2005,18:35:32,1124476532,307,1124476492,335,000:39:00,39,0,192.188.58.61,0,192.188.58.61,0,V,1,N,I
billing.dat.081905-144352.unsent:CALL_BILL,20b4fb58-ff5c-4243-6b67-c5494d22a6ed@10.0.0.228,2000,,6665936463730344,,08/19/2005,19:41:48,1124480508,602,08/19/2005,19:42:32,1124480552,305,1124480508,602,000:43:00,43,0,192.188.58.61,0,192.188.58.61,0,V,1,N,I
billing.dat.081905-151520.unsent:CALL_BILL,ce2c33e2-0563-4a82-a404-0cd484c91f74@10.0.0.238,2005,,6665936507400354,,08/19/2005,20:02:22,1124481742,307,08/19/2005,20:13:30,1124482410,235,0,0,011:07:00,667,0,192.188.58.61,0,192.188.58.61,0,V,1,N,I
billing.dat.081905-160520.unsent:CALL_BILL,40f50872-6c3f-4c86-83d9-6359723da924@10.0.0.228,2000,,6665932123939469,,08/19/2005,20:47:36,1124484456,172,08/19/2005,20:57:08,1124485028,528,1124484456,172,009:32:356,572,356,192.188.58.61,0,192.188.58.61,0,V,1,N,I
billing.dat.081905-161520.unsent:CALL_BILL,3e408d37-774f-48ba-b34c-4c79e9d88ca1@10.0.0.228,2000,,6665939529034818,,08/19/2005,21:08:28,1124485708,902,08/19/2005,21:09:38,1124485778,690,1124485708,902,001:09:00,69,0,192.188.58.61,0,192.188.58.61,0,V,1,N,I
billing.dat.081905-162520.unsent:CALL_BILL,0c84d35e-5ea1-4821-a92e-62d6db2fd53@10.0.0.228,2000,,2005,,08/19/2005,21:14:04,1124486044,501,08/19/2005,21:15:19,1124486119,940,1124486044,501,001:15:439,75,439,192.188.58.61,0,192.188.58.61,0,V,1,N,I
[root@vocal billing]#

```

Figura 4.5. Archivos billing.dat en las pruebas a nivel WAN

Tomando uno de los archivos billing.dat, tenemos lo siguiente:

```

billing.dat.081905-160520.unsent:CALL_BILL,40f50872-6c3f-4c86-83d9-6359723da924@10.0.0.228,2000,6665932123939469,08/19/2005,20:47:36,1124484456,172,08/19/2005,20:57:08,1124485028,528,1124484456,172,009:32:356,572,356,192.188.58.61,0,192.188.58.61,0,V,1,N,I

```

De acuerdo con cada uno de los campos de los archivos billing.dat, se tiene:

01. call\_event, CALL\_BILL
02. call\_id, 40f50872-6c3f-4c86-83d9-6359723da924@10.0.0.228
03. user\_id, 2000
04. ani,
05. dtmf\_called\_number, 6665932123939469
06. el64\_called\_number,
07. formatted\_start\_date, 08/19/2005
08. formatted\_start\_time, 20:47:36

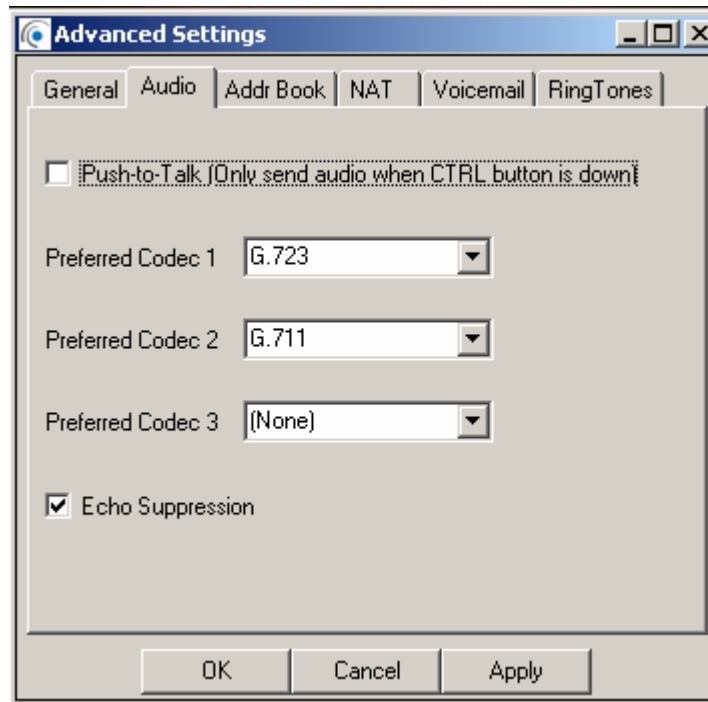
09. start\_date\_time\_secs, 1124484456
10. start\_time\_msecs, 172
11. formatted\_end\_date, 08/19/2005
12. formatted\_end\_time, 20:57:08
13. end\_date\_time\_secs, 1124485028
14. end\_time\_msecs, 528
15. start\_ring\_time\_secs, 1124484456
16. start\_ring\_time\_msecs, 172
17. formatted\_duration, 009:32:356
18. duration\_secs, 32
19. duration\_msecs, 356
20. originator\_ip, 192.188.58.61
21. originator\_line, 0
22. terminator\_ip, 192.188.58.61
23. terminator\_line, 0
24. call\_type, v
25. call\_parties, 1
26. call\_disconnect, N
27. call direction, I

De esta información se desprende que el usuario 2000 con IP 10.0.0.228, llamo al número 2123939469 en USA, el 19 de Agosto del 2005, a las 20:47:36 de Greenwich, termino la llamada el mismo 19 de Agosto del 2005, a las 20:57:08, la duración de la llamada fue de 9 minutos, 32 segundos y 356 milisegundos, la llamada se origino a través de la IP 192.188.58.61 y se termino en la misma IP 192.188.58.61

#### **4.4. ANALISIS DE TRAFICO GENERADO POR UNA LLAMADA A TRAVES DEL SITEMA VOCAL**

Para el análisis del tráfico generado por una llamada a través del sistema VOCAL, se recurrio a una herramienta de software de nombre PRTG Traffic Grapher, que entre algunas de sus potencialidades permite visualizar gráficamente cual es el ancho de banda que esta utilizando una dirección IP, en un protocolo y puerto específico, que para este caso en particular es el protocolo SIP y el puerto 5060.

El análisis se realizo con dos tipos de codificaciones de voz, G 723 y G 711; éstas pueden ser configuradas en el softphone, haciendo click en File, Settings, Audio, como se muestra en la siguiente figura:



**Figura 4.6.** Configuración de los codificadores de voz en el softphone eStara

Los resultados de las pruebas de tráfico, utilizando codificación G 723, muestran que el ancho de banda utilizado con este tipo de codificación es de aproximadamente 11 Kbps, como muestra la siguiente gráfica:



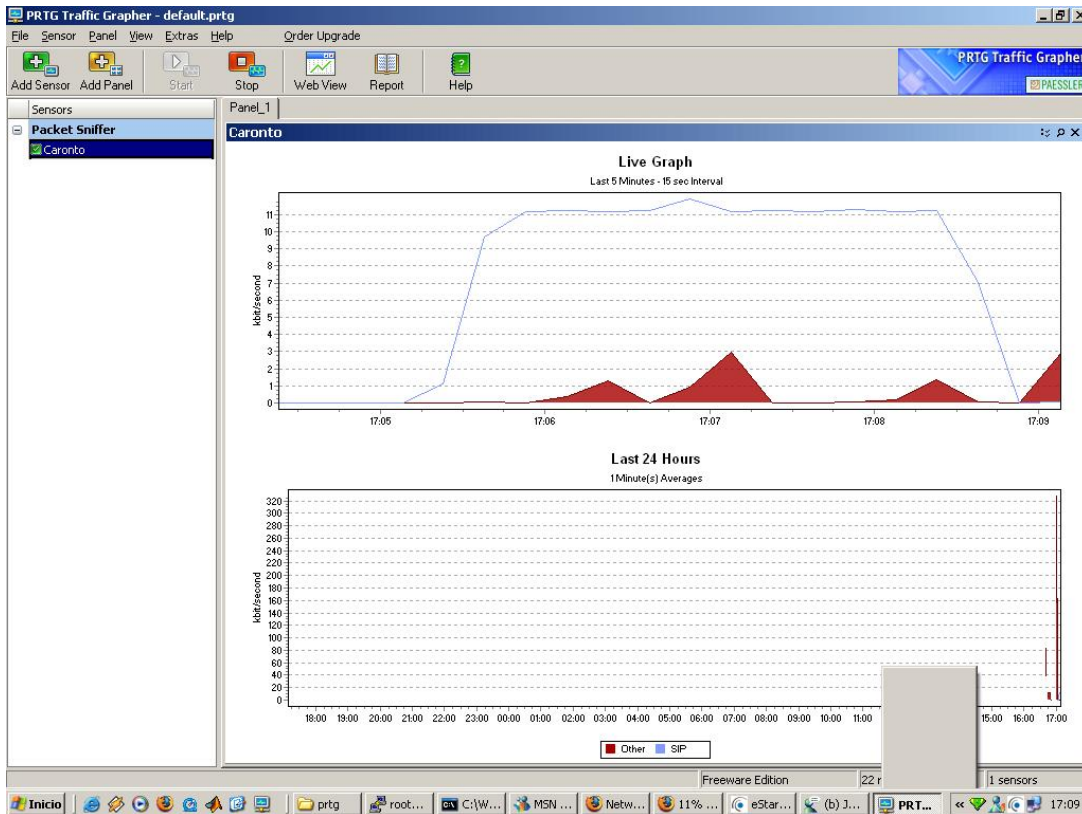


Figura 4.7. Tráfico generado por una llamada con codificación G 723

A diferencia de los resultados de las pruebas de tráfico, utilizando codificación G 723, en las pruebas de tráfico, utilizando codificación G 711, muestran que el ancho de banda utilizado con este tipo de codificación es de aproximadamente 80 Kbps, como muestra la siguiente gráfica:

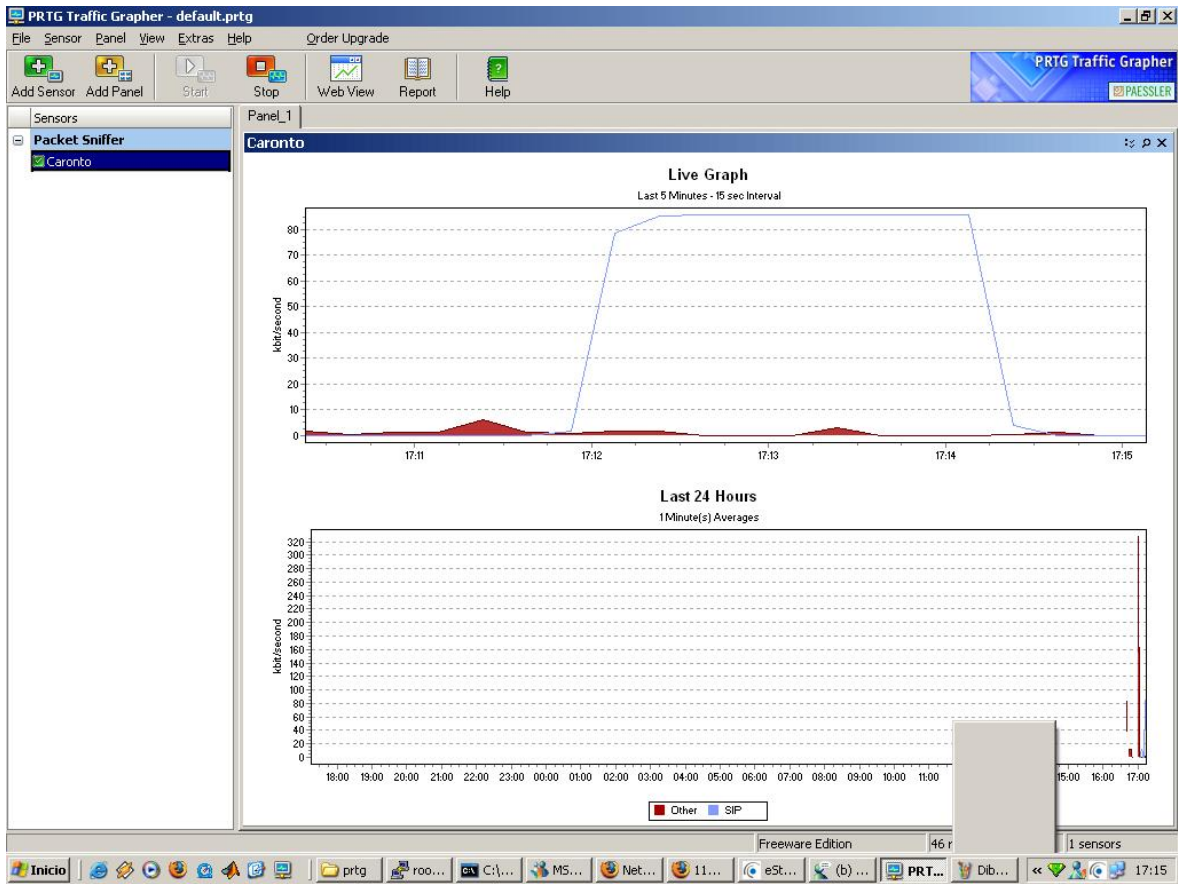


Figura 4.8. Tráfico generado por una llamada con codificación G 711

## CAPÍTULO V

### 5. ANÁLISIS ECONÓMICO

En este capítulo se hará un análisis económico del proyecto.

#### 5.1. INTRODUCCION

Como se aclaró en el segundo capítulo de este documento el software de nombre VOCAL con el que se implemento nuestro sistema emulador de la central telefónica IP, pertenece a la familia de software *open source*, es decir código abierto, cuya principal característica es la de ser de distribución libre (gratuita) y el único costo es el del tiempo que se dedique al estudio e implementación de soluciones a través de este tipo de herramientas.

En este capítulo se muestra la solución de telefonía IP para la FIE, tanto con la utilización del sistema VOCAL que se describe en este documento, así como también una solución a través de tecnología CISCO.

#### 5.2. SOLUCIÓN DE TELEFONÍA IP PARA LA FIE UTILIZANDO VOCAL

Para la realización de este proyecto en particular se dispuso del equipamiento ya existente en la red FIE como es: PCs, HUBs, SWITCHs, RUTEADORES, backbone, etc.

Parte fundamental del sistema emulador es el host donde se instalo el sistema VOCAL, que vendría a ser nuestra central telefónica, pero este item ya fue descrito en el capítulo

número tres de este documento, por lo que resulta redundante volver a describir esta parte del sistema.

Otra parte fundamental del sistema emulador son los teléfonos IP, en este proyecto los softphones; los mismos que pueden ser descargados del Internet en forma gratuita, por lo que no representan gasto alguno.

Tomando en cuenta todo lo anterior se puede decir que la implementación de telefonía IP, a través del sistema emulador no conlleva gastos ya que se dispuso de los elementos de red ya existentes de la FIE.

### **5.3. SOLUCIÓN DE TELEFONÍA IP PARA LA FIE UTILIZANDO TECNOLOGÍA CISCO**

El motivo por el cual se escogió la solución de Cisco, es que permite la utilización de elementos heredados, es decir elementos ya disponibles en la red FIE como teléfonos analógicos y equipos de fax tradicionales, entre otros, para implementación de nuevas redes de Telefonía IP.

La solución de Telefonía IP de Cisco, está dentro de la arquitectura AVVID (Architecture for Voice, Video and Integrated Data) e incluye, entre otros, los siguientes elementos principales: Media Convergence Server, Teléfonos IP, Gateways de Voz, Conmutadores Catalyst y Mensajería Unificada Cisco Unity. La siguiente figura muestra el esquema general de la solución de telefonía IP de CISCO:

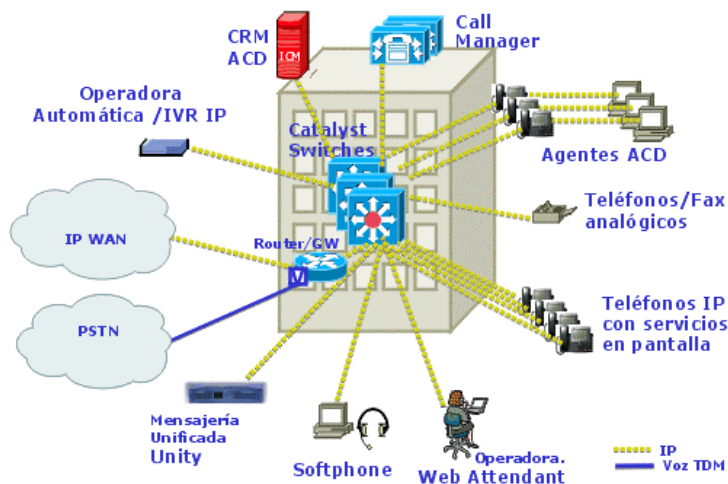
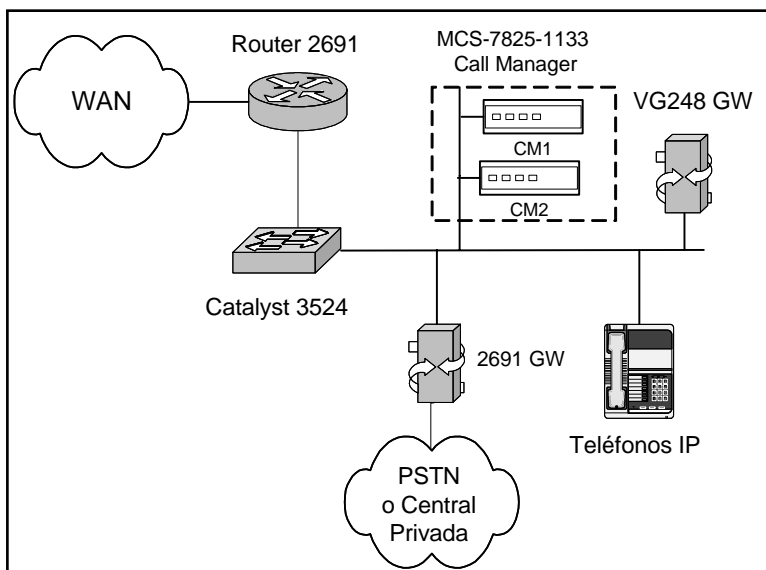


Figura 5.1. Solución de Telefonía IP de CISCO

CISCO tiene tres soluciones de telefonía IP de acuerdo al número de usuarios de la organización que quiera implementar este tipo de tecnología, y estas se dividen en: sitios grandes 3000 usuarios, sitios medianos 1000 usuarios y sitios pequeños 500 usuarios.

En este documento se presenta la solución para sitios pequeños, ya esta cumple de sobra con los requerimientos de la FIE; la misma presenta la siguiente topología:



**Figura 5.2. Topología de un sitio pequeño**

### 5.3.1. MEDIA CONVERGENCE SERVER O MCS

Es el sistema central de procesamiento de llamadas y elemento central de la arquitectura de Telefonía IP de Cisco. Para la solución de un sitio pequeño se utiliza el modelo MSC-7825-1133.

Algunas características del MCS-7825-1133 son:

- Procesador Intel Pentium III de 1.3 GHz.
- Memoria Cache de 256 KB.
- Memoria SDRAM de 1 GB/133 MHz.
- 10/100 TX PCI Fast Ethernet
- Disco Duro de 40 GB Ultra ATA/100 de 7,200 rpm.
- Floppy Disk de 1.44 MB Floppy
- CDROM Removable
- Dimensiones: 4.24 x 42.55 x 54.6 cm.
- Peso: 24.58 lb. (11.15 Kg.)
- Voltaje de entrada: 100 a 120VAC / 220 a 240VAC
- Corriente: 2A (110VAC) a 1A (240VAC)
- Frecuencia: 50 a 60 Hz.
- Potencia: 200 W
- Temperatura de Operación: 10° a 35° C

**Figura 5.3. Media Convergente Server MCS-7825-1133**

### 5.3.2. TELÉFONOS IP

Existen en la solución CISCO varios modelos de teléfonos que pueden ser utilizados para dar telefonía IP en la FIE, y estos pueden ser: Cisco IP Phone 7960, Cisco IP Phone 7940, Cisco IP Phone 7910 y 7910+SW; todos estos teléfonos IP soportan compresión de audio según G.711 y G.729.

#### 5.3.2.1. Cisco IP Phone 7960

El Teléfono IP 7960 de Cisco está diseñado principalmente para gerentes, administradores y ejecutivos. Opera basado en la tecnología del Cisco CallManager, basado en el estándar H.323, o en el Protocolo SIP (Session Initiated Protocol). Este modelo tiene como características principales:

- Seis líneas programables y cuatro teclas interactivas para cada línea.
- Dos puertos RJ-45 que soportan 10/100 Mbps para conexiones half o full duplex a dispositivos externos:
  - Un Conector se conecta al Conmutador Catalyst de forma que el teléfono puede alimentarse a través del conmutador.
  - El otro conecta el teléfono a otro dispositivo de red tal como una computadora.
- 24 Tonos de Timbrado.
- Compresión de Audio según G.711 y G.729a.
- Compatibilidad con H.323 y Microsoft NetMeeting.
- Identificación y categorización de mensajes y llamadas.
- Directorio Telefónico.
- Botones para configuración del volumen, tonos de timbrado, rellamada automática, etc.
- La configuración puede ser automática o manual.
- Puerto EIA/TIA-232 para diferentes opciones tales como expansión de línea y seguridades.
- Cumple con los Estándares IEEE 802.3x, IEEE 802.1D, IEEE 802.1.



**Figura 5.4. Teléfono IP 7960**

### **5.3.2.2. Cisco IP Phone 7940**

El Teléfono IP 7940 de Cisco es un Teléfono para usuarios con bajo y medio tráfico quienes necesitan un mínimo de teléfonos en el directorio. El teléfono opera bajo la tecnología CallManager de Cisco, el estándar H.323 o SIP.



**Figura 5.5. Teléfono IP 7940**

El modelo 7940 se diferencia del modelo 7960 únicamente en que el primero posee dos botones programables en vez de seis que posee el modelo 7960.

### **5.3.2.3. Cisco IP Phone 7910 y 7910+Sw**

El teléfono IP 7910 y el 7910+SW de Cisco son teléfonos básicos para áreas que requieren solo las características básicas de un teléfono. Operan bajo la tecnología CallManager de Cisco, y H.323.





**Figura 5.6. Teléfono IP 7910**

Tiene como principales características:

- Poseen una línea con cuatro botones dedicados para transferencia, configuración, y mantener una llamada
- El modelo 7910 tiene un conector estándar 10BASE-T RJ-45
- El modelo 7910+SW tiene dos interfaces 10/100BASE-T RJ-45
- Compresión de Audio según G.711 y G.729a
- Compatibilidad con H.323 y Microsoft NetMeeting
- Botones para configuración del volumen, tonos de timbrado, rellamada automática, etc.
- La configuración puede ser automática o manual.

#### **5.3.2.4. Softphone IP**

Es un teléfono IP software basado en interfaces estándares y abiertas (TAPI y JTAPI) que permite la integración entre la Telefonía IP y las aplicaciones de usuario corporativas; corre sobre un PC con Windows. Permite al usuario una movilidad y capacidad sin precedentes, al ser posible la integración con Microsoft NetMeeting, Outlook y otras aplicaciones, dotando así al terminal de todas las ventajas y capacidades que proporciona la Telefonía IP. Es posible integrarlo también con directorios LDAP ó agendas personales, de tal forma que hacer una llamada a un usuario, sea tan sencillo como encontrar su nombre en un directorio y arrastrar esa información hacia el Softphone.

Los requisitos para instalar y usar el Cisco IP Softphone en un PC son:

- Microsoft Windows 95, Windows 98 SE (Segunda Edición), Windows NT 4.0 con Service Pack 4 o mayor, o Windows 2000.
- Espacio Libre en Disco de 60 MB.
- Procesador Pentium de 166 MHz. o mayor.
- 64 MB de Memoria RAM



Figura 5.7. IP Softphone de Cisco

### 5.3.3. CONMUTADORES CATALYST

La familia de Conmutadores Catalyst son Conmutadores Ethernet cuya principal característica es la de proporcionar alimentación a través de los puertos y además QoS específicas para Telefonía IP. El modelo escogido para esta solución en particular es el Catalyst 3524.



Figura 5.8. Cisco Catalyst 3524

### 5.3.4. GATEWAYS DE VOZ

Los gateways de voz son dispositivos que proveen el interface entre la Telefonía IP y la telefonía analógica tradicional. El modelo VG248, es el adecuado para esta solución.

### 5.3.4.1. Cisco VG248

El gateway Cisco VG248 es una solución mixta habilitada por AVVID (Arquitectura integrada para la Voz, Video y Datos) que permite a las organizaciones utilizar dispositivos analógicos como teléfonos analógicos, máquinas del fax, módems, sistemas de correo de voz, dentro de un sistema de voz empresarial basado en Cisco CallManager; a través de una red IP. Este ofrece 48 puertos para líneas analógicas.



**Figura 5.9. Cisco VG248**

Los protocolos que soporta este gateway son:

- Skinny Client Control Protocol (SCCP)
- Real-Time Transport Protocol (RTP)
- Trivial File Transfer Protocol (TFTP)
- File Transfer Protocol (FTP)
- Hypertext Transfer Protocol (HTTP)
- Simple Network Management Protocol (SNMP)
- Telnet
- Dynamic Host Configuration Protocol (DHCP)
- Domain Name System (DNS)
- Cisco CallManager 3.3
- Codificador/decodificador según G.711 o G.729<sup>a</sup>

La utilización de este gateway se muestra en la siguiente figura:

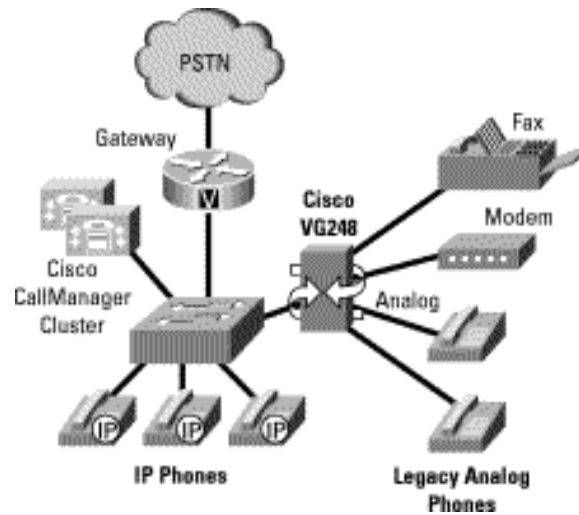


Figura 5.10. Utilización del Gateway Cisco VG248

### 5.3.5. CISCO 2691

El equipo CISCO 2691 actúa como router y como gateway; posee un potente CPU y está diseñado para ofrecer QoS que es requerida especialmente en oficinas pequeñas. Tiene una gran versatilidad y soporta interfaces de alta velocidad. Tiene dos puertos Ethernet 10/100 Base T , 3 zócalos para tarjetas WIC (WAN Integrated Card) y dos zócalos para Módulos de Integración Avanzados.



Figura 5.11. Cisco 2691

### 5.3.6. COSTOS DE EQUIPOS CISCO

El siguiente cuadro muestra el costo unitario de los equipos, así como también el costo total de toda la solución CISCO para telefonía IP, para este proyecto en particular.

<b>CANTIDAD</b>	<b>ITEM</b>	<b>COSTO UNITARIO</b>
1	MSC-7825-1133 (Instalado Cisco CallManager 3.3)	8400
1	Catalyst 3524	4750
1	Gateway VG248	4750
1	Gateway Cisco 2691	2670
1	Router Cisco 2691	3900
<b>TOTAL</b>		<b>24470</b>

**Tabla 5.1. Costos de equipos CISCO**

## CAPÍTULO VI

### 6. CONCLUSIONES Y RECOMENDACIONES

En este capítulo se indicarán las conclusiones y recomendaciones recopiladas a lo largo del proyecto de tesis.

#### 6.1. CONCLUSIONES Y RECOMENDACIONES

- En un inicio los sistemas de telecomunicaciones servían principalmente para llevar tráfico de voz y ocasionalmente tráfico de datos, es decir que sobre una plataforma que era construida para la transmisión de voz, se la adaptaba para transmitir datos, o dicho de otra forma la voz absorbía a los datos; en la actualidad en plataformas construidas para la transmisión de datos, son adaptadas para la transmisión de voz, o dicho de otra forma los datos absorben a la voz.
- La principal diferencia entre la telefonía convencional y la telefonía IP, radica en que la primera utiliza una red conmutada de circuitos y la segunda utiliza una red conmutada de paquetes. En la red conmutada de circuitos se establece o se crea un canal dedicado durante el tiempo que dura una llamada y se libera una vez que se termine la misma, a diferencia de una red conmutada de paquetes, en la cual la voz es paquetizada, estos paquetes son enviados individualmente y pueden tomar distintas rutas, motivo por el cual no utilizan todo el tiempo el canal de comunicación.
- La red de datos con la que cuenta la Facultad de Ingeniería Electrónica FIE, es independiente de la red de datos de la ESPE, por lo tanto esta posee una

administración propia y por consiguiente brinda las facilidades necesarias, tanto para el análisis, así como el desarrollo de nuevas aplicaciones, basadas en nacientes tecnologías.

- El auge de Voz sobre IP o VoIP, hace que cada vez más empresas a nivel mundial que están dedicadas al negocio de las telecomunicaciones quieran desarrollar soluciones, basadas en esta tecnología: tanto para empresas públicas, como para empresas privadas; esto a su vez obliga al Ingeniero en Telecomunicaciones a estar al tanto de lo que es VoIP, para así poder aportar con ideas claras en el momento que amerite brindar soluciones de este tipo.
- El potencial del sistema VOCAL va mucho más allá que el de ser utilizado como una central de Voz sobre IP, sino que además puede ser utilizado para la terminación de llamadas de larga distancia, de hecho muchas empresas de USA y México le dan esta aplicación; así como también de poder ser utilizado como un laboratorio de aplicaciones de VoIP.
- Al tratarse de una tecnología relativamente nueva en nuestro país, es apropiado comenzar estudiando los estándares internacionales que gobiernan la misma; esto nos permitirá tener un panorama más claro con de hasta donde puede esta tecnología dar una solución a un problema en particular.
- Linux es un sistema operativo que cuenta con todas las herramientas necesarias, para brindar estabilidad y un alto performance, en el desarrollo de todo tipo de soluciones, tanto en el campo de las telecomunicaciones y como en el de las redes de datos.