

## **CERTIFICADO**

Director: Ing. Gonzalo Olmedo

Codirector: Ing. Carlos Romero

## **CERTIFICAN**

Que el trabajo titulado “DISEÑO E IMPLEMENTACIÓN DE UN IVR BASADO EN CÓDIGO LIBRE ORIENTADO A TRONCALES SIP E INTERACTUANDO CONTRA BASES DE DATOS MySQL”, realizado por EDWIN JHAIR MENA JÁCOME, ha sido guiado y revisado periódicamente y cumple normas estatutarias establecidas por la ESPE, en el Reglamento de Estudiantes de la Escuela Politécnica del Ejército.

---

Ing. Gonzalo Olmedo  
DIRECTOR

---

Ing. Carlos Romero  
CODIRECTOR

## RESUMEN

El presente proyecto de tesis consiste en el estudio e implementación de un IVR (*INTERACTIVE VOICE RESPONSE*) basado en código libre, orientado al protocolo SIP y que tenga acceso a base de datos MySQL.

Para esto se realizará un estudio teórico de telefonía IP con sus respectivas características, funcionalidades y específicamente el protocolo SIP. Además se estudiarán los sistemas IVR y todo lo referente a sus aplicaciones, funcionamiento y tecnologías anexas que soporta. Finalmente un estudio e introducción a las bases de datos de una forma general.

El sistema consiste en un servidor que contiene la central telefónica IP y la base de datos MySQL. Físicamente este servidor estará conectado directamente a la PSTN a través de un módulo FXO y tendrá salida hacia el internet por medio de una IP pública que nos permitirá tener accesos remotos.

Para la implementación utilizamos el sistema ecuatoriano basado en software libre "Elastix". Este sistema nos permitirá tener una administración web de la central. La central telefónica IP tendrá como equipos finales teléfonos IP o *softphones* y un teléfono analógico conectado a la central por medio de un módulo FXS.

El sistema IVR permitirá tomar información de la base de datos y transmitirla al usuario de forma hablada. Además permitirá tomar información del usuario por DTMF y almacenarlos en la respectiva base de datos.

## DEDICATORIA

Esta tesis está dedicada a mis padres, hermanos,  
abuelitos y mi mami lila por ser mi inspiración,  
mi apoyo y por el amor que me dieron  
durante mi carrera.

## AGRADECIMIENTO

Inicialmente quiero agradecer a Dios por ser mi guía y por darme las bendiciones que me han permitido llegar a este punto de mi vida.

Quiero agradecer a mis padres por su esfuerzo diario y dedicación que me ha permitido tener una excelente educación y que será la base para mi futuro. Además agradecer a mis hermanos por su preocupación y apoyo que siempre me han demostrado y más aún durante mi carrera.

Agradecer a mis tíos, abuelitos, primos y mi ale por hacer mis problemas suyos y apoyarme siempre para solucionarlos de la mejor manera. Agradecer a mis amigos “los mijines” por su apoyo incondicional y por estar a mi lado durante todo este tiempo brindándome alegría y experiencias inolvidables.

Por último quisiera agradecer al Ing. Gonzalo Olmedo por el apoyo y guía durante la carrera. Además agradezco a la empresa PaloSanto Solutions Quito por las oportunidades que me han brindado.

## ÍNDICE DE CONTENIDO

ÍNDICE DE FIGURAS .....	8
ÍNDICE DE TABLAS .....	10
ÍNDICE DE FOTOGRAFÍAS.....	10
GLOSARIO.....	11
INTRODUCCIÓN .....	13
CAPÍTULO I .....	15
1 MARCO TEÓRICO .....	15
1.1 VOZ SOBRE IP .....	15
1.1.1 Definición.....	15
1.1.2 Aplicaciones.....	16
1.1.3 Protocolo de Señalización SIP.....	17
1.1.4 <i>Codecs</i> .....	25
1.2 IVR ( <i>INTERACTIVE VOICE RESPONSE</i> ) .....	27
1.2.1 Definición.....	27
1.2.2 Servicios y Aplicaciones .....	27
1.2.3 Funcionamiento .....	32
1.2.4 Tecnologías anexas a los sistemas IVR .....	34
1.3 BASE DE DATOS MySQL .....	43
1.3.1 Definición.....	43
1.3.2 Estructura de una Base de Datos General .....	45
1.3.3 Modelos de una Base de Datos General .....	46
CAPÍTULO II .....	52
2 ASTERISK COMO PBX .....	52
2.1 ASTERISK COMO PBX .....	52
2.1.1 Definición.....	52
2.1.2 Estructura .....	53
2.1.3 Directorios .....	57

2.1.4	Funcionamiento de Asterisk .....	59
2.1.5	Estructura Modular .....	59
2.1.6	<i>Zaptel</i> , ahora DAHDI .....	60
2.1.7	Plan de Mercado.....	62
2.1.8	Asterisk AGI.....	63
2.1.9	Ventajas.....	64
2.2	SOLUCION ELASTIX.....	65
2.3	HERRAMIENTA GRÁFICA FREE PBX.....	68
2.3.1	Propiedades .....	69
2.3.2	Fundamentos de FreePBX .....	69
2.3.3	Organización de Archivos.....	72
2.3.4	Alcance.....	73
2.4	CONFIGURACIÓN DE ASTERISK CON FREEPBX.....	84
2.4.1	Extensiones SIP .....	84
2.4.2	Configuración de un <i>sofphone</i> (X-Lite) como elemento final del PBX	92
2.4.3	Configuración del teléfono Siemens <i>OptiPoint</i> 150 S como elemento final del PBX.....	98
CAPÍTULO III .....		108
3	DISEÑO DEL IVR .....	108
3.1	ANÁLISIS DE PROBLEMAS COMUNES EN TELEFONIA DENTRO DE LOS CLIENTES DE HIGHTELECOM.....	108
3.1.1	Encuestas.....	109
3.2	DEFINICIÓN DEL PROBLEMA A RESOLVER POR MEDIO DEL IVR..	117
3.2.1	Problema .....	117
3.2.2	Alcances y Limitaciones del Proyecto.....	118
3.3	DISEÑO DE UN IVR GENERAL .....	120
3.3.1	Recomendaciones de diseño .....	120
3.3.2	Fases de Diseño.....	125
3.3.3	Análisis del diagrama de bloques y diagrama de flujo .....	126
3.4	BASES DE DATOS A UTILIZAR EN BASE EL PROBLEMA .....	132
CAPÍTULO IV .....		134
4	CONFIGURACIÓN DEL IVR.....	134
4.1	CONFIGURACIÓN DE ASTERISK PARA EL IVR MODELADO.....	134

4.1.1	CONFIGURACIÓN DEL IVR CON ELASTIX.....	135
4.1.2	INTERACCIÓN DEL IVR CON LA BASE DE DATOS MySQL .....	151
CAPÍTULO V .....		159
5	RESULTADOS.....	159
5.1	ANÁLISIS DEL IVR EN EL MERCADO.....	159
5.2	SOLUCIONES DE TELEFONÍA BASADOS EN SOFTWARE LIBRE VS SOLUCIONES PROPIETARIAS .....	161
5.3	ANÁLISIS DE LA HERRAMIENTA FREEPBX, SUS VENTAJAS Y DESVENTAJAS FRENTE A UNA CONFIGURACIÓN POR CONSOLA. ....	163
5.4	CONCLUSIONES.....	165
5.5	RECOMENDACIONES .....	167
ANEXOS .....		170
	Anexo 1: Contextos en el archivo <i>/etc/Asterisk/extensions_custom.conf</i> .....	170
	Anexo 2: Script <i>agi3334.php</i> .....	171
	Anexo 3: Script <i>agi4445.php</i> .....	172
	Anexo 4: Script <i>agi5556.php</i> .....	173
	Anexo 5: Script <i>agi6667.php</i> .....	174
	Anexo 6: Script <i>agi7778.php</i> .....	175
	Anexo 7: Comandos AGI .....	177
REFERENCIAS.....		179

## ÍNDICE DE FIGURAS

Figura. I.1. <i>Stack</i> de Protocolos .....	18
Figura. I.2. Configuración “Trapezoide” SIP .....	21
Figura. I.3. Estructura del mensaje.....	23
Figura. I. 4. Aplicaciones del IVR basado en un estudio en México .....	29
Figura. I.5. IVR como operadora automática.....	30
Figura. I.6. IVR como gestor de base de datos .....	31
Figura. I. 7. FXS/FXO sin PBX .....	41
Figura. I.8. FXS/FXO con PBX .....	41
Figura. I.9. Puerta de enlace FXO .....	42
Figura. I.10. Comparación de una Base de Datos y un Sistema de Archivos .....	43
Figura. I.11. Estructura de una Base de Datos.....	45
Figura. I.12. Estructura de Base Jerárquica .....	48
Figura. I.13. Estructura de Base de Datos de Red .....	48
Figura. I.14. Ilustración de tipos de relación: El ERD Pata de Gallo.....	50
Figura. I.15. Desarrollo de Modelos de Datos .....	51
Figura. I.16. Principales subsistemas de Asterisk .....	53
Figura. II.17. Interacción de Asterisk con <i>Zaptel</i> .....	61
Figura. II.18. Esquema general de los componentes de Elastix.....	66
Figura. II.19. Interfaz gráfica de Elastix .....	67
Figura. II.20. FreePBX no embebido en Elastix.....	68
Figura. II.21. Interfaz gráfica de Elastix .....	70
Figura. II.22. Ejemplo de Interacción entre módulos de FreePBX.....	73
Figura. II.23. Página principal de FreePBX .....	74
Figura. II.24. Menú de herramientas de FreePBX .....	80
Figura. II.25. Parámetros a configurar para un dispositivo SIP .....	86
Figura. II.26. Interfaz gráfica <i>softphone</i> X-Lite.....	93
Figura. II.27. Ingreso a configuración de extensión SIP en X-Lite.....	94
Figura. II.28. Ventana de configuración en X-Lite .....	94
Figura. II.29. Ventana de propiedades para extensión SIP en X-Lite.....	96
Figura. II.30. <i>Softphone</i> X-Lite registrado.....	97
Figura. II.31. Partes del teléfono OPTIPOINT 150 S.....	99
Figura. II.32. Página principal del teléfono <i>OptiPoint 150 S</i> vía web .....	102
Figura. II.33. Cambio de Idioma del teléfono <i>OptiPoint 150 S</i> .....	103
Figura. II.34. Configuración de red del teléfono <i>OptiPoint 150 S</i> .....	103
Figura. II.35. Configuración de red del teléfono <i>OptiPoint 150 S</i> .....	104
Figura. II.36. Configuración de extensión SIP del teléfono <i>OptiPoint 150 S</i> .....	106
Figura. III.37. Gráfico en barras del resultado de la encuesta .....	111
Figura. III.38. Gráfico en pastel del resultado de la encuesta .....	111
Figura. III.39. Diagrama de bloque del IVR propuesto .....	126



Figura. III.40. Diagrama de flujo del IVR propuesto.....	129
Figura. IV.41. Acceso inicial a la administración web.....	134
Figura. IV.42. Interfaz web para añadir extensión SIP .....	137
Figura. IV.43. Visualización de extensiones configuradas .....	138
Figura. IV.44. Interfaz web para añadir grabaciones.....	138
Figura. IV.45. Interfaz web para configurar el IVR.....	140
Figura. IV.46. Configuración del IVR secundario 1.....	142
Figura. IV.47. Interface web para detección de hardware .....	145
Figura. IV.48. Interface web para configuración de troncales.....	146
Figura. IV.49. Interface web para configuración de rutas salientes .....	147
Figura. IV.50. Interface web para configuración de rutas entrantes .....	149
Figura. IV.51. Interface web para configuración de extensión ZAP.....	150
Figura. IV.52. Herramienta putty para acceso ssh .....	152
Figura. IV.53. Creación de base de datos en la consola MySQL .....	153
Figura. IV.54. Configuración base de datos usuario en la consola MySQL.....	154
Figura. IV.55. Configuración base de datos promo en la consola MySQL .....	154
Figura. IV.57. Interface web para configuración de otros destinos.....	156

## ÍNDICE DE TABLAS

Tabla. I.1. Diferencias entre protocolos SIP y H323.....	24
Tabla. I.2. <i>Codecs</i> para VoIP .....	25
Tabla. II.3. Interfaces No- <i>Zaptel</i> .....	55
Tabla. II.4. <i>Codecs</i> soportados por Asterisk.....	56
Tabla. II.5. Formatos de Archivos de audio .....	57
Tabla. II.6. Directorios de Asterisk.....	57
Tabla. II.7. Comandos para el programa amportal .....	59
Tabla. II.8. Directorios que maneja FreePBX .....	72
Tabla. III.9. Datos de la encuesta-Limitantes de los clientes HIGHTELECOM....	110
Tabla. III.10. Estructura de la base de datos usuario .....	132
Tabla. III.11. Estructura de la base de datos promociones .....	133
Tabla. IV.12. Tabla de grabaciones para el IVR.....	139
Tabla. V.13. Encuesta a clientes sobre preferencias en sistemas IVR .....	160

## ÍNDICE DE FOTOGRAFÍAS

Fotografía. IV.1. Tarjeta PCI de 4 puertos para módulos FXS/FXO.....	143
Fotografía. IV.2. Módulos FXS y FXO .....	143
Fotografía. IV.3. Tarjeta PCI con un módulo FXS y un FXO .....	144

## GLOSARIO

### [A]

- AGI (*Asterisk Gateway Interface*)  
ANI (Número automático de identificación)  
ASR (*Automatic Speech Recognition*)

### [B]

- BRI (Acceso básico ISDN)

### [C]

- CID (*Caller ID*)  
CLI (*Command Line Interface*)  
CTI (*Computer Telephone Integration*)

### [D]

- DBMS (Sistema de Administración de Base de Datos)  
DISA (*Direct Inward System Access*)  
DNS (*Domain Name Server*)

### [E]

- ERD (Diagrama Entidad Relación)

### [F]

- FOP (*Flash Operator Panel*)  
FXO (*Foreign Exchange Office*)  
FXS (*Foreign Exchange Station*)

### [G]

- GPL (*General Public License*)  
GSM (*Global System of Mobile communications*)  
GUI (*Graphic User Interface*)

### [I]

- IAX (*Inter-Asterisk Exchanged*)  
ISP (*Internet Service Provider*)  
ITU (*International Telecommunication Union*)  
IVR (*Interactive Voice Response*)

### [J]

JTAPI (*Java Telephony Application Programming Interface*)

**[M]**

MIPS (*Millones de Instrucciones Por Segundo*)

MMUSIC (*Multiparty Multimedia Session Control*)

**[N]**

NAPTR (*Naming Authority Pointer*)

**[P]**

PBX (*Private Branch Exchange*)

PRI (*Acceso Primario ISDN*)

PSTN (*Red de Telefónica Pública*)

**[R]**

RPE (*Regular Pulse Excited*)

RTP (*Real-time Transport Protocol*)

**[S]**

SGBDR (*Sistema de Gestión de Bases de Datos Relacionales*)

SIP (*Session Initiation Protocol*)

SOHO (*Small Office Home Office*)

SQL (*Structured Query Language*)

STT (*Speech to Text*)

STUN (*Simple Transversal of UDP over NATs*)

**[T]**

TDM (*Multiplexador por División de Tiempo*)

TTS (*Text To Speech*)

**[U]**

UAC (*User Agent Client*)

UAS (*User Agent Server*)

**[V]**

VRU (*Unidad de Respuesta Vocal*)

VSP (*Voice Service Provider*)

**[Z]**

ZAP (*Líneas analógicas*)

## INTRODUCCIÓN

En los últimos años todo sistema analógico ha sido sometido a grandes cambios con el fin de llevarlo a un sistema completamente digital, mejorando así la calidad y buscando la convergencia de servicios.

El sistema de voz ha evolucionando constantemente en todo lo referente a sistemas telefónicos digitales tanto en servicios como aplicaciones. Brindando así una mejor interfaz con el usuario, mejores prestaciones y buscando siempre ofrecer una mejor calidad en la transmisión de la voz. Esta evolución va de la mano con el crecimiento de la tecnología aplicada a redes de datos y ordenadores.

Una de las prestaciones y aplicaciones que presentan estos sistemas telefónicos es el IVR (*Interactive Voice Response*) que permite direccionar las llamadas entrantes de una forma ordenada y amigable para el usuario. Además le permite al usuario tener acceso a bases de datos en busca de información que se defina como accesible.

El IVR ha ido evolucionado y con ello las funciones disponibles, para esto, ha anexado tecnologías a su funcionamiento como: TTS (*Text To Speech*), ASR (*Automatic Speech Recognition*), entre otras.

El desarrollo de este proyecto busca dar un nuevo servicio al sistema “*NetCyclon*”, incrementado así sus prestaciones y haciéndolo mucho más completo. El servicio de telefonía busca dar a los clientes de la empresa HIGHTELECOM una

alternativa a las centrales telefónicas convencionales con un menor presupuesto y mayor simplicidad.

En este proyecto, se hará un breve estudio teórico de los conceptos ligados a la telefonía IP, al protocolo SIP, al IVR y bases de datos MySQL con sus más importantes características y definiciones.

En base a los estudios teóricos, se procederá a hacer el diseño e implementación de un sistema IVR basado en las necesidades comunes de los clientes de la empresa HIGHTELECOM. Para ello se realizará encuestas a los clientes y se buscará satisfacer todas las necesidades que ellos tengan en el aspecto de telefonía.

# CAPÍTULO I

## 1 MARCO TEÓRICO

### 1.1 VOZ SOBRE IP

#### 1.1.1 Definición

Llamamos “Voz sobre IP” (*VoIP, Voice over IP*) a la tecnología que permite la transmisión de la voz a través de redes IP en forma de paquetes de datos.

Se utilizan protocolos para llevar las señales de voz sobre la red IP, estos estándares son conocidos como protocolos de Voz sobre IP o protocolos IP. Además utiliza *codecs* para la codificación de la señal y el encapsulamiento en estos protocolos.

Inicialmente la utilización de VoIP tenía problemas de baja calidad en la voz y problemas de seguridad. Hoy en día el desarrollo de nuevas tecnologías entre ellas *codecs* ha permitido que la comunicación IP ofrezca mayor calidad y seguridad, despertando en usuarios la curiosidad por aprovechar y conocer las ventajas y prestaciones que ofrece esta comunicación.

Las principales ventajas de voz sobre IP son: la integración de servicios como voz, video y datos en una misma estructura, y sobre todo, la reducción de costos en términos de telefonía y comunicación.

La diferencia entre VoIP y telefonía IP es principalmente la función que brindan, es decir, VoIP permite la transmisión de voz sobre el protocolo IP a través de protocolos, dispositivos, etc. mientras que telefonía IP son las nuevas funcionalidades que se brinda a la telefonía gracias al tratamiento de voz como datos.

### **1.1.2 Aplicaciones**

VoIP facilita las tareas en comparación a las redes telefónicas comunes. La aplicación primordial que caracteriza a la telefonía IP es:

- ✓ Las llamadas telefónicas locales pueden ser automáticamente enrutadas a un teléfono VoIP, sin importar dónde se esté conectado a la red. Uno podría en un viaje llevar consigo un teléfono VoIP y conectado a Internet, se podría recibir llamadas, es decir, los usuarios de VoIP pueden viajar a cualquier lugar en el mundo y seguir haciendo y recibiendo llamadas sin ningún costo y con una muy alta calidad.
- ✓ Los usuarios que cuentan con el servicio de mensajería basada en servicios de VoIP pueden viajar a cualquier lugar del mundo y hacer y recibir mensajes.
- ✓ Los equipos terminales de la VoIP o teléfonos VoIP pueden integrarse con otros servicios disponibles en Internet como videoconferencias, intercambio de datos, mensajes con otros, etc.



### 1.1.3 Protocolo de Señalización SIP

Los protocolos de señalización son los encargados del establecimiento y gestión de mensajes de estado entre los puntos extremos que participan en una llamada. Estos protocolos indicarán el paso de los flujos de voz, encapsulados en paquetes RTP/RTCP por la red hasta llegar a su destino.

El protocolo SIP (*Session Initiation Protocol*) o Protocolo de Inicio de Sesiones [SIP2008] es un protocolo de señalización que trabaja en el nivel de aplicación para conferencia, telefonía, presencia, notificación de eventos y mensajería instantánea a través de Internet. Fue desarrollado inicialmente en el grupo de trabajo IETF MMUSIC (*Multiparty Multimedia Session Control*) y, a partir de Septiembre de 1999, pasó al grupo de trabajo IETF SIP.

SIP nace en el año 1996 como respuesta a las dificultades del protocolo H.323 correspondiente a la ITU. Se convierte en un estándar en febrero de 1999 en la [RFC2543]. Se crea el grupo SIP y en junio del 2002 se publica la [RFC3261] que reemplaza al estándar anterior con ciertas modificaciones y mejoras.

- **Arquitectura**

SIP al igual que el protocolo H.323 hace uso de RTP<sup>1</sup> (*Real-time Transport Protocol*) y UDP<sup>2</sup> (*User Datagram Protocol*) para hacer transferencia de voz, usando una única petición para enviar la

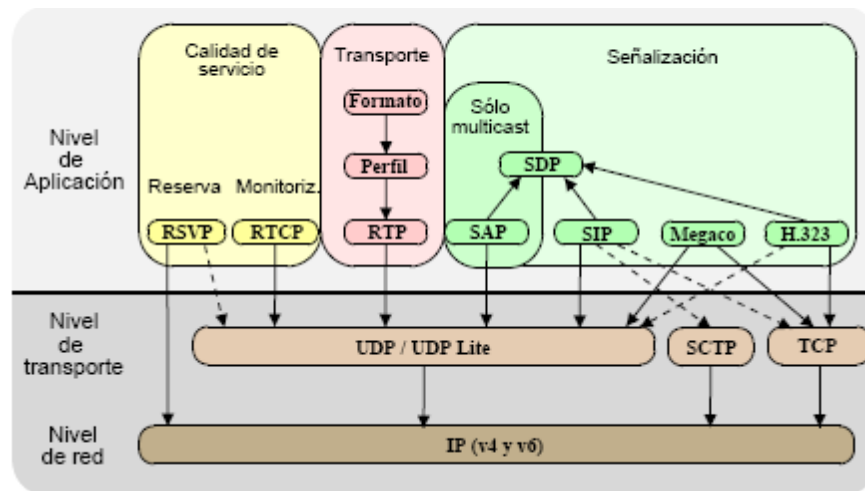
---

<sup>1</sup> RTP Protocolo de Transporte de Tiempo real es un protocolo que trabaja en el nivel de sesión y es utilizado para la transmisión de información a tiempo real como audio y video. Los puertos utilizados se encuentran aproximadamente del 10000 al 20000.

<sup>2</sup> UDP protocolo que trabaja en el nivel de transporte y se basa en el intercambio de datagramas.

información que se requiere, siendo así mucho más eficiente y rápido que otros protocolos.

Además SIP se complementa, entre otros, con el protocolo SDP (*Session Description Protocol*), que describe el contenido multimedia de la sesión, por ejemplo qué direcciones IP, puertos y *codecs* se usarán durante la comunicación antes de empezar el flujo RTP.



**Figura. I.1. Stack de Protocolos**

Fuente: [IRIS2003]

El protocolo SIP adopta el modelo cliente-servidor y es transaccional. Además codifica sus mensajes en modo texto que hace mucho más fácil su interpretación, no así el protocolo H.323 que usa codificación binaria incrementando su complejidad.

Los servidores, por defecto, utilizan el puerto 5060 en TCP y UDP para recibir las peticiones de los clientes SIP.

Se ubica en el nivel de aplicación para establecimiento y gestión de sesiones con múltiples usuarios. La sintaxis de sus operaciones se basa en protocolos utilizados en páginas Web (HTTP) y en protocolos de distribución de e-mails (SMTP). Esta similitud es natural ya que SIP fue diseñado para que la telefonía se vuelva un servicio en la Internet. La limitante del protocolo SIP es que no puede atravesar NATs (Traductores de direcciones de red), esto se debe a que la señalización y los flujos RTP son transmitidos por puertos diferentes, es decir, RTP utiliza puertos aleatorios haciendo difícil la traducción de direcciones IP. Para este problema existen otras traducciones como los NAT Transversal y los servidores STUN. A pesar de esto, en la actualidad SIP es uno de los protocolos de VoIP más utilizados.

- **Componentes de Red**

Una red basada en señalización SIP tiene al menos cinco tipos de componentes de red o llamadas también entidades lógicas. Cada entidad tiene una función determinada y participa en las conversaciones SIP como cliente (inicia solicitudes), como servidor (responde a solicitudes) o de ambas formas.

Los componentes de red principales del protocolo se dividen en dos elementos principales que son:

- ✓ Agentes Usuarios (*UA-User Agent*)
- ✓ Servidores SIP

## Agentes Usuarios

Estos se dividen en dos tipos:

- ✓ *User Agent Client (UAC)*.- es una entidad lógica que genera peticiones SIP y recibe respuestas a esas peticiones.
- ✓ *User Agent Server (UAS)*.- genera respuestas a las peticiones SIP.

Ambos se encuentran en todos los agentes de usuario, así permiten la comunicación entre diferentes agentes de usuario mediante comunicaciones de tipo cliente-servidor.

## Servidores SIP

Estos se dividen en tres tipos:

- ✓ *Servidor Proxy*.- busca que la solicitud del cliente se remita a la entidad más cercana para el usuario de destino. Además verifica ciertas políticas como comprobar si el usuario está autorizado a efectuar una llamada, etc.

A continuación se muestra la configuración básica para el establecimiento de una llamada:

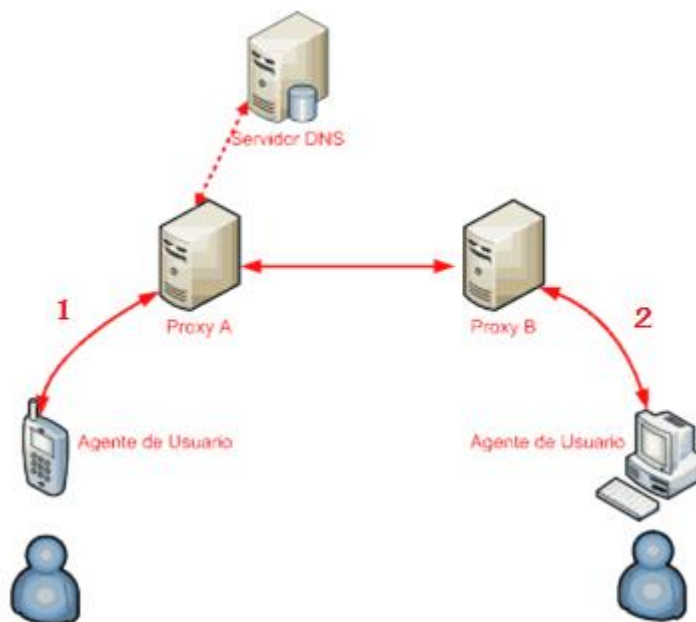


Figura. I.2. Configuración "Trapezoide" SIP

Fuente: [EBSIP]

El Proxy 1 remite la solicitud al proxy del dominio de la persona llamada (dominio B), que es el proxy 2. Y el Proxy 2 remite la llamada al Agente de Usuario de la persona llamada, UA 2. Para conseguir realizar dicho proceso de llamada, el proxy 1 necesita determinar cuál es el servidor SIP del dominio B. Para conseguir saberlo emplea procedimientos DNS, obteniendo registros SRV<sup>3</sup> y NAPTR<sup>4</sup> que apuntan al servidor SIP del dominio B basado en [RFC3263].

<sup>3</sup> **Registro SRV** Es un registro DNS donde se especifica información de servicios disponibles. La definición oficial se encuentra en la RFC2782.

<sup>4</sup> **Registro NAPTR** (*Naming Authority Pointer*) especifica las diferentes formas de contactar a un usuario final o propietario de un número. Los registros NAPTR nos informan de su Web, su e-mail, su teléfono móvil, etc.

- ✓ Servidor de Redirección: reduce la carga del servidor Proxy y mejora la señalización basado en [RFC3261]. Este servidor no pasa las solicitudes a otros servidores.
- ✓ Servidor de Registro: atiende las solicitudes *SIP REGISTER* y actualiza la base de datos de localización, con la información del usuario que se registre. No es obligatorio.

- **Mensajes Básicos SIP**

El protocolo SIP utiliza métodos hacia el servidor similares al protocolo HTTP, el cual responde de igual manera con mensajes tipo HTTP. Los métodos definidos en la especificación SIP básica son:

- ✓ *INVITE*: para iniciar una sesión desde un UA (*User Agent*).
- ✓ *ACK*: para acusar recibo de una solicitud *INVITE*.
- ✓ *BYE*: termina la sesión de quienes estén en llamada.
- ✓ *OPTIONS*: pregunta a un servidor sobre sus capacidades, sin establecer la llamada.
- ✓ *CANCEL*: anula una solicitud que esté pendiente. Por ejemplo cuando un proxy ha reenviado una solicitud a varios proxys y alguno de ellos contesta satisfactoriamente, la parte UA que inició la consulta puede querer cancelarla.
- ✓ *REGISTER*: para registrar una dirección en un servidor SIP, de modo que a partir de entonces sepa a donde hay que

enrutar las solicitudes dirigidas hacia la entidad registrada. Un UA puede enviar una solicitud *REGISTER* a un determinado proxy, pero también podría enviarla a una dirección *multicast*.

- **Funcionamiento**

El cliente realiza peticiones (*requests*) que el servidor atiende y genera una o más respuestas (dependiendo de la petición).

La comunicación se define a través de dos tipos de mensajes, que son:

- ✓ Las solicitudes (métodos)
- ✓ Las respuestas (códigos de estado)

La estructura de los códigos de estado o respuestas constan de una línea inicial seguida de uno o más campos de cabecera (*headers*), una línea vacía que indica el final de las cabeceras, y por último, el cuerpo del mensaje que es opcional.

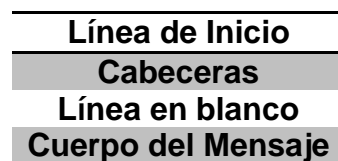


Figura. I.3. Estructura del mensaje

Por ejemplo para iniciar una sesión el cliente realiza una petición con el método *INVITE* en donde indica con qué usuario (o recurso) quiere establecer la sesión. El servidor responde ya sea rechazando o aceptado esa petición en una serie de respuestas. Las respuestas

llevan un código de estado que brindan información acerca de si las peticiones fueron resueltas con éxito o si se produjo un error.

- **Ventajas y Desventajas frente al Protocolo H323**

Uno de los principales protocolos en VoIP junto con SIP es el protocolo H323. El protocolo H323 es un conjunto de estándares pertenecientes a la ITU-T (*International Telecommunication Union*), los cuales definen protocolos para proveer sesiones de comunicación audiovisual sobre una red de computadores. H323 es un protocolo relativamente viejo y está siendo reemplazado por SIP; inicialmente fue creado para proveer un mecanismo de transporte de aplicaciones multimedia en redes LAN.

Las principales diferencias entre estos dos protocolos se muestran en la siguiente tabla:

**Tabla. I.1. Diferencias entre protocolos SIP y H323**

Fuente: [IRIS2003]

	H.323	SIP
Codificación	Binaria (ASN.1)	Textual (SigComp)
Formatos	Series G.XXX y H.XXX, MPEG, GSM	Tipos MIME – IANA
Ampliabilidad	Campos reservados	Métodos, cabeceras
Autenticación	H.235 (puede usar TLS)	Análogo a HTTP
Localización	Gatekeeper (puede usar DNS)	DNS
Transporte	TCP, UDP	TCP, UDP, SCTP, DCCP, etc.



Otras diferencias importantes que podemos nombrar son:

- ✓ Las extensiones estándares de SIP permiten realizar funciones específicas mientras que en H.323 deben ser implementados en elementos no estándares.
- ✓ La señalización simple de SIP da un mínimo retardo mientras que en el peor de los casos H.323 podría incurrir en retrasos de 7 a 8 segundos.
- ✓ SIP mezcla información (media) en una sola sesión mientras que esto es imposible en H.323.

#### 1.1.4 Codecs

Los *codecs* (COdificador – DECodificador) son modelos matemáticos que codifican una señal (proceso de transmisión) y la recuperan o descifran del mismo modo para la manipulación en un formato más apropiado. La mayor parte de *codecs* provoca pérdidas de información para conseguir un tamaño lo más pequeño posible del archivo destino. Estos *codecs* son importantes ya que para alcanzar un tipo de calidad, enviamos únicamente una cantidad de datos suficiente, disminuyendo así la carga de los paquetes de voz.

Los *codecs* más importantes se los menciona en la siguiente tabla:

**Tabla. I.2. Codecs para VoIP**  
Fuente: [AST2007]

Codec	Data bitrate (Kbps)	License required?
G.711	64 Kbps	No
G.726	16, 24, 32, or 40 Kbps	No
G.729A	8 Kbps	Yes (no for passthrough)
GSM	13 Kbps	No
iLBC	13.3 Kbps (30-ms frames) or 15.2 Kbps (20-ms frames)	No
Speex	Variable (between 2.15 and 22.4 Kbps)	No

- **G.711**

Es un estándar desarrollado por la ITU-T para la compresión de audio. Este estándar permite representar señales de audio con frecuencias de la voz humana, mediante muestras comprimidas de una señal de audio digital con una tasa de muestreo de 8000 muestras por segundo, este es el *codec* nativo de la PSTN. Su tasa de transmisión es de 64 [kbps] que usa compansión (comprensión - expansión) [ATT2005], un tipo de compresión que dependiendo de la zona usa la ley  $\mu$  en Norteamérica y la ley A en el resto del mundo.

- **G.729**

Es también un estándar desarrollado por la ITU-T para la compresión de datos de audio para voz que comprime audio de voz en trozos de 10 milisegundos. Este estándar opera a una tasa de 8 [kbps] y ofrece una gran calidad de audio debido al uso del la Predicción Lineal de Código Algebraico de Estructura Conjugada (CSACELP). Es utilizado en su mayoría en aplicaciones de VoIP o telefonía IP por sus bajos requerimientos de ancho de banda pero requiere una gran capacidad computacional, que puede ser un gran inconveniente en sistemas convencionales. También es muy común G.729A el cual es compatible con G.729, pero requiere menos cómputo.

Si bien es un *codec* que requiere licencia, existen implementaciones de uso gratuito.

- **GSM**

El *codec* GSM (*Global System of Mobile communications*) proviene del conocido sistema de comunicaciones móviles y utiliza un *codec*

RPE (*Regular Pulse Excited*). Opera a una tasa de 13 [kbps] y proporciona una buena calidad de voz con gran simpleza de proceso para aplicaciones de tiempo real, aunque no tan buena como la tasa ligeramente superior del *codec* G.728. Su principal ventaja sobre otros *codecs* de baja tasa es su relativa simplicidad [TUB2006].

## 1.2 IVR (*INTERACTIVE VOICE RESPONSE*)

### 1.2.1 Definición

Los IVR (Respuesta de Voz interactiva), conocido también como VRU (Unidad de Respuesta Vocal), son una poderosa plataforma de desarrollo de aplicaciones telefónicas, que permite diseñar, integrar, implementar y administrar sistemas de respuesta interactiva, permitiendo enrutar las llamadas al usuario final en muy corto tiempo. Los sistemas IVR han tenido un gran éxito debido a que su dispositivo de entrada de información es un teléfono convencional y pueden ser atendidas miles de llamadas a la vez. Los IVRs se basan en grabaciones de mensajes y respuestas estáticas y/o dinámicas de acuerdo a los requerimientos del usuario y alcances del propio sistema.

Un sistema IVR bien diseñado requiere solamente que el usuario tenga un teléfono y una línea convencional y debe ser capaz de satisfacer tres requerimientos básicos: Ser usado por cualquier persona, desde cualquier lugar y a cualquier hora [CTI2003].

### 1.2.2 Servicios y Aplicaciones

El servicio primordial del sistema IVR es atender gran cantidad de llamadas y enrutarlas hacia el departamento o usuario final sin la necesidad de

intervención humana, reduciendo así el tiempo de transferencia por llamada. Por ello el IVR es comúnmente implementado en empresas o entidades que poseen un alto número de departamentos o personal que reciben una gran cantidad de llamadas por día. Este sistema permite disminuir el personal que funcionaba como operador y al mismo tiempo reducir los costos que representa ese servicio con mejores resultados y mejor atención al cliente.

Otro servicio muy importante es la capacidad que tiene para acceder a bases de datos disponibles. Este tipo de servicio es visto como una propiedad de consulta de los usuarios hacia cierta entidad por medio de una línea telefónica. Esta información además puede ser filtrada y seleccionada, es decir, dependiendo del usuario acceder a una total o parcial información. Uniendo estos dos servicios se da lugar a un sistema que brinda gran cantidad de beneficios basados en una llamada telefónica, esto ha sido reconocido por las empresas y en la actualidad la mayor parte de las entidades ya estructuradas utilizan un IVR.

Una de las instituciones que utilizan este sistema son las agencias bancarias conocidas como “banca telefónica” que junto con la prestación del IVR para acceder a bases de datos, permiten dar una muy buena y actualizada atención al cliente. Otras instituciones como hoteles, restaurants, hospitales, servicio social, etc. también utilizan este tipo de sistema. Además es un tipo de servicio convergente, es decir, tiene la capacidad de combinarse con otros sistemas como es el caso de SMS que al combinarse con el IVR prestan cualquier clase de servicio ya sea: tele-votación, encuestas, sorteos, acceso a bases de datos, servicios informativos, etc.

Un estudio realizado en México sobre la “Infraestructura Tecnológica de los Centros de Contacto”, nos da una idea más clara de las aplicaciones reales del IVR que se pueden tener y de estas la más popular.

El autoservicio, con todos los beneficios implícitos para el cliente y el Centro de Contacto, cubre en buena medida las principales demandas del mercado. La aplicación más importante del sistema IVR que muestra este estudio es la operadora automática y la justificación predominante es el servicio de comodidad y rapidez que brinda a los usuarios finales. Además de ordenar y estructurar las llamadas entrantes, previniendo la colisión telefónica. [TCC2007]

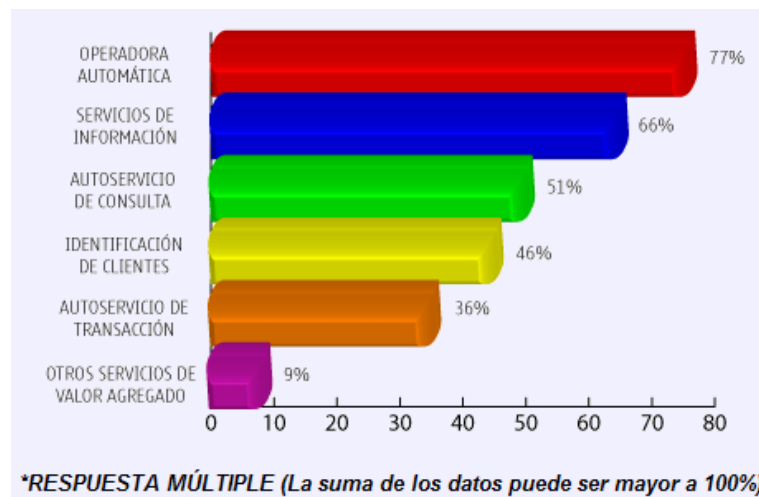


Figura. I. 4. Aplicaciones del IVR basado en un estudio en México

Fuente: [TCC2007]

- **IVR como operadora automática**

En este caso el IVR realiza el papel de una operadora automática, es decir, da un número de extensiones en forma de menú que hacen referencia a cierto departamento por medio de mensajes pregrabados; dependiendo de la selección del usuario enruta la

llamada hacia la opción elegida. Por lo general el usuario es recibido con un mensaje de publicidad y bienvenida de la entidad, además de contener tonos de espera que son reproducidos cuando se realiza una transferencia. Dentro del menú por lo general se da la opción (dígito 0) que lleva directamente a comunicarse con una operadora para tener mayor información o realizar una consulta específica, etc. En la siguiente gráfica se muestra la configuración básica del IVR como operadora automática.

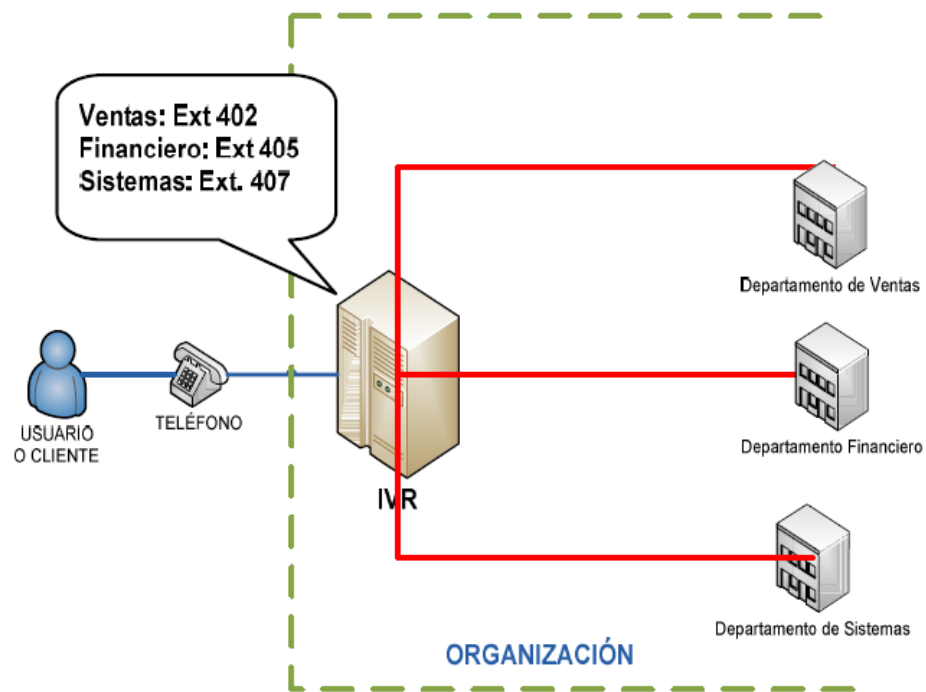


Figura. I.5. IVR como operadora automática

- **IVR como gestor de base de datos**

En este caso el IVR es mucho más robusto y permite tener una mayor cantidad de funciones. Al acceder a una base de datos el IVR es capaz de realizar funciones de identificación de usuarios y consulta de información. Entre esta información disponible podemos

citar: consultas de saldo, consulta de nombres y otro tipo de funciones como reservaciones, compras, alquiler, cambios, etc. Estas opciones dependerán de los servicios que la entidad tenga disponible y de los servicios que puedan ser ofrecidos a través de una llamada telefónica en una interacción de doble sentido entre el usuario y la entidad.

Esta interfaz se realiza enviando comandos por digitación de teclas en diferentes tonos por parte del usuario y comando de voz por parte del IVR, ya sea que han sido grabados o junto con la tecnología TTS<sup>5</sup> dar a conocer información de la base de datos.

En la siguiente gráfica se muestra la configuración básica del IVR como gestor de base de datos.

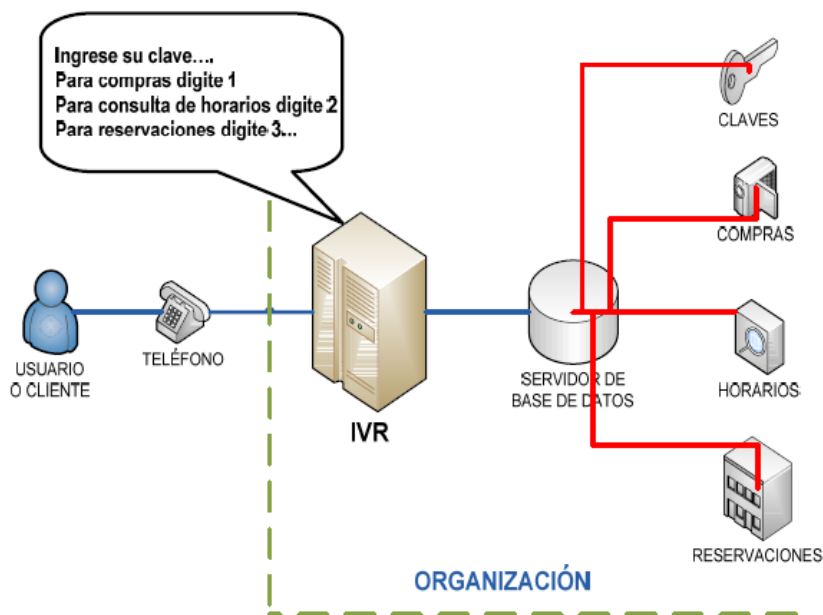


Figura. I.6. IVR como gestor de base de datos

<sup>5</sup> TTS (*Text To Speech*) Tecnología que transforma texto a voz, vista en la sección 2.4

### 1.2.3 Funcionamiento

En general el funcionamiento del IVR empieza cuando un usuario realiza una llamada a un número de teléfono que contenga un IVR por lo general entidades o empresas, el sistema IVR gestionando como una operadora automática contesta la llamada dándole la bienvenida a dicha entidad o empresa y a continuación le presenta al usuario una serie de opciones en forma de menú.

Todos estos sonidos son grabados previamente y reproducidos al entrar una llamada, por ejemplo el saludo inicial puede ser: “Usted se ha comunicado con HIGHTELECOM” y las opciones del menú pueden ser: “Para el departamento de ventas presione uno, para administración presione dos, para servicio técnico presione 3, etc.”.

Por lo general se da una opción al final que permite comunicarse directamente con una operadora la cual le guiará al usuario si no ha encontrado una opción en el menú que le satisfaga o si necesita realizar una consulta específica; por ejemplo la grabación en este caso sería: “Caso contrario espere en la línea que una operadora le atenderá” o “Presione cero para comunicarse con una operadora”.

El usuario después de escuchar el menú elige la opción que desee digitando el número indicado en el teclado del teléfono, así pues le indicará al sistema hacia donde redireccionar su llamada.

Es así como se establece la comunicación, del usuario al sistema IVR por medio de tonos del teléfono que se diferencian uno de otro por la frecuencia fundamental a la que trabajan y la comunicación del sistema IVR al usuario por medio de reproducciones de voz.



Un IVR puede tener más de un menú, es decir, la configuración permite crear submenús para poder abarcar la mayor cantidad de extensiones que una entidad posee de una forma ordenada.

Estos submenús son diferentes ramificaciones que parten siempre de un menú principal; por ejemplo al ingresar al departamento de ventas en el menú principal, el submenú puede indicar “Presione uno para gerencia, dos para asesoramiento, tres compras, etc.”. Es así como el usuario finalmente va navegando por los diferentes menús hasta llegar al destino final requerido.

En el sistema IVR existen ciertos caracteres en especial ‘i’ y ‘t’ que son muy utilizados el momento de la configuración, que al igual que un número permiten redireccionar la llamada dependiendo de la acción:

- ✓ “i”: Cuando el usuario ingresa una opción incorrecta del menú, por ejemplo si tenemos en el menú la opción 1, 2, 3 y el usuario presiona la tecla 4, en este caso se enviará al destino especificado en la letra ‘i’ que por lo general se reproduce nuevamente el menú hasta que se presione una opción correcta y si se presiona más de tres veces una opción incorrecta se cuelga la llamada. [FRE2009]
  
- ✓ t: Este es el destino usado cuando el usuario no ha presionado ninguna tecla, por lo general es enviado hacia una operadora. El tiempo de espera para que el usuario presione una tecla es variable pero por lo general es de diez segundos.

### 1.2.4 Tecnologías anexas a los sistemas IVR

- **CTI (*Computer Telephone Integration*)**

CTI “Integración de telefonía y computadora” es un grupo de componentes de software y hardware que permiten a un ordenador manejar y administrar llamadas telefónicas e integrar servicios telefónicos en computadores de escritorio, servidores, dispositivos PBX y otros equipos computacionales.

El uso de la tecnología CTI se está incrementando alrededor de un 30% anualmente especialmente en el segmento SOHO (*Small Office Home Office*) quien constituye la mayor parte de este crecimiento. [CTI2003]

CTI dispone de funciones adicionales como identificación de llamadas o enrutamiento de llamadas que facilitan la trazabilidad de los contactos realizados con los clientes y la automatización de tareas. Desde una perspectiva de alto nivel los sistemas CTI proveen los siguientes servicios [CTI2003]:

- ✓ Control de llamadas: permite que las llamadas puedan ser controladas desde una simple computadora. Las llamadas pueden ser monitoreadas, iniciadas y respondidas.
- ✓ PC como teléfono (*softphone*): teléfonos implementados en software
- ✓ Reconocimiento automático de llamadas basado en el identificador de llamadas ID (identificador) o ANI (Número automático de identificación).
- ✓ Administración de fax.
- ✓ Servicios activados por voz tal como TTS (*Text to Speech*) o STT (*Speech to Text*).

- ✓ Información mostrada en la pantalla sobre la llamada.
- ✓ Rastreo de llamadas y eventos en un sistema telefónico desde un servidor.
- ✓ Usar herramientas de internet como JTAPI (*Java Telephony Application Programming Interface*).

- **TTS (*Text To Speech*)**

El sistema TTS es el encargado de convertir un texto normal en voz natural en tiempo real. El sistema utilizado para este propósito es conocido como “sintetizador de voz” y puede ser implementado en software o hardware. El sintetizador TTS se encuentra en ordenadores o teléfonos e intentan simular la voz humana.

La voz sintetizada se basa en: el análisis del texto, fonética, fonología, sintaxis, fonética acústica y procesamiento de señales.

TTS ha mejorado significativamente con el pasar de los años la voz de computadora robotizada usualmente asociada a este sistema. Existen actualmente dos técnicas principales usadas para sintetizar la voz, que son [WIE2002]:

- ✓ Formato de síntesis de voz: requiere menos hardware (Aunque todavía 50 MIPS<sup>6</sup>) y es de menor calidad. Se basa en teléfonos individuales (como elementos fonéticos) o teléfonos pares (*diphones*)<sup>7</sup>.
- ✓ Sonidos de síntesis de voz concatenada más natural: donde se basa en frases de palabras. Requiere más recursos para su

---

<sup>6</sup> MIPS acrónimo de “Millones de Instrucciones Por Segundo”, es una forma de medir la potencia de los procesadores.

<sup>7</sup> *Diphone* es un par de teléfonos adyacentes. Es usualmente usado para referirse a la grabación de la transmisión de dos teléfonos.

implementación, haciéndolo más difícil para instalar en equipos móviles. Pero esto se está convirtiendo en una norma para sistemas de servidores base.

Adicionalmente está apareciendo una nueva técnica llamada “*phrase splicing*”. Es el sonido más natural pero es solamente apropiado para aplicaciones con un número de frases muy limitado. Los e-mails son considerados la aplicación más beneficiada con el avance de TTS. Esto también es claramente un esfuerzo muy ambicioso, cuando se refleja en el amplio vocabulario, abreviaturas ambiguas que son comunes en algunos e-mails.

- **ASR (*Automatic Speech Recognition*)**

Los sistemas ASR trabajan analizando la señal acústica recibida a través de un micrófono conectado a una computadora. Los usuarios dictan algún texto y el micrófono captura la señal acústica como datos digitales que son analizados como un modelo acústico o un modelo de lenguaje. Los diferentes sistemas de reconocimiento de voz en el mercado difieren en los bloques de fabricación que emplean para analizar la señal de datos acústicos. Los algoritmos de análisis empleados se basan en el modelo de *Markov*, un tipo de algoritmo que usa modelamiento estocástico para decodificar una secuencia de símbolos para completar los cálculos. [HCI2003]

Tanto estudios fisiológicos y sico-acústicos han mostrado que el reconocimiento de la voz humana se basa en la extracción de información paralela de la señal de voz.

En particular *Greenberg* (1997) describe cómo los diferentes tipos de segmentos de voz determinan el tipo de procesamiento espectral. Un trabajo hecho por *Fletcher* (1994) permite dar una insinuación de la inteligibilidad de voz en banda limitada. El modelo dice que en el reconocimiento humano, los errores de identificación de fonema en una banda de frecuencia dada, son independientes de los errores en otra banda. Mejoras futuras a este modelo fueron sugeridas por *Steeneken*, quien encontró que en condiciones de transmisión restringidas, los humanos requieren diferentes rangos de frecuencia óptimos para un correcto reconocimiento de fonemas independientes. Estos estudios han demostrado que incorporando más procesamiento heterogéneo en los sistemas ASR se puede incrementar la calidad de reconocimiento de estos sistemas disminuyendo errores por ruido e interferencia entre bandas. El modelo matemático de los sistemas ASR lo podemos encontrar en la siguiente referencia [IPKST].

Si le dirigimos al sistema ASR directamente a la aplicación en un IVR, encontramos que el ASR le permite al usuario enviar comandos a través de voz hacia el IVR en lugar de tonos por tecla pulsados desde su teléfono. Esta nueva forma de comunicación entre el IVR y el usuario, basada únicamente en voz permite optimizar significativamente el proceso de transacción, evita niveles del menú en el árbol de programación, mejora la interfaz y el proceso del usuario para llegar a su destino. Estas ventajas se dan ya que los usuarios ven más fácil y rápido pronunciar dígitos o palabras en lugar de pulsar las teclas de su teléfono. Como vimos anteriormente este sistema necesita de gran precisión para identificar la voz ya que se puede cometer varios errores y disminuir su confiabilidad.

Estos sistemas al tener una arquitectura compleja por requerir una alta precisión poseen un costo elevado lo que no les ha permitido tener éxito en el mercado. Pero las mejoras que se han encontrado en programación y sistemas de reconocimiento, han hecho al sistema ASR una herramienta muy importante en los actuales sistemas de reconocimiento y particularmente en los IVR.

- **PBX (*Private Branch Exchange*)**

El PBX (*Private Branch Exchange* o Central Secundaria Privada) es una central telefónica conectada directamente a la red pública de teléfono que permiten realizar la conmutación de las llamadas entrantes y salientes del sistema, además gestionar las llamadas internas sin intervención del proveedor telefónico, permitiendo a las empresas mantener redes internas con comunicación libre de facturación. El nombre como lo indica “privado” hace referencia a que esta central telefónica y su administración le pertenece a la empresa que lo tiene instalado y no a la compañía telefónica.

En las PBX los usuarios no se conectan al exterior por medio de líneas telefónicas convencionales, sino que el PBX es el que gestiona y enruta las llamadas hasta el destino final por medio de enlaces unificados de transporte de voz llamados líneas troncales. Así se realiza la conmutación de las extensiones internas PBX para que puedan realizar o recibir llamadas del exterior.

Un PBX, al igual que una central híbrida, puede automatizar los procesos de tráfico de llamadas de una oficina gracias a sus múltiples funciones, como el IVR que permite eliminar en algunos

casos la necesidad de que la recepcionista o secretaria atienda la totalidad de las llamadas entrantes. Entre las funciones ya descritas del PBX se pueden nombrar tres esenciales como: Establecimiento de llamadas internas o externas, mantenimiento de la comunicación y dar información para contabilidad y/o facturación de llamadas.

Además existen determinados servicios adicionales, como:

- ✓ Marcado Automático
- ✓ Contestador automático (IVR)
- ✓ Distribuidor automático de tráfico de llamadas
- ✓ Servicio de directorio automatizado (usuarios pueden ser ruteados a la extensión deseada tecleando o diciendo verbalmente las iniciales o el nombre del empleado)
- ✓ Cuentas con códigos para registrar llamadas
- ✓ Desvío de llamadas (al estar ocupado, no contesta, o incondicional)
- ✓ Contestar llamadas de otra extensión timbrando
- ✓ Transferencia de llamadas
- ✓ Llamada en espera
- ✓ Aviso mediante timbre cuando una línea externa/extensión está libre.
- ✓ Conferencia entre 3 o más usuarios.
- ✓ Mensaje de Bienvenida
- ✓ Marcación Abreviada (*Speed Dialing*)
- ✓ Marcado de una extensión desde el exterior del sistema
- ✓ Música en espera
- ✓ Contestador automático de buzón de voz
- ✓ Anuncio por altavoces, entre otras.

Existen también PBX–IP que son centrales desarrolladas en software integrando esta y más funciones en un computador, capaz de transmitir voz sobre IP basándose en protocolos previamente descritos como SIP, H323, etc. Esta PBX-IP reemplaza completamente a una PBX normal, de igual manera se conecta a red telefónica pública conmutada (PSTN) para el manejo de llamadas entrantes o salientes hacia el exterior.

Las interfaces con las que cuenta una PBX dependen del tipo de líneas de telefonía utilizadas, que pueden ser analógicas como digitales.

Para líneas analógicas se utiliza:

✓ FXO (*Foreign Exchange Office*)

“Interfaz de central externa” es el puerto que recibe la línea analógica. Es un enchufe del teléfono o aparato de fax, o el enchufe de su centralita telefónica analógica. Envía una indicación de colgado/descolgado (cierre de bucle). Como el puerto FXO está adjunto a un dispositivo, tal como un fax o teléfono, el dispositivo a menudo se denomina “dispositivo FXO”.

✓ FXS (*Foreign Exchange Station*)

“Interfaz de abonado externo” es el puerto que efectivamente envía la línea analógica al abonado. En otras palabras, es el “conector de la pared” que envía tono de marcado, corriente para la batería y tensión de llamada



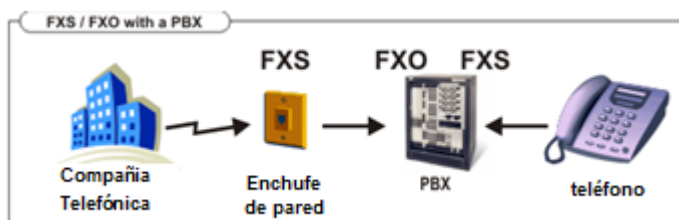
A continuación se muestra la configuración de estas interfaces sin una PBX, en este caso el equipo terminal (teléfono) se conecta directamente a la FXS dada por la compañía telefónica:



**Figura. I. 7. FXS/FXO sin PBX**

Fuente: [FXM2009]

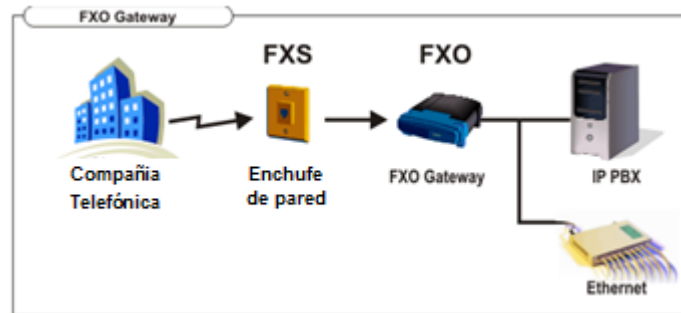
Si se cuenta con un PBX, tanto los equipos terminales como las líneas telefónicas se conectan al PBX. El PBX debe tener tanto puertos FXO como FXS.



**Figura. I.8. FXS/FXO con PBX**

Fuente: [FXM2009]

Para conectar líneas analógicas a un sistema de teléfonos IP necesitamos una puerta de enlace FXO. Esto permite conectar el puerto FXS con el puerto FXO de la puerta de enlace, el cual transforma la línea de teléfono analógica a llamadas VOIP.



**Figura. I.9. Puerta de enlace FXO**  
Fuente: [FXM2009]

Las interfaces digitales son:

- ✓ BRI (Acceso básico ISDN)
- ✓ PRI (Acceso Primario ISDN)

Y la interfaz que funciona como troncal para unir a dos centrales es:

- ✓ E&M (Conexión entre centrales)

Es la interface en un dispositivo VOIP que le permite ser conectado a las líneas troncales analógicas de un PBX. Tiene dos o cuatro hilos para señalización y dos o cuatro hilos para la voz.

En esta interfaz se utiliza el nombre derivado de los hilos que hacen referencia a: E (*Ear* - Recibe), M (*Mouth* - Transmite), para señalización como: cuelgue, descuelgue, etc.

Las PBX-IP usan solamente interfaces de este tipo para comunicarse con la PSTN. Estas interfaces se adquieren a manera de tarjetas o módulos y deben ser compatibles con la PC que administra el PBX y con el programa que se esté manejando.

## 1.3 BASE DE DATOS MySQL

### 1.3.1 Definición

Preámbulo:

Una base de datos, en su definición más sencilla, es una colección de archivos relacionados. Una base de datos y el software que controla la base de datos se denomina sistema de administración de base de datos (DBMS). La mayor parte de las bases de datos actuales son de tipo relacional. Se denominan así porque utilizan tablas de datos relacionadas por un campo en común. La generación actual de software de DBMS no solo guarda las estructuras de los datos, sino también las relaciones entre ellas y las rutas de acceso a estas, todas en una ubicación central. La generación actual de software de DBMS, también se ocupa de definir, guardar y administrar todas las rutas de acceso a dichos componentes. A continuación se muestra un gráfico donde se analiza la diferencia entre una base de datos y un sistema de archivos:

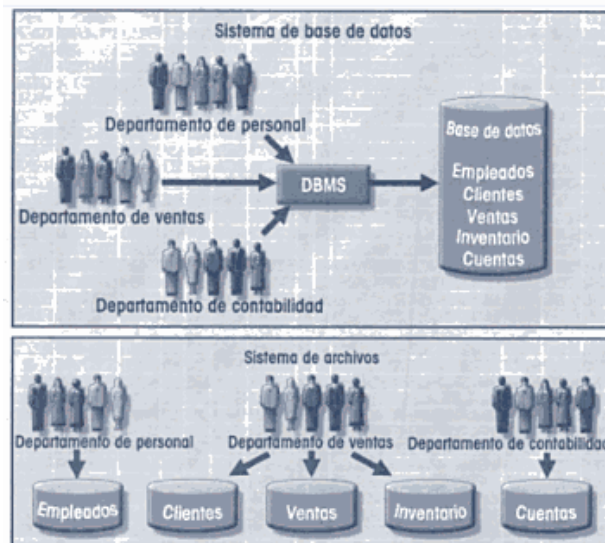


Figura. I.10. Comparación de una Base de Datos y un Sistema de Archivos

Fuente: [SBD2003]

---

Hay que recordar que el DBMS es solo uno de varios componentes vitales de un sistema de base de datos.

MySQL es un Sistema de Gestión de Bases de Datos Relacionales (SGBDR) rápido, robusto y fácil de usar. Se adapta bien a la administración de datos en un entorno de red, especialmente en arquitecturas cliente/servidor. Se proporciona con muchas herramientas y es compatible con muchos lenguajes de programación. Es el más célebre SGBDR del mundo *Open Source*, en particular gracias a su compatibilidad con el servidor de páginas Web Apache y el lenguaje de páginas Web dinámicas PHP. MySQL al difundirse como *Open Source*, permite acceder al código fuente y modificar el código para adaptarlo a las necesidades que requiera el usuario. Debido a su licencia GNU *General Public License* (GPL), toda persona puede descargar, instalar y utilizar MySQL libremente sin ninguna aportación económica

El origen del nombre MySQL no está claramente definido. Muchas bibliotecas y herramientas desarrolladas por la empresa TCX llevaban el prefijo “My”, lo que había llevado a MySQL. Otra versión dice que MySQL proviene del nombre de la hija de uno de los principales desarrolladores (*Monty*), cuyo nombre era “My”. Oficialmente, la propia empresa editora no sabe realmente cuál de estas versiones dio origen al nombre de esta base de datos.

El equipo en el que se ejecuta MySQL y que almacena los datos se denomina servidor MySQL. Este servidor de base de datos es interrogable por SQL (*Structured Query Language*), el lenguaje estándar más popular para interrogar bases de datos. SQL permite manipular los datos muy

fácilmente. El lenguaje SQL aporta una serie de funciones uniformes que permiten trabajar con los datos sin que el desarrollador necesite escribir sus propias funciones.

### 1.3.2 Estructura de una Base de Datos General

La base de datos nos permite guardar los datos con independencia del formato físico de los archivos de almacenamiento, el usuario solo tiene que gestionar el formato lógico, optimizando así el rendimiento. Además ofrece posibilidades de control de la integridad de los datos.

Una base de datos permite guardar y relacionar datos por medio de tablas compuestas con un gran número de columnas, llamadas campos y de líneas, llamadas registros, que permiten guardar datos uniformes.

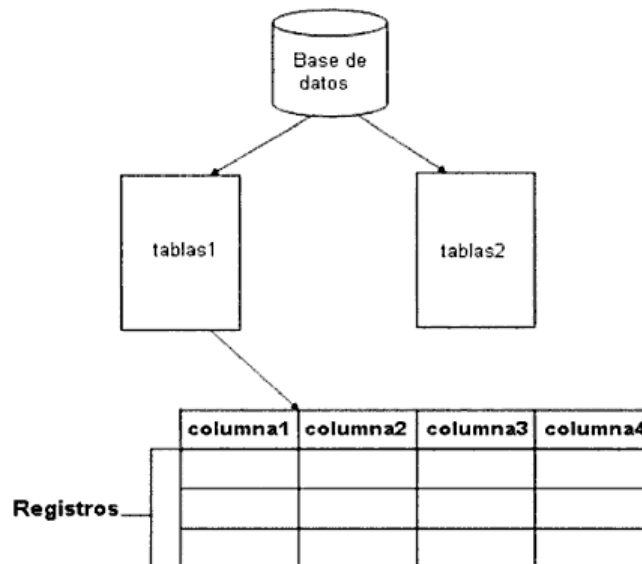


Figura. I.11. Estructura de una Base de Datos  
Fuente: [MySQL5]

Dado este tipo de estructura, la base de datos nos brinda una forma de manipular los datos SQL. SQL es una potente herramienta para realizar búsquedas sobre registros o realizar cambios. Prácticamente todos los DBMS (Sistema de Administración de Base de Datos) utilizan este lenguaje, aunque la mayoría ha agregado sus propios elementos.

Una base de datos MySQL está constituida por un directorio y archivos. El nombre del directorio es el nombre de la base de datos MySQL. El directorio contiene archivos correspondientes a las diferentes tablas de la base de datos. Una tabla se compone de varios archivos, uno con los datos, otro con la estructura, otro con los índices, etc.

MySQL limita el tamaño de una tabla a más de 8 millones de terabytes desde la versión 3.23. Anteriormente el tamaño máximo era de 4 gigabytes. Debido a que el sistema se compone de directorios y archivos, el sistema operativo puede imponer sus propios límites que pueden ser inferiores a los de MySQL.

### **1.3.3 Modelos de una Base de Datos General**

Un modelo de base de datos es un conjunto de ideas lógicas utilizadas para representar la estructura de datos y las relaciones entre ellos dentro de la base de datos. Estos modelos se pueden agrupar en dos categorías: modelos conceptuales y modelos de ejecución. [SBD2003]

- ✓ El modelo conceptual se enfoca en la naturaleza lógica de la representación de datos. Por consiguiente este modelo está comprometido con lo que está representado en la base de datos, y

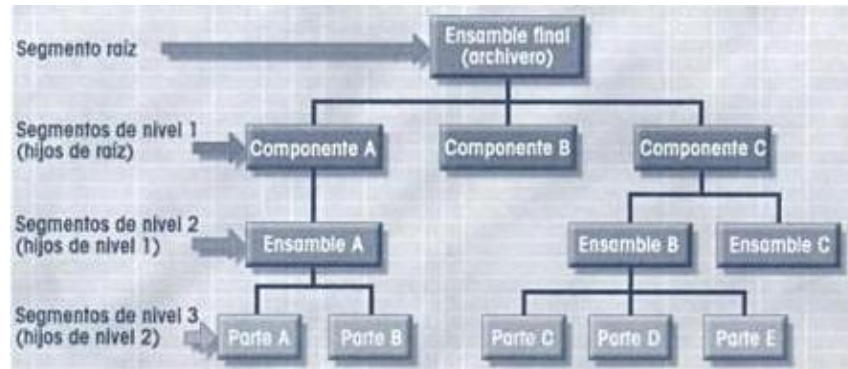
no en cómo está representado. En este modelo se incluye el modelo de Entidad-Relación y el modelo orientado a objetos.

- ✓ El modelo de ejecución hace énfasis en como los datos están representados en la base de datos o en cómo se ejecutan las estructuras de datos para representar lo que está modelado. En este modelo se incluye el modelo de base de datos jerárquico, el de base de datos de red, de base de datos relacional y el modelo de base de datos orientado a objetos.

- **Modelo de Bases de Datos Jerárquico**

El modelo de bases de datos jerárquico está basado en una estructura de árbol compuesto de un segmento raíz, segmentos padre y segmentos hijos. El segmento es el equivalente a un tipo de registro de archivo. Este modelo ilustra un conjunto de relaciones uno a muchos (1:M) entre un padre y sus hijos. Además utiliza una secuencia jerárquica o ruta pre-ordenada para navegar por sus estructuras, misma que siempre inicia por el lado izquierdo del árbol.

Gracias a su estructura padre/hijo, el modelo jerárquico produce una base de datos integra y consistente: no puede existir un registro hijo sin su padre. Además una base de datos bien diseñada es muy eficiente cuando maneja grandes cantidades de datos o muchas transacciones.



**Figura. I.12. Estructura de Base Jerárquica**

Fuente: [SBD2003]

- **Modelo de Base de Datos de Red**

El modelo de red intenta ocuparse de muchas de las limitaciones del modelo jerárquico. Si bien, el modelo de red se parece al modelo jerárquico, su estructura fácilmente acomoda la estructura hijo con múltiples padres. Incluso los componentes del modelo básico son similares; el miembro de red es el equivalente del hijo jerárquico y el propietario de la red es el equivalente al padre jerárquico.

Aunque el modelo de bases de datos de red es eficiente, su complejidad estructural a menudo limita su efectividad.



**Figura. I.13. Estructura de Base de Datos de Red**

Fuente: [SBD2003]



- **Modelo de Bases de Datos Relacional**

El modelo de bases de datos relacional es el estándar de ejecución de las bases de datos actuales. El Sistema de Administración de Bases de Datos Relacional (RDBMS) es tan complejo que el usuario o el diseñador, solo tienen que preocuparse de la visualización lógica de la base de datos; los detalles de almacenamiento físico, rutas de acceso y estructuras de datos son manejados por el RDBMS. Por lo tanto, el diseño de bases de datos relacionales es mucho más simple que el jerárquico o el de red.

Como el RDBMS le oculta la complejidad del sistema tanto al usuario como al diseñador, una base de datos relacional exhibe al mismo tiempo independencia de los datos e independencia estructural. De tal forma que el manejo de los datos es más fácil que con los primeros modelos. Se requiere mucha menos programación en el ambiente relacional porque la base de datos relacional dispone de un tipo muy poderoso de lenguaje de consulta llamado lenguaje de consulta estructurado (SQL, por sus siglas en inglés), el cual hace posible las consultas ad hoc.

- **Modelo de Relación de Entidad**

El modelo Entidad-Relación (E-R) es particularmente valioso porque está basado en una presentación visual de los datos y de sus relaciones. Además el modelo E-R permite que el diseñador de la base de datos capture diferentes vistas de datos tal como las ven ellos, los programadores y los usuarios finales de la base de datos. Por lo tanto, el modelo E-R permite que los diseñadores de base de datos integren las diferentes vistas de los datos en un marco de

referencia común. El modelo E-R produjo una nueva generación de herramientas de diseño de bases de datos; la más utilizada es el diagrama Entidad- Relación (ERD).

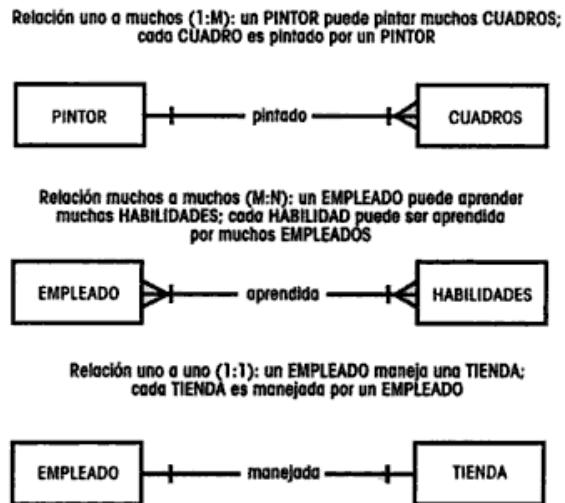


Figura. I.14. Ilustración de tipos de relación: El ERD Pata de Gallo<sup>8</sup>

Fuente: [SBD2003]

- **Modelo de Bases de Datos Orientada a Objetos**

El ambiente de datos complejo ha llevado a la búsqueda de un modelo de datos diferente que facilitara el diseño, administración y ejecución. El primero de ellos fue el modelo de datos semántico (SDM, por sus siglas en inglés). El SDM es la base para el modelo de bases de datos orientado a objetos (OODBMS). La estructura de modelado básica del SDM es el objeto. Un objeto se parece a una entidad que incluye los hechos que lo definen. Pero a diferencia de una entidad, el objeto también incluye información sobre las relaciones entre los hechos, así como también sobre las relaciones con otros objetos, por lo que sus datos tienen más significado.

<sup>8</sup> La designación “pata de gallo” se deriva del símbolo de tres puntas utilizado para representar el lado “muchos” de la relación.

A pesar de las ventajas del modelo de datos orientado a objetos, su falta de estándares, su difícil ambiente de navegación de acceso a los datos y la forma relativamente lenta en la que realizan transacciones, por los elevados requerimientos del sistema, han impedido su amplia aceptación como estándar de base de datos.

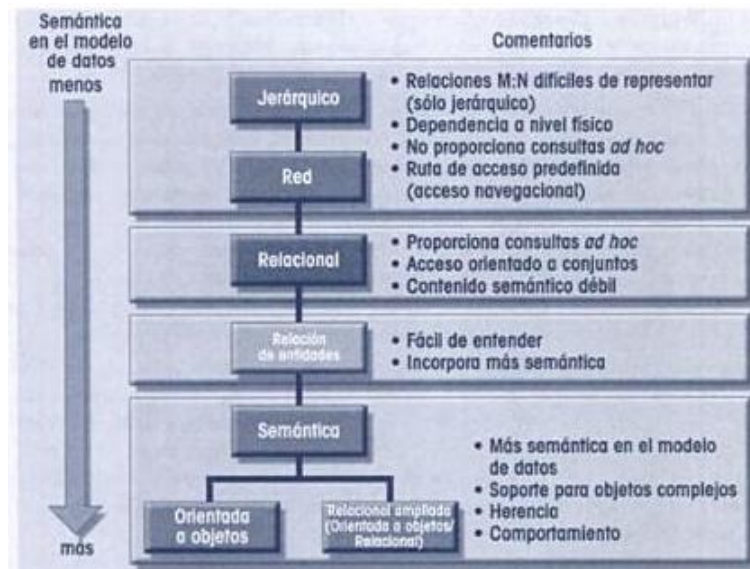


Figura. I.15. Desarrollo de Modelos de Datos

Fuente: [SBD2003]

## CAPÍTULO II

### 2 ASTERISK COMO PBX

#### 2.1 ASTERISK COMO PBX

##### 2.1.1 Definición

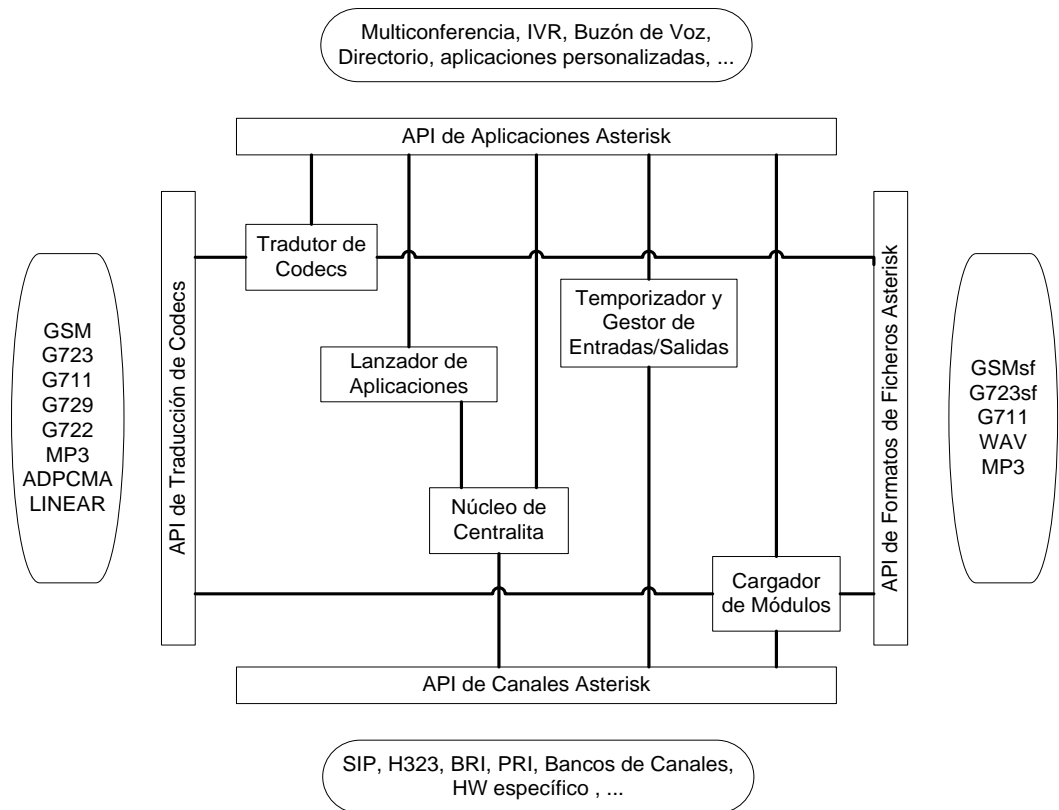
Asterisk es una aplicación *Open Source*, que se desenvuelve en un ambiente de telefonía, el cual es designado principalmente para correr bajo Linux. Asterisk combina más de 100 años de telefonía en un sólido conocimiento estrechamente integrada con aplicaciones de telecomunicaciones. Con Asterisk, nadie le dice la forma en que su sistema de teléfono debería funcionar, o las limitaciones que tiene por la tecnología. Asterisk abarca el concepto de cumplimiento de las normas, al tiempo que disfrutan de libertad para desarrollar sus propias innovaciones, es decir, Asterisk no le impone límites para sus aplicaciones. [AST2007]

Asterisk es una implementación en software de un PBX el cual proporciona las mismas funcionalidades de un PBX normal como buzón de voz, conferencias, IVR, distribución automática de llamadas, reconocimiento de llamadas, servicio de fax, desvío de llamadas, llamada en espera, entre otras. Además la ventaja de ser una aplicación en software y de código libre permite a los usuarios crear nuevas funcionalidades escribiendo un *dialplan*

en el lenguaje de *script* de Asterisk o añadiendo módulos escritos en lenguaje C o en cualquier otro lenguaje de programación soportado por Linux. Asterisk fue creado por *Mark Spencer*, de *Digium* el cual es actualmente su principal desarrollador. Asterisk originalmente fue desarrollado para el sistema operativo GNU/Linux, pero en la actualidad ya existen diferentes versiones para los sistemas operativos como MacOSX, Microsoft Windows, aunque la plataforma nativa (GNU/Linux) es la mejor soportada de todas.

### 2.1.2 Estructura

En la siguiente gráfica se muestran los principales subsistemas que posee Asterisk:



**Figura. I.16. Principales subsistemas de Asterisk**

Fuente: [VoIPTA]

- **Protocolos IP**

Asterisk soporta toda clase de tecnología telefónica, incluyendo las aplicaciones de VoIP. Maneja los siguientes protocolos:

- ✓ IAX (*Inter-Asterisk Exchanged*)

Es el protocolo de VoIP específico de Asterisk.

- ✓ SIP (*Session Initiation Protocol*)

Es el protocolo de VoIP fabricado por la IETF. Visto en el capítulo I

- ✓ H.323

Es el protocolo de VoIP fabricado por la ITU. Visto en el capítulo I

- **Interfaces de Hardware**

Asterisk soporta una gran variedad de interfaces para conectar canales de telefonía al servidor Asterisk o PBX Asterisk.

A continuación revisamos algunas de las interfaces que tiene Asterisk:

- **Interfaces *Zaptel Pseudo TDM***

Todo el hardware *Digium* presenta drivers comunes y usan una librería común. Los drivers *Digium* están basados en el driver de telefonía Zapata.

Este grupo de drivers son usualmente llamados "*Zaptel*". La interfaz *Zaptel* utiliza el procesador del equipo para simular el bus multiplexador por división de tiempo (TDM) típicamente construido

dentro de otras interfaces de telefonía. Las tarjetas de interface *Zaptel* están disponibles en *Digium* para una variedad de interfaces de red incluido PSTN, POTS, T1, E1, PRI, BRI, entre otras.

Para conectar teléfonos estándar analógicos son necesarias tarjetas electrónicas telefónicas FXS o FXO fabricadas por *Digium* u otros proveedores, ya que para conectar el servidor a una línea externa no basta con un simple módem.

- **Interfaces *No-Zaptel***

Las Interfaces aceptadas para la conexión a los servicios telefónicos tradicionales que soportan conmutación *Pseudo-TDM* se muestran en la siguiente tabla:

**Tabla. II.3. Interfaces *No-Zaptel***

Fuente: [VoIPTA]

Interface	Description
ISDN4Linux	Basic Rate ISDN interface for Linux
OSS/Alsa	Sound card interfaces
Linux Telephony Interface (LTI)	Quicknet Internet Phonejack/Linejack
Dialogic	Full-duplex Intel/Dialogic hardware

- **Interfaces de Telefonía Linux**

Las interfaces de telefonía Linux han sido desarrolladas principalmente por *Quicknet, Inc.* con la ayuda de *Alan Cox*. Esta

interface es orientada hacia la simple interface analógica y da soporte para *codecs* con baja tasa de transmisión.

Los siguientes productos son conocidos por trabajar con Asterisk, aunque no pueden trabajar tan bien como los dispositivos *Digium*.

- ✓ *Quicknet Internet Phonejack* (ISA, FXS)
- ✓ *Quicknet Internet Phonejack PCI* (PCI, FXS)
- ✓ *Quicknet Internet Linejack* (ISA, FXO or FXS)
- ✓ *Quicknet Internet Phonecard* (PCMCIA, FXS)
- ✓ *Creative Labs VoIP Blaster* (*limited support*)

- **Codecs**

Asterisk provee una traducción transparente entre cualquiera de los siguientes *codecs*:

**Tabla. II.4. Codecs soportados por Asterisk**  
Fuente: [VoIPTA]

Codec	Rate
16-bit linear	128 kbps
G.711u (micro-law)	64 kbps
G.711a (A-law)	64 kbps
IMA-ADPCM'	32 kbps
GSM 6.10	12 kbps
MP3	variable, decode only
LPC-10	2.4 kbps



- **Formato de Archivos**

Asterisk usa archivos para almacenar datos de audio incluido buzones de voz y música en espera. En la siguiente tabla se muestran las extensiones de archivos soportados:

**Tabla. II.5. Formatos de Archivos de audio**

Fuente: [VoIPTA]

format	description
raw	16-bit linear raw data
pcm	8-bit micro-law raw data
vox	4-bit IMA-ADPCM raw data
wav	16-bit linear WAV file at 8000 Hz
WAV	GSM compressed WAV file at 8000 Hz
gsm	raw GSM compressed data

### 2.1.3 Directorios

La siguiente tabla muestra los archivos relacionados con Asterisk (directorios) y la ubicación donde son almacenados, además de una breve descripción de cada uno:

**Tabla. II.6. Directorios de Asterisk<sup>9</sup>**

Directorio	Descripción
/etc/Asterisk	Todos los archivos de configuración excepto <i>/etc/Zaptel.conf</i>
/usr/sbin	Ejecutables y <i>scripts</i> incluyendo Asterisk, <i>astman</i> , <i>astgenkey</i> y <i>safe_asterisk</i> .
/usr/lib/Asterisk	Arquitectura de Asterisk específicamente objetos binarios.

<sup>9</sup> Traducido de [VoIPTA] páginas: 29-30 y con referencias de [AST2007] - Capítulo3

/usr/lib/asterisk/modules	Contiene varias aplicaciones, <i>codecs</i> , formatos y canales usados por Asterisk.
/usr/include/Asterisk	Cabeceras de archivos requeridas para construir aplicaciones, canales y otros módulos cargables.
/var/lib/Asterisk	Datos variables usados por Asterisk durante la operación normal.
/var/lib/asterisk/agi-bin	Contiene los <i>scripts</i> de costumbre. AGI <i>scripts</i> usados por las aplicaciones AGI dial plan.
/var/lib/asterisk/astdb	Base de datos de Asterisk, mantiene información de configuración. Este archivo nunca es cambiado a mano, se usa las funciones de líneas de comandos para cambiar, añadir o modificar este archivo.
/var/lib/asterisk/images	Imágenes referenciadas para aplicaciones o para el dial plan.
/var/lib/asterisk/keys	Claves públicas y privadas usadas dentro de Asterisk para autenticación RSA. IAX usa claves almacenadas aquí.
/var/lib/asterisk/mohmp3	Archivos mp3 usados para música en espera. La configuración para la música en espera es encontrada en el directorio /var/lib/Asterisk/sounds
/var/lib/asterisk/sounds	Archivos de audio, <i>promts</i> , etc. usados por aplicaciones de Asterisk. Algunas aplicaciones pueden tener sus archivos en subdirectorios.
/var/run	Contiene la información del proceso ID (PID) para todos los procesos activos en el sistema, incluyendo Asterisk.
/var/run/asterisk.pid	Identificador del proceso primario (PID) del corrimiento de Asterisk.
/var/run/asterisk/ctl	Llamado pipe usado por Asterisk para habilitar la operación remota.
/var/spool/Asterisk	Duración de archivos en cola para el buzón de voz, llamadas salientes, etc. Contiene algunos subdirectorios incluyendo <i>dictate/</i> , <i>meetme/</i> , <i>monitor/</i> , <i>outgoing/</i> , <i>system/</i> , <i>tmp/</i> , y <i>voicemail/</i> .
/var/spool/asterisk/outgoing	Asterisk monitorea este directorio para llamadas salientes. Una llamada

	saliente resulta en un archivo en este directorio. Asterisk analiza el archivo creado y los intentos de hacer una llamada. Si la llamada se contesta, se pasa al PBX Asterisk.
/usr/spool/asterisk/qcall	Utilizado por la aplicación obsoleta <i>gcall</i> . Se recomienda no usarlo.
/var/spool/asterisk/vm	Cajas de voz, anuncios y carpetas.

#### 2.1.4 Funcionamiento de Asterisk

A pesar de que Asterisk puede ser controlado directamente con su *script* de inicio (ubicado en */etc/rc.d/init.d/asterisk*) lo recomendable en la solución Elastix (sección 2.2) es hacerlo mediante el programa “*amportal*”. Esto es así debido a que existen otros programas relacionados con Asterisk que también se deben sincronizar con el encendido y apagado de Asterisk. [CUE2008]

Tabla. II.7. Comandos para el programa *amportal*

Acción	Comando
Apagar	<i>amportal stop</i>
Encenderlo	<i>amportal start</i>
Reiniciarlo	<i>amportal restart</i>

#### 2.1.5 Estructura Modular

Asterisk utiliza la idea de módulos para extender su funcionalidad. Es tan importante la funcionalidad residente en los módulos que sin ellos Asterisk por sí solo no cumpliría ninguna función interesante. Estos módulos residen en la carpeta */usr/lib/asterisk/modules/* y son archivos con extensión *.so*

Los módulos pueden ser controlados a través del archivo de configuración *modules.conf*. En este archivo puedo decirle a Asterisk que cargue o no cargue un módulo específico a través de las directivas *load* y *noload*. Adicionalmente se pueden cargar módulos en tiempo real a través de la consola de Asterisk con el comando “*module load*”. [CUE2008]

### 2.1.6 *Zaptel*, ahora DAHDI

A pesar de ser un gran producto de software Asterisk necesita hardware para complementar sus servicios y salir a la PSTN. Uno de los paquetes de software que lo complementa se llama *Zaptel*. Básicamente *Zaptel* es un conjunto de *drivers* para controlar hardware telefónico como tarjetas PCI que nos permiten conectarnos a la PSTN.

Estos *drivers* se comunican con Asterisk a través de un módulo de Asterisk llamado *chan\_zap.so* que se configura a través del archivo *zapata.conf*. Adicionalmente *Zaptel* tiene su propio archivo de configuración, independiente de Asterisk, llamado *Zaptel.conf* y ubicado en la carpeta */etc/*. A continuación se muestra un diagrama explicativo de la interacción de Asterisk con *Zaptel*. [CUE2008]

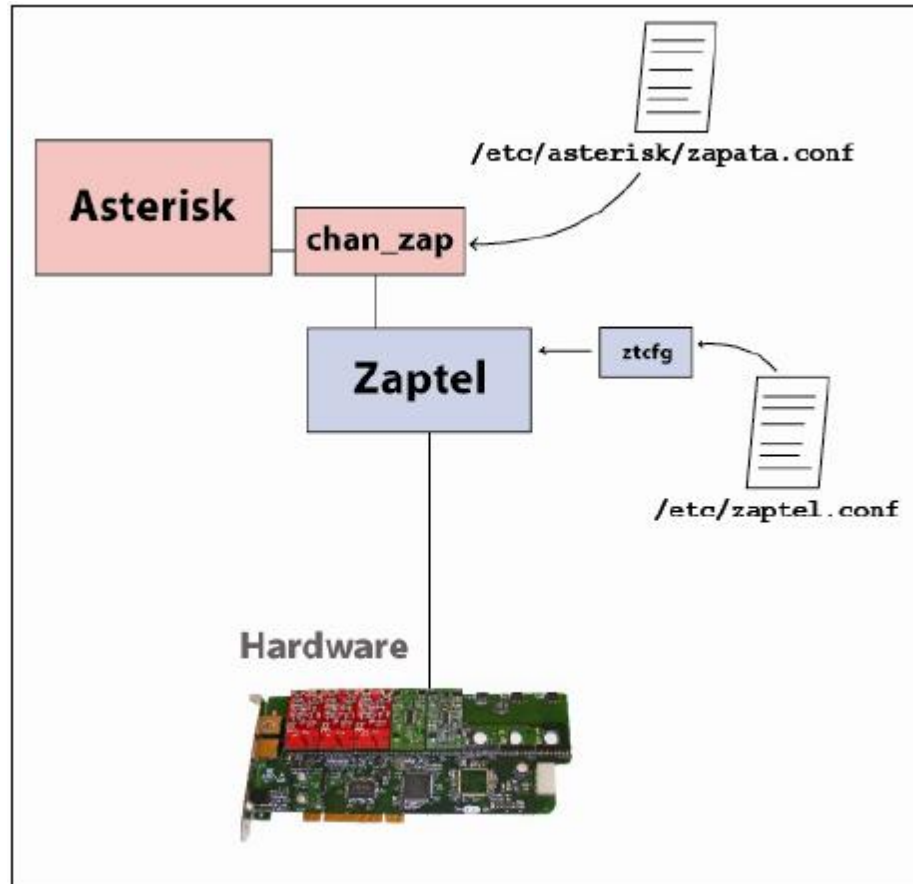


Figura. II.17. Interacción de Asterisk con Zaptel

Fuente: [CUE2008]

Inicialmente *Zaptel* contenía *drivers* para tarjetas Tormenta fabricadas por Zapata Telecom, pero pronto Digium comenzó a mejorar los *drivers* y extender el soporte para nuevos modelos de hardware, entre ellos los que el mismo Digium fabricaba. Pronto Digium se convirtió en el principal desarrollador de *Zaptel*. Luego de algunos años de mantener *Zaptel*, Digium cayó en cuenta de que *Zaptel* era una marca registrada de Zapata Telecom y para evitar cualquier posible confusión futura decidió cambiar de nombre a sus *drivers*. El nombre elegido es DAHDI.

DAHDI ya se encuentra disponible para descarga pero muchos continúan usando *Zaptel* por razones históricas o esperando a que este nuevo paquete se establezca un poco más. Digium ha manifestado que en su nueva versión de Asterisk 1.6 ya no soportarán *Zaptel* oficialmente.

### 2.1.7 Plan de Mercado

El plan de marcado en Elastix reside principalmente en el archivo *extensions.conf*. Desde este archivo se incluyen otros importantes que contienen también porciones del plan de marcado. Los principales son:

- ✓ *extensions\_additional.conf*

Aquí reside el plan de marcado variable escrito por FreePBX. Si se escribe manualmente en este archivo los cambios se perderán cuando FreePBX sobrescriba este archivo.

- ✓ *extensions\_custom.conf*

Aquí se puede escribir el plan de marcado definido manualmente por el usuario. Este archivo no es tocado por FreePBX por lo que es seguro escribirlo sin temor a perder los cambios.

Todos estos archivos de plan de marcado están escritos en texto plano en un formato definido por Asterisk. Estos archivos se encuentran divididos en secciones llamadas contextos.

Básicamente los contextos sirven para agrupar lógica de plan de marcado. Los contextos también pueden incluir otros contextos mediante la cláusula *"include"*.

### 2.1.8 Asterisk AGI

AGI (Asterisk Gateway Interface) es una manera de interactuar con Asterisk desde un programa de línea de comandos. Este programa puede ser escrito en prácticamente cualquier lenguaje y es invocado por Asterisk desde el *dialplan*.

Al momento de la invocación de un programa AGI se le pasan ciertos parámetros y este a su vez responde con comandos AGI que son entendidos por Asterisk.

La utilidad de AGI es enorme y nos permite desarrollar funcionalidad extra para nuestra central telefónica. Un ejemplo muy común de esto es el desarrollo de aplicaciones de consulta telefónica a bases de datos.

Los programas o *scripts* AGI son muy populares y se encuentran en la carpeta */var/lib/asterisk/agi-bin/*.

Los *scripts* AGI se comunican con Asterisk a través de STDIN, STDOUT y STDERR. Esto quiere decir que deberían poderse ejecutar desde la misma línea de comandos.

Los *scripts* AGI deben ser ejecutables por el usuario Asterisk, para ello ejecutamos las siguientes sentencias a cada *script agi*:

```
# chown asterisk.asterisk script_prueba.agi
# chmod 755 script_prueba.agi
```

Con esto ya podemos ejecutar el *script* AGI desde el *dialplan*, esto se lo hace gracias a la función AGI() que puede ser llamada desde el *dialplan* de Asterisk.

Por omisión Asterisk asume que cualquier *script* se encuentra en la ruta */var/lib/asterisk/agi-bin/* así que si lo ubicamos en un lugar diferente habrá que especificar la ruta completa.

En el momento de la invocación del *script*, Asterisk le envía cierta información. A continuación un ejemplo de la información enviada cuando ejecutamos un *script* llamado `dialparties.agi`, un componente de FreePBX.[CUE2008]

```
agi_request: dialparties.agi
agi_channel: SIP/4444-08d6ab60
agi_language: en
agi_type: SIP
agi_uniqueid: 1222117320.271
agi_callerid: 4444
agi_calleridname: Edgar Landivar
agi_callingpres: 0
agi_callingani2: 0
agi_callington: 0
agi_callingtns: 0
agi_dnid: 216 6
agi_rdnis: unknown
agi_context: macro-dial
agi_extension: s
agi_priority: 3
agi_enhanced: 0.0
agi_accountcode:
```

### 2.1.9 Ventajas

Asterisk al funcionar como una PBX basada en software brinda una gran cantidad de prestaciones, las cuales se las puede especificar en: Funcionalidad, Escalabilidad, Reducción y Competitividad de costo, Interoperabilidad y Flexibilidad.

✓ En Funcionalidad:

Asterisk dispone de todas las funcionalidades de las grandes centrales telefónicas propietarias (Cisco, Avaya, Alcatel, Siemens, etc.). Desde las más básicas (desvíos, capturas, transferencias,



multiconferencias, etc.) hasta las más avanzadas (Buzones de voz, IVR, etc.).

✓ En Escalabilidad:

El sistema puede dar servicio desde 10 usuarios en una sede de una pequeña empresa, hasta 10.000 de una multinacional repartidos en múltiples sedes.

✓ En Reducción y Competitividad de costo:

No solo por ser un sistema de código abierto (*Open Source*) sino gracias a su arquitectura de bajo hardware, es decir, utiliza una plataforma servidor estándar (de propósito no específico) y tarjetas PCI para los interfaces de telefonía, que por la competencia del mercado se han ido abaratando progresivamente. Además permite a las empresas disminuir el gasto de adquisición de costosas centrales telefónicas y en muchos casos la reducción de personal como operadoras y secretarias.

✓ En Interoperabilidad y Flexibilidad:

Asterisk ha incorporado la mayoría de estándares de telefonía del mercado, tanto los tradicionales (TDM) con el soporte de puertos de interfaz analógicos (FXS y FXO) y RDSI (básicos y primarios), como los de telefonía IP (SIP, H.323). Eso le permite conectarse a las redes públicas de telefonía tradicional e integrarse fácilmente con centrales tradicionales y otras centrales IP.

## 2.2 SOLUCION ELASTIX

Elastix es una distribución ecuatoriana de “Software Libre” de Servidor de Comunicaciones Unificadas que integra en un solo paquete algunas tecnologías de comunicaciones claves como:

- ✓ VoIP PBX
- ✓ Fax
- ✓ Mensajería Instantánea
- ✓ Email
- ✓ Colaboración

Elastix implementa gran parte de su funcionalidad sobre 4 programas de software muy importantes como son Asterisk, Hylafax, Openfire y Postfix. Estos brindan las funciones de PBX, Fax, Mensajería Instantánea e Email, respectivamente. La parte de sistema operativo se basa en CentOS, una popular distribución Linux orientada a servidores.

A continuación una figura un poco más ordenada donde se pueden observar los componentes de Elastix y su relación entre sí.

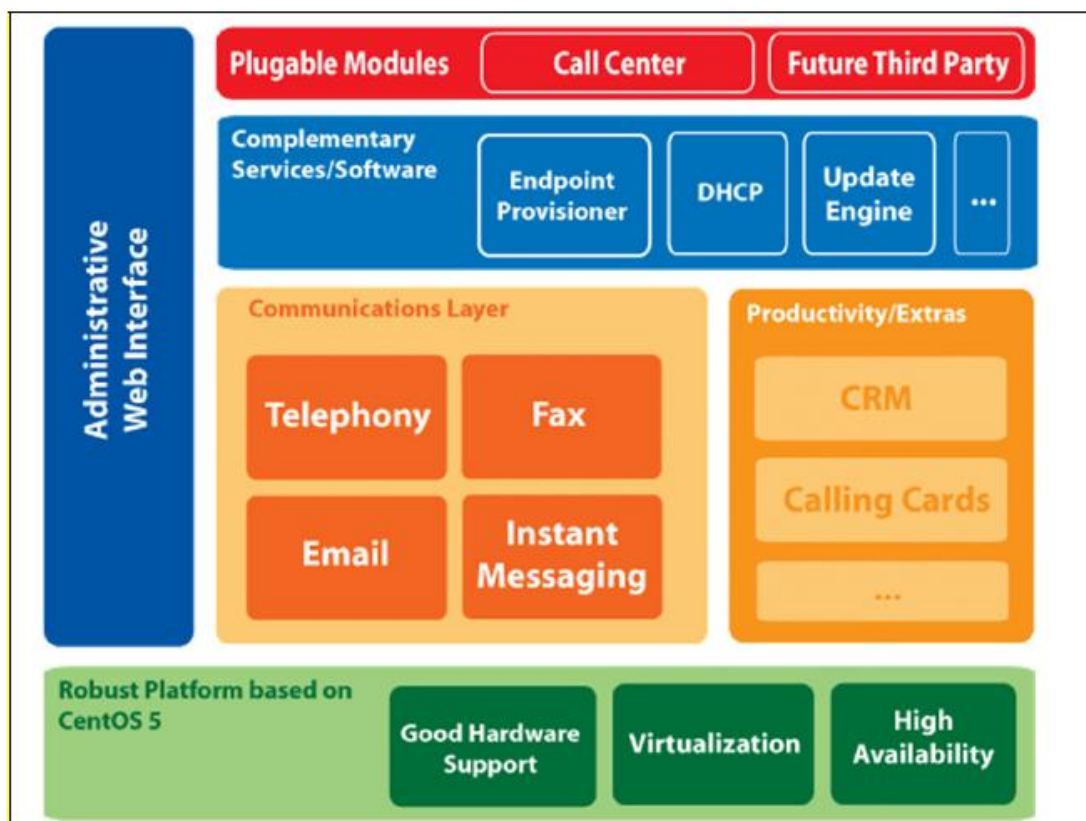
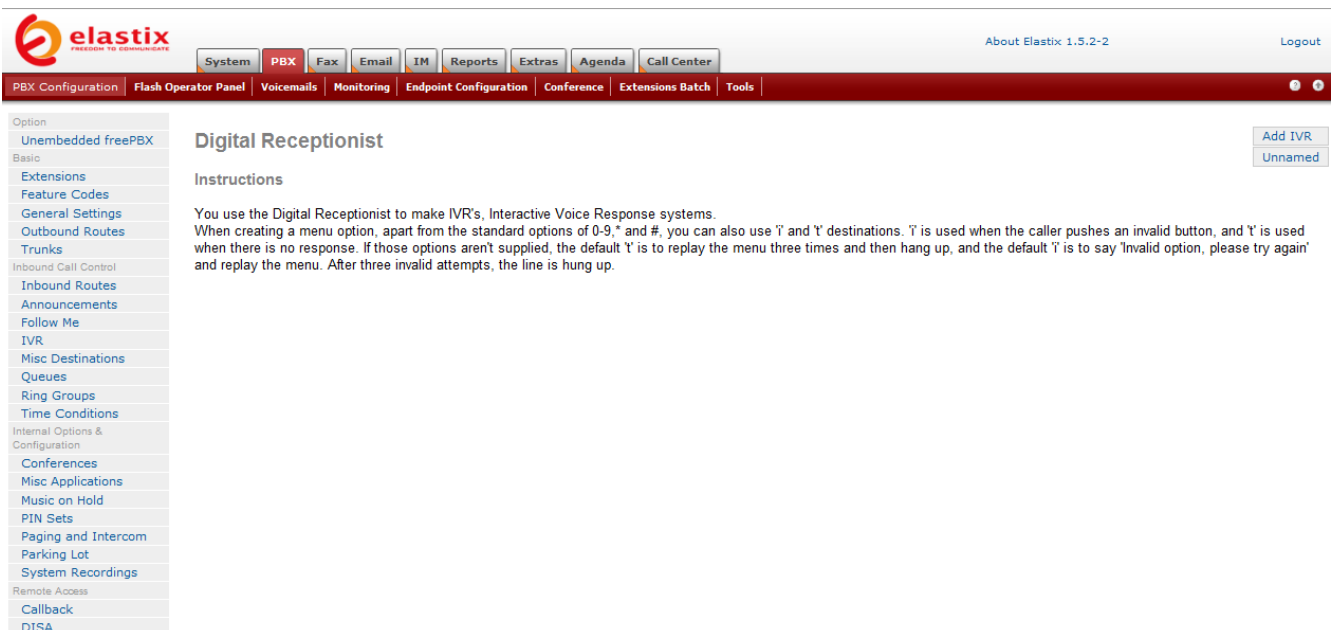


Figura. II.18. Esquema general de los componentes de Elastix  
Fuente: [CUE2008]

Elastix es software libre distribuido bajo licencia GPL versión 2. Es decir que puede ser usado, copiado, estudiado, modificado y redistribuido libremente según los lineamientos de esta licencia.

La interface Web de Elastix es una aplicación completa de administración del servidor de comunicaciones unificadas escrita en su mayoría en lenguaje PHP. Está basado en la herramienta FreePBX, en su mayoría toma los módulos más importantes de esta herramienta y los presenta en su interfaz web de una forma personalizada.



**Figura. II.19. Interfaz gráfica de Elastix**

Fuente: [CUE2008]

Elastix da la facilidad de acceder a la interfaz web propia de FreePBX donde están todos los módulos configurables de esta herramienta. Para acceder a esta interfaz vamos al menú PBX y dentro de este a la opción “FreePBX Sin embeber”.



Figura. II.20. FreePBX no embebido en Elastix  
Fuente: [ELXCJM]

La instalación paso a paso de Elastix está en la referencia [CUE2008] sección 5, o se la puede descargar de la página web: <http://www.elastix.org>.

### 2.3 HERRAMIENTA GRÁFICA FREE PBX

Como Elastix está basado en FreePBX, haremos un análisis de FreePBX y de todos los módulos y características que esta herramienta presenta.

Para nuestra aplicación en específico, la interfaz web de Elastix como FreePBX son las mismas, manejan los mismos módulos con sus respectivos campos.

### 2.3.1 Propiedades

FreePBX es un proyecto *Open Source* desarrollado completamente en código PHP y de forma modular. Su finalidad es facilitar la administración de un sistema Asterisk a través de una interfaz gráfica. Es un GUI web que se encarga de realizar por el usuario todos los cambios necesarios en los archivos de configuración del Asterisk, cuando este modifique de forma gráfica la configuración de la central.

Con esto, el administrador no necesariamente debe ser un experto en Asterisk para manejar la central, solo basta conocer conceptos básicos de telefonía y funcionamiento de una central telefónica como llamadas entrantes y salientes, extensiones analógicas, extensiones digitales, extensiones SIP, IVR, Correo de Voz, Fax, desvío de llamadas, etc.

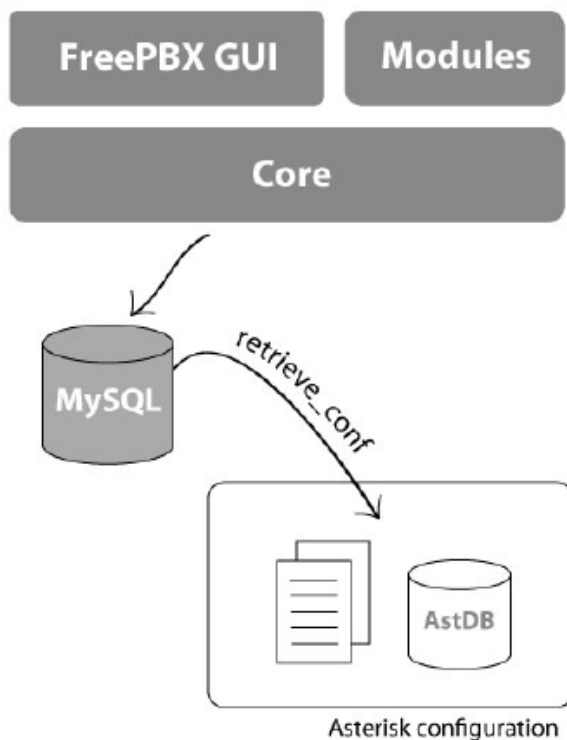
FreePBX es una herramienta muy potente desde un punto de vista empresarial, al ofrecer una herramienta gráfica al cliente y por la velocidad de respuesta ante peticiones. Además, FreePBX es bastante flexible en cuanto a la edición de ficheros adicionales que no limiten la funcionalidad de Asterisk.

### 2.3.2 Fundamentos de FreePBX

FreePBX es un poderoso software de configuración de Asterisk que ha sido integrado como parte de Elastix. Básicamente es el software que se encarga de generar los archivos de configuración de Asterisk incluido el plan de marcado en base a la configuración ingresada por el usuario desde la Web.

Para realizar esta difícil tarea FreePBX la divide en dos pasos:

- ✓ Recaba la configuración deseada por el usuario en una base de datos de configuración. En el caso de Elastix se usa el motor de base de datos MySQL.
- ✓ Luego convierte esa información almacenada en MySQL principalmente en archivos de configuración a través de una herramienta de *parsing* llamada *retrieve\_conf* y envía esta información a la base de datos por defecto de Asterisk llamada “AstDB”.



**Figura. II.21. Interfaz gráfica de Elastix**  
Fuente: [CUE2008]

- **Relación entre FreePBX y Asterisk por el programa**

*retrieve\_conf* es un programa escrito en PHP que se encarga de generar el plan de marcado a partir de la información almacenada en

la base de datos. Este programa puede ser invocado inclusive desde la línea de comandos aunque usualmente se ejecuta desde la Web a través de un *script* PHP cada vez que aplicamos los cambios en la interfaz web (dando clic en la barra roja de FreePBX “aplicar cambios”).

- **Base de datos de Asterisk (*AstDB*)**

Asterisk proporciona un potente mecanismo para el almacenamiento de los valores, se conoce como base de datos de Asterisk (*AstDB*). El *AstDB* proporciona una forma sencilla de almacenar datos para el uso dentro del dialplan.

La base de datos de Asterisk no es una base de datos relacionales tradicionales. Se trata de una base de datos Berkeley DB, versión 1.

La base de datos almacena los datos de los usuarios de Asterisk en grupos llamados familias, con *values identified* por claves. Dentro de una familia, una clave puede ser utilizada sólo una vez. Cada valor almacenado debe estar asociado con una familia.

- ***Script dialparties.agi***

Se trata de un *script* AGI, escrito en PHP, que se invoca desde el plan de marcado cuando se quiere hacer una llamada. Este *script* se encarga de revisar el estado de la extensión a la que se está tratando de alcanzar, así como también ciertas opciones que pudieran estar configuradas en dicha extensión (como *call waiting*, *call forward*, *do not disturb*, etc) y retorna dicha información para que se pueda tomar una acción acorde como por ejemplo devolver al llamante un tono de ocupado o enrutar la llamada a otra extensión en el caso de *call*

*forward*. Si la extensión está disponible y no se encuentran opciones activadas la llamada transcurre sin inconvenientes.

Para realizar su trabajo *dialparties.agi* consulta la base de datos de Asterisk (la que se encuentra almacenada en el archivo */var/lib/asterisk/astdb*).

### 2.3.3 Organización de Archivos

FreePBX escribe o lee información de algunos directorios para su correcto funcionamiento. A continuación una lista de los más importantes.

**Tabla. II.8. Directorios que maneja FreePBX**

Fuente: [CUE2008]

Directorio	Descripción
<i>/var/www/html/admin/</i>	El directorio principal donde reside el GUI de FreePBX
<i>/var/www/html/panel/</i>	Directorio del Flash Operador Panel. En realidad se trata de otro proyecto independiente pero relacionado.
<i>/etc/asterisk/</i>	El directorio donde residen los archivos de configuración de Asterisk que escribe FreePBX
<i>/var/lib/asterisk/agi-bin/</i>	Aquí residen los <i>scripts</i> AGI de Asterisk. FreePBX cuenta con algunos <i>scripts</i> en esta carpeta como el <i>dialparties.agi</i>
<i>/var/lib/asterisk/bin/</i>	Algunos binarios utilizados por FreePBX. Algunos de ellos son links simbólicos a las carpetas de los módulos
<i>/usr/local/sbin/</i>	Aquí reside el <i>script amportal</i> . Se encuentra en esta carpeta porque puede ser manipulado solo por <i>root</i>
<i>/var/lib/asterisk/sounds/</i>	Aquí residen los sonidos que se utilizan en el plan de marcado por FreePBX
<i>/var/lib/asterisk/sounds/custom/</i>	Aquí residen los sonidos que han sido creados por el usuario. Por ejemplo cuando se crea un IVR de bienvenida
<i>/var/spool/asterisk/voicemail/</i>	Directorio donde se almacenan los <i>Voicemails</i>
<i>/var/spool/asterisk/monitor/</i>	En este directorio se almacenan las grabaciones de voz
<i>/var/log/asterisk/cdr-csv/</i>	Directorio donde se escriben los <i>CDRs</i> en formato texto



### 2.3.4 Alcance

FreePBX está compuesto por módulos que permiten administrar la central telefónica. No existe un orden preestablecido en el que deben configurarse, ya que depende de las necesidades del cliente o empresa. En general, la configuración de un solo módulo no tiene efectos en la configuración del sistema por sí solo, en la mayor parte de los casos, necesita complementarse con otros. Por ejemplo, si se configuran troncales para recibir y/o hacer llamadas, no podrán utilizarse sin configurar rutas salientes y/o entrantes.

En el siguiente gráfico se muestra un ejemplo de configuración donde puede apreciarse la coparticipación entre los módulos de FreePBX.

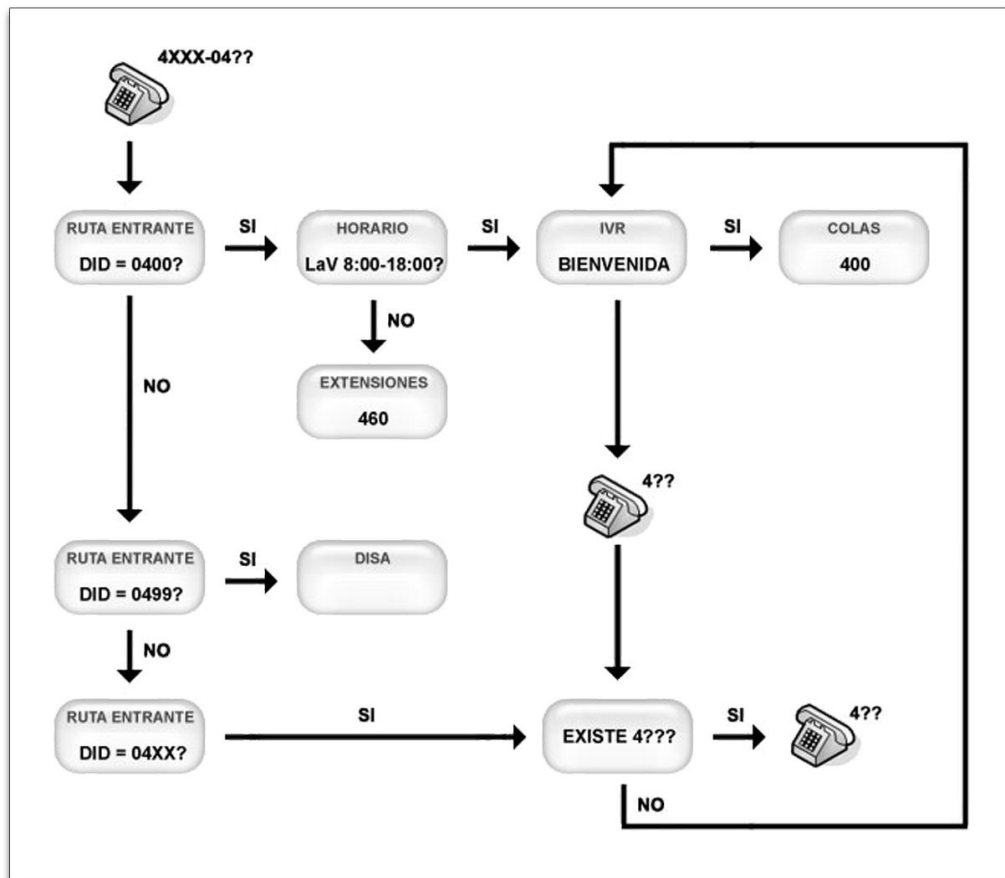


Figura. II.22. Ejemplo de Interacción entre módulos de FreePBX

Fuente: [ADS2.0]

- **Sistema y Módulos de FreePBX**

En la siguiente figura se muestra la página principal de la herramienta FreePBX y todos los posibles módulos a configurarse [ADS2.0]:

The screenshot displays the FreePBX System Status page. On the left is a navigation menu with categories like Setup, Tools, Admin, and various system modules. The main content area is titled 'FreePBX System Status' and includes several informational sections: 'FreePBX Notices' with warnings about upgrades and passwords; 'FreePBX Statistics' showing zero active calls and channels; 'FreePBX Connections' listing online IP phones, trunks, and registrations; 'Uptime' showing system and Asterisk uptime; 'System Statistics' with processor, memory, and disk usage; and 'Server Status' with a list of services (Asterisk, Op Panel, MySQL, Web Server, SSH Server) all marked as 'OK'.

Figura. II.23. Página principal de FreePBX

Fuente: [ELXCJM]

Brevemente se explicarán los módulos que manejan FreePBX y todos los servicios que presta:

- ✓ Extensiones (*Extensions*)

Esta área es para *handsets*, *softphones*, sistemas de *pager* o cualquier otro dispositivo que pueda ser considerado como

---

extensión de un sistema PBX. Existen cuatro tipos de dispositivos como SIP, IAX2, ZAP y personalizado (*custom*).

✓ *Feature Codes*

Este módulo permite personalizar los números que se deben marcar para acceder a las funciones de Asterisk. También pueden deshabilitarse funciones, como por ejemplo, aquellas que puedan comprometer la privacidad.

✓ Configuraciones Generales (*General Settings*)

En este módulo se encuentran todo lo referente a:

- Opciones de Marcado
- Correo de Voz
- Directorio de la empresa
- Máquina de Fax
- Configuraciones Internacionales
- Configuraciones de Seguridad

✓ Rutas Salientes (*Outbound Routes*)

Las llamadas salientes son enviadas por troncales de la manera en que estén configuradas en la página de rutas salientes. Esto está diseñado para ser tan flexible como sea posible, y permite mapeos de bajo costo.

✓ Troncales (*Trunks*)

Las Troncales se utiliza para llevar una ó varias llamadas a un VSP (*Voice Service Provider*) ó a algún dispositivo que responda al número que se marcó (por ejemplo, otra central Asterisk o la red PSTN). Existen cinco tipos de Troncales: ZAP

(líneas analógicas), IAX2 (Inter Asterisk), SIP (*Session Initiation Protocol*), ENUM y Personalizada (*Custom trunk*).

Todas las troncales son configuradas de la misma forma

✓ Lista Negra (*Blacklist*)

En la sección de configuración se puede ingresar o reemplazar un número a la lista de números bloqueados conocida como lista negra.

✓ Fuentes de búsqueda de nombres de llamados (*Caller Name Lookup Sources*)

Este módulo permite identificar el nombre del llamante a través del número de teléfono. Este tipo de búsqueda permite especificar una fuente para resolver identificaciones numéricas de las llamadas entrantes, con esto se puede vincular una ruta entrante específica a una fuente CID (*callerID* - número desde el cual el llamante realiza la llamada) particular. Se puede también instalar el módulo de directorio telefónico (*phonebook*) para tener asociado un número a un nombre.

✓ Rutas entrantes (*Inbound Routes*)

Este módulo permite configurar que destino usa Asterisk para llamadas entrantes desde troncales. Cuando una llamada es recibida por Asterisk desde una troncal, se identifica el DID (número marcado por el llamante) y el CID (*callerID* - número desde el cual el llamante realiza la llamada) y la llamada es derivada de acuerdo a su configuración.

✓ *Follow Me*

La configuración de “*follow me*” es como un mini grupo de extensiones, pero está ligada directamente a una extensión. Se configura de la misma manera en la que se configura un

grupo de extensiones, incluyendo la prestación de anunciar a la persona que llama que está siendo transferida a otra parte.

Existen otros usos para la función de “*follow me*”. Como algo simple, puede ingresar en la extensión del rastreo una opción para ir al correo de voz si no es respondida la llamada. La diferencia es que así se puede variar el tiempo de llamado, introducir un anuncio, ir a otro correo de voz ó si el destino no es alcanzado, puede agregar múltiples números y estrategias de llamado, para cuando esa extensión sea solicitada.

El lugar donde la función de rastreo no es prioritaria es cuando la extensión local es seleccionada como un destino explícito desde varias pantallas.

✓ Colas (*Queues*)

Las colas le permiten administrar un gran número de llamadas entrantes, como es de esperarse en un *call center* ó en un *contact center*. Esta es una aplicación muy completa, y como tal, tiene muchas opciones de configuración.

✓ DISA

DISA (*Direct Inward System Access*) le permite proveer un tono de discado interno a llamantes externos. Cuando se configura un destino DISA, se puede usarlo como un menú de destino de una recepcionista digital, para poder obtener un tono de discado interno de Asterisk. Esto significa que se podría llamar a su Asterisk y hacer una llamada desde el mismo Asterisk como si se estuviera utilizando una extensión conectada con el mismo Asterisk dentro de la empresa.

✓ Grupos de extensiones (*Ring Groups*)

Esto define una extensión “virtual” que llama a un grupo de teléfonos simultáneamente, deteniéndose cuando uno de ellos es contestado. Esto es básicamente una versión simplificada de las colas para aquellos que no necesitan la funcionalidad extra de ellas.

✓ *Callback*

Esta opción consiste en llamar a un número que haya intentado previamente comunicarse con la empresa. La central devuelve el tono de ocupado a la llamada entrante y pasados los segundos fijados en “*delay before callback*” marca el número ingresado en “*callback number*” para establecer la comunicación anteriormente intentada por el llamante.

Una vez configurado el módulo, es necesario crear una ruta de entrada para que desvíe el número del llamante al *callback* configurado.

Esta función puede ser útil para el ahorro en llamadas a teléfonos móviles.

✓ Grabaciones de sistema (*System Recordings*)

Las grabaciones del sistema son utilizadas para los grupos de llamados y conferencias, para hacer anuncios, etc.

✓ IVR

Es la interface utilizada para configurar su auto-respuesta cuando las personas llaman a su empresa ó a su hogar. Normalmente está constituida por un saludo seguida de un menú de opciones como “Gracias por llamar a (nombre de la empresa), para ventas marque uno, para servicios marque dos”, etc.

---

Cuando se selecciona IVR, la primera página es una pequeña serie de instrucciones sobre cómo manejarlo. Se puede editar un IVR existente, ó crear uno nuevo.

✓ Salas de conferencia (*Conferences*)

Las conferencias son una facilidad preestablecida que está disponible como un destino.

✓ Horarios (*Time Conditions*)

Las condiciones de tiempo son módulos que parecen destinos cuando están instalados. Permiten condicionar las llamadas basándose en el horario. Hasta ahora su funcionamiento es muy básico y no funciona con adiciones ó exclusiones, pero se puede encadenar las condiciones de tiempo para simular condiciones.

✓ Destinos Misceláneos (*Misc Destinations*)

Los destinos misceláneos permiten usar cualquier cosa que marque desde una extensión estándar como un destino.

✓ Configuración de Contraseñas (*PIN sets*)

La configuración de contraseñas es un módulo que permite usar un rango de contraseñas en lugar de sólo una. Esto es únicamente utilizable por las troncales, pero puede ser usado en DISA ó en cualquier otra cosa que requiera el uso de contraseñas para la autenticación.

✓ Gestión de usuarios (*Administrators*)

Este módulo permite crear usuarios que podrán ingresar al FreePBX para modificar ciertas configuraciones a razón de los privilegios que se le hayan otorgado.

✓ Música en espera (*Music on Hold*)

Aquí se puede configurar los archivos de música en espera que serán reproducidos.

- **Herramientas de FreePBX**

En la siguiente gráfica se muestra las herramientas con las que cuenta FreePBX y la página donde se configuran:

The screenshot shows the FreePBX System Status page. On the left, a navigation menu is visible with 'Tools' highlighted in red. The main content area is titled 'FreePBX System Status' and includes several sections:

- FreePBX Notices:** A list of three notices: 'There are 3 modules available for online upgrades', 'Default SQL Password Used', and 'Default Asterisk Manager Password Used'. A 'show all' link is provided below the list.
- FreePBX Statistics:** A table showing call and channel statistics:
 

Total active calls	9
Internal calls	3
External calls	5
Total active channels	17
- FreePBX Connections:** A table showing connection counts:
 

IP Phones Online	11
IP Trunks Online	2
IP Trunk Registrations	2
- Uptime:** A section displaying system and Asterisk uptime:
  - System Uptime:** 2 days, 5 hours, 51 minutes
  - Asterisk Uptime:** 17 hours, 1 minute
  - Last Reload:** 0 minutes

At the bottom of the page, the FreePBX logo is displayed with the tagline 'Freedom to Connect' and the text: 'FreePBX is a registered trademark of Atengo, LLC. FreePBX 2.4.0 is licensed under GPL.'

Figura. II.24. Menú de herramientas de FreePBX

Fuente: [ELXCJM]



- 
- ✓ **Gestor Módulos (*Module Admin*)**

Administra los módulos que se usarán para configurar el Asterisk. Permite activar, desactivar y actualizar módulos con las últimas versiones disponibles en el sitio oficial de FreePBX.
  
  - ✓ **Asterisk *Logfiles***

Muestra las últimas dos mil líneas del archivo “log” ubicado en la carpeta `/var/log/asterisk`. En este archivo se guardan acciones generales realizadas por las extensiones y las llamadas entrantes. También registra los errores que pueden inducir a un malfuncionamiento, en el caso de encontrarse con un error.
  
  - ✓ **Asterisk API**

Este módulo permite configurar los permisos para las conexiones de algunos programas al Asterisk Manager. Asterisk manager permite que ciertos programas puedan conectarse a una instancia Asterisk para ejecutar comandos o leer eventos a través de un *stream* sobre TCP/IP. Las conexiones de programa-cliente se realizan mediante un usuario y contraseña. La configuración de estas conexiones se encuentra en el archivo `/etc/asterisk/manager.conf`.
  
  - ✓ **Asterisk CLI**

Esto permite correr comandos como si fueran ingresados en el Asterisk CLI (*Command Line Interface*). Se puede acceder con el comando `Asterisk -r` desde la consola de Linux. Para ver ciertos comandos generales y de mantenimiento del servidor revise la referencia [ADS2.0] en las páginas 53-55.

- ✓ *Asterisk Info*

Muestra la información de Asterisk, como: Versión de Software, *uptime* del sistema, canales activos, extensiones, troncales SIP y usuarios de correo de voz (*voicemail*), etc.

- ✓ Directorio telefónico (*Asterisk Phonebook*)

Con esta herramienta se conseguirá establecer una base de datos de fácil acceso y uso, que permite localizar efectivamente los nombres y números establecidos para el discado rápido.

- ✓ *Backup & Restore*

Se puede configurar una programación regular para *backup* para asegurarse de tener una copia de la configuración del Asterisk y FreePBX, correo de voz, y grabaciones en CD de sus grabaciones. Se puede también restaurar un estado previo al *backup* en el caso de pérdida de datos ó una masiva falla de configuración. Los *backups* son almacenados en */var/lib/asterisk/backups*.

- ✓ *PHP Info*

Muestra la información sobre el módulo de PHP instalado en el equipo.

- ✓ *Sys Info*

Muestra información completa sobre el sistema. Es importante controlar el espacio en el disco y el uso de la memoria en forma periódica para prevenir un malfuncionamiento del sistema.

- ✓ Java SSH

Muestra la consola del Asterisk en un navegador web en forma segura.

✓ *Edit Configuration*

La forma en que Asterisk deba trabajar está regulada por una serie de archivos de texto. Estos archivos guardan la información de las extensiones, colas, IVR y hasta de la configuración del hardware.

Esta herramienta permite editar los archivos de configuración ubicados en cuatro carpetas sin la necesidad de administrar la central desde la consola.

Estas carpetas son:

- */etc/asterisk*: Contiene los archivos de configuración del hardware y del plan de marcado.
- */var/www/html/panel*: Contiene los archivos para manejar el FOP<sup>10</sup> (*Flash Operator Panel*).
- */etc*: Contiene los archivos de la configuración del sistema en general.
- */tftpboot*: No contiene archivos por el momento.

Luego de editar alguno de estos archivos, es necesario hacer clic en “*re-read configs*” para que los cambios hagan efecto.

✓ *phpMyAdmin*

Es una interfaz gráfica web para administrar de forma sencilla la base de datos MySQL.

---

<sup>10</sup> **FOP** (*Flash Operator Panel*) El panel de operador es una aplicación para Asterisk PBX, funciona en un explorador de Internet. Es capaz de mostrar información sobre el funcionamiento de su PBX en tiempo real y hacer operaciones sobre las comunicaciones.

- ✓ *System Status*

Muestra el tiempo transcurrido desde el último inicio del sistema. Permite además reiniciar el sistema (*reboot*) o apagarlo (*shutdown*).

## 2.4 CONFIGURACIÓN DE ASTERISK CON FREEPBX

### 2.4.1 Extensiones SIP

En este punto indicaremos como crear extensiones en nuestra central basándonos en la herramienta FreePBX. Actualmente existen cuatro tipos de dispositivos que soporta Asterisk como SIP, IAX2, ZAP y personalizado (*custom*). En la página principal de FreePBX, al elegir la opción “*Extensions*” del menú *Setup* nos aparecerá opciones del tipo de dispositivo al que se le asignará la extensión, una breve descripción de las opciones de dispositivos es [ADS2.0]:

- ✓ SIP es “Protocolo de Inicialización de Sesiones”, para teléfonos IP o *softphones*.
- ✓ IAX2 es el “*Inter Asterisk Protocol*”, es un protocolo manejado por sólo unos pocos dispositivos como: teléfonos de base PA1688, el IAX2 y ATA
- ✓ ZAP es un hardware conectado a la central Asterisk, como: un TDM400, TE11P. Se utiliza para manejar teléfonos analógicos.
- ✓ Personalizada (*custom*) es un adaptador para cualquier dispositivo no estandarizado, como: H323.  
También puede ser usado para trazar un mapa de una extensión a un número externo.

Al elegir un dispositivo SIP la página que despliega FreePBX es:

### Añadir SIP Extensión

Añadir Extensiones

---

**Extensión**

**Nombre asociado**

**CID Num Alias**

**SIP Alias**

Opciones de la extensión

---

**Direct DID**

**DID Alert Info**

**Music on Hold**

**CallerID de Salida**

**Ring Time**

**Call Waiting**

**CID de emergencia**

Opciones del dispositivo

---

This device uses sip technology.

**secret**

**dtmfmode**

Configuración del Fax

---

**Extensión del Fax**

**Email del Fax**

**FaxDetection Type**

**Pausa después de responder**

Privacidad

---

**Aplicar Privacidad**

Language

---

**Language Code**

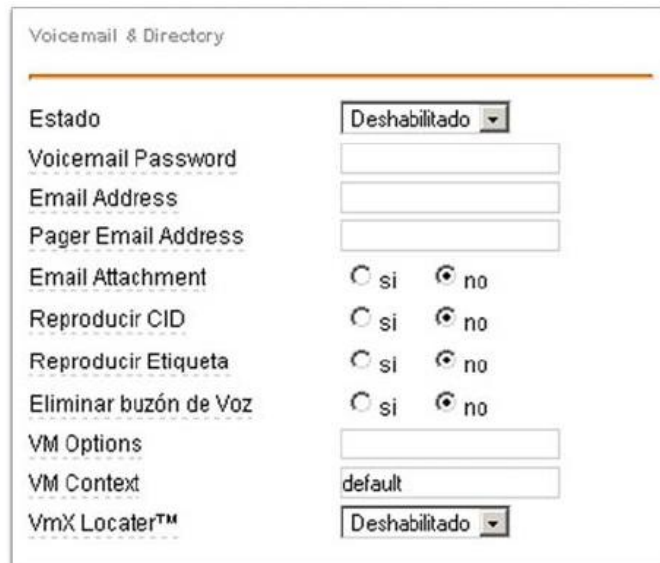
Recording Options

---

**Grabaciones entrantes**

**Grabaciones salientes**

Voicemail & Directory



Voicemail & Directory	
Estado	Deshabilitado
Voicemail Password	
Email Address	
Pager Email Address	
Email Attachment	<input type="radio"/> si <input checked="" type="radio"/> no
Reproducir CID	<input type="radio"/> si <input checked="" type="radio"/> no
Reproducir Etiqueta	<input type="radio"/> si <input checked="" type="radio"/> no
Eliminar buzón de Voz	<input type="radio"/> si <input checked="" type="radio"/> no
VM Options	
VM Context	default
VmX Locater™	Deshabilitado

Figura. II.25. Parámetros a configurar para un dispositivo SIP

Fuente: [ELXCJM]

- **Parámetros a Configurar**

#### Añadir Extensiones

- ✓ Extensión

El valor ingresado en este campo debe ser único. Este es el número que puede ser marcado desde otra extensión cualquiera, ó directamente desde la recepcionista digital (IVR), si está activado este módulo. Este número puede tener cualquier largo, pero convencionalmente es utilizado un número de tres ó cuatro dígitos.

- ✓ Nombre Asociado

Esto es el texto del identificador de llamada que se presenta a los que son llamados. Debe ser cualquier línea de texto ASCII, pero también acepta UTF-8<sup>11</sup>. Es importante mencionar que

<sup>11</sup> **UTF-8** "8-bit Unicode Transformation Format", es un formato de codificación de caracteres Unicode e ISO 10646 utilizando símbolos de longitud variable. Está definido como estándar por la RFC 3629.

algunos teléfonos tendrán problemas con nombres de varios bytes, por ejemplo los UTF-8, que pueden colapsar. Si se utiliza nombres de varios bytes, se recomienda experimentar de forma moderada y progresiva.

✓ *CID Num Alias*

El valor colocado en este campo, sobrescribe el ya configurado en “nombre asociado” solo cuando se realizan llamadas internas. Por ejemplo, puede colocar como alias el número de la cola a la que pertenece el interno y así, las llamadas devueltas, serán hacia la cola y no directamente al interno.

### Opciones de la Extensión

✓ *Direct DID*

Aquí se coloca el número directo (PSTN o VoIP) tal como es recibido por la central, al que se quiere asociar con este interno. Por ejemplo, si tenemos una línea con el número 2XXXXXX y las llamadas a este que suenen directamente en este interno, debemos colocar 2XXXXXX.

✓ *DID Alert Info*

Esta función es para configurar los *ringtones* de los teléfonos IP. No tiene efecto si no se configura un número en *Direct DID*.

✓ *Music on Hold*

Esta opción especifica que categoría de música usará este interno cuando necesite enviar música en espera a la parte

llamante. Las categorías se configuran previamente en el módulo “*Music on Hold*”

✓ *Outbound CID*

Debe ingresarse un valor con el formato “Nombre” <#####>. Este valor sobrescribirá el *Caller ID* de la línea cuando este interno realice una llamada al exterior.

✓ *Ring Time*

Aquí se configuran los segundos de espera antes de derivar al llamante al *voicemail*. La opción default toma el valor ingresado en el módulo “Configuraciones Generales”.

✓ *Call Waiting*

Permite activar o desactivar la función de llamada en espera para el interno que se está creando. Esta función puede activarse o desactivarse posteriormente utilizando \*70 o \*71 respectivamente.

✓ CID de emergencia

El valor que se ingrese en este campo, sobrescribirá todos los ajustes anteriores referentes a la identificación de llamada cuando se utilice una ruta saliente marcada como “Llamadas de emergencia”.

### Opciones del Dispositivo

✓ Contraseña (*secret*)

Esta es la contraseña (*password*) utilizada por el dispositivo telefónico para autenticarse al servidor de Asterisk cuando se



crea una extensión SIP o IAX2. Esto es generalmente configurado por el administrador antes de dar el teléfono al usuario, y comúnmente no conocida por el usuario. Si el usuario está utilizando un *softphone*, entonces necesita saber esta contraseña para configurar el programa.

### Configuración del Fax

✓ Extensión del Fax

Puede seleccionarse el interno donde se harán llegar las llamadas originadas desde un Fax. Si se selecciona FreePBX default, se utilizarán los valores indicados en el módulo “Configuraciones Generales”. Si en cambio, es seleccionado “sistema”, los faxes recibidos serán enviados por mail.

✓ Email del Fax

En este campo se indica el mail de destino de los faxes recibidos para el interno que se está creando. Esta función tiene efecto solo si el parámetro “Extensión del Fax” es configurado como “sistema”.

✓ *Fax detection type*

Seleccionar el modo en que Asterisk intenta determinar si el origen de la llamada es de un equipo de fax. Colocar *Zaptel* si el interno está vinculado a una línea directa *Zap*. Si está configurado para troncales VoIP (SIP o IAX2) colocar *NVFax*.

✓ Pausa después de responder

Ajustar el tiempo en segundo que desea reproducir el sonido de señal de fax al llamante.

---

## Privacidad

- ✓ Aplicar privacidad

Si la persona que llama no tiene identificador de llamada (número privado), se le pedirá que ingrese los 10 números de su número telefónico.

## *Language*

- ✓ *Language Code*

Aquí puede especificar en idioma de los paquetes de sonido que Asterisk utilizará para este interno. Por ejemplo, “en” para inglés, “es” para español, “it” para italiano, etc.

## *Recording Options*

- ✓ Grabación Entrante

Opciones para grabar las llamadas recibidas en la extensión.  
Existen tres opciones:

- Siempre
- Nunca
- A pedido (el usuario puede presionar ‘\*’1 para activarlo durante cualquier llamada)

- ✓ Grabación Saliente

Funciona de la misma manera que el anterior, pero con llamadas salientes.

---

## Buzón de Voz (*Voicemail*) y directorio

✓ Estado

Al seleccionar “habilitado”, se activa la casilla de mensajes para el interno que se está creado. Si la casilla ya estaba habilitada y se la deshabilita, se borrarán todos los valores de configuración ingresados.

✓ Contraseña del correo de voz (*voicemail password*)

Esta es la contraseña para acceder al sistema de correo de voz (*voicemail*). Puede ser cambiada por el usuario cuando ingresa en su buzón de voz marcando \*98. Para hacer esto, luego de ingresar, se debe presionar cero y luego cinco.

✓ Dirección de e-mail

Las direcciones a las que el correo de voz (*voicemail*) enviará las notificaciones cuando haya un nuevo correo almacenado.

✓ Dirección de e-mail de *pager*

Esta es la dirección de e-mail a la que se enviará una pequeña notificación al momento de registrarse un nuevo mensaje en la casilla (*voicemail*), adaptable para un servicio de e-mail a *pager*.

✓ Reproducir CID

Reproduce el número que llamó antes de reproducir el mensaje, e inmediatamente después anuncia la fecha y hora (*envelope*) en la que fue grabado el mensaje.

✓ Reproducir fecha y hora (*envelope*)

Esta opción controla si el sistema reproducirá ó no la fecha y hora (*envelope*) del mensaje antes de reproducir el mensaje. Esta configuración no tiene efecto sobre la operación de la opción de *envelope* en el menú “*advance*” del buzón de voz (*voicemail*).

✓ Borrar buzón de voz (*voicemail*)

Si está seleccionado en “*yes*” el mensaje será borrado de la casilla de correo de voz después de que se haya enviado por e-mail. Esta función provee la funcionalidad que le permite al usuario recibir su correo de voz únicamente por e-mail, en lugar de recuperar el mensaje desde la web ó la extensión.

Precaución: Se debe tener la configuración de e-mail en “*yes*” si no quiere que el sistema le envíe una notificación que diga “Usted tiene un correo de voz” e inmediatamente después borre el buzón de voz. Asegúrese de haber revisado *voicemail-to-e-mail* antes de activar este comando.

#### 2.4.2 Configuración de un *softphone* (X-Lite) como elemento final del PBX

Preámbulo:

Un *softphone* es un programa que emula un teléfono convencional para ser usado en una computadora. En general, utilizan los protocolos SIP o IAX2 de la misma manera que los utiliza Asterisk a la hora de crear extensiones. La única diferencia, es que el *softphone* trabaja como cliente y Asterisk como un proveedor del servicio de comunicación.

Existe una gran cantidad de compañías que ofrecen este software. Se pueden encontrar tanto versiones gratuitas como pagas para trabajar en distintas plataformas.

En nuestro caso, vamos a trabajar con el *softphone* X-Lite que es una herramienta gratuita para plataforma Windows desarrollado por *CounterPath*. Del sitio oficial (<http://www.counterpath.com>), puede descargarse la última versión de X-Lite, en este caso trabajaremos con la versión 3.0. Una vez descargado el archivo, basta ejecutarlo e instalarlo.

Ya instalado el *softphone* la interfaz gráfica que muestra es la siguiente:



Figura. II.26. Interfaz gráfica *softphone* X-Lite

Para ingresar una extensión SIP que sea propietaria de este teléfono nos colocamos en la opción “*SIP Account Settings*” que se encuentra en la parte superior como lo indica la siguiente figura:



Figura. II.27. Ingreso a configuración de extensión SIP en X-Lite

Seguido de esto se desplegará la siguiente ventana:

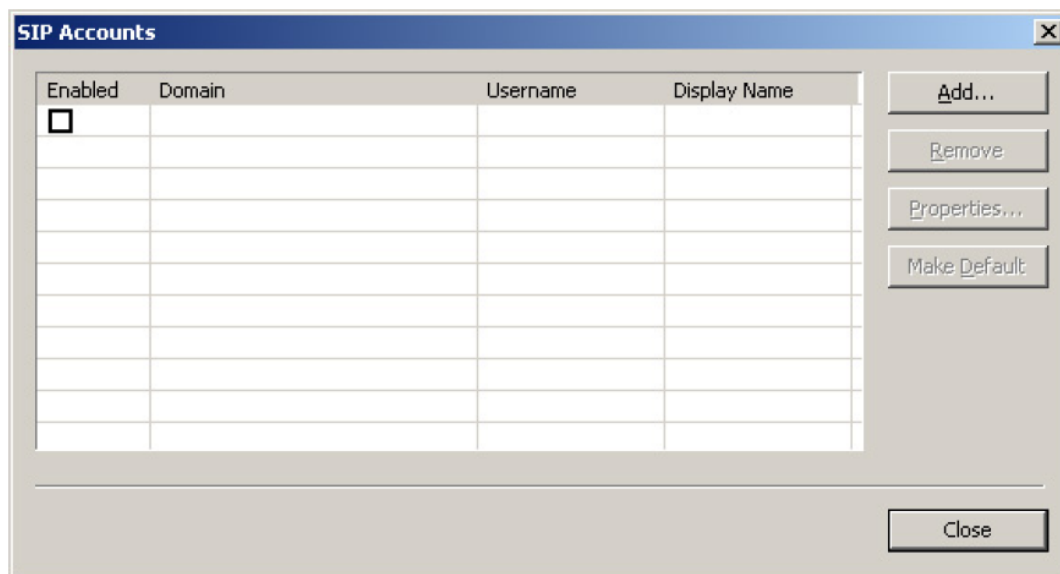


Figura. II.28. Ventana de configuración en X-Lite

---

Hacemos clic en “*add*” para así configurar X-Lite con los datos correspondientes a la extensión SIP creada previamente en Asterisk.

A continuación se llenan los campos que aparecen en la siguiente ventana de la siguiente manera:

- ✓ *Display name*: El nombre que se envía como CID.
- ✓ *User name*: El número de la extensión que se configuró previamente en Asterisk.
- ✓ *Password*: El mismo valor ingresado en el campo “*secret*” cuando se configuró la extensión.
- ✓ *Authorization user name*: El mismo valor ingresado anteriormente en “*user name*”.
- ✓ *Domain*: La dirección IP de la central Asterisk, es decir, colocamos la IP privada correspondiente de nuestra red. En el caso que el usuario esté fuera de la oficina, este campo debe ser completado con la IP pública otorgada por el ISP.

La página que se despliega con los campos llenos es la siguiente:

**Properties of Account 1**

Account | Voicemail | Topology | Presence | Advanced

User Details

Display Name: José

User name: 116

Password: ●●●●●

Authorization user name: 116

Domain: 192.168.0.1

Domain Proxy

Register with domain and receive incoming calls

Send outbound via:

domain

proxy Address: \_\_\_\_\_

target domain

Dialing plan: #1\a\a.T;match=1;prestrip=2;

Aceptar Cancelar Aplicar

Figura. II.29. Ventana de propiedades para extensión SIP en X-Lite

Aceptamos y volvemos automáticamente a la pantalla anterior donde vemos la extensión que acabamos de configurar, finalmente hacemos clic en “close”.



La versión pagada de X-lite nos permite manejar varias extensiones a la vez.

Es importante mencionar que en el caso de tener un firewall activo tenemos que desbloquear la aplicación X-Lite para que pueda comunicarse la central con el *softphone*.

Si la configuración fue realizada con éxito tanto en el *softphone* como en la central, Asterisk reconocerá la extensión, y en el teléfono gráfico se mostrará un mensaje “Ready” junto con el “Username” asignado.



Figura. II.30. *Softphone* X-Lite registrado

### 2.4.3 Configuración del teléfono Siemens *OptiPoint* 150 S como elemento final del PBX

La gama de terminales *OptiPoint* en su conjunto forman parte del tradicional sistema de comunicaciones *HiPath* de Siemens.

El teléfono *OptiPoint* 150 S es un modelo básico de coste-efectiva para telefonía de voz sobre IP. Se puede acceder a todas las funciones clave a través de teclas de función.

Está diseñado para una perfecta integración en el sistema de comunicación Siemens *HiPath* 8000 y otras plataformas estándar basadas en SIP como Asterisk para soluciones de comunicación en la empresa y el mercado móvil. Este teléfono SIP cuenta con funciones clave como conferencia, altavoz, silencio (*mute*), volver a marcar (*redial*) y el protocolo STUN<sup>12</sup>. Soporta *codecs* de audio como G.711 a/ $\mu$ -law, G.723.1A, G.729 AB. Es fácil de operar y fácil de administrar [OPT150].

A continuación se muestra el teléfono y las partes que lo conforman:

---

<sup>12</sup> **STUN** “*Simple Transversal of UDP over NATs*”, es un protocolo de red del tipo cliente/servidor que permite a clientes NAT encontrar su dirección IP pública, el tipo de NAT en el que se encuentra y el puerto de Internet asociado con el puerto local a través de NAT.



**Figura. II.31. Partes del teléfono OPTIPOINT 150 S<sup>13</sup>**  
Fuente: [OPT150]


La configuración del teléfono se puede realizar de diferentes maneras, ya sea por el teléfono directamente o por un explorador de internet accediendo de forma remota al teléfono para su administración a través de su dirección IP.



En nuestro caso nombraremos la manera más fácil a nuestra forma de ver: La configuración la hacemos accediendo al teléfono a través de un explorador, para ello primeramente necesitamos que el teléfono tenga una dirección IP que sea parte de nuestra red local. Este proceso lo

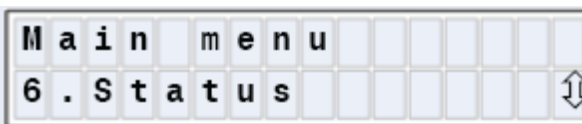
<sup>13</sup> Traducido del manual de usuario del teléfono OPTIPOINT 150 S. Página 10.  
Enlace: [http://wiki.siemens-enterprise.com/images/4/49/User\\_Manual\\_optiPoint\\_150\\_S.pdf](http://wiki.siemens-enterprise.com/images/4/49/User_Manual_optiPoint_150_S.pdf)

realizaremos a través de las teclas del teléfono, para ello seguimos el siguiente proceso:

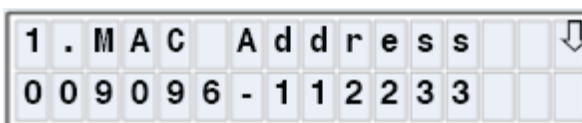
- **Configuración de la dirección IP**


Inicialmente presionamos la tecla Menú  que se encuentra dentro de las teclas de función indicadas previamente.

Usamos la tecla  hasta llegar a la opción “6.Status” y presionamos la tecla  para confirmar la selección:



Ya en esta opción se nos indicará la configuración del equipo, en primer lugar la dirección MAC del equipo:



Presionamos la tecla  para avanzar a la opción “2.IP Address” donde nos muestra la IP del equipo. Si deseamos cambiar la



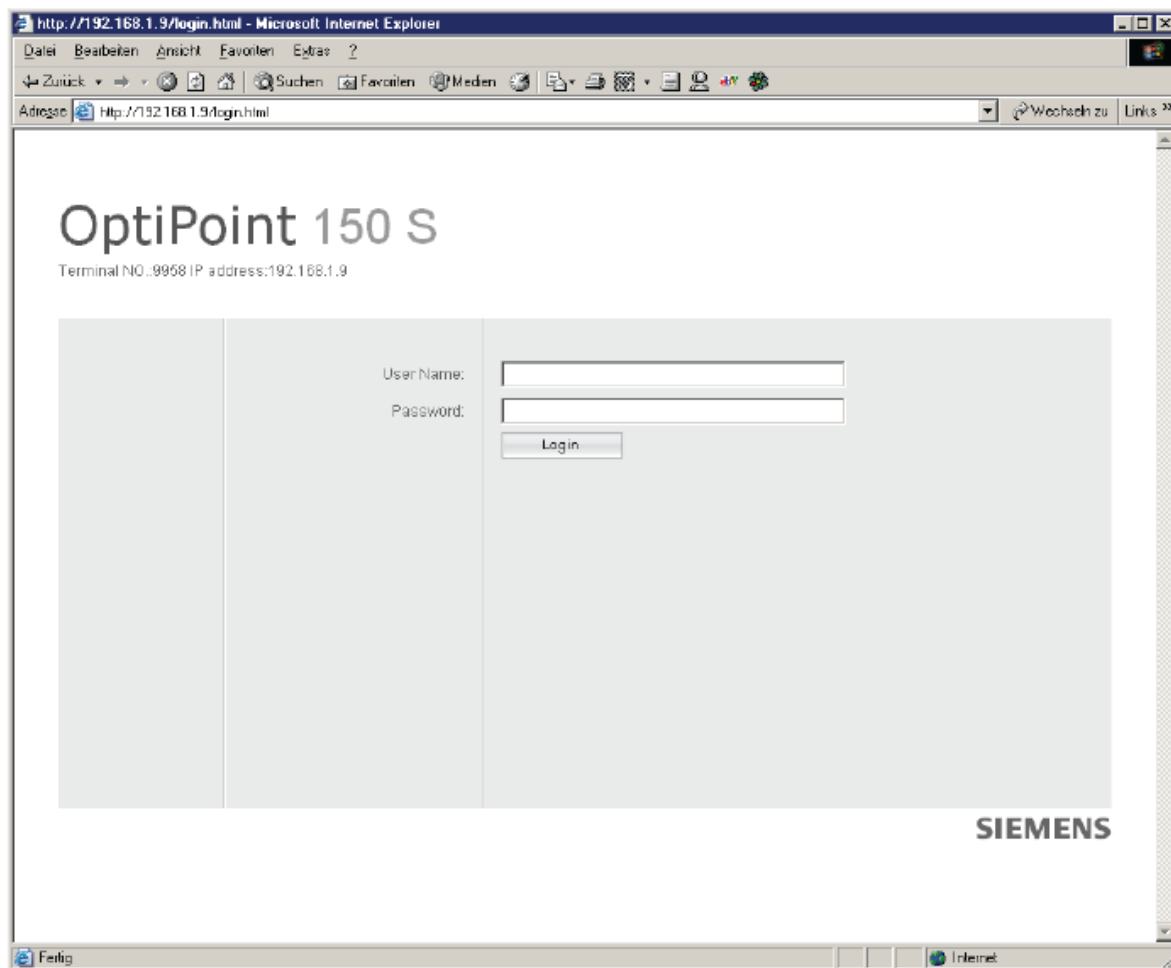


Figura. II.32. Página principal del teléfono *OptiPoint 150 S* vía web

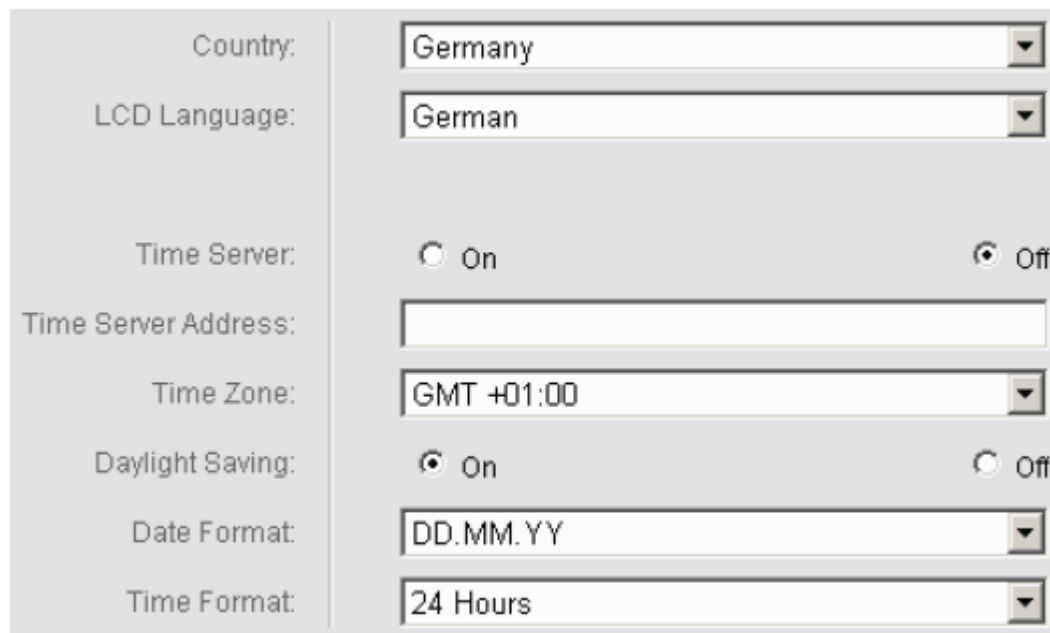
Por defecto de fábrica el teléfono *OptiPoint 150 S* tiene como *user name* “*admin*” y como *password* “*123456*”.

Ya en la administración del teléfono podemos modificar todas las configuraciones del teléfono. Las más importantes para nuestro fin son:

- **Lenguaje**

De fábrica el teléfono viene con estándares de Alemania, los cuales no concuerdan con los de nuestro país. Estos estándares son los tiempos de tono y marcado, etc. Para solucionar esto en

la opción “*Localization*” modificamos el país y el lenguaje LCD a un país disponible con estándares parecidos al nuestro como Brasil.

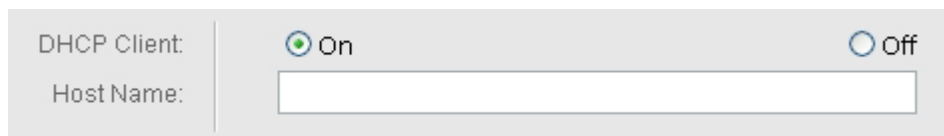


Country:	Germany
LCD Language:	German
Time Server:	<input type="radio"/> On <input checked="" type="radio"/> Off
Time Server Address:	
Time Zone:	GMT +01:00
Daylight Saving:	<input checked="" type="radio"/> On <input type="radio"/> Off
Date Format:	DD.MM.YY
Time Format:	24 Hours

Figura. II.33. Cambio de Idioma del teléfono *OptiPoint 150 S*

#### o Red

En la opción “*Network*” vamos a la interface y configuramos al teléfono para que funcione bajo el protocolo DHCP (direccionamiento IP dinámico) o direccionamiento IP fijo.

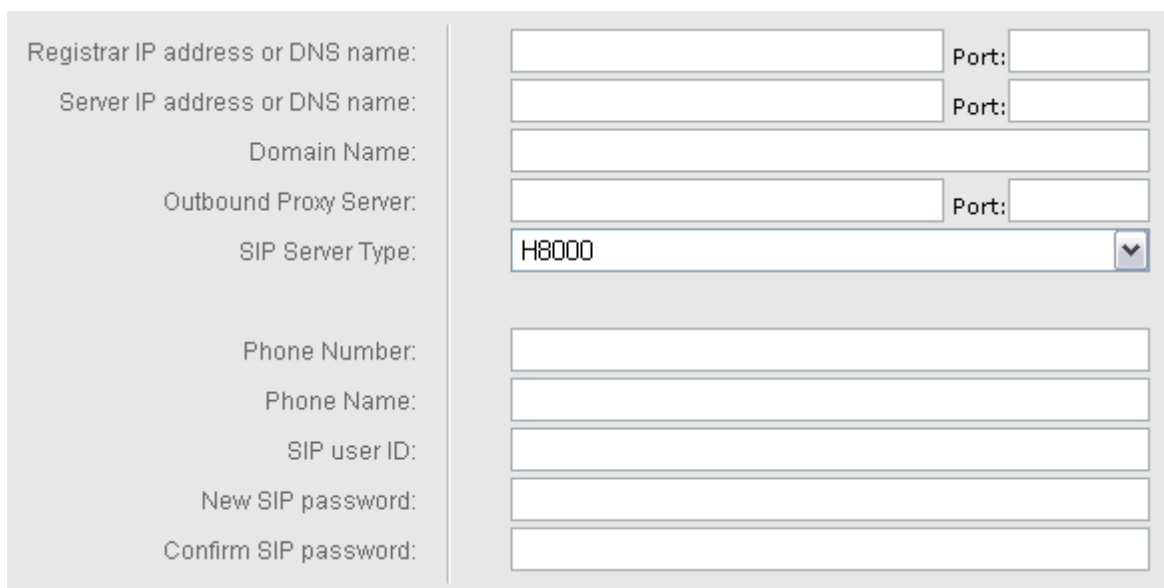


DHCP Client:	<input checked="" type="radio"/> On <input type="radio"/> Off
Host Name:	

Figura. II.34. Configuración de red del teléfono *OptiPoint 150 S*

- **Configuración SIP**

Al igual que en el *softphone* necesitamos configurar la extensión del dispositivo SIP que previamente se creó en la central Asterisk. Para ello vamos a la opción “*SIP Setup*” y luego a “*SIP Settings*”, la siguiente página se desplegará:



Registrar IP address or DNS name:	<input type="text"/>	Port:	<input type="text"/>
Server IP address or DNS name:	<input type="text"/>	Port:	<input type="text"/>
Domain Name:	<input type="text"/>		
Outbound Proxy Server:	<input type="text"/>	Port:	<input type="text"/>
SIP Server Type:	H8000 <input type="button" value="v"/>		
Phone Number:	<input type="text"/>		
Phone Name:	<input type="text"/>		
SIP user ID:	<input type="text"/>		
New SIP password:	<input type="text"/>		
Confirm SIP password:	<input type="text"/>		

**Figura. II.35. Configuración de red del teléfono *OptiPoint 150 S***

- ✓ *Registrar IP address or DNS name*: Dirección IP o *host name* del “SIP Registrar”<sup>14</sup>. Por lo general se coloca la dirección de nuestro servidor Asterisk.
- ✓ *Server IP address or DNS name*: Dirección IP o *host name* del servidor SIP.
- ✓ *Domain Name*: Nombre del dominio del proveedor de VoIP.

<sup>14</sup> **SIP Registrar** es un servidor en una red SIP que acepta y procesa los requerimientos SIP REGISTER. Provee un servicio de localización el cual registra una o más direcciones IP a un cierto SIP.



- 
- ✓ *Outbound Proxy Server*: Dirección IP o *host name* del servidor proxy en caso de que se posea.
  - ✓ *Phone Number*: Se ingresa el SIP ID, es decir, la extensión asociada a este dispositivo.
  - ✓ *Phone Name*: Se ingresa CID o Nombre Asociado, es decir, el nombre que aparecerá en el identificador de la persona llamada.
  - ✓ *SIP user ID*: Se ingresa el SIP ID, es decir, la extensión.
  - ✓ *New SIP password*: Se ingresa la clave especificada para la extensión.
  - ✓ *Confirm SIP password*: Se ingresa nuevamente la clave para confirmar.

Finalmente se guardan las configuraciones realizadas y se reinicia el teléfono automáticamente para aplicar los cambios.

- **Extensión SIP**

Los siguientes campos son importantes para la configuración de la extensión SIP:

Register Expires:	<input type="text" value="1800"/>	(SECS)
Session Timer:	<input type="text" value="0"/>	(SECS)
Min-SE:	<input type="text"/>	(SECS)
RTP Base Port:	<input type="text" value="8000"/>	
SIP Transport:	<input checked="" type="radio"/> UDP	<input type="radio"/> TCP
PRACK Support:	<input checked="" type="radio"/> ON	<input type="radio"/> OFF
Appearance Line-Seize Support:	<input type="radio"/> ON	<input checked="" type="radio"/> OFF

**Figura. II.36. Configuración de extensión SIP del teléfono *OptiPoint 150 S***

- ✓ *Register Expires*: El tiempo que el teléfono espera por la confirmación de conexión por el servidor IP después del marcado. Si el tiempo termina antes que la conexión sea establecida, el teléfono termina la acción y marca otra vez. Se recomienda 3600.
- ✓ *Session Timer*: Intervalo de tiempo para el refrescamiento de la conexión con el servidor. Si el valor es muy alto, se puede desconectar. Recomendado 180.
- ✓ *Min-SE*: Duración mínima de una duración SIP en segundos. Recomendado 100.
- ✓ *RTP Base Port*: Número de puerto IP para ser usado para el envío de paquetes de voz. Este depende de la configuración de su servidor, en nuestro caso esta un rango de puertos del 10000 al 20000.
- ✓ *SIP Transport*: Protocolo de transporte utilizado, en nuestro caso es UDP.

- ✓ *PRACK Support*: PRACK asegura el intercambio de información media SIP (*codecs*) ya antes de la conexión.

Se guarda la configuración y automáticamente el teléfono se reiniciará para guardar los cambios.

Los otros parámetros no necesitan ser tomados en consideración y pueden permanecer con la configuración de fábrica.

## **CAPÍTULO III**

### **3 DISEÑO DEL IVR**

#### **3.1 ANÁLISIS DE PROBLEMAS COMUNES EN TELEFONIA DENTRO DE LOS CLIENTES DE HIGHTELECOM**

HIGHTELECOM es una empresa que tiene dos frentes de mercado, uno de Telefonía siendo distribuidor autorizado de la marca SIEMENS en lo referente a centrales telefónicas y todos los dispositivos anexos a esta rama. Se encarga de la comercialización e instalación de todos estos dispositivos.

El otro frente de mercado es el de datos, donde comercializan una herramienta gráfica desarrollada por ellos llamada NetCyclon que se encarga de la administración total en lo referente a redes, seguridad y otras prestaciones por medio de servidores proxy, DNS, de correo, firewall, etc. En esta herramienta no se maneja por el momento nada referente a telefonía y VoIP. Es este estudio el que dará inicio a la telefonía dentro del sistema NetCyclon.

Es aquí donde la empresa quiere apuntar, nuevos clientes que adquieran el sistema NetCyclon y a la vez tener su central telefónica de VoIP, aumentando así las prestaciones y calidad del sistema. Con esta nueva aplicación estaremos realizando ya una convergencia de servicios tanto voz como datos sobre una misma red, lo cual permite disminuir el gasto de la empresa y optimizar los recursos.

### 3.1.1 Encuestas

Los clientes de HIGHTELECOM están conformados por todo tipo de empresas e instituciones tanto públicas y privadas bien conformadas y organizadas.

La encuesta fue realizada a 20 clientes potenciales de HIGHTELECOM. Se buscó hacer algo sencillo de corto tiempo y que cumpla con nuestras expectativas. La encuesta constaba de dos partes:

- ✓ La primera era sobre los limitantes que tenía la empresa con su sistema telefónico.
  - Primero se pedía al cliente que de una lista de limitantes planteadas escoja el que más se ajustaba a su situación.
  - Segundo se pedía que nos diga la mayor limitante que tiene con su central.
  
- ✓ La segunda parte era sobre requerimientos.
  - Primero se pedía que nos diga los requerimientos que necesita de una central telefónica.
  - Segundo, se planteaba el caso de tener implementado un sistema IVR completo, es decir, con acceso a bases de datos, etc. Se pedía dar los requerimientos que le haría a este sistema, dependiendo de las necesidades de su empresa.

- **Limitantes de las centrales telefónicas**

Tanto a los clientes de datos como de telefonía se les planteo la siguiente lista de limitantes:

- ✓ Alto costo
- ✓ Baja calidad
- ✓ Difícil administración
- ✓ Pocos servicios y aplicaciones
- ✓ Difícil mantenimiento
- ✓ Dispositivos extras

De estas opciones cada cliente escogió la que más se ajusta a su realidad y los resultados mostrados son los siguientes:

**Tabla. III.9. Datos de la encuesta-Limitantes de los clientes HIGHTELECOM**

LIMITACIÓN	Nº CLIENTES
Alto costo	7
Dispositivos extras	5
Difícil administración	3
Pocos servicios y aplicaciones	3
Difícil mantenimiento	1
Baja calidad	1
<b>TOTAL</b>	<b>20</b>

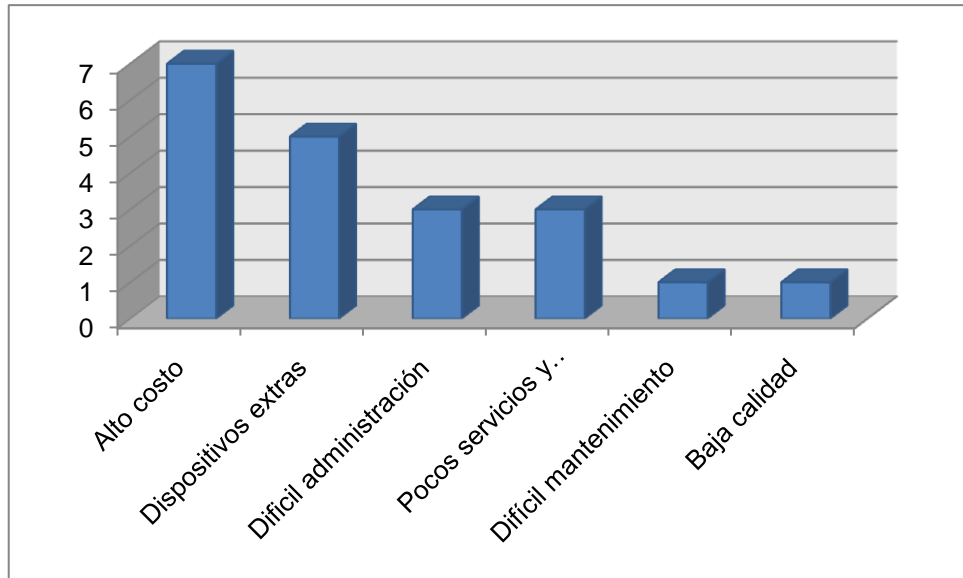


Figura. III.37. Gráfico en barras del resultado de la encuesta

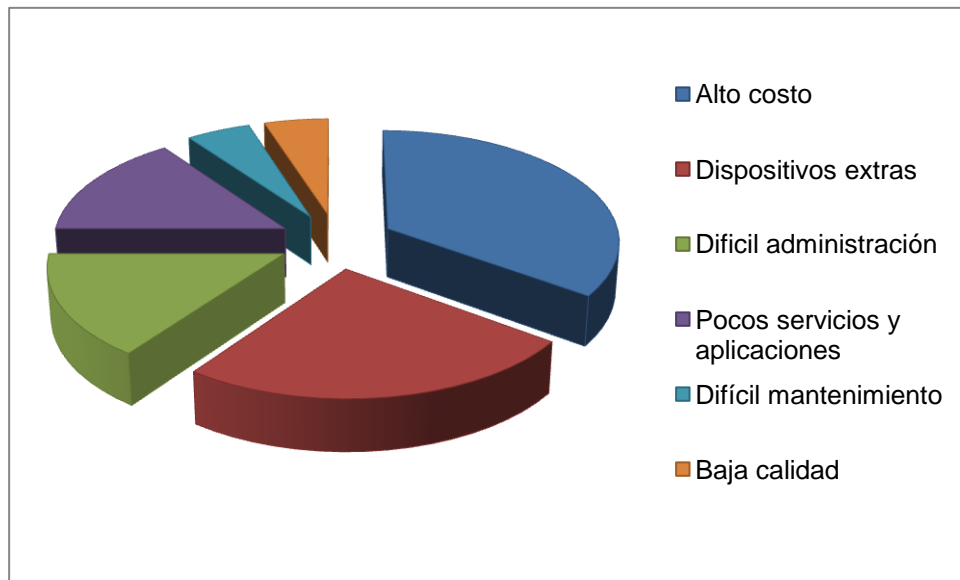


Figura. III.38. Gráfico en pastel del resultado de la encuesta

Las limitaciones nombradas por los clientes se las ha diferenciado entre clientes de datos y telefonía.

Por un lado los clientes de telefonía han adquirido una central SIEMENS que administra en su totalidad todo lo relacionado con voz. HIGHTELECOM fue el encargado de la venta y es el encargado del mantenimiento y administración de la central.

Las limitantes nombradas por este tipo de clientes son:

- ✓ El aspecto económico es una de las limitantes más grandes que tienen este tipo de centrales, debido a que su costo inicial es muy alto al igual que los dispositivos anexos como tarjetas de expansión digitales y analógicas, teléfonos, etc.
- ✓ Se necesita una persona dedicada para el mantenimiento y administración de este tipo de centrales, ya que su dificultad es muy alta para una persona que no tenga conocimientos básicos sobre centrales. Además al poner una persona dedicada se incrementa el gasto de la empresa.
- ✓ Tiene sus extensiones tanto analógicas como digitales limitadas, es decir, inicialmente la central viene con 8 extensiones analógicas y 4 digitales (dependiendo de la central) en la tarjeta madre y su expansión se hace por hardware mediante tarjetas de expansión que son muy costosas y estandarizadas, es decir, solo existen tarjetas de expansión de 4, 8 y 24 extensiones. Además los slots para las tarjetas de expansión son limitadas por la arquitectura de cada central.
- ✓ El equipo como central físicamente necesita un área de trabajo amplia donde exista una muy buena ventilación y un



ambiente frío. Además las conexiones a realizar desde la central a los teléfonos deben ser realizadas muy ordenada y cuidadosamente, caso contrario se corre el riesgo de cortar los cables que son muy pequeños de alguna extensión.

- ✓ Para realizar aplicaciones adicionales como IVR's dentro de estas centrales, dependerá de la central que se maneje, ya que las más económicas como la Hípate 1200 no maneja estas aplicaciones y las siguientes a estas como la Hípate 3500 necesitan dispositivos adicionales que incrementan su costo. Centrales como la *Hipath* 3800 y 4000 ya manejan estos servicios pero su costo es mucho mayor.
- ✓ Estas centrales no tienen la capacidad de acceder a ningún tipo de información como bases de datos.
- ✓ Al no tener un IVR y un sistema que pueda acceder a bases de datos, se gasta dinero en impresiones y en envíos para que los clientes conozcan de información propia de ellos como saldos y nuevos servicios a ofrecer.
- ✓ No se puede brindar un servicio de información de 24 horas para los clientes.

Con respecto a los clientes de datos, el sistema NetCyclon les ayuda a administrar por completo su red pero con respecto a telefonía tienen centrales privadas de marcas como Panasonic, Siemens, Alcatel entre otras, que presentan limitaciones semejantes a las antes mencionadas.

Existen algunos clientes de datos que como central telefónica poseen en otros equipos centrales de voz sobre IP. Las limitantes que presentan estos clientes son:

- ✓ La calidad de voz deja mucho que desear.
- ✓ El hardware utilizado no es inmune al ruido y causa mucha interferencia.
- ✓ La mala administración y configuración causa problemas en la central haciéndola colgar y reiniciar.
- ✓ La administración de la central se la hace únicamente por consola lo que aumenta su dificultad. No se tiene un administrador gráfico que ayude con la configuración.
- ✓ El IVR no permite hacer consulta a bases de datos, lo cual disminuye su funcionalidad.

Todas las limitaciones mencionadas son las que cada cliente nombro y en los casos que coincidían, se formuló una idea general.

- **Requerimientos de telefonía**

Todos los clientes de HIGHTELECOM buscan siempre abaratar costos sin dejar a un lado la calidad. La telefonía en una empresa es un aspecto muy importante y fundamental ya que de esta depende la comunicación de todo su personal, la de sus clientes hacia su empresa y viceversa. Con la telefonía se pueden brindar varios servicios como consultas, compras, modificaciones, etc. que dan mayor comodidad al cliente y es un punto a favor para la empresa.

---

Basándonos en los problemas que tienen los clientes en telefonía y en sus necesidades, nos han planteado los siguientes requerimientos:

- ✓ Una central telefónica que no sea costosa ni su arquitectura ni sus dispositivos.
- ✓ Una central que brinde una buena calidad de voz, y que sus servicios sean confiables y seguros.
- ✓ Una central que no tenga problemas ni limitantes con las extensiones.
- ✓ Una central que sea fácil de administrar y configurar con herramientas gráficas que faciliten esta labor.
- ✓ Una central que preste la mayor cantidad de servicios como IVR, correo de voz, llamada en espera, identificador de llamadas, etc.
- ✓ Una central con la que se pueda tener acceso a bases de datos y permita que los clientes hagan consultas de sus datos y de información a la cual tengan acceso.
- ✓ Una central que optimice los recursos y que pueda utilizar el mismo cableado de red para no tener que aumentar un nuevo cableado en su empresa.
- ✓ Una central a la que se pueda acceder remotamente para su administración en forma de VPN o vía web.

- **Requerimientos del IVR**

Los requerimientos expuestos para el IVR planteado fueron:

- ✓ Hacer el reconocimiento de un usuario mediante una clave, con validaciones de intentos.
- ✓ Tener acceso a la una base de datos que contiene información sobre el cliente y que este pueda acceder a preguntar por ejemplo el saldo que tiene en la empresa.
- ✓ Dar a conocer productos y servicios que ofrece la empresa.
- ✓ Permitir hacer reservas y compras por medio del teléfono.
- ✓ Acceder a una base de datos que dependiendo del producto muestre que disponibilidad tiene la empresa actualmente, es decir, la cantidad en stock.
- ✓ Permitir conectarse con un ejecutivo de servicio al cliente.
- ✓ Mostrar el día y hora disponible que la empresa tiene para realizar un servicio al cliente.
- ✓ Conocer el valor que un cliente debe a la empresa y su fecha límite para el pago.
- ✓ Permita acceder a cualquier extensión de la empresa.
- ✓ Un tema que contenga la mayor información de la empresa como origen, nombres de directivos, etc.

- ✓ Dependiendo del usuario pueda dejar mensajes de voz en cierta extensión.
- ✓ En el caso que el cliente no tenga respuesta, tener grabaciones que expliquen al cliente la causa y darle soluciones para ser atendido.
- ✓ Poner a disposición la mayor cantidad de información que el cliente pueda requerir y pueda tener acceso.
- ✓ Permitir en el caso de un colegio que los estudiantes consulten sus notas por el teléfono.

### **3.2 DEFINICIÓN DEL PROBLEMA A RESOLVER POR MEDIO DEL IVR**

Al haber recolectado los mayores requerimientos de los clientes de HIGHTELECOM vamos a formular un problema en general, tratando de abarcar la mayor cantidad de requerimientos expuestos.

El IVR se enfocará a una empresa que brinde diferentes servicios y a la vez que tenga cuentas por cobrar a sus clientes.

#### **3.2.1 Problema**

La empresa en cuestión será hipotéticamente una empresa distribuidora de computadores y equipos electrónicos. Al ser un distribuidor mayoritario la empresa brindará créditos a sus clientes y ofrecerá promociones a sus clientes registrados.

La empresa necesita que los usuarios por medio del teléfono puedan conocer las promociones que la empresa brinda, la cuenta a pagar que tiene cada cliente y que nuevos usuarios puedan crear una cuenta con la empresa. Además de administrar y redireccionar las llamadas entrantes a los respectivos departamentos. El requerimiento se da ya que no quieren tener una persona dedicada que realice este trabajo.

Para que el usuario conozca su cuenta a pagar y las promociones que la empresa brinda necesita ser identificado. La identificación se la realizará por medio de la cédula y una clave personal de cuatro dígitos. Esta clave inicialmente se la dará la empresa pero para mayor seguridad el usuario podrá cambiar su clave y personalizarla.

Para la creación de nuevas cuentas el usuario ingresará su cédula y su clave, después de esto se enviará a una operadora que le permitirá llenar sus datos personales como nombres, dirección teléfono, pedido si existe, etc.

Todas estas opciones se complementarán con funciones básicas como acceder a extensiones de los diferentes departamentos si el usuario lo requiere y dar la opción de conocer sobre los servicios que ofrece la empresa y sobre algunas políticas y características de la empresa.

### **3.2.2 Alcances y Limitaciones del Proyecto**

Esta empresa al tener varios servicios nos permitirá cubrir la mayor cantidad de requerimientos.

Los clientes de esta empresa estarán identificados por su cédula y una clave que la empresa le proporcionará. Dicha clave puede ser personalizada y cambiada a través de una llamada telefónica, almacenando su nueva clave en la base de datos.

Con esto al realizar una llamada el usuario será identificado y podrá revisar su cuenta a pagar y las promociones disponibles que la empresa le brinda por ser cliente. En este caso tenemos un acceso a la base de datos en doble sentido, es decir, se leerán datos al identificar si la cédula ingresada esta en nuestra base de datos y se introducirán nuevos datos al cambiar la clave del usuario.

Al crear nuevos usuario e ingresar su cédula y clave, se estarán ingresando también nuevos datos a nuestra base. Esta operación reducirá a la operadora la adquisición de datos a los nuevos usuarios, además de tener ya configurada la clave personalizada del usuario.

Se podrá también acceder directamente a una operadora de servicio al cliente o a extensiones de los diferentes departamentos de la empresa. El IVR ofrecerá también una opción que le permitirá al usuario conocer de la empresa, de sus servicios y de su manera de trabajo.

El IVR responderá a tonos de marcado multifrecuencia DTMF<sup>15</sup> y obtendrá la información de su base de datos y la proporcionará al usuario en forma de voz por medio de la herramienta TTS (*Text To Speech*).

---

<sup>15</sup> **DTMF** (*Dual-Tone Multi-Frequency*), es una identificación de las teclas de un teclado de teléfono, cuando se presiona una tecla, se envían dos tonos de distinta frecuencia, uno por columna y otro por fila en la que esté la tecla. La central descodifica a través de filtros especiales, detectando instantáneamente que dígito se marcó.

### 3.3 DISEÑO DE UN IVR GENERAL

El 95% del éxito del diseño de los sistemas IVR Asterisk se basa en el funcionamiento y en el poder de satisfacer a la empresa con todas sus necesidades para que los usuarios finales se sientan más cómodos y satisfechos con los servicios de la empresa. El 5% restante es solo programación, pero de este depende la estructura y organización del sistema IVR para su éxito. Aún así se presentan errores de diseño y de ergonomía como:

- ✓ Menús demasiado largos
- ✓ La pérdida de los usuarios en el sistema.
- ✓ Los usuarios no encuentran lo que necesitan y no comprenden cómo usar el sistema
- ✓ Tiene lógica confusa y "callejones sin salida"
- ✓ Ineficiente recuperación ante fallos
- ✓ No existe suficiente canales de atención
- ✓ Carencia de salida de un submenú hacia un operador humano

Por estos problemas citados que pueden marcar el fracaso de un sistema IVR se toma en cuenta ciertas recomendaciones para el diseño.

#### 3.3.1 Recomendaciones de diseño

Primeramente se tienen en cuenta aspectos como el dimensionamiento de líneas entrantes y la cantidad óptima de menús. [VOXDAT]

- Dimensionamiento adecuado de líneas entrantes  
Basándonos en el libro "La teoría de las probabilidades y las conversaciones telefónicas" de *A.K. Erlang*. Para el



dimensionamiento de las líneas entrantes el autor sugiere tomar en cuenta 3 factores:

- Tráfico ofrecido: cuánto tráfico de llamadas quiere acceder a la central.
  - Recursos de red: cantidad de llamadas entrantes estamos listos para atender.
  - Calidad o grado de servicio: porcentaje de llamadas no podremos atender.
- Cantidad óptima de menús

Un estudio realizado por George Miller titulado como "El número mágico siete, más o menos dos: Algunos límites sobre nuestra capacidad de procesar información", documenta el hecho que la mayoría de las personas sólo pueden procesar alrededor de siete ítems de información en forma simultánea. Dependiendo del tipo de información y de características del propio individuo, a veces esta habilidad se reduce a cinco y en algunos pocos casos se eleva hasta nueve. La mayoría de los estudios realizados en esta dirección se enfocaban en responder o recordar estímulos visuales, por lo tanto las limitaciones de memoria a corto plazo son más evidentes ante estímulos auditivos y presentados en forma secuencial.

Aunque generalmente se adoptan menús de cinco ítems, se recomienda una media óptima de tres alternativas por cada menú. Recordemos que este tipo de interfaz es lenta por naturaleza y la información desaparece una vez expuesta, lo que provoca cierto nivel de fatiga y ansiedad en el usuario final.

Siguiendo con las recomendaciones de diseño se tiene algunos puntos específicos:

- Principios Generales
  - Los menús no tienen que reproducir la estructura organizativa departamental de la empresa (eso es genérico para cualquier tipo de interfaz, sea de voz o no)
  - Si el servicio tiene alguna restricción, es preciso avisar al usuario inicialmente.
  - Proporcionar sólo la información que se necesita y de la forma más simple posible
  - Diseñar para la mayoría de los usuarios evitando que tengan que recorrer opciones que sólo interesan a unos pocos.
  - Hacer que el sistema trabaje por el usuario, es decir, no pidiendo el mismo dato varias veces y evitando preguntar información que el sistema puede conocer de sus bases de datos.
  - Ser consistente y no cambiar a menudo las opciones de los menús.
  
- Saludo inicial
  - La bienvenida debe ser breve.
  - No repetir el saludo inicial en caso que el usuario vuelva al inicio.
  - Evitar mensajes promocionales y en caso de haberlos sólo si el usuario los puede encontrar justificados y, sobretodo, han que ser breves permitiendo acceder a la información en aquel mismo momento.

- 
- Evitar remitir al usuario a otro servicio.
- 
- Menú principal
    - El primer menú debe ser el principal a no ser que se tenga que pedirse identificación.
    - Máximo debe tener 20 segundos de duración total, contando el saludo inicial y el menú principal.
- 
- Identificación
    - Pedir identificación sólo cuando sea necesaria.
    - Pedir el identificador más fácil de recordar.
    - Aceptar que el usuario dé cadenas variables de dígitos como identificador.
    - Gestionar el final de la entrada de datos por medio de *timeout*, no solicitar ningún tecleo adicional
- 
- Menús
    - Deben estar provistos de un breve recordatorio como el título del menú que indicará al usuario donde se encuentra y de qué tratarán las opciones que escuchará.
    - Debe existir máximo 4 opciones por menú, si se utiliza la respuesta por tonos.
    - Debe existir máximo 3 pasos para hacer una tarea.
    - Poner las opciones de navegación al final de todo.

- 
- No hay que cortar nunca la comunicación si el usuario se equivoca. Es preciso dar caminos alternativos
  - Ordenar las opciones de las más específicas a las más generales
- 
- Opciones de navegación recomendables
    - Volver atrás
    - Menú principal
    - Repetir
    - Ayuda (sólo si lo necesita la aplicación)
    - Si existe una opción de paso al operador, debe ser la última.
- 
- Preguntas Sí/No
    - Se usan para recuperar errores y para confirmar acciones
    - Tienen que ser preguntas directas que induzcan a decir de forma natural sólo Sí o No. Por ejemplo:  
Lo incorrecto es: Por favor, confirme si su Identificador es: XXXXXXXX  
Lo correcto sería: Identificador: XXXXXXXX. ¿Es correcto?
    - Debe ser una única pregunta, no dos en una.
    - Si además de la pregunta hay texto informativo, poner la pregunta al final de la frase. Por ejemplo:  
Para obtener más información sobre las condiciones del contrato, ¿desea ser comunicado con un operador?
    - Evitar concatenaciones de preguntas Si/No.

### 3.3.2 Fases de Diseño

- **Consultoría IVR**  
Es la etapa de inicio de un proyecto IVR, en este periodo, la empresa que realiza la consultoría determina si el sistema IVR puede beneficiar a la organización y al usuario final.
- **Diseño IVR**  
Es la etapa más importante en la implementación de un sistema IVR, donde los requerimientos son definidos y a la vez complacidos por un desarrollador de IVR, el resultado de esta fase es el diagrama de flujo del IVR.
- **Desarrollo IVR**  
Es la etapa del proyecto en donde se distribuye las diferentes tareas del IVR a grupos de programación.
- **Programación del IVR**  
Es la creación del código del sistema IVR, esta puede ser acompañado del uso de lenguajes de programación o puede ser creado utilizando herramientas de programación de IVR de alto nivel dependiendo de las funciones que deba cumplir el IVR expuestas en el diagrama de flujo.
- **Pruebas del IVR**  
Es un elemento clave en el desarrollo del proyecto, en esta fase donde se verifica el correcto funcionamiento del IVR y se corrigen errores de funcionamiento.

- Administración del IVR

Es el control, configuración y administración sobre la operación del IVR creado. Aquí se incluyen las diferentes herramientas que se adiciona para tener una administración más fácil y ordenada.

### 3.3.3 Análisis del diagrama de bloques y diagrama de flujo

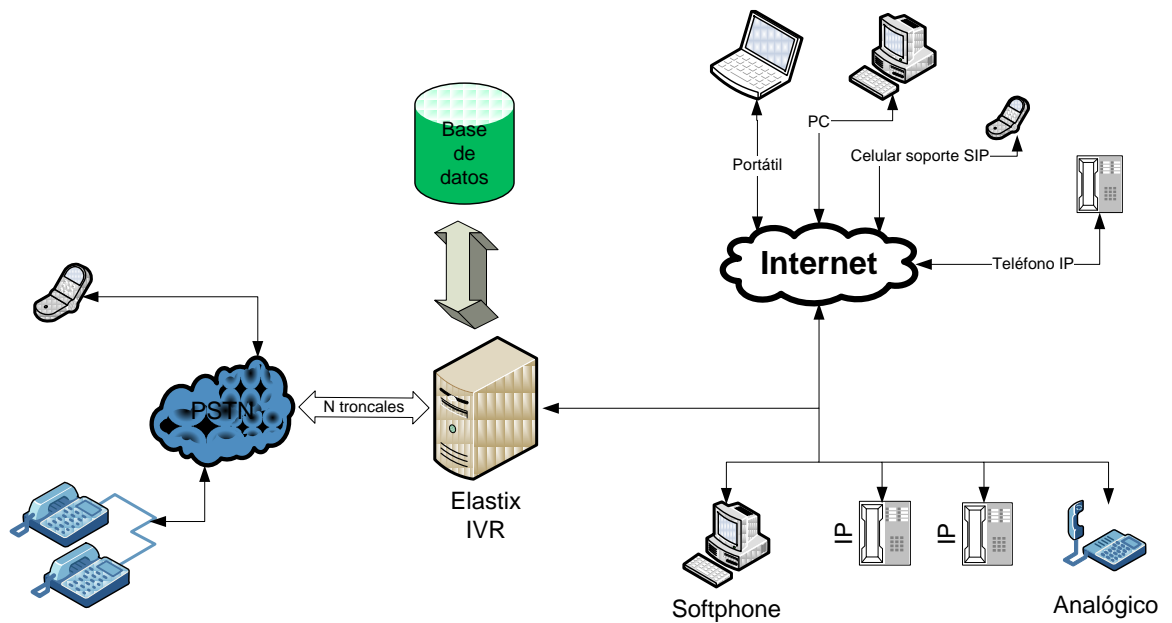


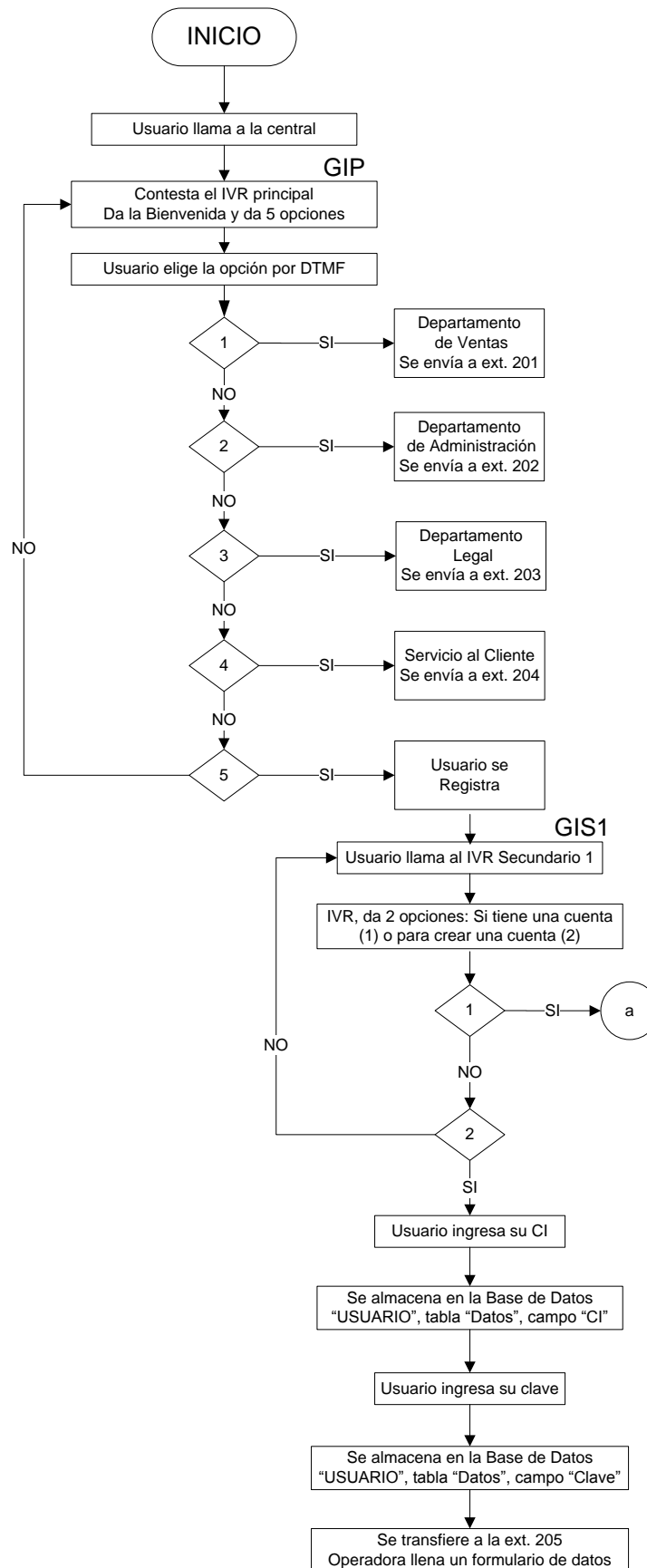
Figura. III.39. Diagrama de bloque del IVR propuesto

La central telefónica está conectada a la PSTN por medio de una línea telefónica a través de una tarjeta *OpenVox FXO*, esto nos permite realizar llamadas desde la PSTN hacia nuestra central de tal manera que conteste nuestro IVR y administre y redireccione las llamadas ya sea hacia nuestra red telefónica interna o hacia otros dispositivos que pueden encontrarse en una ubicación diferente pero a través del internet pueden ser localizados. Para tener esta prestación de salir hacia el internet y tener un acceso hacia

dispositivos que no estén en nuestra red local, necesitamos que nuestra central tenga una IP pública donde pueda ser alcanzada por todos los dispositivos. Con esta funcionalidad se pueden llamar entre extensiones de forma gratuita por medio del internet. Además si algún usuario o extensión requiere salir a la PSTN lo puede hacer ya sea que este en la red local o no, es decir, si un usuario está en otra red, puede comunicarse a la central por medio del internet y realizar una llamada hacia la PSTN que está conectada a la central, con esto la llamada realizada es hecha por medio de la central con un valor local.

En nuestra red telefónica local conectamos también un teléfono analógico por medio de una tarjeta *OpenVox FXS*, el cual nos permite utilizar equipos analógicos y registrarlos a una central de Voz IP, optimizando así los equipos que la empresa tiene en la actualidad.

Los equipos terminales pueden ser teléfonos IP, teléfonos analógicos con un equipo o tarjeta de adaptación, *softphones* que pueden ser instalados en ordenadores de escritorio, portátiles o celulares que soporten el protocolo SIP y tengan la configuración para una cuenta SIP.





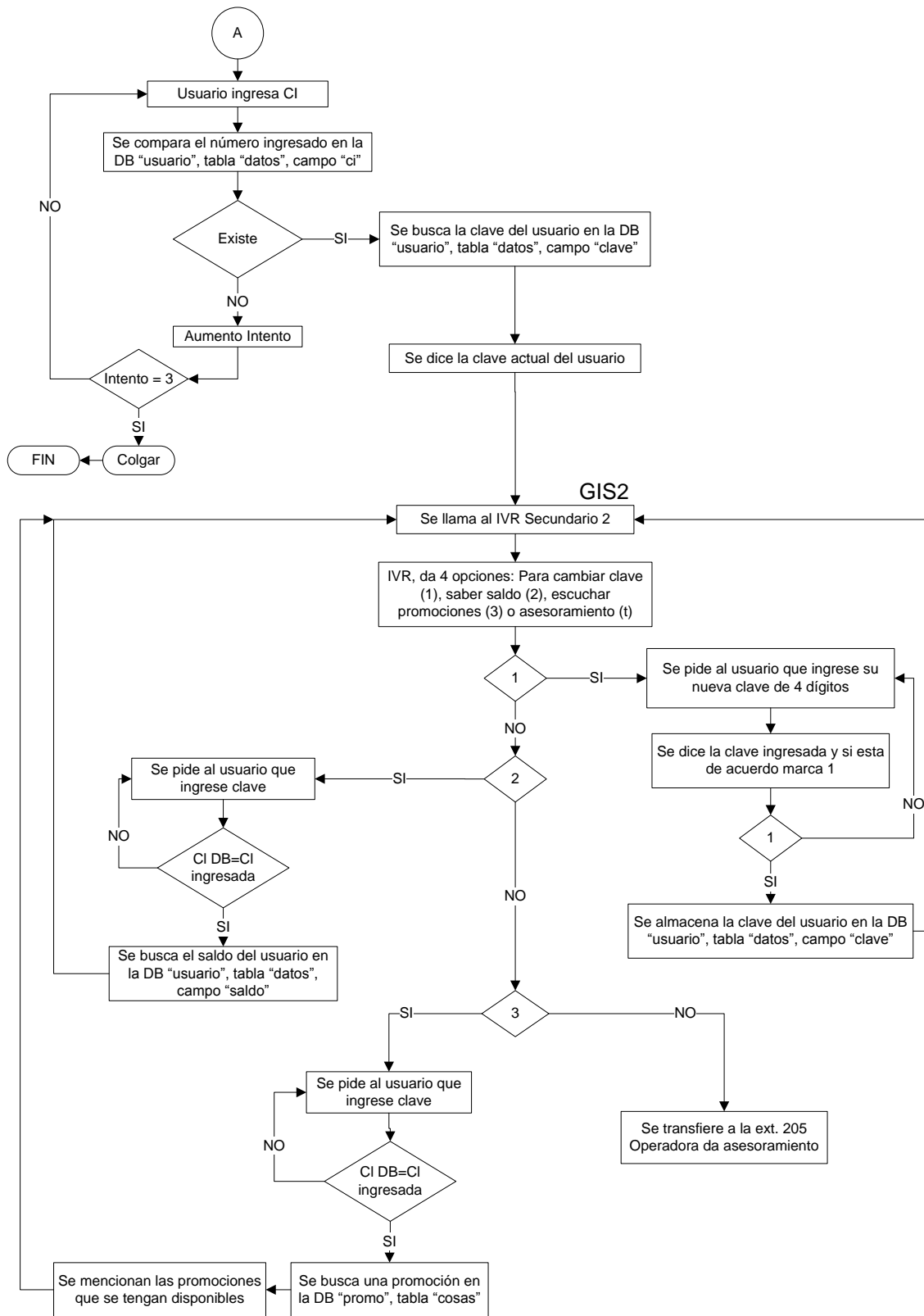


Figura. III.40. Diagrama de flujo del IVR propuesto

El diagrama de bloques muestra la funcionalidad, todas las posibilidades y opciones del IVR propuesto.

- **IVR principal**

Inicialmente el usuario llama al IVR principal, el cual da la bienvenida a la empresa y muestra cinco opciones. El usuario escoge la opción requerida presionando la tecla del número correspondiente. Si en 10 segundos después de dicho el mensaje el usuario no ha presionado ninguna tecla la llamada es colgada (opción t) y en el caso en que el usuario presione una tecla equivocada, el IVR se repite (opción i). Esta configuración será utilizada para todos los IVR's que se utilicen en la solución.

La opción uno del IVR principal nos permite ir a la extensión 201 que se asignó al departamento de ventas, la opción dos nos permite ir a la extensión 202 que se asignó al departamento de administración, la opción tres nos permite ir a la extensión 203 que se asignó al departamento legal, la opción cuatro nos permite ir a la extensión 204 que se asignó a servicio al cliente y la opción cinco que se asignó al IVR secundario 1 para que el usuario pueda registrarse.

- **IVR secundario 1**

El usuario al escoger la opción cinco entra a un nuevo IVR el cual da dos opciones. En estas opciones se distingue a los usuarios que tienen una cuenta y los que desean crear una nueva cuenta. La opción uno es para los usuarios que ya tienen una cuenta y desean

registrarse, para ello se verifica la cédula y clave en la base de datos “usuario”. Después de registrarse es enviado al IVR secundario2.

La opción dos es para los usuarios que quieren crear una nueva cuenta, los valores insertados se colocan en la misma base de datos; el usuario ingresa su cédula y clave de tres dígitos, después de esto el usuario es enviado a la extensión 204 donde una operadora tomará otros datos.

Si el usuario presiona una tecla equivocada el IVR se vuelve a repetir y n el caso de que no presione una tecla por 10 segundos la llamada es colgada.

- **IVR secundario 2**

En este IVR se dan tres opciones. La primera le permite al usuario cambiar su clave, estos cambios se guardaran en la base de datos “usuario”, la segunda le permite al usuario consultar su cuenta a pagar, la cual consulta en la misma base de datos pero en el campo “saldo”, la tercera opción le al usuario ver las promociones que la empresa dispone, estas promociones están guardadas en la base de datos “promo”, si existen promociones se las dice al usuario todas las disponibles, caso contrario se informa al usuario que actualmente no hay promociones.

Al finalizar cada opción se envía nuevamente al IVR secundario 2.

Por último si el usuario espera en la línea es enviado a la extensión 205 que está asignado a una operadora de asesoramiento. Con esta opción se busca que el usuario indique cualquier inquietud que no se ha topado en el IVR, inquietud de las promociones o una ayuda de la operación del IVR total. Además por medio de esta operadora se puede acceder a cualquier extensión si el usuario desearía contactarse con una persona en especial. Si el usuario presiona una tecla equivocada el IVR se vuelve a repetir.

### 3.4 BASES DE DATOS A UTILIZAR EN BASE EL PROBLEMA

En el IVR propuesto se utilizan dos bases de datos; una que hace referencia a los usuarios y su identificación y otra que hace referencia a los equipos y promociones disponibles.

- **Base de Datos USUARIOS**

La base de datos de los usuarios tiene la siguiente estructura:

Nombre de la base de datos: usuario

Nombre de la tabla: datos

Nombre de los campos: ci, clave, saldo.

La forma de la base de datos se muestra en el siguiente ejemplo:

**Tabla. III.10. Estructura de la base de datos usuario**

<b>Ci</b>	<b>clave</b>	<b>saldo</b>
1716490410	1234	100
1716490402	5678	200

- **Base de Datos PROMOCIONES**

La base de datos de las promociones tiene la siguiente estructura:

Nombre de la base de datos: promo

Nombre de la tabla: cosas

Nombre de los campos: des, valor, dias.

---

La forma de la base de datos se muestra en el siguiente ejemplo:

**Tabla. III.11. Estructura de la base de datos promociones**

<b>Descripción</b>	<b>valor</b>	<b>días</b>
Todos los monitores de 17 pulgadas tienen el 50 por ciento de descuento.	80	15
Las memorias RAM de un gigabyte tienen 25 por ciento de descuento.	20	30

## CAPÍTULO IV

### 4 CONFIGURACIÓN DEL IVR

#### 4.1 CONFIGURACIÓN DE ASTERISK PARA EL IVR MODELADO

Para la configuración en general de nuestra central y específicamente del IVR necesitamos primero acceder a la administración web. Para ello se abre un explorador y en la barra de direcciones colocamos la dirección IP pública de nuestra central o su dominio e ingresamos el usuario y la contraseña que por defecto son “*admin*” y “*palosanto*” respectivamente.



**Figura. IV.41. Acceso inicial a la administración web**  
Fuente: [ELXCJM]

### 4.1.1 CONFIGURACIÓN DEL IVR CON ELASTIX

- **Procedimiento**

- **Creación de Usuarios**

Inicialmente necesitamos crear las extensiones de los usuarios. Basándonos en el diagrama de flujo, necesitamos de seis extensiones a las cuales se les asignará los diferentes departamentos, servicio al cliente y operadora de ayuda.

- **Grabaciones para el IVR**

Todos los mensajes del IVR deben ser previamente grabados. Estas grabaciones pueden ser exportadas desde un lugar específico o grabarlas desde una extensión. Los formatos para estas grabaciones son *wav* y *gsm*.

- **Creación del IVR**

Como vimos en el diagrama de flujo, necesitamos configurar tres IVR's, uno principal y dos secundarios. Cada IVR tendrá sus respectivas opciones indicadas en el diagrama de flujo hacia los destinos indicados.

- **Hardware**

La configuración del hardware que realizamos es para poder salir a la PSTN, esto se realiza por medio de una tarjeta FXO. Además colocamos una tarjeta FXS que nos permitirá conectar un teléfono analógico como extensión.

- 
- **Troncales**

Una vez configurado el hardware es importante crear las troncales que permiten direccionar las llamadas entrantes como salientes por el módulo correspondiente.
  
  - **Rutas Salientes**

Las rutas salientes son el plan de marcado que se tomara en cuenta para realizar las llamadas salientes hacia la PSTN, además se especifica la troncal a utilizar.
  
  - **Rutas Entrantes**

La ruta entrante indica la troncal por la cual se atenderán las llamadas entrantes desde la PSTN.
  
  - **Extensión Analógica**

Esta extensión se configuró para indicar el alcance que puede tener Elastix y la VoIP. Con esto queremos indicar la manera como se adapta la telefonía analógica con la telefonía IP.
  
  - **Configuración**
    - **Creación de Usuarios**

Una vez en la interfaz web vamos a “PBX” en el menú principal y dentro de este vamos a extensiones. Una vez en extensiones añadimos un equipo genérico SIP y presionamos en “Submit”. Se nos desplegará la siguiente página:



Opción
<b>freePBX Sin embeber</b>
Básico
<b>Extensiones</b>
Códigos de funcionalidades
Configuración General
Rutas Salientes
Troncales
Control de Llamadas entrantes
<b>Rutas Entrantes</b>
Anuncios
Sígueme
IVR
Otros Destinos
Colas
Grupos de Timbrado
Condiciones de Tiempo
Opciones Internas & Configuración
<b>Conferencias</b>
Otras Aplicaciones
Música en Espera
Conjuntos de PIN
Paginación e Intercomunicación
Estacionamiento
Grabaciones del Sistema

## Add SIP Extension

Add Extension

---

User Extension

Display Name

CID Num Alias

SIP Alias

Extension Options

---

Outbound CID

Ring Time

Call Waiting

Call Screening

Emergency CID

Assigned DID/CID

---

DID Description

Add Inbound DID

Add Inbound CID

Device Options

---

This device uses sip technology.

secret

dtmfmode

**Figura. IV.42. Interfaz web para añadir extensión SIP**  
Fuente: [ELXCJM]

Para una configuración básica llenamos los campos: “*User Extension*”, “*Display Name*” y “*secret*”. El significado de cada campo esta detallado en el capítulo 2 sección 2.3.

Una configuración adicional es el idioma con el que se manejarán los mensajes de Asterisk desde una extensión, por defecto están en inglés, pero para configurarlos al español en el campo “*Language Code*” colocamos “es”.

Una vez configuradas todas las extensiones con sus respectivos campos, la visualización de las extensiones es:

**Figura. IV.43. Visualización de extensiones configuradas**  
Fuente: [ELXCJM]

- **Grabaciones para el IVR**

Las grabaciones del IVR lo haremos desde la extensión 206 que esta asignada a un teléfono analógico. Para ello en el submenú derecho del menú “PBX” elegimos grabaciones del sistema. En el campo de extensión colocamos 206 para indicar la extensión desde donde se van hacer las grabaciones y damos clic en “go”.

**Figura. IV.44. Interfaz web para añadir grabaciones**  
Fuente: [ELXCJM]

En el teléfono de la extensión 206 marcamos “\*77” y sonará un *beep* indicándonos el inicio de la grabación, para finalizar la grabación podemos colgar o marcar numeral y colgar.

Para escuchar la grabación marcamos “\*99”, si estamos de acuerdo colgamos y en la interfaz web en el campo “*Name this Recording*” colocamos el nombre de la grabación y damos clic en “*Save*”.

Las grabaciones que realizamos en nuestro IVR son:

**Tabla. IV.12. Tabla de grabaciones para el IVR**

<b>Código</b>	<b>Nombre Archivo</b>	<b>Dialecto</b>
GIP	menu_principal	“Bienvenido a la empresa Intel Solutions, si desea comunicarse con ventas marque 1, con administración marque 2, con el departamento legal marque 3, con servicio al cliente marque 4 o si desea registrarse marque 5”
GIS1	crear_cuenta	“Si tiene una cuenta marque 1, si desea crear una cuenta marque 2”
GIS2	menu_secundario1	“Para cambiar la clave marque 1, para consultar su saldo marque 2, para escuchar promociones a clientes marque 3, caso contrario espere en la línea que una operadora le atenderá”

- **Creación del IVR**

La configuración del módulo IVR se la hace en el submenú derecho del menú “PBX”. Elegimos la opción IVR y damos clic en “*Add IVR*”. La pantalla de configuración para el IVR se presenta a continuación:

## Digital Receptionist

### Edit Menu Unnamed

Delete Digital Receptionist Unnamed

---

Change Name:

Announcement:

Timeout:

Enable Directory:

VM Return to IVR:

Directory Context:

Enable Direct Dial:

Loop Before t-dest:

Timeout Message:

Loop Before i-dest:

Invalid Message:

Repeat Loops:

---

Increase Options Save Decrease Options

---

Return to IVR

Phonebook Directory:

Terminate Call:

Extensions:

Misc Destinations:

IVR:

Ring Groups:

**Figura. IV.45. Interfaz web para configurar el IVR**  
Fuente: [ELXCJM]

Los campos principales del módulo IVR son:

➤ *Change Name*

Este es el título *descriptivo* del IVR.

➤ *Announcement*

Es el sonido que previamente debe ser agregado en “grabaciones de sistema” y es reproducido a los usuarios cuando ingresan en el IVR. Este campo no es obligatorio.

➤ *Timeout*

Es la cantidad de tiempo que el sistema espera antes de enviar la llamada al destino 't'.

➤ *Enable directory*

Si se activa esto, los usuarios serán capaces de marcar el código de prestaciones para el directorio desde el IVR y acceder al servicio de directorio.

➤ *Enable direct dial*

Si se activa esto, los usuarios además de poder discar las opciones del IVR, serán capaces de discar directamente un número de extensión.

➤ Opciones del IVR

En el casillero a la izquierda, se ingresa el número de opción para el usuario. Esto puede ser uno, ó una serie de números.

i: Esto es el destino usado cuando un llamante ingresa una opción inválida.

t: Este es el destino utilizado cuando se supera el "tiempo fuera".

Se usa "Incrementar opciones" (*Increase Options*) o "Disminuir opciones" (*Decrease Options*) para alterar el número de opciones disponibles. Para borrar una opción, simplemente deje la sección en blanco.

La configuración del IVR secundario 1 se muestra a continuación:

---

Return to IVR

Leave blank to remove

Phonebook Directory: Phonebook Directory  
 Terminate Call: Hangup  
 Extensions: <201> Ventas  
 Misc Destinations: Si tiene cuenta  
 IVR: IVR\_principal  
 Ring Groups: Prueba grupo <600>

---

Return to IVR

Leave blank to remove

Phonebook Directory: Phonebook Directory  
 Terminate Call: Hangup  
 Extensions: <201> Ventas  
 Misc Destinations: Crea cuenta  
 IVR: IVR\_principal  
 Ring Groups: Prueba grupo <600>

---

Return to IVR

Leave blank to remove

Phonebook Directory: Phonebook Directory  
 Terminate Call: Hangup  
 Extensions: <201> Ventas  
 Misc Destinations: cambia clave  
 IVR: IVR\_secundario\_1  
 Ring Groups: Prueba grupo <600>

---

Return to IVR

Leave blank to remove

Phonebook Directory: Phonebook Directory  
 Terminate Call: Hangup  
 Extensions: <201> Ventas  
 Misc Destinations: cambia clave  
 IVR: IVR\_principal  
 Ring Groups: Prueba grupo <600>

---

**Figura. IV.46. Configuración del IVR secundario 1**  
Fuente: [ELXCJM]

Cuando se termine la configuración hacer clic en “Guardar” (Save).

- **Hardware**

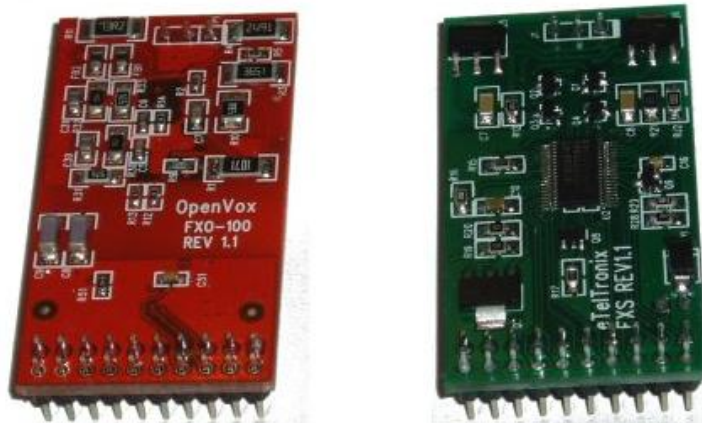
Una interfaz FXS y FXO se maneja por medio de módulos y una tarjeta PCI.

La tarjeta PCI utilizada es la *OpenVox A400P*.



Fotografía. IV.1. Tarjeta PCI de 4 puertos para módulos FXS/FXO

Esta tarjeta es de cuatro puertos en los cuales se pueden incluir módulos FXS o FXO. Los módulos FXO se distinguen por su color rojo, mientras los módulos FXS son verdes:



Fotografía. IV.2. Módulos FXS y FXO

En nuestro caso utilizamos un módulo FXS y un modulo FXO, la siguiente figura muestra físicamente los módulos colocados en la tarjeta:



Fotografía. IV.3. Tarjeta PCI con un módulo FXS y un FXO

Una vez colocada la tarjeta con los módulos en la máquina, Elastix los reconocerá automáticamente y en la administración web se podrá visualizar el estado de los módulos. Para ello una vez en a interfaz web vamos a “sistema” y dentro de este a “Detección de *hardware*”. Si el *hardware* no ha sido detectado, presionamos en “Detectar nuevo *hardware*”, caso contrario si el *hardware* ya ha sido detectado nos aparecerá la ubicación de los módulos dentro de la tarjeta, como se indica en la figura.





Figura. IV.47. Interface web para detección de hardware

Fuente: [ELXCJM]

Es importante identificar y recordar el canal en el cual se han conectado los módulos. Para nuestro caso el módulo FXS está en el canal 1 y el módulo FXO está en el canal 3.

Una vez detectados los módulos, damos clic en la opción “Reemplazar el archivo *chan\_dahdi.conf*”. Este es el archivo de configuración de los módulos conectados a Asterisk, contiene los drivers de los módulos y permite el correcto funcionamiento entre la tarjeta y Asterisk.

Si se da el caso en que los módulos aparezcan en estado de “No en uso”, se debe reiniciar el programa *amportal*, para ello por consola colocamos “*amportal restart*”, finalizando así todos los programas relacionados con Asterisk y sincronizándolos en el encendido.

- **Troncales**

Para la configuración de las troncales, vamos a “Troncales” en el menú PBX.

Para el modulo FXO añadimos una troncal ZAP<sup>16</sup>, para ello presionamos en “*Add Zap Trunk (DAHDI compatibility mode)*”. En esta configuración únicamente se especifica el canal de la troncal, los otros campos se los deja por defecto ya que son dirigidos hacia otras aplicaciones. En el campo “*Zap Identifier (trunk name)*” colocamos el número de canal, en este caso 3 y damos clic en “*Submit Changes*” como se indica en la figura:

### Add ZAP Trunk (DAHDI compatibility mode)

#### General Settings

Outbound Caller ID:   
 Never Override CallerID:   
 Maximum Channels:   
 Disable Trunk:  Disable  
 Monitor Trunk Failures:  Enable

#### Outgoing Dial Rules

Dial Rules:

Clean & Remove duplicates

Dial Rules Wizards: (pick one)

Outbound Dial Prefix:

#### Outgoing Settings

Zap Identifier (trunk name):

**Figura. IV.48. Interface web para configuración de troncales**

Fuente: [ELXCJM]

De la misma manera creamos una troncal ZAP para el módulo FXS variando únicamente el número del canal que sería el 1.

<sup>16</sup> **Troncales ZAP.** son troncales físicamente conectadas a la Red de Telefonía Pública (PSTN), las cuales pueden ser líneas telefónicas analógicas, BRI ISDN o PRI ISDN.

- **Rutas Salientes**

Para la configuración de las llamadas salientes, vamos a “Rutas Salientes” en el menú PBX. Esta configuración solo la realizaremos al módulo FXO ya que es el que se conecta a la PSTN.

La configuración de la ruta saliente se muestra a continuación:

### Add Route

Route Name:	<input type="text" value="Salida"/>
Route Password:	<input type="text"/>
PIN Set:	<input type="text" value="None"/>
Emergency Dialing:	<input type="checkbox"/>
Intra Company Route:	<input type="checkbox"/>
Music On Hold?	<input type="text" value="default"/>
Dial Patterns	<div style="border: 1px solid #ccc; padding: 5px; min-height: 80px;"> XXXXXXXX  5  XXXXXXXXXX </div>
	<input type="button" value="Clean &amp; Remove duplicates"/>
Dial patterns wizards:	<input type="text" value="(pick one)"/>
Trunk Sequence	<input type="text" value="ZAP/3"/>
	<input type="text"/>
<input type="button" value="Submit Changes"/>	

**Figura. IV.49. Interface web para configuración de rutas salientes**

Fuente: [ELXCJM]

La configuración básica es el nombre, el plan de marcado y la secuencia de troncales.

En nuestro caso se la ruta llamará “Salida”.

En el plan de marcado asignado, “X” significa cualquier número, por lo tanto “XXXXXXXX” significa cualquier

combinación de 7 números puede salir por esta ruta, es decir, números locales como “28611XX”. Por otro lado “5|XXXXXXXXXX” significa que lo que sea marcado inicialmente con 5 y seguido de 9 números salga por esta ruta, es decir teléfonos celulares como “0870027XX”. El símbolo “|” permite separar un indicador de un número, es decir, lo que esta antes será el indicador para salir a celulares.

La secuencia de troncales, quiere decir la troncal por la que se saldrá hacia la PSTN, si se tuvieran más líneas se podría direccionar las llamadas, pero en este caso solo elegimos la troncal 3 que es el modulo FXO.

#### ○ Rutas Entrantes

Para la configuración de las llamadas entrantes, vamos a “Rutas Entrantes” en el menú PBX. Esta configuración solo la realizaremos al módulo FXO ya que es el que se conecta a la PSTN.

La configuración de la ruta entrante se muestra a continuación:

### Route: Entradas

[Delete Route Entradas](#)

Edit Incoming Route

Description:   
 DID Number:   
 Caller ID Number:   
 CID Priority Route:

Options

Alert Info:   
 CID name prefix:   
 Music On Hold:   
 Signal RINGING:   
 Pause Before Answer:

Source:  ▼

Set Destination

---

Phonebook Directory:  ▼  
 Terminate Call:  ▼  
 Extensions:  ▼  
 Misc Destinations:  ▼  
 IVR:  ▼  
 Ring Groups:  ▼

**Figura. IV.50. Interface web para configuración de rutas entrantes**

Fuente: [ELXCJM]

La configuración básica para una ruta entrante es el nombre y el destino. Las rutas entrantes se las puede hacer para un número en específico o para todos los números entrantes, si el campo “*DID Number*” y “*Caller ID Number*” lo dejamos vacío tomará todos los números entrantes de la PSTN y los enviará al destino seleccionado.

En nuestro caso el nombre es “Entradas” y el destino será el IVR principal. Finalmente presionamos “*Submit*”.

#### ○ **Extensión Analógica**

Para la configuración de la extensión analógica, vamos a “Extensiones” en el menú PBX. Esta configuración solo la realizaremos al módulo FXS ya que es el que se conecta al teléfono analógico.

Dentro de “Extensiones ” escogemos un equipo ZAP genérico, ya que es un teléfono analógico.

La configuración de una extensión ZAP se muestra a continuación:

## Add ZAP Extension

Add Extension

---

User Extension

Display Name

CID Num Alias

SIP Alias

Extension Options

---

Outbound CID

Ring Time

Call Waiting

Call Screening

Emergency CID

Assigned DID/CID

---

DID Description

Add Inbound DID

Add Inbound CID

Device Options

---

This device uses zap technology. (Via DAHDI compatibility mode) channel

**Figura. IV.51. Interface web para configuración de extensión ZAP**  
Fuente: [ELXCJM]

Los parámetros básicos a configurar son el número de extensión, el nombre y el canal por el que se va a usar la extensión.

En nuestro caso la extensión es la 206, el nombre es “Telefono Analogico” y el canal es el 1 ya que corresponde al módulo FXS.

Una configuración adicional es el idioma con el que se manejarán los mensajes de Asterisk desde esta extensión, por defecto están en inglés, pero para configurarlos al español en el campo “*Language Code*” colocamos “es”. Finalmente damos clic en “*Submit*”.

## 4.1.2 INTERACCIÓN DEL IVR CON LA BASE DE DATOS MySQL

- **Procedimiento**

- **Bases de datos**

En nuestro sistema existen dos bases de datos, de usuarios y promociones. La configuración de las bases de datos se hará de acuerdo a la estructura indicada anteriormente. Las bases de datos son MySQL y se crearán con lenguaje SQL. Estas bases de datos están dentro del mismo servidor ya que Elastix trae preinstalado el servidor de base de datos MySQL.

- **Creación de los contextos y otros destinos**

Los contextos son los agrupadores u organizadores del plan de marcado, básicamente los contextos sirven para agrupar lógicamente el plan de marcado. Los contextos pueden ser anillados y llamados desde otro contexto mediante la sentencia “*include*”.

Cada contexto lleva un nombre único y este nombre va encerrado entre corchetes “[ ]”.

Para llamar a un contexto desde la central, se asigna un número al cual responde. Para poder llamar desde el IVR a este número del contexto es necesario crear otros destinos desde la administración web.

- **Creación del *script* PHPAGI**

PHPAGI es un *framework* para realizar *scripts* AGI de una manera más óptima con funciones características complementarias, es decir, utiliza funciones específicas de phpagi que se basan en una combinación de comandos agi. Con

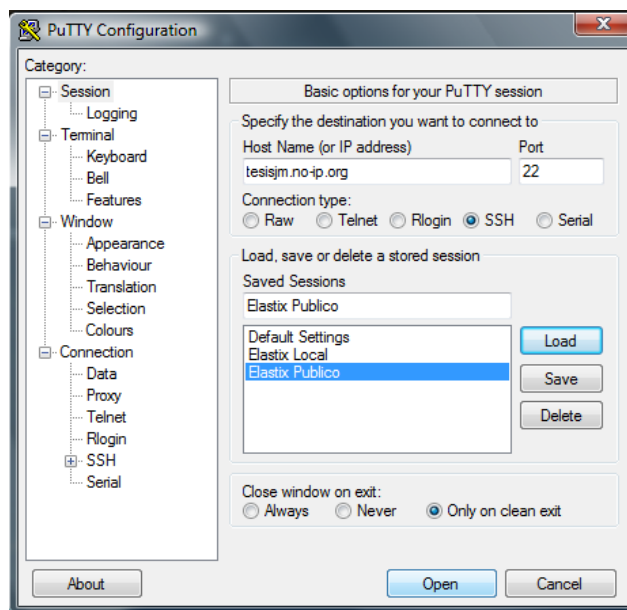
esto facilitamos la creaciones de scripts con funciones predefinidas. Además utiliza festival un programa TTS para tener también un formato hablado si es necesario.

Con esto podemos manipular la información de cualquier base de datos, tanto extraer datos como guardarlos y por medio del TTS transformarlos a voz.

- **Configuración**

- **Bases de datos**

La configuración de las bases de datos se la hizo por consola en el servidor donde se encuentra la central. Para configurar la base de datos accedemos via *ssh* a la central, en nuestro caso utilizamos el programa “*putty*”<sup>17</sup>. Colocamos la dirección IP del servidor o el dominio del mismo y accedemos.



**Figura. IV.52. Herramienta putty para acceso ssh**

<sup>17</sup> **PuTTY**: es un cliente SSH, Telnet, rlogin, y TCP raw con licencia libre. Disponible originariamente sólo para Windows, ahora también disponible en varias plataformas Unix y Mac OS clásico y X. Se lo puede descargar de <http://www.putty.org/>.



Una vez en la central, nos registramos y accedemos al servidor MySQL. El usuario es *root* y la clave “*eLaStIx.2oo7*”. El ingreso a MySQL se muestra a continuación:

```
login as: root
root@tesisjm.no-ip.org's password:
Last login: Sun Nov  8 13:33:46 2009 from gprsinternet125.porta.net

Welcome to Elastix
-----

For access to the Elastix web GUI use this URL
http://190.154.214.44

[root@tesisjm ~]# mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 146
Server version: 5.0.77 Source distribution

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> create database usuario;
Query OK, 1 row affected (0.00 sec)

mysql> create database promo;
Query OK, 1 row affected (0.00 sec)

mysql> show databases;
+-----+
| Database          |
+-----+
| information_schema |
| asterisk          |
| asteriskcdrdb    |
| mya2billing       |
| mysql             |
| promo            |
| roundcubedb       |
| usuario           |
+-----+
8 rows in set (0.00 sec)
```

Figura. IV.53. Creación de base de datos en la consola MySQL

Configuración de la base de datos “usuario”:

```

mysql> use usuario;
Database changed
mysql> create table datos (ci int(10) null, clave int(4) null, saldo float null);
Query OK, 0 rows affected (0.05 sec)

mysql> select * from datos;
Empty set (0.00 sec)

mysql> insert into datos (ci,clave,saldo) values (1716490410,1234,100);
Query OK, 1 row affected (0.00 sec)

mysql> insert into datos (ci,clave,saldo) values (1716490402,1234,200);
Query OK, 1 row affected (0.00 sec)

mysql> select * from datos;
+-----+-----+-----+
| ci          | clave | saldo |
+-----+-----+-----+
| 1716490410 | 1234  | 100   |
| 1716490402 | 1234  | 200   |
+-----+-----+-----+
2 rows in set (0.00 sec)

```

Figura. IV.54. Configuración base de datos usuario en la consola MySQL

Configuración de la base de datos “promo”:

```

mysql> select * from cosas;
Empty set (0.01 sec)

mysql> insert into cosas (descripcion,valor,dias) values ('Los monitores de 15 pulgadas tienen el 25 por ciento de descuento',55,15);
Query OK, 1 row affected (0.00 sec)

mysql> insert into cosas (descripcion,valor,dias) values ('Los discos duros de 160 gigas tienen el 50 por ciento de descuento',62,5);
Query OK, 1 row affected (0.00 sec)

mysql> select * from cosas;
+-----+-----+-----+
| descripcion                                     | valor | dias |
+-----+-----+-----+
| Los monitores de 15 pulgadas tienen el 25 por ciento de descuento | 55    | 15   |
| Los discos duros de 160 gigas tienen el 50 por ciento de descuento | 62    | 5    |
+-----+-----+-----+
2 rows in set (0.00 sec)

mysql> quit;
Bye
[root@tesisjm ~]#

```

Figura. IV.55. Configuración base de datos promo en la consola MySQL

- **Creación de los contextos y otros destinos**

Los contextos se crean en el archivo:

“*/etc/asterisk/extensions\_cutom.conf*”.

En los contextos lo que hacemos es llamar a las respectivas interfaces AGI. Los contextos creados para el IVR se muestran en los anexos.

El contexto [agi3334] es utilizado para ingresar la cédula de un usuario que este en la base de datos y es llamado marcando 3334. Al finalizar y regresar del *script* envía la llamada al IVR secundario 2.

El contexto [agi4445] es utilizado para crear una nueva cuenta y es llamado marcando 4445. Al finalizar y regresar del *script* envía la llamada a la extensión 205.

El contexto [agi5556] es utilizado para que el usuario cambie su clave y es llamado marcando 5556. Al finalizar y regresar del *script* envía la llamada al IVR secundario 2.


El contexto [agi6667] es utilizado para que el usuario revise su cuenta a pagar y es llamado marcando 6667. Al finalizar y regresar del *script* envía la llamada al IVR secundario 2.

El contexto [agi7778] es utilizado para revisar las promociones disponibles y es llamado marcando 7778. Al finalizar y regresar del *script* envía la llamada al IVR secundario 2.

Estos contextos deberán ser llamados por el IVR cuando corresponda, para ello se añaden “otros destinos” desde la interfaz web. En el menú “PBX” escogemos “Otros Destinos” (*Misc Destination*). Para la configuración únicamente colocamos la descripción y en el campo “Dial” el número al que se debe marcar para acceder al contexto.

Por ejemplo para acceder al contexto [agi3334], la configuración es:

## Misc Destination: Si tiene cuenta

 Delete Misc Destination Si tiene cuenta  
 Used as Destination by 2 Objects:  
[Edit Misc Destination](#)

---

Description:   
 Dial:

**Figura. IV.56. Interface web para configuración de otros destinos**  
Fuente: [ELXCJM]

### ○ Creación del *script* PHPAGI

Cada contexto llama a un *script phpagi*. Los *scripts* respectivamente para cada contexto son *agi3334.php*, *agi4445.php*, *agi5556.php*, *agi6667.php* y *agi7778.php*.

Los *scripts* se guardan en la carpeta “*/var/lib/asterisk/agi-bin*”.

#### ✓ *agi3334.php*

Inicialmente con la ayuda del TTS enviamos un mensaje al usuario para que ingrese su cédula, este ingreso solo se lo hará cuando suene un *beep* al final del mensaje y está limitado para 10 dígitos. Una vez ingresada la cédula verificamos en la base de datos si existe dicho usuario, si este número de cédula no existe se vuelve a pedir la cédula de identidad; para esto el usuario tiene tres intentos o caso contrario se colgará la llamada indicando un mensaje que los intentos de ingreso han caducado.

Si el usuario existe se busca en la base de datos “usuario” la clave actual y por medio del TTS indicamos la clave dígito por dígito.

✓ agi4445.php

Este *script* nos permite almacenar los datos ingresados por el usuario en la base de datos “usuario”. Inicialmente enviamos un mensaje de forma hablada al usuario para que ingrese su cédula, este ingreso solo se lo hará cuando suene un *beep* al final del mensaje y está limitado para 10 dígitos. Una vez ingresada la cédula almacenamos en la base de datos y pedimos que ingrese la clave de cuatro dígitos.

✓ agi5556.php

Este *script* nos permite cambiar la clave de un usuario en la base de datos “usuario”.

Para ello se pide al usuario que ingrese su nueva clave de cuatro dígitos, este ingreso está validado en tamaño. Una vez ingresada la clave, de forma hablada se indica al usuario la clave ingresada dígito por dígito y se pregunta si está de acuerdo. Para afirmar su clave ingresada el usuario presiona 1, caso contrario presiona otra tecla y se pedirá al usuario nuevamente que ingrese su clave.

Una vez aceptada la nueva clave se almacena en la base de datos y se envía un mensaje hablado de la confirmación del cambio.

✓ agi6667.php

Este *script* nos permite consultar la cuenta a pagar de un usuario y utiliza la base de datos “usuario”.

Para que el usuario pueda conocer su cuenta el *script* pide inicialmente que ingrese la clave de usuario.

La clave debe ser la correspondiente al usuario que se registro en el *script* agi3334.php con su cédula.

Si la clave ingresada es igual a la clave de la base de datos, el usuario es registrado y puede escuchar su saldo. El saldo se busca en la base de datos y se lo envía al usuario en un mensaje de forma hablada. De igual manera el saldo corresponderá al usuario que se registro en el *script* agi3334.php con su cédula.

✓ agi7778.php

Este *script* nos permite consultar las promociones disponibles y utiliza la base de datos “promo”.

Para que el usuario pueda conocer las promociones el *script* pide inicialmente que ingrese la clave de usuario.

La clave debe ser la correspondiente al usuario que se registro en el *script* agi3334.php con su cédula.

Si la clave ingresada es igual a la clave de la base de datos, el usuario es registrado y puede escuchar las promociones disponibles. Si existen promociones se las va nombrando de una en una con su respectiva descripción, precio y tiempo de duración.

## CAPÍTULO V

### 5 RESULTADOS

#### 5.1 ANÁLISIS DEL IVR EN EL MERCADO

El sistema IVR actualmente es una herramienta muy poderosa que permite a las empresas y organizaciones administrar de mejor manera sus llamadas entrantes y salientes.

Este sistema cada vez ha ido evolucionando y junto con otras aplicaciones como base de datos, TTS, ASR, etc. han permitido que el sistema IVR se convierta en más que una simple operadora automática. El avance de los IVR's en el campo tecnológico permiten que las empresas puedan brindar a sus clientes mejoras en atención y calidad de los servicios.

Las aplicaciones principales del IVR se dan en las áreas de comunicación, servicios financieros, servicios de salud entre otros.

Una de las empresas más fuertes según Beasty Collin [BEA2006] es "Avaya" que estableciendo una alianza con "Intervoice", alcanzó en el año 2005 un crecimiento del 20% en ventas. En el 2005 Avaya lanza el "Avaya Voice Portal", una solución basada en el protocolo IP que soporta SIP y es perfecto para la actual transición a telefonía IP.

Por otro lado Nortel Networks es una empresa que también ha incluido soporte para VoIP y VXML (*Voice XML*) pero no ha tenido la suficiente aceptación por su bajo rendimiento.

La empresa más reconocida actualmente es *Genesys Telecommunications Laboratory*, derrocando a Avaya e Intervervoice. La plataforma *Genesys Voice Plataform* ofrece interconexión con sistemas *open source* e incluye VXML.

La siguiente tabla muestra la estadística de tres empresas representativas en sistemas IVR:

**Tabla. V.13. Encuesta a clientes sobre preferencias en sistemas IVR**

Fuente: [ BEA2006]

<b>IVR</b>	<b>Satisfacción del cliente</b>	<b>Funcionalidad</b>	<b>Dirección Empresarial</b>	<b>Top 3 Verticales</b>
Avaya	3.5	3.6	4	Servicios financieros, de salud y comunicaciones.
Genesys Telecommunications Labs	4	4.2	3.9	Comunicaciones, finanzas, salud
Nortel Networks	3.3	4	4	Comunicaciones, servicios financieros, salud, tecnología

El mercado de los sistemas IVR desde su inicio se ha incrementado alrededor de un 90% en la última década. La integración con nuevas tecnologías han elevado las ventas dado la funcionalidad y ahorro que produce en el sector empresarial. [FRO2006]



Los sistemas IVR en Estados Unidos se incrementaron notablemente en los años 2004 y 2005. En el 2005 el mercado de los sistemas IVR generó aproximadamente \$564.8 millones de dólares y en los próximos años tuvo una tasa de crecimiento anual cercana a 13.4 por ciento.

Los pronósticos sufrieron un crecimiento permanente del mercado en el 2008 y el 2009, sufrieron una estabilización experimentando en este proceso ligeros descensos. [FRO2006]

China es otro mercado aún más prometedor dada la densidad poblacional. En este país los servicios IVR han alcanzado un auge como servicios de valor agregado a las empresas de telefonía celular, como por ejemplo consultar sobre el clima, noticias de deporte, consultar su saldo, modificar información de la cuenta del usuario, ingresar a una sala de Chat con comandos de voz o realizar compras.

## **5.2 SOLUCIONES DE TELEFONÍA BASADOS EN SOFTWARE LIBRE VS SOLUCIONES PROPIETARIAS**

Los “*Softswitches*” son los dispositivos encargados del control y procesamiento de llamadas sobre una red de conmutación de paquetes. Uno de los *softswitches* más populares es “Asterisk” de Digium, el cual nos da la ventaja de implementar centrales basadas en software, reduciendo los precios y cambiando radicalmente la telefonía manejando la voz como paquetes de datos, convirtiéndose en tan sólo otro servicio corriendo en un servidor.

Asterisk es un sistema de implementación libre de una central telefónica o PBX que permite todas las funcionalidades de ésta pudiendo servir también como enlace a la PSTN. Para usar Asterisk sólo se requiere

como mínimo una computadora personal o servidor al que se le pueden añadir periféricos como tarjetas de telefonía *PCI* o *PCI Express* para poder conectarnos a la red telefónica. Asterisk funciona y fue creado para principalmente para GNU/Linux convirtiéndose de ésta forma una solución basada en software libre, libre de necesidad de licencias y amplias posibilidades de configuración de servicios como los IVR.

Otras soluciones para telefonía IP pero privadas son las soluciones de Microsoft como el “*Speech Server 2007*”, que corre bajo *Windows Sever 2003* o similar. A diferencia de estas soluciones Asterisk nos brinda una solución de costo mucho menor debido a las licencias de uso que las soluciones de Microsoft necesitan.

En especial haciendo referencia a sistemas IVR las interfaces de Microsoft son mucho más amigables y comprensibles, mientras que nativamente en Asterisk sólo se configuraban manipulando los archivos de configuración bajo consola. Actualmente existen muchas interfaces web como FreePBX las cuales han mejorado la administración y la han hecho mucho más amigable y sencilla, superando la presentada por Microsoft.

Por otro lado, Asterisk también tiene debilidades como lo son la incertidumbre sobre la cantidad de CPU disponible para llevar a cabo las tareas de conmutación y servicios telefónicos en un determinado instante o la variación de calidad de la voz y calidad de servicio ya que depende del estado, tráfico en la red y hardware disponible.

Otras soluciones son las centrales de marcas privadas como Siemens, Alcatel, Huawei, Panasonic, entre otras. Su principal desventaja es el

alto costo y su difícil administración. A pesar de que prestan en su mayoría las mismas prestaciones de una central bajo software, estas centrales están limitadas por hardware.

### **5.3 ANÁLISIS DE LA HERRAMIENTA FREEPBX, SUS VENTAJAS Y DESVENTAJAS FRENTE A UNA CONFIGURACIÓN POR CONSOLA.**

Como hemos visto FreePBX se encarga de crear todos los cambios necesarios en los archivos de configuración del asterisk `/etc/asterisk/*`.

FreePBX crea archivos adicionales de configuración los cuales sobrescribe cada vez que se haga un cambio en la interfaz web.

Con esto es posible modificar de una manera sencilla la configuración del sistema de telefonía, evitando tener que editar los ficheros de configuración manualmente.

Prácticamente el 90% de las personas que manejan centrales de VoIP utilizan interfaces web para administrar Asterisk, utilizan algún tipo de FreePBX (el que viene con Elastix por ejemplo) el cual se ha convertido en la interfaz web más utilizada para gestionar Asterisk.

FreePBX es cierto que permite simplificar ciertas acciones de administración, no obstante, el problema viene cuando se necesita hacer algo no previsto por la interfaz web, y es entonces donde podemos ver una limitante de la interfaz.

Los usuarios de distribuciones controladas por interfaces web suelen olvidar que toda interfaz web es creada con un objetivo: simplificar la configuración y gestión de una aplicación (en este caso, de Asterisk), esta simplificación tiene un efecto muy negativo, impide realizar tareas que no han sido previamente preparadas por los creadores de dicho interfaz o incluso han sido desechadas por su complejidad y poco útil

para un público general, se dice entonces que esa distribución está a merced de lo que permita hacer el interfaz web.

Este tipo de problemas son los que han hecho ver a FreePBX una herramienta limitada pero hay que tener en cuenta que FreePBX no limita tan solo a una administración web, más bien da la oportunidad de hacer modificaciones por consola en archivos donde FreePBX respeta y nos los topa por lo que es seguro escribirlos sin temor a perder los cambios. Estos archivos de configuración se los conoce como “*custom*” y son utilizados únicamente por el usuario.

Esta funcionalidad deja a un lado todas las confusiones que puede haberse creado con respecto a esta interfaz web; FreePBX lo que hace es ayudar en la administración de una manera fácil y sencilla pero si es necesario y la interfaz está limitada en cierta operación, permite escribir los archivos personales del usuario para crear las operaciones que no se las pudo realizar por la interfaz web.

Por ello pese a lo “útil y práctico” que parece FreePBX es importante aprender a configurar Asterisk por consola ya que la limitante no estará en la herramienta FreePBX sino en las capacidades del administrador para dar solución a nuevas operaciones.

Un caso en común es el desarrollo de esta tesis en la cual creamos *scripts* y contextos en los archivos *extensions\_custom.conf* y *sip\_general\_custom.conf*. La implementación total no se la hizo por medio de la interfaz web, el acceso, consulta y escritura en la base de datos se la hizo por consola; pero la interfaz web nos ayudo y facilitó la creación de usuarios, rutas salientes, entrantes y troncales.

## 5.4 CONCLUSIONES

- Se realizó el estudio de los sistemas IVR, sus ventajas y funcionalidades las cuales le han mantenido a este sistema como uno de los más usados en el mercado de telecomunicaciones, especialmente en las centrales telefónicas. Las prestaciones que presta al organizar las llamadas entrantes y salientes, optimización para el intercambio de información y reducción de costos en operación y mantenimiento han hecho de este sistema una herramienta fundamental en las organizaciones para brindar mejores servicios a sus consumidores finales.
- Se estudió el protocolo SIP desde su estructura básica hasta su funcionamiento práctico y se lo aplico en una plataforma de software libre para el desarrollo del IVR. Este protocolo nos permitió tener una comunicación de voz sobre una red de datos, convirtiendo la señal en paquetes de datos y con esto aprovechar todos los beneficios que brinda una comunicación basada en paquetes.
- La aplicación se la realizó bajo la distribución de software libre Elastix y se aprovecho todas los módulos y aplicaciones que brinda como la administración web basada en FreePBX y sus sistemas de servicio preinstalados.
- El servidor sobre el que se monto la central IP posee una dirección IP pública la cual nos permite realizar pruebas por el internet. Además tiene un módulo FXO que es una interfaz física que nos

permite conectarnos directamente a la PSTN. Con esto la central tiene la capacidad de responder llamadas tanto por el internet como por la PSTN. Todas estas características hicieron de nuestra central una solución robusta desarrollándose en un ambiente de trabajo real y con soluciones adicionales como la realizada en este trabajo de tesis.

- Las extensiones configuradas se basaron en el protocolo SIP y tuvieron como elementos finales teléfonos IP y *softphones*. Además de las extensiones SIP se configuró una extensión ZAP la cual tenía como elemento final un teléfono analógico, esto se realizó por medio de un módulo FXS y una tarjeta PCI de cuatro puertos. Con esto logramos visualizar el gran alcance que tiene la telefonía IP al combinar diferentes tecnologías de telefonía en una sola central.
- Se comprobó y estudió la exitosa interacción entre tecnologías a través de interfaces que permiten brindar mayor utilidades a un servicio. Se implementó la interfaz AGI para la conexión entre una base de datos MySQL y la PBX Asterisk, dando una visión de futuras aplicaciones que se pueden generar. Además se integro con un sistema TTS que permitía presentar datos y textos de forma hablada hacia los llamante, incrementando la funcionalidad de la nuestra solución.
- Se implementó exitosamente el sistema IVR-IP con todos los objetivos planteados inicialmente, usando una aplicación en código libre como Asterisk, con protocolo SIP e interactuando con bases de datos MySQL. La arquitectura se compone de un solo servidor donde se encuentra la base de datos y la central, tiene salida hacia la PSTN

e Internet y extensiones que son utilizadas por el IVR como ya se ha descrito a lo largo del presente trabajo, demostrando así su viabilidad y su enorme potencial de aplicación en empresas de diversos ámbitos que requieran brindar una interfaz telefónica completa y amigable a sus clientes complementándose con servicios de consultoría e identificación de usuarios automática.

## 5.5 RECOMENDACIONES

- Se recomienda el uso de centrales IP ya que tienen mayores utilidades que una central privada como la administración web, su bajo costo, funcionalidades de interconexión con otros sistemas, no está limitada por hardware y tiene la capacidad de soportar dispositivos que utilicen otra tecnología
- Refiriéndonos a la parte lógica de la red, se recomienda el uso del protocolo SIP ya que nos permite transportar la voz a través de una red de datos y aprovechar todas las prestaciones que esto trae como realizar llamadas por internet sin ningún costo, tener extensiones remotas que puedan realizar llamadas hacia la PSTN, etc.
- Con respecto al hardware del servidor donde se implemento la central se recomienda realizar un análisis de las llamadas concurrentes y el tráfico en el sistema. En nuestro caso la central se monto sobre un ordenador que tiene 512 (MB) de RAM, sin embargo se recomienda la ampliación a 1GB, para garantizar el correcto desempeño del servidor TTS y brindar calidad de servicio a las llamadas. Con respecto al disco duro el servidor actual tiene una

capacidad de 40 GB de disco, esto únicamente para lo que es el sistema operativo, los archivos de configuración y las grabaciones del sistema, sin embargo si se van a realizar grabaciones de llamadas se recomienda colocar un disco externo para este fin.

- Dada la importancia del ancho de banda y la calidad de voz en un sistema de comunicación, se recomienda utilizar un *codec* que realice una compresión para que no se consuma mucho ancho de banda y que la calidad de la voz no se vea afectada.

El *codec* recomendado para redes LAN es el G.711, que no realiza ninguna compresión y su calidad es muy alta, pero la limitantes es su alto consumo de ancho de banda, por lo que no se recomendaría para enlaces remotos. Otro protocolo recomendable para redes LAN es el G.722, aunque es nuevo en el mercado las ventaja que presenta es la frecuencia de muestreo que es el doble que la del *codec* G.711, 16 [kHz] y consumen el mismo ancho de banda, con esto se tiene más muestras de voz que mejoran la fidelidad y calidad de voz.

Por otro lado un *codec* recomendado para enlaces remotos es el G.729 que realiza una compresión ocho veces mayor, es decir, utiliza 8 [kbps] por canal, lo cual disminuye el consumo de ancho de banda ocho veces, la limitante de este *codec* es ser licenciado. Existe una versión liberada <sup>18</sup> que puede ser usada en ciertos países dependiendo de sus políticas. Como respuesta a esto y si el dispositivo final lo soporta se recomienda para enlaces remotos el *codec* GSM que de la misma manera realiza una compresión y disminuye el consumo del ancho de banda.

---

<sup>18</sup> Enlace para descargar la versión libre del *codec* G.729 <http://asterisk.hosting.lv/>



- 
- Para la implementación del servidor PBX en un ambiente real de producción se recomienda el uso de sistemas como Elastix que tiene un administrador web, sistemas preinstalados como TTS, compatibilidad con otras tecnologías y hardware, facilidad de personalización de los módulos, etc.
  - El administrador web utilizado en el presente trabajo fue FreePBX y nos facilitó la administración básica de la central como creación de extensiones, rutas salientes, rutas entrantes, configuración del IVR, etc. Además nos permite ver el estado del sistema, el hardware instalado y ciertos reportes de las extensiones en forma gráfica. Esta administración web no nos limita únicamente a la capacidad de configuración web, también nos da la oportunidad de realizar nuevas aplicaciones en archivos de configuración personales que son respetados por la interfaz web, sin realizar ningún cambio sobre ellos.

## ANEXOS

### Anexo 1: Contextos en el archivo */etc/Asterisk/extensions\_custom.conf*

; This file contains the contexts the agents login for the module call center.  
; and contains the context conferences for module conferences of elastix 1.0.

```
[from-internal-custom]
exten => 1234,1,Playback(demo-congrats) ; extensions can dial 1234
exten => 1234,2,Hangup()
exten => h,1,Hangup()
include => agentlogin
include => conferences
include => calendar-event
include => weather-wakeup
include => custom-agi
include => agi3334
include => agi4445
include => agi5556
include => agi6667
include => agi7778
```

```
[agi3334]
exten => 3334,1,Answer()
exten => 3334,n,AGI(agi3334.php)
exten => 3334,n,goto(ivr-6,s,1)
```

```
[agi4445]
exten => 4445,1,Answer()
exten => 4445,2,AGI(agi4445.php)
exten => 4445,3,goto(from-did-direct,205,1)
```

```
[agi5556]
exten => 5556,1,Answer()
exten => 5556,n,AGI(agi5556.php)
exten => 5556,n,goto(ivr-6,s,1)
```

```
[agi6667]
exten => 6667,1,Answer()
exten => 6667,n,AGI(agi6667.php)
exten => 6667,n,goto(ivr-6,s,1)
```

```
[agi7778]
exten => 7778,1,Answer()
exten => 7778,2,AGI(agi7778.php)
exten => 7778,n,goto(ivr-6,s,1)
```

## Anexo 2: Script agi3334.php

```
#!/usr/bin/php -q
<?php

//llamamos a las librerias de phpagi
require("phpagi/phpagi.php");
//activamos la bandera de enviar todo lo que se realice a consola como modo depuracion
error_reporting(E_ALL);
//creamos una instancia de la clase AGI
$agi = new AGI();
//generamos el comando Answer de Asterisk, este comando siempre lo ejecutamos al inicio de un AGI
$agi->answer();
//Generamos una conexion a la base de datos de mysql
$conexion = mysql_connect("localhost","root","eLaStIx.2oo7") or die("Error: El servidor no puede conectar con la base
de datos");

//Referenciamos el cursor a la base de datos usuario
mysql_select_db("usuario",$conexion);
$query="update datos set id=0";
$_result=mysql_query($query,$conexion);

$aux=0;
//repeticion para 3 intentos
do{
    $aux=$aux+1;
    //Mensaje para por TTS para que introduzca el numero de cedula
    $agi->text2wav('Por favor, ingrese su numero de cedula');
    //enviamos un beep y esperamos 20 segundos en la espera de una respuesta y limitamos el tama ingresado a 10
digitos
    $_result = $agi->get_data('beep', 20000, 10);
    //obtenemos la informacio introducida desde el telefono
    $keys = $_result['result'];
    //si introdujo un valor el cliente entonces creamos el string la consulta mysql y la ejecutamos
if($keys!=""){
    //Buscamos la clave del usuario
    $query = "select clave from datos where ci=\"\$keys\" ";
    $_result = mysql_query($query, $conexion);
    //verificamos si existe un resultado
if($_re = mysql_fetch_array($_result)) {
    //Regresamos de forma hablada la informacion encontrada
    $agi->text2wav("Ha sido encontrado en nuestra base de datos, su clave actual es");
    $agi->say_digits($_re[clave]);
    $query="update datos set id=1 where ci=\"\$keys\"";
    $_result=mysql_query($query,$conexion);
    $aux=5;}
    else{
    //de lo contrario enviamos un mensaje de que la cedula no se encontro
    $agi->text2wav("No ha sido encontrado en nuestra base de datos");}
    mysql_close();}
}while($aux<3);

//Si se finalizaron los 3 intentos
if($aux==3){
    $agi->text2wav('Han finalizado sus intentos. Gracias por llamar, Adios');
    $agi->hangup();}
?>
```

### Anexo 3: Script agi4445.php

```
#!/usr/bin/php -q
<?php

//llamamos a las librerías de phpagi
require("phpagi/phpagi.php");
//activamos la bandera de enviar todo lo que se realice a consola como modo depuracion
error_reporting(E_ALL);
//creamos una instancia de la clase AGI
$agi = new AGI();
//generamos el comando Answer de Asterisk, este comando siempre lo ejecutamos al inicio de un AGI
$agi->answer();
//Generamos una conexion a la base de datos de sql
$conexion = mysql_connect("localhost","root","eLaStIx.2oo7") or die("Error: El servidor no puede conectar con la base
de datos");

//Referenciamos el cursor a la base de datos usuario
mysql_select_db("usuario",$conexion);

$aux=$aux+1;
//enviamos un mensaje para que el usuario ingrese su cedula
$agi->text2wav('Por favor, ingrese su numero de cedula para cargar en nuestra base de datos');
//enviamos un beep y esperamos 20 segundos en la espera de una respuesta y limitamos tamaño a 10 digitos
$_result = $agi->get_data('beep', 20000, 10);
//obtenemos la información introducida desde el telefono
$keys = $_result['result'];

//si introdujo un valor el cliente entonces creamos el string mysql
if($keys!=""){
    //almacenamos la nueva cedula en la base de datos usuario
    $query = "insert into datos (ci,clave,saldo) values (\\"$keys\\", \" \", \" \");
    $_result = mysql_query($query, $conexion);
}

//enviamos un mensaje para que el usuario ingrese su clave
$agi->text2wav('Por favor, ingrese su contrasenia de cuatro digitos');
//enviamos un beep y esperamos 10 segundos en la espera de una respuesta y limitamos el tamaño a 4 digitos
$_result = $agi->get_data('beep', 10000, 4);
//obtenemos la información introducida desde el telefono
$key = $_result['result'];

//si introdujo un valor el cliente entonces creamos el string mysql
if($key!=""){
    //almacenamos la nueva clave en la base de datos usuario
    $query = "update datos set clave=$key where ci=$keys";
    $_result = mysql_query($query, $conexion);
}
mysql_close();

//regresamos al contexto
$agi->text2wav('A continuacion le transferiremos a una operadora para completar nuestra hoja de datos');

?>
```

## Anexo 4: Script agi5556.php

```
#!/usr/bin/php -q
<?php
//llamamos a las librerias de phpagi
require("phpagi/phpagi.php");
//activamos la bandera de enviar todo lo que se realice a consola como modo depuracion
error_reporting(E_ALL);
//creamos una instancia de la clase AGI
$agi = new AGI();
//generamos el comando Answer de Asterisk, este comando siempre lo ejecutamos al inicio de un AGI
$agi->answer();
//Generamos una conexion a la base de datos de mysql
$conexion = mysql_connect("localhost","root","eLaStIx.2oo7") or die("Error: El servidor no puede conectar con la base
de datos");

//Referenciamos el cursor a la base de datos usuario
mysql_select_db("usuario",$conexion);

do{
//Mensaje para que el usuario ingrese la nueva clave
$agi->text2wav('Ingrese su nueva clave de 4 digitos ahora');
$_result = $agi->get_data('beep', 10000, 4);
$cl = $_result['result'];
//Decimos de forma hablada la clave ingresada
$agi->text2wav("Su clave ingresada es");
//Clave hablada digito por digito
$agi->say_digits($cl);
//Mensaje para aceptar la nueva clave o no
$agi->text2wav("Si esta de acuerdo marque uno, caso contrario presione cualquier tecla");
$_result = $agi->get_data('beep', 1000, 1);
$aux1 = $_result['result'];
//Validamos si el usuario acepta la nueva clave ingresada o no
}while($aux1!=1);

//Almacenamos la nueva clave en la base de datos
$query="update datos set clave=$cl where id=1";
$_result=mysql_query($query,$conexion);
//Mensaje de cambio correcto de clave
$agi->text2wav('Su cambio se ah realizado con exito');

?>
```

## Anexo 5: Script agi6667.php

```
#!/usr/bin/php -q
<?php

//llamamos a las librerias de phpagi
require("phpagi/phpagi.php");
//activamos la bandera de enviar todo lo que se realice a consola como modo depuracion
error_reporting(E_ALL);
//creamos una instancia de la clase AGI
$agi = new AGI();
//generamos el comando Answer de Asterisk, este comando siempre lo ejecutamos al inicio de un AGI
$agi->answer();
//Generamos una conexion a la base de datos de mysql
$conexion = mysql_connect("localhost","root","eLaStIx.2oo7") or die("Error: El servidor no puede conectar con la base
de datos");

//Referenciamos el cursor a la base de datos usuario
mysql_select_db("usuario",$conexion);

$aux=1;
//VALIDACION DE USUARIO POR MEDIO DE LA CLAVE
do{
//Mensaje para que el usuario ingrese la clave
if($aux==1)$agi->text2wav('Por favor ingrese su clave');
else $agi->text2wav('Clave mal ingresada, Por favor ingrese su clave');
//Usuario ingresa clave
$_result=$agi->get_data('beep',10000,4);
$keys=$_result['result'];
//Buscamos la clave del usuario en la base de datos
$query="select clave from datos where id=1";
$_result=mysql_query($query,$conexion);
$_re=mysql_fetch_array($_result);
$aux=0;
//Comparamos clave ingresada con la clave de la base de datos
}while($keys!=$_re[clave]);

//Usuario registrado
$agi->text2wav('Usuario Registrado');

//BUSCA DEL SALDO
//Referenciamos el cursor a la base de datos promo
$query = "select saldo from datos where id=1 ";
$_result = mysql_query($query, $conexion);
//verificamos si existe un resultado
if($_re = mysql_fetch_array($_result)) {
//Regresamos de forma hablada la informacion encontrada
$agi->text2wav("Su saldo a pagar durante el proximo mes es $_re[saldo] dolares");
}

?>
```

## Anexo 6: Script agi7778.php

```
#!/usr/bin/php -q
<?php

//llamamos a las librerias de phpagi
require("phpagi/phpagi.php");
//activamos la bandera de enviar todo lo que se realice a consola como modo depuracion
error_reporting(E_ALL);
//creamos una instancia de la clase AGI
$agi = new AGI();
//generamos el comando Answer de Asterisk, este comando siempre lo ejecutamos al inicio de un AGI
$agi->answer();
//Generamos una conexion a la base de datos Mysql
$conexion = mysql_connect("localhost","root","eLaStIx.2oo7") or die("Error: El servidor no puede conectar con la base
de datos");

//Referenciamos el cursor a la base de datos usuario
mysql_select_db("usuario",$conexion);

$aux=1;
//VALIDACION DE USUARIO POR MEDIO DE LA CLAVE
do{
//Mensaje para que el usuario ingrese la clave
if($aux==1)$agi->text2wav('Por favor ingrese su clave');
else $agi->text2wav('Clave mal ingresada, Por favor ingrese su clave');
//Usuario ingresa clave
$_result=$agi->get_data('beep',10000,4);
$keys=$_result['result'];
//Buscamos la clave del usuario en la base de datos
$query="select clave from datos where id=1";
$_result=mysql_query($query,$conexion);
$_re=mysql_fetch_array($_result);
$aux=0;
//Comparamos clave ingresada con la clave de la base de datos
}while($keys!=$_re[clave]);

//Usuario registrado
$agi->text2wav('Usuario Registrado');

//BUSCA DE PROMOCIONES

//Referenciamos el cursor a la base de datos promo
mysql_select_db("promo",$conexion);
$aux=1;
$query = "select descripcion from cosas where id=$aux;";
$_result = mysql_query($query, $conexion);
$_re = mysql_fetch_array($_result);
while($_re != 0){
if ($aux==1)$agi->text2wav("Las promociones con las que cuenta la empresa son");
$agi->say_digits("$aux");
//Busca las promociones disponibles en la base de datos
$query = "select descripcion,valor,dias from cosas where id=$aux;";
$_result = mysql_query($query, $conexion);
$_re = mysql_fetch_array($_result);

//Regresamos de forma hablada la informacion encontrada
$agi->text2wav("$_re[descripcion], el valor por unidad es $_re[valor] dolares, la promocion dura por los proximos
$_re[dias] dias");
```

```
$aux=$aux+1;
$query = "select descripcion from cosas where id=$aux;";
$_result = mysql_query($query, $conexion);
$_re = mysql_fetch_array($_result);
}
//Mensajes para finalizar la consulta
if ($aux!=1) $agi->text2wav("No existen mas promociones");
if ($aux==1) $agi->text2wav("No existen promociones actualmente");

?>
```



## Anexo 7: Comandos AGI

Una lista de los principales comandos de la función AGI.

Comando	Descripción
answer	Answer channel
channel status	Returns status of the connected channel
database del	Removes database key/value
database deltree	Removes database keytree/value
database get	Gets database value
database put	Adds/updates database value
exec	Executes a given Application
get data	Prompts for DTMF on a channel
get full variable	Evaluates a channel expression
get option	Stream file, prompt for DTMF, with timeout
get variable	Gets a channel variable
hangup	Hangup the current channel
noop	Does nothing
receive char	Receives one character from channels supporting it
receive text	Receives text from channels supporting it
record file	Records to a given file
say alpha	Says a given character string
say digits	Says a given digit string
say number	Says a given number
say phonetic	Says a given character string with phonetics
say date	Says a given date
say time	Says a given time
say datetime	Says a given time as specified by the format given

send image	Sends images to channels supporting it
send text	Sends text to channels supporting it
set autohangup	Autohangup channel in some time
set callerid	Sets callerid for the current channel
set context	Sets channel context
set extension	Changes channel extension
set music	Enable/Disable Music on hold generator
set priority	Set channel dialplan priority
set variable	Sets a channel variable
stream file	Sends audio file on channel
control stream file	Sends audio file on channel and allows the listner to control the stream
tdd mode	Toggles TDD mode (for the deaf)
verbose	Logs a message to the asterisk verbose log

## REFERENCIAS

- [SIP2008] Columbia University – Computer Science  
Protocolo SIP  
URL: <http://www.cs.columbia.edu/sip/>
- [RFC2543] Handley, et al, ‘SIP: Session Initiation Protocol’  
IETF, Junio de 1999  
URL: <http://www.ietf.org/rfc/rfc2543.txt>
- [RFC3261] Rosenberg, et. al, ‘SIP: Session Initiation Protocol’  
IETF, Junio del 2002  
URL: <http://www.ietf.org/rfc/rfc3261.txt>
- [RFC3263] Rosenberg, et. al, ‘SIP: Session Initiation Protocol’  
IETF, Junio del 2002  
URL: <http://www.ietf.org/rfc/rfc3263.txt>
- [EBSIP] Entidades Básicas SIP  
URL: [http://es.wikitel.info/wiki/Entidades\\_b%C3%A1sicas\\_SIP](http://es.wikitel.info/wiki/Entidades_b%C3%A1sicas_SIP)
- [GTI2003] SIP – IRIS-MMEDIA  
XV Grupo de Trabajo de RedIRIS Mayo 2003  
Contacto: Federico Montesino Pouzols <[fedemp@altern.org](mailto:fedemp@altern.org)>
- [ATT2005] Van Meggelen Jim et al. ‘Asterisk The future of Telephony’  
O’Reilly, 2005
- [BER2006 ] Technical University of Berlin – GSM Lossy Speech Library  
URL: <http://user.cs.tu-berlin.de/~jutta/toast.html>  
Información de contacto: [jutta@pobox.com](mailto:jutta@pobox.com)

- [CTI2003] Yarberry William A., 'Computer Telephony Integration'  
Segunda Edición  
CRC Press, 2003
- [TCC2007] Estudio sobre la Infraestructura Tecnológica de los Centros de  
Contacto en México. Octubre – Noviembre 2007.  
URL:[http://www.mundo-contact.com/revistas/mundocontact/55\\_investigacion.php](http://www.mundo-contact.com/revistas/mundocontact/55_investigacion.php)  
Consejo: Argüelles R, Ayala F, Corona J, Suárez C, Ugalde V, Valle.  
Mundo – Contact.
- [FRE2009] Sitio oficial de FreePBX  
URL: <http://www.FreePBX.org/support/>  
Secciones: documentation, module-documentation, ivr-digital-  
receptionist
- [WIE2001] Rhoton John, 'The Wireless Internet Explained'  
Capítulo III  
Digital Press, 2001-11-15
- [IPKST] Barry William J., van Dommelen Wim A., 'The Integration of Phonetic  
Knowledge in Speech Technology'  
Pag: 67-87  
Springer
- [HCI2003] Jacko Julie, Sears Andrew, 'The Human Computer Interaction  
Handbook'  
Pag: 173-186  
Human Factors and Ergonomics

- [FXM2009] Software based PBX for Windows – 3CX  
URL: <http://www.3cx.com/PBX/FXS-FXO.html>  
2009, 3CX
- [MySQL5] Thibaud Cyrill, ‘MySQL 5: Instalación, Implementación,  
Administración y Programación’  
Capitulo I  
Ediciones ENI  
Ref. ENI : RIT5MYS
- [SBD2003] Rob Peter, Coronel Carlos, ‘Sistemas de bases de datos: diseño,  
implementación y administración’  
Capitulo I  
Editorial : INTERNATIONAL THOMSON PARANIN  
Fecha de Publicación : 01/01/2003
- [AST2007] Van Meggelen Jim, Madsen Leif, Smith Jared, ‘Asterisk The Future of  
Telephony’  
Publicado por O’Reilly Media, Inc.  
Agosto 2007 – Segunda Edición
- [VoIPTA] Mahler Paul, ‘VoIP Telephony with Asterisk’  
ISBN 09759992-0-6  
Mahler, P.S.
- [ADS2.0] Viegas Eduardo, Correa Facundo, ‘Asterisk Desconsolado’  
Versión 2.0  
Asterio
- [ELXCJM] Central Local Elastix  
Dirección IP Local: 192.168.1.254/24

Dirección IP Pública: DHCP  
Dominio: tesisjm.no-ip.org  
Herramienta web: <https://tesisjm.no-ip.org>

- [OPT150] Sitio Oficial de Siemens Enterprise  
Teléfono Optipoint 150S  
[http://wiki.siemens-enterprise.com/index.php/optiPoint\\_150\\_S](http://wiki.siemens-enterprise.com/index.php/optiPoint_150_S)
- [VOXDAT] Voxdata Comunicaciones IP  
IVR basados en Asterisk  
URL: <http://www.voxdata.com.ar/voxivr.html>
- [CCT2009] Call-Center-Tech is Trademarked by Database Systems Corp.  
Última modificación: 2009  
IVR Management.  
URL: <http://www.call-center-tech.com/ivr-management.htm>
- [BEA2006] Beasty Colin, 'INTERACTIVE VOICE RESPONSE  
Customer Relationship Management'; Apr 2006; 10, 4; ABI/INFORM  
Global. pag. 27
- [FRO2006] Frost & Sullivan, 'U.S. IVR (Interactive Voice Response) System  
Markets'  
URL: <http://www.researchandmarkets.com/reports/358843>  
Frost & Sullivan, Febrero del 2006
- [CUE2008] Edgar Landívar, 'Comunicaciones Unificadas con Elastix' Volumen 1  
Primera Edición, versión Beta  
Descarga: [http://www.elastix.org/index.php?option=com\\_content&task=view&id=137&Itemid=60](http://www.elastix.org/index.php?option=com_content&task=view&id=137&Itemid=60)

Fecha de entrega:

Sangolquí, \_\_\_\_ de Noviembre del 2009

---

Edwin Jhair Mena Jácome

AUTOR

---

Ing. Gonzalo Olmedo

DIRECTOR DE CARRERA