INFORME FINAL
PRACTICA EMPRESARIAL EN TYT LTDA
TECNOLOGIAS DE INFORMACION Y TELECOMUNICACIONES

JULIO CESAR RUIZ RUIZ
CÓDIGO 1991161

UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FISICOMECANICAS
ESCUELA DE INGENIERÍA DE SISTEMAS
BUCARAMANGA
2005

INFORME FINAL
PRACTICA EMPRESARIAL EN TYT LTDA
TECNOLOGIAS DE INFORMACION Y TELECOMUNICACIONES

PRACTICANTE EMPRESARIAL:
JULIO CESAR RUIZ RUIZ
CÓDIGO 1991161

Este proyecto es presentado como requisito para optar al título de Ingeniero de Sistemas e Informática

DIRECTOR - TUTOR EMPRESARIAL:
OSCAR GONZALEZ PIMENTEL
PROFESIONAL SENIOR DE SISTEMAS

UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FISICOMECANICAS
ESCUELA DE INGENIERÍA DE SISTEMAS
BUCARAMANGA
2005

AGRADECIMIENTOS

La verdad, se hace tedioso nombrar un determinado número de personas, puesto que muy es muy probable, que siendo alto el esfuerzo realizado, la misma lista no llegue a quedar completa. Sin embargo, con el ánimo de iniciar por alguien, creo que Dios todopoderoso es quien debe encabezar mis agradecimientos, ya que estoy totalmente convencido de que sin su ayuda, mis metas no hubiesen llegado a feliz término.

Es preciso, también dar crédito a los accionistas de TYT Ltda, encabezados por el Ingeniero Sergio Oswaldo Cajías. Todos y cada uno de ellos, significan un sinnúmero de conocimientos altamente profesionales y morales, quienes no permitieron que ésta práctica empresarial fuera simple y sin resultados reales. Gracias Cajías por su amistad y mi más sincero reconocimiento a la positiva influencia que ejerció sobre mi, el equipo de trabajo de TYT Ltda.

Dado que un equipo de trabajo, no se compone solamente de accionistas, pienso que el contacto frecuente con mis compañeros practicantes Marino Pérez y Gerson Samaniego, también debe ser agradecido, ya que surgió una positiva realimentación entre nosotros tanto a nivel profesional como personal. Gracias a ellos y también a Linderman Moncada, Oscar Mauricio Murillo y Jason Fabián Becerra (compañeros de EISI - UIS) por acompañarme en la empresa y brindarme su amistad.

Es importante también recordar que mi familia, mis amigos de infancia y mi novia, se constituyeron en un soporte vital para lo que se conoce como inteligencia emocional; por eso, genial que permanecieran conmigo mis padres y hermanas, José Miguel Aguilera y Libia Mogollón.

Simplemente gracias a ellos… y a todos los que estuvieron conmigo.

TITULO: PRACTICA EMPRESARIAL EN TYT LTDA *

AUTOR: JULIO CESAR RUIZ RUIZ **

PALABRAS CLAVE: VoIP, Voz sobre IP, ORACLE, PL/SQL, Visual Basic, Contact Center, RUP.

DESCRIPCIÓN

Una práctica empresarial tiene diferentes definiciones. Se trata principalmente de lograr una aproximación al quehacer profesional, por parte de un estudiante de pregrado que tiene acumulada una serie de conocimientos teóricos, logrando una confrontación entre la teoría académica y el ejercicio práctico en el sector productivo. Dicha "aproximación", tuvo relación principalmente con tres áreas de trabajo, que se pueden resumir como se muestra a continuación:

La primera área, *Investigación Descriptiva* está relacionada directamente con Voz sobre IP. Esta investigación, de acuerdo con el cronograma de actividades del Plan de Proyecto de Grado, acaparó gran parte del tiempo total de la práctica. Se investigó entonces sobre diferentes ítems relacionados, tales como: Antecedentes, tendencias, ventajas, estándares, regulación, entre muchos otros que sin lugar a duda contribuyeron con el objetivo principal del Trabajo de Grado. La segunda parte, *Diseño de Software* se centró en la definición de la arquitectura, levantamiento de requerimientos, investigación sobre metodologías de desarrollo, entre otras tareas, orientadas específicamente hacia la implementación de dos aplicaciones: "Generador de Código Oracle" y "Software VoIP". Finalmente la tercera parte, *Desarrollo de Software* hace alusión a la implementación de los dos sistemas diseñados previamente, en donde se hizo uso de tecnologías de punta como Oracle 9i y Microsoft Visual Basic. Se siguieron siempre los lineamientos de las "mejores prácticas" de programación. El código fuente se documentó y se realizó una serie de pruebas al software, para así garantizar que el producto final mostraba el desempeño esperado.

Lógicamente, hubo otras actividades adicionales a las anteriormente mencionadas que se realizaron durante los seis meses de la práctica empresarial (desde 02/07/2004 hasta 31/01/2005) tales como capacitación, acercamientos con profesionales reconocidos, entre otras que vale la pena tener en cuenta.

* Trabajo de Grado

** Escuela de Ingeniería de Sistemas e Informática.
Director/Tutor Empresarial: Ing. Oscar González Pimentel.

TITLE: ENTERPRISE PRACTICE IN TYT LTDA *

AUTHOR: JULIO CESAR RUIZ RUIZ **

KEY WORDS: VoIP, Voice over IP, ORACLE, PL/SQL, Visual Basic, Contact Center, RUP.

DESCRIPTION

One enterprise practice has different definitions. It is mainly to achieve an approach to the professional chore, by a predegree student who has a lot of teoric knowledge, achieving a confrontation between academic teoric and practice exercise in the productive sector. This “approach” had a relationship mainly with three work areas:

The *first part*, Descriptive Investigation is related directly with Voice Over IP. This investigation was agree with the activities chronogram, shown in “Plan de Proyecto de Grado”, and it took so much time from total time in practice. Then, there was investigation about different related issues such as: Successful cases, tendencies, advantages, standards, reglamentation and so much others that without doubt, helped to the main target of Work Grade. The *second part*, Software Design was centered in architecture definition, requeriments construction, investigation about development methodologies and other tasks, specificly oriented to implementation of “Generador de Código Oracle” and “Software VoIP”. Finally the *third part*, Software Development is focused to the implementation of two systems previously designed, where last technologies like Oracle 9i and Microsoft Visual Basic were used. The better practices principies always were followed. The source code was documented and a test-group was done over the software, to warrant the wished performance.

Obviously, there were other activities apart to previously said, that were realized through the six months in the enterprise practice (since 02/07/2004 to 31/01/2005), such as capacitation, nearings with important professionals, and other things no less fantastic.

* Graduation Work

** Escuela de Ingeniería de Sistemas e Informática.
Director/Enterprise Tutor: Ing. Oscar González Pimentel.

CONTENIDO

Pág.

INTRODUCCIÓN.

1. EL PROYECTO HERRAMIENTA SOFTWARE PARA CONTACT CENTER BASADO EN EL PROTOCOLO IP. 13

1.1. RESUMEN DEL PROYECTO. 13
1.2. PLANTEAMIENTO DEL PROBLEMA. 14

2. MARCO TEORICO SOBRE VOZ IP. 18

2.1. ANTECEDENTES. 18
2.1.1. Trabajos Realizados. 18
2.1.2. Historia de Éxito: Norstar Al Rescate. 21
2.2. TENDENCIAS. 23
2.3. VENTAJAS. 24
2.4. TELEFONIA IP. 26
2.4.1. ¿Qué es Telefonía IP? 26
2.4.2. ¿Cómo funciona la Telefonía IP? 26
2.4.3. Telefonia IP Vs PSTN. 27
2.4.4. ¿Por qué es más barata la Telefonía IP? 27
2.4.5. Historia de Éxito: Pontificia Universidad Católica. 28
2.4.6. Ventajas de la Telefonía IP. 31
2.5. ESTÁNDARES ASOCIADOS. 31
2.5.1. H. 323. 32
2.5.2. SIP. 33
2.5.3. Comparacion entre H.323 Y SIP. 35
2.6. REGULACION. 36
2.6.1. En Colombia. 36
2.6.2. En Otros Países. 37
2.7. CALIDAD DE LA VOZ. 38
2.7.1. Parámetros Involucrados. 40
2.7.2. Soluciones a Nivel de Aplicación. 41
2.8. ELEMENTOS DE REDES DE VOZ SOBRE IP. 42

3. SOFTWARE DISEÑADO Y DESARROLLADO. 45

3.1. SOFTWARE VOIP. 46
3.1.1. Definición de Archivo DLL. 48
3.1.2. Acerca de WaveStream.dll. 49
3.1.3. Interfaz de Software VoIP. 50
3.2. GENERADOR DE CÓDIGO ORACLE. 51
3.2.1. Introducción a las Herramientas Generadoras de Código. 51
3.2.2. Características de las Herramientas Generadoras de Código. 53
3.2.3. Paquetes en Oracle. 54
3.2.4. Acerca de GeneraTYT. 55
3.2.4.1. Algunos Aspectos Técnicos. 56
3.2.4.2. Algunos Aspectos Generales para el Usuario. 57

4. OTRAS ACTIVIDADES REALIZADAS. *60*

4.1. CAPACITACION ORACLE. 60
4.2. INVESTIGACION METODOLOGIAS DE DESARROLLO. 62
4.2.1. Proceso Unificado de Desarrollo de Software (RUP). 62
4.2.2. Programación Extrema. 63
4.3. SINTESIS DE LOS INFORMES DE AVANCE. 64

5. GLOSARIO. 71

5.1. ACRONIMOS. 71
5.2. TERMINOS. 72

CONCLUSIONES 74
BIBLIOGRAFIA 76
ANEXOS 79

LISTA DE TABLAS

Pág.

Tabla 1.	Componentes de H. 323.	33
Tabla 2.	Partes de GeneraTYT.	56
Tabla 3.	Componentes de la arquitectura de GeneraTYT v.1.0.	56
Tabla 4.	Campos de Desempeño del Ingeniero de Sistemas, según la UIS.	66

LISTA DE FIGURAS

Pág.

Figura 1.	CC Tradicional.	15
Figura 2.	Ejemplo de configuración sobre el software de Brujula.NET.	20
Figura 3.	Nashville Area Chapter de la Cruz Roja Americana.	22
Figura 4.	Resultados principales arrojados por encuesta de IDC Latinoamérica.	25
Figura 5.	Resultados secundarios arrojados por encuesta de IDC Latinoamérica.	25
Figura 6.	Intercambio de mensajes para establecer una comunicación con protocolo SIP.	34
Figura 7.	Ejemplo de red con conexión de centralitas a routers CISCO que disponen de soporte VoIP.	43
Figura 8.	Elementos de una red VoIP.	43
Figura 9.	Esquema básico de funcionamiento del software VoIP.	47
Figura 10.	Soporte para Audio.	49
Figura 11.	Interfaz de Software VoIP.	50
Figura 12.	Campo de Acción o ámbito de RUP.	62

LISTA DE ANEXOS

Pág.

ANEXO A. Funciones de TCP y Chat utilizadas en Sofware VoIP. 79

ANEXO B. Specification del Query Package de GeneraTYT v.1.0. 83

INTRODUCCION

TYT Ltda es una empresa de base tecnológica que principalmente se dedica al desarrollo de software. Se identificó una necesidad en el mercado nacional e internacional, la cual tenía que ver con las Telecomunicaciones y los Centros de llamadas y/o Centros de Contacto.

Un Contact Center (CC en adelante) puede asociarse con un centro de acopio, al cual se lleva cierta información de diferentes empresas o entidades, para que una población objetivo determinada pueda realizar consultas sobre esa información, básicamente por PSTN o INTERNET buscando siempre convergencia entre las redes de voz y datos de dicha central, además de brindar velocidad para el usuario que finalmente experimenta la veracidad de los resultados.

En el trascurso de la práctica empresarial se cubrió la fase de investigación que tenía prevista TYT Ltda, con el objeto de posteriormente realizar una herramienta software que permitiese reemplazar en alta medida, el hardware utilizado normalmente en un CC.

Con este objetivo en mente y también el de obtener un título profesional por parte del practicante, se logró adquirir experiencia y afianzar los conocimientos teóricos que se traían desde el pregrado universitario, entre otras cosas como perder el temor que normalmente se tiene al enfrentar el medio laboral y así finalmente prestar un servicio al sector empresarial y a la sociedad.

Siendo lo anterior parte del propósito integral del presente trabajo de grado, debe hacerse énfasis en que ésta práctica empresarial se ha hecho además con el objeto de conocer acerca de la revolución VoIP, reconocer a los grandes productores de VoIP, tener en cuenta los adelantos hechos en Colombia, y así poder tomar este conocimiento y adaptarlo a la solución que se va planeó desarrollar.

1. EL PROYECTO HERRAMIENTA SOFTWARE PARA CONTACT CENTER BASADO EN EL PROTOCOLO IP

1.1. RESUMEN DEL PROYECTO.

El mercado altamente competitivo de hoy en día hace que las empresas le den un trato personalizado a cada cliente ya que los costos de mantener o adquirir uno nuevo son considerables, además la relación debe ser bidireccional para que exista retroalimentación. Estas relaciones se entablan a través de diferentes canales de comunicación, sin embargo cuando el número de clientes es considerable se hace necesario administrarlos en forma eficiente haciendo uso de tecnología especializada como la que se emplea en los Call Center.

Un Call Center es un centro especializado que une la telefonía tradicional y los diferentes medios de comunicación con la tecnología de los sistemas de información, permitiendo manejar grandes volúmenes de llamadas tanto de entrada como de salida, con personal especializado, software e información oportuna, administrando eficientemente la relación de las empresas con sus clientes.

Con el avance que ha tenido últimamente Internet surgen nuevos medios de comunicación con los clientes rompiendo las barreras geográficas y horarias, es decir, los clientes están en cualquier parte del mundo y se comunican a cualquier hora utilizando medios tales como el chat, el video metting, el mail, el fax, etc. lo que ha provocado que los Call Center evolucionen y no solo atiendan contactos a través del teléfono sino también a través de estos otros medios, cambiado su denominación hacia Centros de Contacto o Contact Center (en adelante CC).

A pesar de los grandes beneficios que un CC le puede brindar a cualquier tipo de Empresa, estos solo han sido implementados por una fracción del sector empresarial debido principalmente a los costos de adquisición y mantenimiento de la tecnología requerida.

La mayor parte de los CC que existen en Colombia están conformados por módulos hardware separados que se integran a través de aplicaciones y sistemas propietarios, por lo cual su implementación requiere el desarrollo de interfaces que incrementan los costos de la solución, la complejidad, los tiempos de montaje y el mantenimiento creando fuertes lasos de dependencia con los proveedores extranjeros de las maquinas. Además a la hora de crecer es necesario adquirir más equipos, actualizar el software e incrementar el licenciamiento de las

aplicaciones. El presente es un proyecto de desarrollo tecnológico que va a mejorar la tecnología existente en el país de soluciones CC a través de la creación de una herramienta software modular que realice las funciones tanto operativas como administrativas aprovechando las ventajas de la voz sobre IP y la puesta en el mercado por parte de los grandes fabricantes de equipos para telecomunicaciones de maquinas que conectan la telefonía publica con el mundo IP.

Una solución para CC de las características aquí descritas se basa en su mayoría en componentes software, lo que reduce los costos de implementación, facilita el montaje, la administración, el crecimiento y disminuye la dependencia existente con los proveedores de hardware.

Para el desarrollo del proyecto se ha conformado un equipo de trabajo que no solo cuenta con experiencia reconocida en el desarrollo de productos software de propósito industrial, sino que también tiene experiencia en el montaje y administración de un Call Center en Telebucaramanga y pleno conocimiento de la tendencia tecnológica especialmente en telecomunicaciones, lo que ha permitido dimensionar los objetivos del proyecto y estimar los recursos necesarios para su consecución.

1.2. PLANTEAMIENTO DEL PROBLEMA

La mayoría de los CC existentes en Colombia están conformados por la integración de diversos componentes que cumplen funciones determinadas (ver Figura 1) así:

- ✓ **PSTN – Red de Telefonía de servicio Público:** Es el servicio publico que ofrece uno o varios operadores y permite la comunicación de un cliente con el CC a través de una llamada local o de larga distancia.

- ✓ **Planta Telefónica:** Son los equipos que reciben los canales de comunicación PBX o canales E1 suministrador por el operador local y permiten administrar las redes y comunicaciones de voz al interior del CC.

- ✓ **CTI – Integración entre la Telefonía y la Computación:** Es quien relaciona la telefonía con los sistemas de información relacionando los datos de un cliente con la información asociada a una llamada tales como el número marcado, el número de donde se llama, etc. y permite que a un agente se le despliegue en pantalla toda la información correspondiente al contacto que va a atender.

- ✓ **IVR – Respuesta Interactiva de Voz:** Es un sistema con la capacidad de capturar o entregar información en forma automática a través del teléfono con el que se comunica un cliente, interpretando la voz o los tonos marcados y realizando búsquedas en bases de datos para suministrar la información requerida.

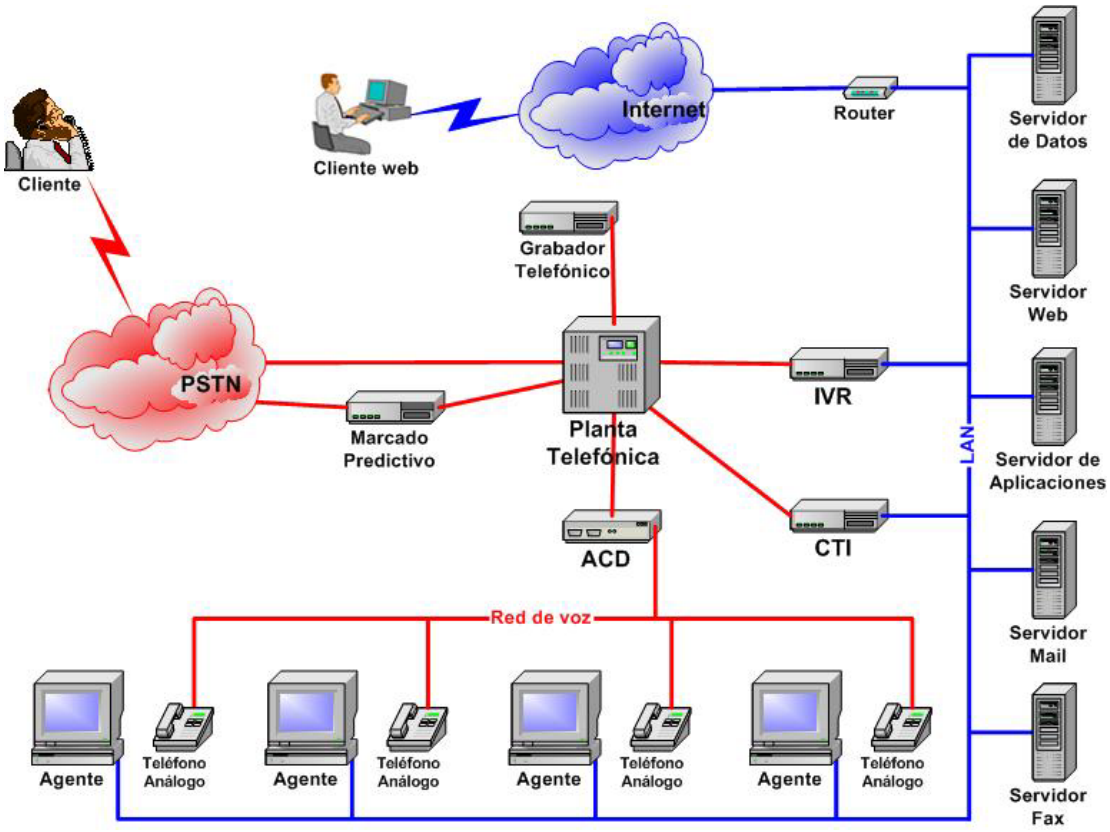

***Figura 1**. CC Tradicional.*

- ✓ **ACD – Distribuidor Automático de Llamadas:** Distribuye automáticamente las llamadas entrantes y salientes a un pool de agentes disponibles. Esta distribución es inteligente de tal forma que los agentes presenten una carga de trabajo homogénea y se realiza con base en algoritmos que dependen del tipo de campaña al que pertenezca cada llamada.

- ✓ **Marcador Predictivo (PD Predictive Dialer):** Es quien administra las estrategias de colas y enrutamiento del tráfico de llamadas. En general, en un entorno manual los agentes destinados a generar llamados para telemarketing o telecobranzas pierden mucho tiempo al discar, escuchar

tonos ocupados, rings de espera, grabaciones, etc. Estos tiempos muertos provocan índices de productividad estimados en el orden del 30% (fracción del tiempo que los agentes se encuentran hablando por teléfono y actualizando registros). Los sistemas de discado predictivo elevan dicha productividad a rangos superiores al 90 %. La filosofía de un Marcador predictivo esta basada en su algoritmo de predicción del resultado de cada llamada a efectuar.

- ✓ **Grabador de Llamadas:** Es el sistema que permite grabar y administrar la grabación de las llamadas que sostienen los agentes con los clientes según el tipo de campaña o controles que se le definan.

- ✓ **Router:** Es la máquina que permite el enrutamiento del trafico tanto entrante como saliente vía Internet.

- ✓ **Servidores:** Son los equipos de computo que a través de software y bases de datos cumplen funciones específicas. Pueden funcionar bajo una o varias máquinas dependiendo del volumen de información y trafico que manejen.

- ✓ **LAN - Red de Área Local:** Es la red de comunicaciones IP por la que se transmiten datos al interior del CC y permite la interconexión de los equipos de los agentes con los distintos servidores de información y aplicaciones.

- ✓ **Red de Voz:** Es la red que permite interconectar cada teléfono con la planta telefónica.

En términos generales estas soluciones están basadas en hardware ya que la mayoría de sus componentes son maquinas, que cada una cumple una función específica e interactúan a través de interfaces propietarias que suministran los mismos proveedores, lo cual genera los siguientes problemas:

a) **Inversiones cuantiosas:** Son soluciones costosas en su implementación (El valor promedio de implementación inicial de un CC con la arquitectura descrita es de 10.000 Us por puesto de trabajo[1])

b) **Costo por obsolescencia tecnológica:** Altos costos de actualización por cuanto son tecnologías con vida útil media de cinco años.

[1] Ricardo Durán, presidente de la Asociación Colombiana de *Call Center*, hizo un análisis del estado del sector en nuestro país durante el Primer Congreso Andino de Call Centers y Contact Centers Marzo/2004. "Las empresas prestadoras de este servicio ofrecen hoy en día más de 10.000 estaciones de trabajo con inversiones superiores a los cien millones de dólares"

c) **Crecimiento costoso:** Incorporar mas agentes o aumentar la funcionalidad implica altas inversiones ya que esto se realiza adquiriendo más equipos y más licenciamiento de software.

d) **Dependencia del proveedor:** Dependencia total de la tecnología extranjera por cuanto son maquinas que no se producen en el país y el soporte y mantenimiento depende totalmente del proveedor

e) **Limitación funcional:** Limitación en la funcionalidad debido a que el Hardware viene con unas especificaciones estrictas y difícilmente se pueden implementar nuevos requerimientos con las mismas máquinas.

f) **Soluciones complejas:** Son soluciones complejas debido a la multiplicidad de tecnologías e interfaces requeridas

g) **Dificultad administrativa:** La administración la debe realizar personal especializado el cual es un recurso escaso.

h) **Administración de varias redes:** Son soluciones que manejan dos tipos de redes distintas (voz y datos) cuya integración se realiza con componentes propietarios.

Dentro de la operación de un *Contact Center* es necesario intercomunicar un grupo determinado de agentes (al interior del Contact Center) con un grupo específico de usuarios potenciales (al exterior del mismo). Una forma de establecer tal comunicación es utilizando los distintos métodos para transportar y manipular la *Voz sobre IP*.

Los dos puntos anteriormente mencionados no pretenden extender mucho el contenido de éste capítulo, pero si se quiere brindar una idea acerca del contexto sobre el cual se concibió la práctica empresarial.

2. MARCO TEORICO SOBRE VOZ IP

Al interior del presente capítulo lo que se pretende no es precisamente abarcar todos los subtemas relacionados con Voz IP. Si es deseable, en otro orden de ideas, tratar en este libro ciertos ítems de alta relevancia, tanto para TYT Ltda como para la UIS, que se trabajaron ampliamente durante el período en práctica empresarial y que en una forma muy resumida se mostrarán a continuación.

2.1. ANTECEDENTES.

[1] Iniciando los años ochentas, la prestación de servicios de voz a través de redes fijas, era la base fundamental de las empresas de telecomunicaciones. La mayoría de los países invertían grandes rubros de dinero y tiempo en lograr el mayor cubrimiento posible de servicios básicos y de hecho se tenían redes de comunicaciones separadas para el transporte de voz y de datos (principalmente para el transporte de voz).

A finales de los ochentas y principios de los noventas, aparece Internet junto con un cambio radical en los productos y servicios que ofrecen las empresas de telecomunicaciones. Esto último debido a que los centros de investigación, los proveedores de equipos, entre otros entes redirigen su atención, ahora al protocolo IP, el cual logra un protagonismo envidiable. Se le dio tal importancia, ya que el término convergencia tomó mucha fuerza haciendose posible la integración de prácticamente todo el flujo de información empresarial en una sola red, por medio de la digitalización y la conmutación de paquetes.

Durante los noventas, con el auge de “las .com” aumentó la popularidad del Internet y se hacía más y más común escuchar el concepto de convergencia de redes a nivel de empresas y organizaciones, ya que esto ahorraba en gran parte los costos para las comunicaciones en cualquier red de datos interna. Era evidente la necesidad de unir voz y datos como paquetes de información.

2.1.1. Trabajos Realizados

NetVoice

[2] NetVoice de Avantel es el servicio de telecomunicaciones más innovador en el mercado ya que utiliza al 100% la tecnología IP, lo que permite navegar en

INTERNET y hablar por teléfono (ya sea local o de larga distancia) al mismo tiempo. Funciona a través de una conexión de INTERNET de banda ancha de última generación, lo que da como resultado tarifas sumamente atractivas.

Brújula

[3] Se puede adquirir una suscripción y hablar con cualquier usuario de VoIP.brujula.net gratis! Si se recomienda este servicio voip a los amigos se puede hablar con ellos sin cargo aunque estén en países distintos. Incluye contestador de llamadas con redirección a email. Llamadas en conferencia.

Administración de las llamadas por Web y más beneficios para suscribirse gratis. Asocie una cuenta VoIP.brujula.net a un número telefónico entrante en USA Gratis. Se puede ahorrar en el servicio telefónico. Se puede llamar a teléfonos fijos en cualquier parte del mundo, con tarifas muy reducidas y sin abonos mensuales. Simplemente se agrega crédito a una cuenta cuando es necesario.

¿Qué Teléfonos VoIP puede usar?

- ✓ X - Ten Lite Sip SoftPhone. Se puede descargar gratis para Windows, Mac OS X o Linux.
- ✓ Más opciones de telefonos SIP en software o hardware

¿Cómo Configurar el Servicio?

En el programa X-Ten-Lite. Haga per en MENU (el boton a la derecha de CLEAR y a la izquierda del boton verde), System Settings, SIP Proxy, Default. E ingrese los siguientes datos de ejemplo:

Figura 2. *Ejemplo de configuración sobre el software de Brujula.NET.*

Skype

[4] La tecnología puerto-a-puerto (P2P) fue ampliamente extendida y popularizada por las aplicaciones de compartición de ficheros tales como Napster y KaZaA. En este contexto, la tecnología P2P permite a los usuarios compartir, buscar y descargar ficheros. El nombre P2P ha sido utilizado ampliamente e incluso a menudo abusado. Tanto grandes como pequeñas compañías, autoproclamados expertos y otros, intentando conseguir dinero con el negocio de P2P utilizan este término mientras ello involucra comunicaciones directas entre usuarios o nodos. Esta descripción de P2P olvida por completo el fin de esta tecnología.

Un sistema P2P real, en nuestra opinión, es aquella en la que todos los nodos en una red se unen dinámicamente para participar en el enrutamiento de tráfico, procesamiento y tareas de gestión intensiva del ancho de banda que de otra forma serían realizadas por servidores centrales.

Para los fundadores de KaZaA la telefonía P2P se convirtió en el siguiente paso donde el P2P podría significar un impacto disruptivo y Skype fue fundado para desarrollar la primera red de telefonía P2P.

La telefonía basada en Internet (voz sobre IP, VoIP) ha estado evolucionando durante años pero no ha alcanzado el mercado masivo. Las razones de esto son bastante claras para aquellos que han probado el software VoIP:

- ✓ Los productos que consiguen un ahorro de costes real en comparación con la telefonía tradicional no tienen una calidad comparable.
- ✓ El porcentaje de llamadas completadas es muy bajo debido a *firewalls* y el uso del protocolo NAT (Network Address Translation) lo que produce que más del 50% de los ordenadores domésticos no sean capaces de comunicarse con el software de VoIP tradicional.
- ✓ El interfaz de usuario está normalmente exagerado y es necesario una compleja configuración y elevados conocimientos técnicos.

Aprovechando la experiencia en la creación de la red descentralizada P2P prácticamente más popular de la historia (KaZaA), se puede pensar que con Skype se ha creado la interfaz de usuario más amistosa hasta la fecha. Algunas de las técnicas que Skype emplea para proporcionar telefonía basada en IP de última generación son:

- ✓ Firewall and NAT (Network Address Translation).
- ✓ Directorio global de usuarios descentralizado.
- ✓ Enrutamiento Inteligente.
- ✓ Seguridad.
- ✓ Interfaz de Usuario fácil de usar.

2.1.2. Historia de Éxito: Norstar Al Rescate.

[5] Cuando los desastres golpean, la comunicación es crítica. Eric Rhinehardt nunca olvidará el tornado que arrasó con Nashville en Abril 16 de 1998.

“Iba hacia la avenida Charlotte, tomé una de las paredes de la casa, movida a través de todo el distrito comercial y también sobre algunas vecindades alrededor del río”. Eric es Director de Desarrollo en el Nashville Area Chapter de la Cruz Roja Americana. Para soportar el tornado, realmente el esfuerzo económico alcanzó los U$1’400.000. Adicionalmente, la Cruz Roja Americana consiguió cientos de voluntarios extras para ayudar. Con esto se consiguieron 186.000 comidas en dos semanas; y ésto cobijó a familias que perdieron sus casas. Durante el esfuerzo real, el Nashville Area Chapter aprendió que las tecnologías de las comunicaciones influyen sobre la preparación de desastres y tiempo de

respuesta. Como Eric dijo: “Esa experiencia nos permitió encontrar un sistema telefónico que podía saber sobre nuestras especiales necesidades.”

Figura 3. *Nashville Area Chapter de la Cruz Roja Americana.*

Cuando Eric escuchó sobre el Norstar Modular Integrated Communications System (ICS), él llamó a Nortel Networks inmediatamente. “Nosotros sabemos de otras compañías y de las desventajas que tuvo el sistema Norstar. Todos están contentos con eso”, él anotó. El sistema Nashville Area Chapter incluía voz por mail, 60 conjuntos de teléfonos (incluyendo una consola de atención), y un rango de aplicaciones de telefonía integradas. La configuración simple del sistema y la flexibilidad fue una gran sorpresa. “Es lo suficientemente fácil como para que cualquiera lo use”, dijo jocosamente Eric.

Las operaciones de ayuda son complejas. Después de un desastre, el capítulo corre a asesorar el daño, diseña contingencias, responde inquietudes propias de un desastre (cuando residentes de afuera llaman a confirmar el estado de sus familiares), y realizan muchas otras actividades. “Dependiendo de la escala del desastre”, anota Eric, “necesitaríamos de uno a tres trabajadores de la Cruz Roja para coordinar cada una de aquellas actividades con voluntarios en cada uno de los campos”. El ACD (Automatic Call Distribution) de Norstar percató al capítulo para crear un call center que inteligentemente ordenara las llamadas entrantes. El ACD reconoce las llamadas y automáticamente las distribuye a grupos apropiados o personas individuales, basado en las especificaciones únicas de ruteo de la Cruz Roja. “El sistema de Norstar nos permite administrar la operación”, dijo Eric.

Los desastres no siempre ocurren en horario laboral. Con el forwarding de las llamadas Norstar redirecciona llamadas críticas a voluntarios de los alrededores. “Cuando el staff va a casa, los voluntarios de puertas hacia afuera están llamando para colaborar en después de horas”, explicó Eric. “Nuestro sistema Norstar les regresa las llamadas”. Con Norstar, el Nashville Area Chapter puede desplegar un equipo de acción contra desastres siempre a cualquier hora, día o noche y entregar mucha ayuda más rápidamente.

Así como el Nashville Area Chapter evoluciona, el Norstar crece con él. El Modulo ICS puede escalar el teléfono del capítulo para una capacidad de hasta 224 sets. Para proveer frecuentemente un rápido soporte, Eric quiere integrar su sistema Módulo ICS del capítulo con los computadores de su oficina. Activando la capacidad de integración de voz y datos de Norstar, Eric ayudará a muchos más con menos recursos. “Cuando se genera una situación de emergencia, debo ir adentro del desastre encontrando formas rápidas de hacerlo”, dijo. Usando la interfaz de datos de Norstar y el Internet instantáneo, Eric puede emitir requerimientos para gestionar a la Cruz Roja soportadores y donantes a través del computador (vía IP) más rápido que usando una tradicional máquina de fax.

Desde 1917, el Nashville Area Chapter de la Cruz Roja Americana ha ayudado residentes en 10 regiones medias de Tennessee sobrellevando muchas dificultades durante el último siglo (incluyendo el Tornado de Abril 16 de 1998). Sin duda, los desastres golpearán de nuevo. Cuando esas cosas vienen, la Cruz Roja, y Norstar, estarán listos. “Los desastres a gran escala no vendrán muy frecuentemente gracias a Dios”, comenta Eric. “Pero cuando ello ocurre, se debe tener un sistema telefónico como éste. Nosotros aprendimos esa lección en 1998”.

2.2. TENDENCIAS.

[6] Como tecnología, la Voz sobre IP (VoIP) lleva varios años de presencia en el mercado. Sin embargo, no ha sido hasta la emergencia de nuevos e innovadores servicios basados en esta tecnología que la integración de datos y voz se ha hecho realidad, lo que, para las empresas, ha significado un ahorro de costes y unas comunicaciones más eficientes y efectivas.

Se prevé que en el año 2005 el mercado VoIP moverá más de 10.000 millones de dólares, especialmente desde el momento en que las percepciones del mundo corporativo le son más favorables.

En la actualidad, son cada vez más numerosas las compañías que ven esta tecnología como una herramienta de comunicaciones comercialmente viable. Como las primeras experiencias con la tecnología VoIP estaban relacionadas con Internet, muchas personas la consideran un simple método económico y de poca

calidad para transportar comunicaciones de voz por la Web. En realidad, se trata de algo mucho más importante. Esta tecnología puede cambiar el modo de crear, distribuir, comercializar y vender aplicaciones de voz. Y al combinar los sistemas de voz y de tráfico en tiempo real, se abre un nuevo mundo de fantásticas oportunidades multimedia e interacción para consumidores individuales y empresas.

Al enlazar VoIP con otras aplicaciones en línea, las empresas pueden permitirse el lujo de realizar videoconferencias a través de sistemas PC, llevar a cabo presentaciones interactivas y compartir documentos en tiempo real. James Walker, director de producto para VoIP Multimedia de BT Ignite, comenta: “La tecnología Voice over IP sirve de base a numerosos servicios: desde sencillos y económicos sistemas de voz a través de Internet para consumidores hasta los servicios de voz y datos de alta calidad para empresas que dan lugar a nuevas aplicaciones en el entorno empresarial”.

Se ha ideado una nueva aplicación IP que se basa en la tecnología VoIP actual. Esto significa que cuando una persona llama, su tarjeta de negocio parpadea en el PC. Asimismo, incorpora una aplicación en el dispositivo IP de bolsillo que identifica la persona que llama en cualquier momento.

Richard Swale, experto en tecnología VoIP de Btexact Technologies, afirma que la tecnología VoIP puede revolucionar las comunicaciones internas: “VoIP brinda nuevas oportunidades para quienes sean capaces de preverlas y actúen con la rapidez suficiente para superar la confusión que envuelve esta extraordinaria tecnología.

2.3. VENTAJAS.

[7] El 28% de las empresas de Latinoamérica está implementando o utilizando sistemas de Telefonía IP en el 2004, versus un 13% en el 2003 y un 37% de las empresas de la región está planeando implementar Telefonía IP, según se desprende de la encuesta realizada por IDC (http://www.idclatin.com/), la cual fue denominada “Telefonía IP En Latinoamérica, Perspectiva del usuario Final 2004”. Se obtuvo la siguiente gráfica:

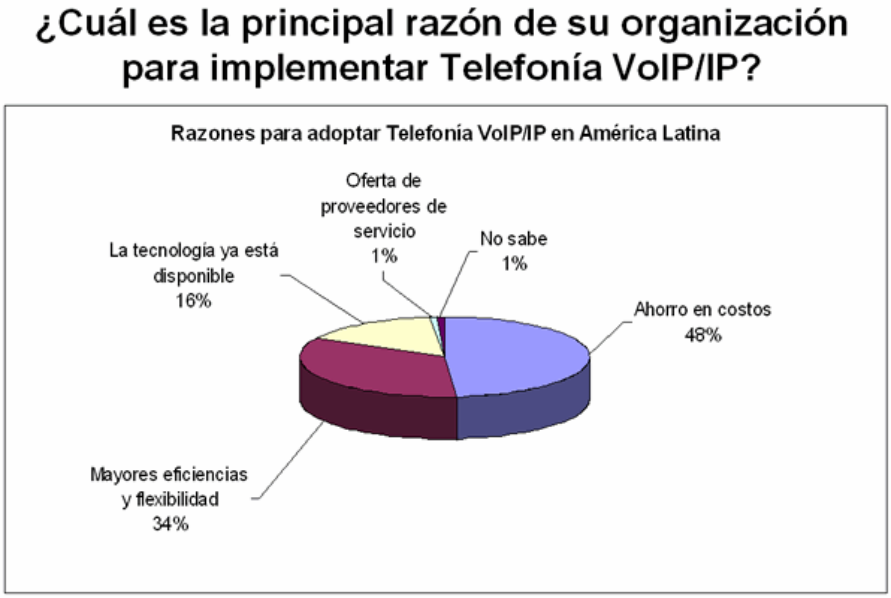

Figura 4. *Resultados principales arrojados por encuesta de IDC Latinoamérica.*

De acuerdo con el estudio de IDC, la principal razón que lleva a las empresas a migrar de sus soluciones de voz tradicionales a Telefonía IP, es el ahorro en costos, en la medida en que las empresas de todos los tamaños de la región tienen una presión fuerte para reducir sus inversiones en capital en telecomunicaciones y sus gastos operativos. Por esto la convergencia (integración de voz y datos en una misma red) emerge como una solución real para reducir costos y generar aumentos en productividad. Se obtuvo la siguiente gráfica:

Figura 5. *Resultados secundarios arrojados por encuesta de IDC Latinoamérica.*

Entre el 37% de las empresas que planean implementar Telefonía IP, el 46% planean hacerlo en el 2004, el 28% en el 2005 y el 4% en el 2006 o luego. Un 22% de los encuestados no sabe cuándo implementará la solución. Para las empresas que están usando en la actualidad Telefonía IP, 62% opina que el servicio es "valioso" y el 37% opina que es "muy valioso". Entre aquellos que respondieron no implementar Telefonía IP, 22% manifestaron que están contentos con su plataforma actual de comunicaciones y 20% manifestaron que la tecnología es aún muy costosa.

[8] Otras de las ventajas a tener en cuenta son:

- ✓ Integración sobre su Intranet de la voz como un servicio más de su red, tal como otros servicios informáticos.
- ✓ Las redes IP son la red estándar universal para la Internet, Intranets y extranets.
- ✓ Estándares efectivos (H.323).
- ✓ Interoperabilidad de diversos proveedores.
- ✓ Uso de las redes de datos existentes.
- ✓ Independencia de tecnologías de transporte (capa 2), asegurando la inversión.
- ✓ Menores costos que tecnologías alternativas (voz sobre TDM, ATM, Frame Relay).

2.4. TELEFONÍA IP

2.4.1. ¿Qué es Telefonía IP?

[9] La Voz sobre IP (VoIP, Voice over IP) es una tecnología que permite la transmisión de la voz a través de redes IP en forma de paquetes de datos. La Telefonía IP es una aplicación inmediata de esta tecnología, de forma que permita la realización de llamadas telefónicas ordinarias sobre redes IP u otras redes de paquetes utilizando un PC, gateways y teléfonos estándares. En general, servicios de comunicación (voz, fax, aplicaciones de mensajes de voz) que son transportadas vía redes IP, Internet normalmente, en lugar de ser transportados vía la red telefónica convencional.

2.4.2. ¿Cómo Funciona la Telefonía IP?

[9] Los pasos básicos que tienen lugar en una llamada a través de Internet son: conversión de la señal de voz analógica a formato digital y compresión de la señal a protocolo de Internet (IP) para su transmisión. En recepción se realiza el proceso

inverso para poder recuperar de nuevo la señal de voz analógica. Cuando hacemos una llamada telefónica por IP, nuestra voz se digitaliza, se comprime y se envía en paquetes de datos IP. Estos paquetes se envían a través de Internet a la persona con la que estamos hablando. Cuando alcanzan su destino, son ensamblados de nuevo, descomprimidos y convertidos en la señal de voz original. Hay tres tipos de llamadas:

- ✓ PC a PC, siempre gratis.
- ✓ PC a Teléfono, gratis en algunas ocasiones, depende del destino.
- ✓ Teléfono a Teléfono, muy baratas.

2.4.3. Telefonia IP Vs PSTN

[9] En una llamada telefónica normal, la centralita telefónica establece una conexión permanente entre ambos interlocutores, conexión que se utiliza para llevar las señales de voz. En una llamada telefónica por IP, los paquetes de datos, que contienen la señal de voz digitalizada y comprimida, se envían a través de Internet a la dirección IP del destinatario. Cada paquete puede utilizar un camino para llegar, están compartiendo un medio, una red de datos. Cuando llegan a su destino son ordenados y convertidos de nuevo en señal de voz.

2.4.4. ¿Por qué es más barata la Telefonía IP?

[9] Una llamada telefónica normal requiere una enorme red de centralitas telefónicas conectadas entre sí mediante fibra óptica y satélites de telecomunicación, además de los cables que unen los teléfonos con las centralitas. Las enormes inversiones necesarias para crear y mantener esa infraestructura las tenemos que pagar cuando realizamos llamadas, especialmente llamadas de larga distancia. Además, cuando se establece una llamada tenemos un circuito dedicado, con un exceso de capacidad que realmente no estamos utilizando.

Por el contrario, en una llamada telefónica IP estamos comprimiendo la señal de voz y utilizamos una red de paquetes sólo cuando es necesario. Los paquetes de datos de diferentes llamadas, e incluso de diferentes tipos de datos, pueden viajar por la misma línea al mismo tiempo. Además, el acceso a Internet cada vez es más barato, muchos ISPs lo ofrecen gratis, sólo tienes que pagar la llamada, siempre con las tarifas locales más baratas. También se empiezan a extender las tarifas planas, conexiones por cable, ADSL, etc.

2.4.5. Historia de Éxito: Pontificia Universidad Católica.

[10] La Pontificia Universidad Católica fue fundada el 18 de Marzo de 1917. Fue la primera universidad privada y es una de las más prestigiosas del Perú. Ha sido casa de estudios de importantes personalidades del ámbito político-económico y reconocidos embajadores del arte.

La Universidad da ahora un gran paso adelante en la modernización de sus procesos de gestión en comunicaciones al implementar la más innovadora solución de Comunicaciones IP en el Perú, la cual comprende la instalación de 1,500 teléfonos IP, iniciativa que los coloca a la vanguardia como la primera universidad del país que cuenta con una solución de esta magnitud.

Necesidades de Integración Comunicacional

Antes de implementar la solución de Telefonía IP, la universidad venía utilizando una central telefónica analógica con capacidad para 487 anexos, de los cuales sólo quedaban libres 30 aproximadamente. La central contaba con 34 líneas troncales de salida y 28 de entrada.

Asimismo, habían implementado pequeñas centrales en diferentes facultades o edificios, las cuales se fueron instalando para satisfacer las necesidades de la infraestructura nueva. Tanto en el campus como en algunos de los locales externos a él, habían instaladas 41 minicentrales de más de 6 marcas diferentes.

Adicionalmente se contaba con dos infraestructuras independientes para la atención de sus sistemas de comunicación, es decir, una para la telefonía convencional y otra para las redes de datos.

El Contar con una PBX tradicional, con una diversidad de centrales instaladas en diferentes ambientes y locales de la universidad, así como la falta de integración de las comunicaciones de voz y datos, no les permitía atender las necesidades básicas de comunicación que demandaban los usuarios.

Con la red de comunicación de voz tradicional no les era posible realizar transferencias de llamadas entre unidades, se requerían operadoras telefónicas en aquellas unidades que tenían una “minicentral”, y además se producía una congestión del servicio telefónico en aquellas unidades que contaban con un elevado número de personal que compartían una cantidad pequeña de anexos y líneas directas.

Camino a la Optimización de las Comunicaciones

Con una demanda actual de 1.500 usuarios dedicados a labores administrativas y de docencia, tanto dentro como fuera del campus universitario y con una proyección de contar con los 6 locales principales interconectados (Plaza Francia, Instituto Riva Agüero, INIPUCP Camacho, CC San Isidro – INIPUCP San Isidro, Centrum y Campus Universitario) la Pontificia Universidad Católica toma la decisión de optimizar sus comunicaciones eligiendo a Cisco Systems como proveedor y a IBM del Perú como integrador de la solución.

En una primera etapa se procedió al planeamiento del proyecto con el objetivo de definir y programar los pasos a seguir en la instalación del nuevo equipamiento, así como la configuración necesaria para responder a los requerimientos del cliente.

Tras el proceso de planeamiento, se llevó a cabo en forma inmediata el repotenciamiento de la infraestructura de datos existente. Para ello se añadió un Switch de Core idéntico al que tenían en funcionamiento y se cambiaron absolutamente todos los hubs por switches, algunos de los cuales fueron in line power, es decir, con la capacidad de alimentar eléctricamente los teléfonos a través de la red. Paralelamente se reforzó la protección eléctrica de los gabinetes de cableado con equipos UPS. De esta manera se logró asegurar que la plataforma sobre la cual corriera la solución de telefonía IP fuera la más adecuada y segura.

En la segunda etapa del proyecto se procedió a la implementación de la solución y al retiro de las centrales antiguas. La implementación de la solución de Telefonía IP en el Campus de la universidad y 8 locales remotos incluyó el reemplazo de las centrales telefónicas antiguas por un único sistema de Telefonía IP. La solución consistió en:

- ✓ Un cluster de CallManagers que le dan a la PUCP doble redundancia y configuración centralizada.
- ✓ Un sistema de Voicemail para 1.000 usuarios.
- ✓ Dos Switches Catalyst 6513 con velocidad de conmutación de 720 Gigas que incluyen una tarjeta de 6 E1s c/u de tal manera que se tiene también redundancia de E1s.
- ✓ Cerca de 100 Switches LAN de 24 puertos 10/100 que brindan energía a los Teléfonos IP.
- ✓ Teléfonos IP modelos Cisco 7970, Cisco 7920, Cisco 7960, Cisco 7940, Cisco 7912, ATA 188 y VG248.
- ✓ Ocho equipos de comunicación inalámbrica para la solución de Telefonía IP wireless.

- ✓ Ocho routers para las sedes remotas. Estos equipos incluyen puertos de datos, puertos de voz, recursos de conferencia y función de Telefonía IP para evitar que las sedes remotas pierdan comunicación si la red de datos no está activa.

Dada la magnitud de la solución y el número de personas que tendrían acceso al nuevo sistema de telefonía, el proyecto consideró también la capacitación de los usuarios, operadores, soporte técnico y administradores, previa a la puesta en funcionamiento del nuevo sistema.

Asimismo, para asegurar el éxito de esta ambiciosa implementación, Cisco Systems e IBM convocaron a diferentes especialistas de la región de Latinoamérica para que participen y apoyen al equipo local en los procesos de planeamiento, configuración y afinamiento del proyecto.

Beneficios Obtenidos

Las soluciones de comunicaciones IP no sólo mejoran las comunicaciones, sino también la eficacia ya que están creadas sobre una base abierta y convergente con inteligencia incorporada, beneficios con los que ahora cuenta la universidad.

Para la Universidad Católica ha redundado además en una importante reducción de costos. Resultado logrado en parte por la eliminación de gastos que ocasionaba el mantenimiento de las centrales antiguas, sumado al ahorro que ofrece el contar con un único sistema de cableado para datos y voz además de la centralización de las comunicaciones que permite optimizar los gastos de llamadas y eliminar la duplicidad de funciones que ocasionaban las múltiples minicentrales instaladas en los distintos locales.

Por otro lado, en términos de eficiencia y productividad, los usuarios de la nueva Telefonía IP gozan ahora de aplicaciones que garantizan una experiencia de las comunicaciones más sofisticada y moderna, al mismo tiempo que les ofrecen acceso oportuno a información y recursos que les permite aumentar su productividad como es el caso de la mensajería unificada, softphone, movilidad de la extensión, aplicaciones XML, directorios personales, información de llamadas, etc.

Las soluciones de Comunicaciones IP son fundamentales en la optimización del servicio de atención al público y en el caso de la universidad, les ha permitido poder manejar un único número para sus diferentes áreas. En la actualidad la PUCP tiene un sólo número telefónico que permite acceder a cualquiera de los 8 locales y 1.500 usuarios.

2.4.6. Ventajas de la Telefonía IP

[11] La telefonía IP ofrece significantes beneficios cuando es comparada con las redes conmutadas:

- ✓ La Telefonía IP usa mucho menos ancho de banda que la telefonía de red conmutada, generalmente por un factor de 8 o más. Un circuito de voz estándar usa 64 Kilobits por segundo, pero la telefonía IP usa de 6 a 8Kb/s, y algunas veces 2.4Kb/s.

- ✓ Desde que la telefonía IP envía comunicaciones en paquetes, los encabezados de los paquetes pueden contener información de control valiosa y ésta es transmitida simultáneamente, mejorando notablemente la flexibilidad.

- ✓ La Telefonía IP proporciona nuevos servicios sobre una infraestructura existente, proveyendo un flujo de mejoras renovadas y costo de transporte más bajo.

2.5. ESTÁNDARES ASOCIADOS.

[12] Se habla hoy por hoy de estándares evolutivos. Los estándares ITU (Sector de Estándares de Telecomunicaciones de la Unión Telegráfica Internacional) para VoIP han evolucionado de manera significativa. Hace dos años, parecía que el H.323, un estándar desarrollado en 1996 para multimedia, se encaminaba hacia la aceptación universal como el estándar principal para la señal y la configuración de llamadas VoIP. Este estándar no se utilizó originalmente con este propósito, sin embargo, algunas de sus partes tuvieron mucho éxito entre los desarrolladores de productos VoIP. La segunda versión de H.323 agregó soporte WAN para VoIP; el estándar H.323 más actual, incluye un nuevo procedimiento para mejorar el tiempo de configuración.

A medida que los fabricantes de productos VoIP realizaron pruebas de interoperación para las operaciones más complejas, se percataron que necesitaban un estándar sencillo y adaptable para el manejo de llamadas y el protocolo de transmisión. Con este fin, el IETF (Grupo de Trabajo de Ingeniería Internet) desarrolló un Protocolo de Iniciación de Sesión (SIP). SIP contiene menor código de computadora que el H.323, lo que cual lo hace menos pesado. Debido a que es más pequeño que el HTML, utilizando texto ASCII para configuración, se puede adaptar con mayor facilidad a sistemas VoIP específicos.

Tanto H.323 como SIP se consideran “clientes gruesos”, donde la inteligencia se

conserva en los dispositivos finales como teléfonos IP. Asimismo, el H.323 tuvo varias ventajas, ya que la mayoría de los productos de primera generación, diseñados con estándares compatibles, soportan el H.323. Sin embargo, muchos proveedores desarrollan productos para soportar SIP, y en muchos casos, los sistemas VoIP soportan tanto H.323 como SIP. Se explica cada uno de ellos a continuación:

2.5.1. H. 323.

[13] H.323 es un estándar ITU (Unión Internacional de Telecomunicaciones) que provee especificaciones para computadores, equipos y servicios para comunicaciones multimedia sobre redes que no proveen una calidad de servicio garantizada. Los computadores y equipos H.323 pueden transportar vídeo en tiempo real, audio y datos o cualquier combinación de éstos elementos. Este estándar se basa en el IETF Real-Time Protocol (RTP) y en el Real-Time Control Protocol (RTCP), con protocolos adicionales para señalización de llamadas y comunicaciones de datos y audiovisuales.

Los usuarios pueden conectarse con otra gente sobre el INTERNET y usar diferentes productos que soporten H.323, así como la gente usa diferentes creaciones y modelos de teléfonos que pueden comunicar sobre líneas PSTN (Public Switched Telephone Network). H.323 define cómo la información de audio y vídeo es formateada y empaquetada para transmisión sobre la red. CODECs estándar de audio y vídeo codifican y decodifican entradas/salidas de recursos de audio y vídeo para comunicaciones entre nodos. Un codec (codificador/decodificador) convierte señales de audio y vídeo entre formas digitales y análogas.

También el H.323 especifica servicios T.120 para comunicaciones de datos y conferencias dentro y próximo a una sesión H.323. Más importante aun, éste soporte T.120 significa que la manipulación de los datos puede ocurrir en conjunto con audio y vídeo H.323 o separadamente.

Microsoft y más de 120 compañías líderes han anunciado sus intentos de soportar e implementar H.323 en sus productos y servicios. Este amplio soporte, establece a H.323 como el estándar para audio y vídeo conferencia sobre INTERNET.

Ventajas

Los productos y servicios H.323 ofrecen las siguientes ventajas a los usuarios:

- ✓ Los productos y servicios desarrollados por múltiples creadores bajo el estándar H.323 pueden interoperar sin limitaciones de plataforma. Los Clientes de conferencia H.323, bridges, servidores y gateways soportan esta interoperabilidad.

- ✓ H.323 provee múltiples códecs de audio y vídeo con formato de datos acorde a los requerimientos de varias redes, usando diferentes tasas de transmisión, retardos y opciones de calidad. Los usuarios pueden escoger los códecs que mejor soporte su computador y red.

- ✓ La adición de datos y conferencias T.120 soportan las especificaciones H.323 y eso significa que los productos desarrollados bajo H.323 pueden ofrecer un rango de funciones multimedia, con soporte para datos y vídeo conferencias.

[14] Este estandar cuenta con las siguientes recomendaciones:

Recommendación	*Descripción*
H.225	Especifica mensajes para el control incluyendo señalización, registro y admisiones, además de paquetización y sincronización de archivos multimedia.
H.245	Especifica mensajes para apertura y cierre de canales para archivos multimedia y otros comandos, requerimientos e indicaciones.
H.261	Vídeo codec para servicios audiovisuales en P x 64Kbps.
H.263	Especifica un nuevo video codec para video sobre POTS.
G.711	Audio Codec, 3.1 KHz en 48.56 y 64Kbps (Telefonía normal).
G.722	Audio Codec, 7 KHz en 48.56 y 64Kbps.
G.728	Audio Codec, 3.1 KHz en 16 Kbps.
G.723	Audio Codec, para modos 5.3 y 6.3 Kbps
G.729	Audio Codec.

Tabla 1. *Componentes de H. 323.*

2.5.2. SIP

[15] El IETF ha generado un set de protocolos que simplifican las funciones de H.323, el cual tiene previstas funciones dentro de una red corporativa y en

multimedia. SIP es un protocolo más simple que H.323 y está basado en HTTP. En H.323 se utiliza el GK, mientras que en SIP se usa el SIP-Server, el cual tiene mejores aspectos de escalabilidad para grandes redes. En H.323 para grandes redes se recurre a definir zonas de influencia y colocar varios GK. Para la interoperatividad de protocolos se requiere un GW de borde que realice la conversión.

SIP es un protocolo basado en texto (de acuerdo con RFC-2279 para la codificación del set de caracteres) y el mensaje basado en http (RFC-2068 para la semántica y sintaxis). La dirección usada en SIP se basa en un localizador URL (Uniform Resource Locater) con un formato del tipo sip: roberto@192.190.132.31 (o mediante el dominio Domain: teleinfo.com.ar). De esta forma SIP integra su servicio a la Internet. En este modelo se requiere el auxilio de un server de resolución de dominio DNS (Domain Name Server).

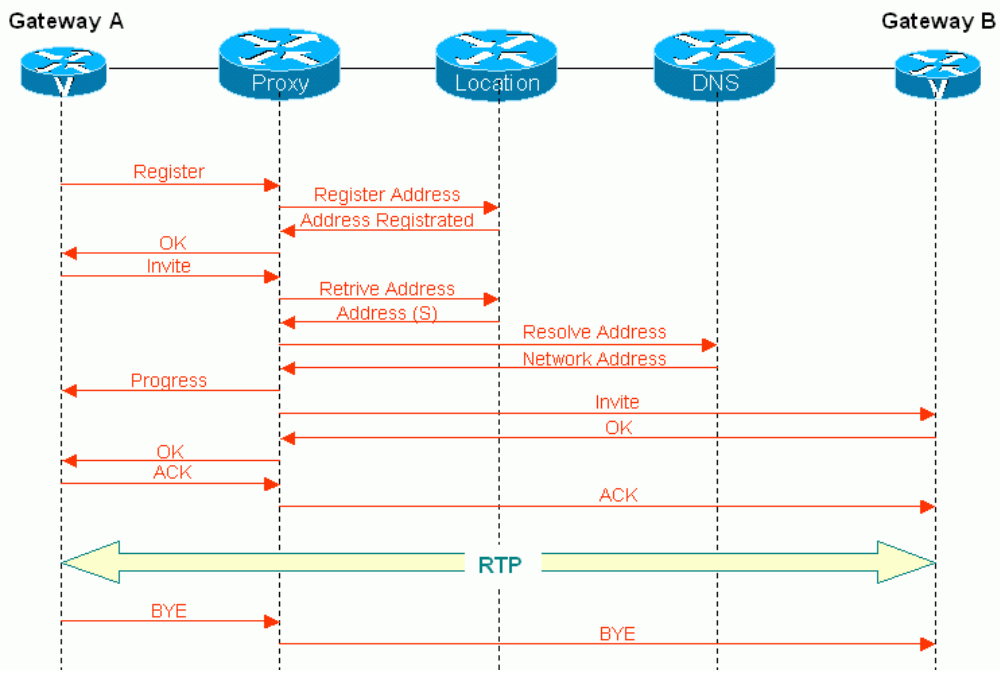

Figura 6. *Intercambio de mensajes para establecer una comunicación con protocolo SIP.*

El protocolo SIP incorpora también funciones de seguridad y autentificación, así como la descripción del medio mediante el protocolo SDP. Para el proceso de facturación billing se puede recurrir a un server RADIUS. Las fases de comunicación soportadas en una conexión unicast mediante el protocolo SIP, son las siguientes:

- ✓ User location. En esta fase se determina el sistema terminal para la comunicación.
- ✓ User capabilities: Permite determinar los parámetros del medio a ser usados.
- ✓ User availability: Para determinar la disponibilidad del llamado para la comunicación.
- ✓ Call setup: (“ringing”); Para el establecimiento de la llamada entre ambos extremos.
- ✓ Call handling: Incluye la transferencia y terminación de la llamada.

El protocolo SIP tiene dos tipos de mensajes: Request y Response. El mensaje de Request es emitido desde el cliente terminal al Server terminal. El encabezado del mensaje request y response contiene campos similares:

- ✓ *Start Line.* Usada para indicar el tipo de paquete, la dirección y la versión de SIP.
- ✓ *General Header.* Contiene el Call-ID (se genera en cada llamada para identificar la misma); Cseq (se inicia en un número aleatorio e identifica en forma secuencial a cada request); From (es la dirección del origen de la llamada); To (es la dirección del destino de la llamada); Via (sirve para recordar la ruta del request; por ello cada proxy en la ruta añade una línea de vía) y Encryption (identifica un mensaje que ha sido encriptado para seguridad).
- ✓ *Additionals.* Además del encabezado general se pueden transportar campos adicionales. Por ejemplo: Expire indica el tiempo de valides de registración; Priority indica la prioridad del mensaje; etc.

Se han definido 6 métodos para los mensajes de request-response.

- ✓ *Invite.* Para invitar al usuario a realizar una conexión. Localiza e identifica al usuario.
- ✓ *Bye.* Para la terminación de una llamada entre usuarios.
- ✓ Options. Información de capacidades que pueden ser configuradas entre agentes o mediante un server SIP.
- ✓ *ACK.* Usado para reconocer que el mensaje Invite puede ser aceptado.
- ✓ *Cancel.* Termina una búsqueda de un usuario.
- ✓ *Register.* Emitido en un mensaje multicast para localizar al server SIP.

2.5.3. Comparacion entre H.323 Y SIP.

[16] Aunque ambos estándares pueden ser usados para aplicaciones VoIP, su enfoque original es muy diferente. Mientras SIP ha sido diseñado como un

protocolo de transacción genérico para inicio de sesión no limitado a cualquier medio específico tal como audio o vídeo, el foco de H.323 ha sido manipular voz y llamadas multimedia, incluyendo servicios suplementarios.

Mirando un poco la estandarización del control de llamadas y especialmente los servicios suplementarios, SIP aun muestra algunos defectos relacionados con los servicios suplementarios. Tal estandarización es importante para la interoperabilidad con los sistemas de telefonía tradicional. Mientras tanto, el IETF ha identificado la necesidad en su alcance, de mirar los requerimientos relacionados con telefonía y servicios tales como interdependencia SIP-H.323, SIP-PSTN o servicios suplementarios. Como una consecuencia, SIP va a ser extendido para obtener y mantener la funcionalidad de H.323.

2.6. REGULACION.

2.6.1. En Colombia.

[17] El Gobierno Nacional busca reglamentar el servicio de llamadas de voz sobre Internet. El Ministerio de Comunicaciones puso a disposición del público su Cuaderno de Políticas de Voz sobre IP, que busca abrir la discusión acerca de la forma como deben ser implementados en el país los servicios de Voz sobre IP (VoIP) y los cambios normativos que se requerirán para garantizar su correcta adopción.

En el documento, además de describirse la manera en que dicha tecnología modificará la prestación de servicios de comunicación en Colombia, se contemplan las recomendaciones realizadas por la Unión Internacional de Telecomunicaciones (UIT) sobre la materia y se presenta el estado actual de la regulación nacional en lo que a VoIP se refiere.

La publicación está disponible en la página web del Ministerio de Comunicaciones (http://www.mincomunicaciones.gov.co/Archivos/Documentos/vozipjun2004.pdf) y se recibirán comentarios hasta el 30 de julio de 2004 en el siguiente correo electrónico desarrollodelsector@mincomunicaciones.gov.co. El 13 de agosto de 2004 se publicará la segunda versión del cuaderno, del que se recibirían nuevos comentarios hasta el 10 de septiembre de 2004, y finalmente se presentará el documento de política definitivo el 8 de octubre de 2004.

En la actualidad, las regulaciones gubernamentales en Colombia permiten el uso de Voz sobre IP siempre y cuando sea empleada por las empresas para sus comunicaciones internas.

2.6.2. En Otros Países

[1] Se presentan a continuación los casos de algunos de los países más representativos en cuanto a VoIP:

Estados Unidos

Los Estados Unidos son tal vez el país mas adelantado en cuanto a la implantación de este tipo de servicios, sin embargo aun no existe una posición oficial en cuanto al tema. Reci-+entemente la FCC se pronunció con respecto al caso específico de ATT, el cual realiza una gran parte de la transmisión de llamadas de larga distancia utilizando un backbone con tecnología IP.

Comunidad Europea

La Comunidad Europea, al igual que muchos otros países en el mundo aun no tienen claro el panorama acerca de la implantación de la VoIP, por tal motivo vienen adelantando estudios que les permita tener los elementos de juicio necesarios para lograr una eficiente implementación de estos nuevos servicios.

España

El nuevo marco normativo propuesto por la Ley General de Telecomunicaciones permite la diferenciación de los diferentes servicios de VoIP y la aplicación de una regulación consecuente con sus características y fines, en tanto no privilegia al tradicional servicio telefónico sobre los demás servicios de comunicaciones electrónicas, sino que lo concibe como uno más de éstos. La ausencia de este privilegio (presente en los cuerpos normativos anteriores) se traduce en una propuesta convergente que permite la aplicación de las mismas reglas a todos los servicios capaces de competir como sustitutos del servicio telefónico, con total neutralidad tecnológica, mientras que otros servicios de transmisión de voz que proponen soluciones diferentes a las del servicio telefónico contarán con sus reglas particulares.

Canadá

Se permite el uso de la telefonía IP, el marco regulatorio establece la no regulación de Internet. En cuanto a la telefonía IP, en mayo de 1997 la entidad regulatoria, CRTC, estableció que los sistemas de telefonía IP que generan

llamadas Teléfono a Teléfono eran susceptibles de pagar contribuciones por cada minuto causado, esta política fue confirmada e implementada en enero de 1998. Luego de esto se reguló que las llamadas PC a PC y PC a teléfono no eran candidatas a pagar contribuciones. De lo anterior se deduce que el régimen se basa en la naturaleza de la terminal utilizada para originar la llamada y por ende de cómo se causa el tránsito de la comunicación a través de la red, determinando que si es originada en la RTPBC y a través de cualquier dispositivo de hardware enrutada a través de Internet, se sigue considerando como telefonía convencional.

China

La telefonía IP está permitida a través de proveedores autorizados. Inicialmente se tenía un marco regulatorio por medio del cual se prohibía la prestación de estos servicios, sin embargo, este marco cambio a raíz de una decisión judicial, la cual implicaba a los propietarios de un café Internet en donde se prestaba el servicio de telefonía IP; estos fueron arrestados y sus equipos confiscados. Al final los propietarios del café terminaron ganando en segunda instancia la demanda que entablaron contra el ente regulador, provocando un cambio radical de la política, hasta tal punto en que la China se propuso como política de Estado incentivar la telefonía IP e invertir capital en I+D, con el fin de desarrollar el estándar chino de Telefonía IP a la mayor brevedad ya que se encuentran convencidos de la posibilidad de reemplazar la telefonía convencional por completo.

2.7. CALIDAD DE LA VOZ.

[18] El servicio de una aplicación es la funcionalidad proporcionada al usuario (hablar, ver un vídeo, transferir un fichero, etc.). La descripción de su rendimiento es la calidad de servicio de la aplicación (Q_0S, por sus siglas en inglés). Se trata de una medida cualitativa de la satisfacción del usuario, la diferencia entre lo que el usuario espera y lo que percibe. Por ello, la calidad de servicio de la aplicación requerida depende tanto del servicio o tipo de aplicación como de los deseos de los usuarios.

La calidad de servicio podría referirse a la fidelidad del mensaje original (la intangibilidad de la voz, su calidad, el movimiento de la imagen en vídeo, los errores en la transferencia de un fichero…), a la interactividad (tiempo de espera de la respuesta durante una conversación telefónica, tiempo de espera de recepción de una página web…) o a la sincronización entre varios flujos (la voz y el movimiento de un lápiz en una pizarra compartida, las voces de varios interlocutores en una audio conferencia…). Naturalmente esta calidad del servicio debe traducirse a parámetros cuantitativos o medibles (un número de imágenes reproducidas por segundo, un máximo tiempo de espera).

La calidad del servicio de aplicación ofrecida depende de dos factores. Por un lado está el comportamiento de la red o calidad de servicio de la red que hace referencia a como la red entrega el flujo de paquetes, las pérdidas y los retardos que se producen, este aspecto es variable, la misma red se puede comportar de diferentes maneras en función de su estado de carga. Por otro lado está la implementación de la aplicación. Hay aplicaciones que, a partir de una misma calidad del servicio de la red, son capaces de ofrecer un mejor rendimiento a sus usuarios que otras. Como ejemplo sencillo podemos mencionar que si durante la visualización de un vídeo no se dispone de una imagen actual, es mejor repetir la anterior que mostrar una oscura. Este apartado corresponde a un comportamiento estático, la aplicación siempre se comportará de manera similar.

La calidad de servicio (Q_0S) es la habilidad de los los elementos de red (Aplicaciones, Hosts, Routers, etc.) para proveer un cierto nivel de predicibilidad. En la actualidad no es posible garantizar la calidad de servicio de aplicaciones que requieren enlaces en tiempo real, como por ejemplo:

- ✓ Telefonía sobre IP.
- ✓ Vídeo conferencia.
- ✓ Telecontrol.

Adicionalmente en la actualidad, la industria no tiene definidos los mecanismos (protocolos) para estandarizar el manejo de la calidad de servicio, pero es claro que estos cambios no deben afectar en mayor medida la infraestructura de la INTERNET.

Entonces el objetivo de los protocolos de calidad de servicio es proveer cierto nivel de certidumbre y control en los enlaces sobre TCP/IP. Una de las formas de describir estos protocolos es estudiarlos de acuerdo a las aplicaciones así:

- ✓ *Reservación de Recursos (Servicio Integrado):* Los elementos de red son reservados según el requerimiento de calidad de servicio de las aplicaciones y el ancho de banda es manejado con alta prioridad.

- ✓ *Priorización (Servicio Diferencial):* El tráfico es clasificado y los recursos de la red son asignados según las políticas de manejo de ancho de banda.

- ✓ *Por Flujo:* Se define un flujo como una cadena de datos unidireccional entre dos aplicaciones única, identificada por: protocolo de transporte, fuente de datos, número de puerto origen, dirección de destino, puerto destino.

- ✓ *Flujos Agregados:* Un Flujo Agregado consta de dos o más flujos simples. Típicamente los flujos tienen algo en común, por ejemplo: Algunos de los

parámetros mencionados, una etiqueta o número de prioridad o quizás alguna información de autenticación.

Dependiendo de la topología de red, las aplicaciones y las políticas de calidad de servicio, en términos particulares se tienen diferentes protocolos y algoritmos que permiten dar un tratamiento de acuerdo a las necesidades:

- ✓ *ReSerVation Protocol (RSVP):* Provee la señalización para reservar recursos en la red, típicamente utilizado para flujos simples, aún cuando es posible utilizarlo para flujos agregados.

- ✓ *Differentiated Services (DiffServ):* Provee un simple y rudimentario método de categorizar y prioritizar los tráficos agregados de la red.

- ✓ *Multi Protocol Labeling Switching (MPLS):* Provee un manejo del ancho de banda para flujos agregados por medio de los enrutamientos de la red, haciendo uso de los encabezados de los paquetes.

- ✓ *Gestión del Ancho de Banda de la Subred (SBM):* Permite la priorización del nivel 2 del modelo OSI en las redes compartidas y switchadas IEEE 802. A pesar de presentarse de manera independiente los protocolos de calidad de servicio en el mundo real son diseñados para ser utilizados de manera conjunta de manera que permitan proveer la calidad de servicio top-to-bottom y end-to-end.

En general, la mayoría de las especificaciones para la interoperabilidad de estos protocolos está en desarrollo.

2.7.1. Parámetros Involucrados

[18] Los principales parámetros de la calidad de servicio en una red son:

La *velocidad de transferencia*, corresponde al volumen de datos por unidad de tiempo que una conexión puede cursar.

El *retardo*, es la demora que sufre una unidad de información (en general un paquete) entre origen y destino. Este retardo está formado por dos componentes: una parte fija y una variable.

- ✓ La parte fija viene dada por los procesos de empaquetado y desempaquetado, y el retardo de propagación de la señal (que solo es significativo en transmisión a larga distancia, también típico de las comunicaciones por satélite).

- ✓ La parte variable, se produce fundamentalmente en las colas de espera de los elementos de encaminamiento. El retardo depende de la carga de la red, a más carga más retardo.

2.7.2. Soluciones a Nivel de Aplicación.

[18] Un objetivo importante es construir aplicaciones que den el mejor rendimiento posible para cada uno de los distintos estados de la red. Son las llamadas aplicaciones adaptativas. Estas aplicaciones utilizan diversas técnicas que, a partir de la monitorización de la "calidad de servicio de red", consiguen la mejor "calidad de servicio de aplicación" posible. Esto también incluye la posibilidad de obtener provecho de una posible "Calidad de servicio de red" extra. La adaptación a las condiciones cambiantes también se refiere a tener en cuenta los recursos disponibles en el computador (host) para que sean asignados a la aplicación (CPU, memoria, tarjetas de comunicaciones, etc.). Esta cuestión está relacionada con el sistema operativo.

Lo que se comentaba anteriormente de las aplicaciones adaptativas puede ser visto de otro modo: a partir de una "Calidad de Servicio de la aplicación" requerida, hay aplicaciones que soportan una peor "Calidad de Servicio de la red". Naturalmente la sola utilización de aplicaciones adaptativas no basta. Si el usuario requiere o desea una calidad mínima, la red debe proporcionar una calidad de servicio mínima. Además hay aplicaciones que, por ser muy exigentes, no se pueden construir con técnicas adaptativas. Pero el estudio de las aplicaciones adaptativas es muy interesante por diversas razones. Por un lado, tan solo requieren una "Calidad de Servicio de red" mínima (de baja calidad), lo que seguramente implicará unos servicios de red más baratos, ya que serían más eficientes en el uso de los recursos de la red.

Según el tipo de flujos de paquetes que manejan, las aplicaciones pueden clasificarse en aplicaciones de tiempo real y en aplicaciones elásticas. Además pueden tener diferentes niveles de interactividad. Las aplicaciones de tiempo real manejan flujos dependientes del tiempo, por ejemplo los flujos de audio y de vídeo. Cada paquete tiene asociado un instante de reproducción. Si el paquete llega antes de ese instante, es utilizado, pero si llega más tarde, no sirve, es como si se hubiera perdido. Por este motivo, este tipo de aplicaciones es muy sensible al retardo y más concretamente a la variación del retardo (Jitter). Aunque normalmente estas aplicaciones tienen una cierta tolerancia a las pérdidas, también se podría dar el caso de flujos de tiempo real sensibles a las pérdidas, por ejemplo, cuando se requiere una gran fidelidad. Las aplicaciones elásticas manejan flujos independientes del tiempo. Se habla de flujos elásticos porque pueden ser "estirados" en el tiempo. Estas aplicaciones siempre esperan a que llegue un paquete y no prosiguen hasta que lo han recibido. Para poder actuar de

esta manera es imprescindible que sean tolerantes al retardo. En cambio, normalmente no toleran las pérdidas pues requieren una fidelidad total.

Como ejemplo de estos tipos de flujo podemos mencionar una transferencia de fichero mediante el protocolo FTP, la transmisión de correo electrónico o el acceso a un servidor web. Por otro lado, la interactividad de una aplicación exige mucho de la red, ya que requiere garantizar una cota máxima de retardo. Esta cota depende de cada aplicación. Por ejemplo, mientras que en telefonía retardos menores a 400 ms son aceptables, en la recepción de audio y vídeo bajo demanda los tiempos de respuesta de varios segundos son aceptables. Otras aplicaciones como la web, telnet o las herramientas de trabajo colaborativo tienen definidos sus propios niveles de interactividad.

2.8. ELEMENTOS DE REDES DE VOZ SOBRE IP.

[8] Nuevas tecnologías para transmitir voz sobre redes de datos representan oportunidades para que las empresas experimenten ahorros significativos en llamadas telefónicas, y desarrollen nuevas aplicaciones de e-business.

Muchas empresas han realizado inversiones en equipos de comunicaciones para el manejo específico de datos que están directamente relacionados con el negocio, al mismo tiempo mantienen toda una infraestructura de teléfonos y PBX's para el manejo de voz. Dos infraestructuras en lugar de una significan diferentes tipos de servicios, soporte y personal requerido para administrarlas.

Ya que los equipos de comunicación de datos tienen la capacidad para manejar tráfico adicional, en lugar de manejar dos redes paralelas, las empresas tienen la oportunidad de utilizar esta infraestructura, con poca o mínima inversión adicional, para el manejo de voz, e inclusive fax.

Transmitir Voz sobre IP en redes diseñadas para datos no solamente puede ser útil para realizar llamadas telefónicas, sino también para otro tipo de aplicaciones que las empresas necesitan para competir en el nuevo mundo de la INTERNET.

Realmente la integración de voz y datos en una misma red es una idea antigua, pues desde hace tiempo han surgido soluciones desde distintos fabricantes que, mediante el uso de multiplexores, permiten utilizar las redes WAN de datos de las empresas (típicamente conexiones punto a punto y frame-relay) para la transmisión del tráfico de voz. La falta de difusión de los estándares, así como la falta de regulación que permita promover el uso de VoIP en Colombia, no ha despertado el desarrollo de aplicaciones en las que los usuarios finales observen su amplia difusión e implantación de las mismas.

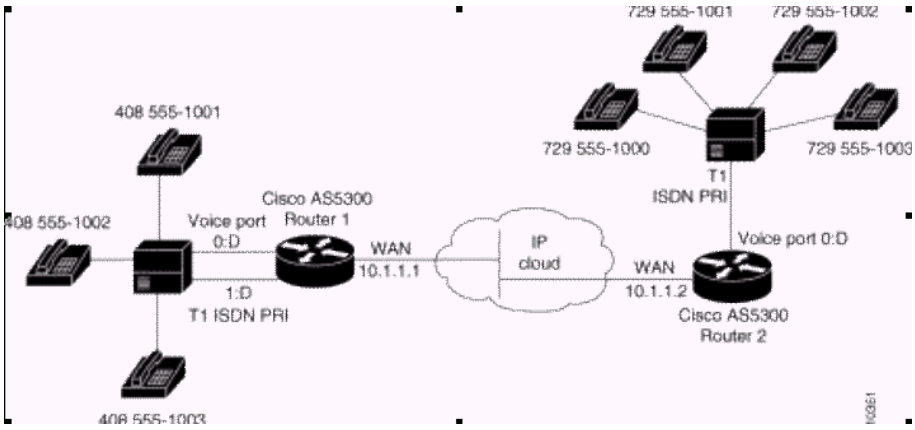

Figura 7. *Ejemplo de red con conexión de centralitas a routers CISCO que disponen de soporte VoIP.*

Actualmente podemos partir de una serie de elementos ya disponibles en el mercado y que, según diferentes diseños, nos permitirán construir las aplicaciones VoIP. Estos elementos son:

- ✓ Teléfonos IP.
- ✓ Adaptadores para PC.
- ✓ Hubs Telefónicos.
- ✓ Gateways (pasarelas entre PSTN e IP).
- ✓ Gatekeeper.
- ✓ Unidades de Control Multipunto (MCU).
- ✓ Servicios de Directorio.

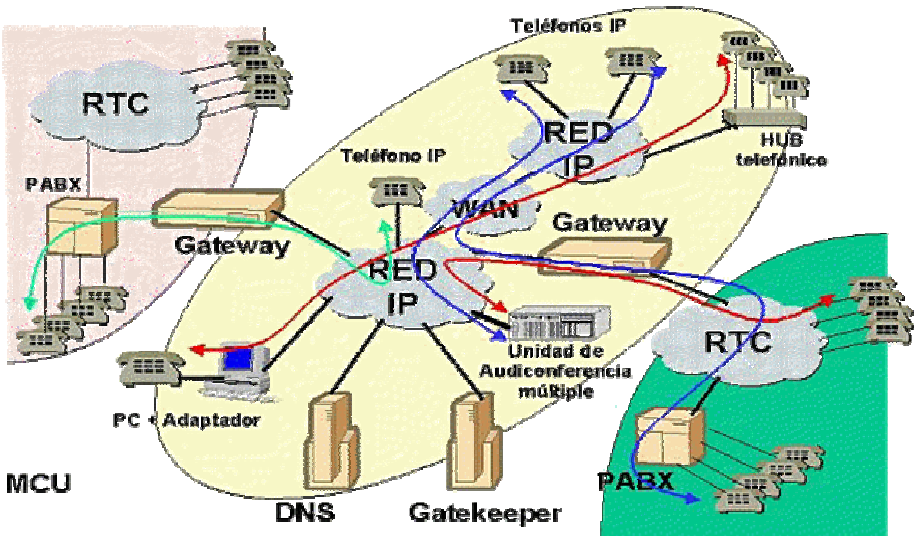

Figura 8. *Elementos de una red VoIP.*

Las funciones de los distintos elementos son fácilmente entendibles a la vista de la figura 7, si bien merece la pena recalcar algunas ideas. El Gatekeeper es un elemento opcional en la red, pero cuando está presente, todos los demás elementos que contacten dicha red deben hacer uso de aquel. Su función es la de gestión y control de los recursos de la red, de manera que no se produzcan situaciones de saturación de la misma.

El Gateway es un elemento esencial en la mayoría de las redes pues su misión es la de enlazar la red VoIP con la red telefónica analógica o RDSI. Podemos considerar al Gateway como una caja que por un lado tiene un interface LAN y por el otro dispone de uno o varios de los siguientes interfaces:

FXO. Para conexión a extensiones de centralitas ó a la red telefónica básica.
FXS. Para conexión a enlaces de centralitas o a teléfonos analógicos.
E&M. Para conexión específica a centralitas.
BRI. Acceso básico RDSI (2B+D)
PRI. Acceso primario RDSI (30B+D)
G703/G.704. (E&M digital) Conexión especifica a centralitas a 2 Mbps.

Los distintos elementos pueden residir en plataformas físicas separada, o nos podemos encontrar con varios elementos conviviendo en la misma plataforma. De este modo es bastante habitual encontrar juntos Gatekeeper y Gateway. También podemos ver en la figura 6 cómo Cisco ha implementado las funciones de Gateway en el router.

Un aspecto importante a reseñar es el de los retardos en la transmisión de la voz. Hay que tener en cuenta que la voz no es muy tolerante con estos. De hecho, si el retardo introducido por la red es de más de 300 milisegundos, resulta casi imposible tener una conversación fluida. Debido a que las redes de área local no están preparadas en principio para este tipo de tráfico, el problema puede parecer grave.

Hay que tener en cuenta que los paquetes IP son de longitud variable y el tráfico de datos suele ser a ráfagas. Para intentar obviar situaciones en las que la voz se pierde porque tenemos una ráfaga de datos en la red, se ha ideado el protocolo RSVP, cuya principal función es trocear los paquetes de datos grandes y dar prioridad a los paquetes de voz cuando hay una congestión en un router. Si bien este protocolo ayudará considerablemente al tráfico multimedia por la red, hay que tener en cuenta que RSVP no garantiza una calidad de servicio como ocurre en redes avanzadas tales como ATM que proporcionan QoS de forma estándar.

3. SOFTWARE DISEÑADO Y DESARROLLADO

Se mencionó en el Plan de Proyecto de Grado de la práctica empresarial, un item relacionado con la investigación sobre UML. Se hacía con el objeto de facilitar la comunicación entre los analistas y los programadores con que cuenta TYT Ltda. Para el presente caso el practicante empresarial tendría el rol de programador.

Se hizo entonces una recopilación de información sobre el UML, siendo esta técnica de modelado, objeto de estudio por parte del practicante en los tiempos que señala el cronograma de actividades. Sin embargo, para efectos prácticos los sistemas desarrollados durante la práctica empresarial no requerían algún tipo de modelado relacionado con UML. No fue necesario crear alguno de los tipos de diagramas que señala dicha técnica, esto por la complejidad y arquitectura requerida.

De acuerdo con el cronograma de actividades del Plan de Proyecto de Grado en ningún momento había compromiso alguno en cuanto a implementar el Software VoIP y el Generador de Código Oracle. Ambos sistemas son un valor agregado de la práctica empresarial.

Para tener una referencia, el diseño y desarrollo, así como también las pruebas realizadas sobre cada uno de los sistemas implementados, fueron hechos sobre un computador personal con las siguientes características, obtenidas gracias a un software de diagnóstico:

Informe (editado) Generado por *Everest Home Edition*

Ordenador:

Sistema operativo	Microsoft Windows XP Professional
Service Pack del Sistema Operativo	Service Pack 2
Internet Explorer	6.0.2900.2180 (IE 6.0 SP2)
DirectX	4.09.00.0904 (DirectX 9.0c)
Nombre del sistema	EQUIPO1 (Equipo1)
Nombre de usuario	Cesar Ruiz

Placa base:

Tipo de procesador	Intel Pentium 4ª, 2766 MHz (21 x 132)
Nombre de la Placa Base	ASRock GE Pro-M2 (2 PCI, 1 AGP, 1 AMR, 2 DDR DIMM, Audio, Video, LAN)
Chipset de la Placa Base	SiS 650GX
Memoria del Sistema	480 MB (DDR SDRAM)
Tipo de BIOS	AMI (11/04/03)
Puerto de comunicación	Puerto de comunicaciones (COM1)
Puerto de comunicación	Puerto de impresora ECP (LPT1)

Monitor:	
Tarjeta gráfica	SiS 650_650GL_650GX_651 (32 MB)
Acelerador 3D	SiS 315 Integrated
Monitor	AOC Spectrum 5E(A)/5En(A)/5Elr(A) [15" CRT] (1337686569)
Multimedia:	
Tarjeta de sonido	SiS 7012 Audio Device
Almacenamiento:	
Controlador IDE	SiS 5513 IDE UDMA Controller
Disquetera de 3 ½	Unidad de disquete
Disco duro	Maxtor 4R080L0 (80 GB, 5400 RPM, Ultra-ATA/133)
Disco duro	ST340016A (40 GB, 7200 RPM, Ultra-ATA/100)
Lector óptico	ATAPI-CD ROM-DRIVE-56MAX (56x CD-ROM)
Estado de los discos duros SMART	OK
Dispositivos de entrada:	
Teclado	Teclado estándar de 101/102 teclas o Microsoft Natural PS/2 Keyboard
Ratón	Mouse PS/2 de Microsoft
Red:	
Tarjeta de Red	SiS 900-Based PCI Fast Ethernet Adapter (192.168.0.35)
Modem	Smart Link 56K Voice Modem
Dispositivos:	
Impresora	Lexmark Z600 Series
Impresora	Microsoft Office Document Image Writer
Controlador USB1	SiS 7001 PCI-USB Open Host Controller
Controlador USB1	SiS 7001 PCI-USB Open Host Controller
Controlador USB2	SiS 7002 USB 2.0 Enhanced Host Controller
Dispositivos USB	Compatibilidad con impresoras USB

Se habla entonces del desarrollo de dos sistemas, uno de ellos (Software VoIP) tuvo relación directa con el principal tema de investigación y el otro (Generador de Código Oracle) tuvo más relación con el proyecto Herramienta Software para Contact Center. Ambos se describen a continuación.

3.1. SOFTWARE VOIP.

Se tenía planificado hacia el interior de la empresa, plasmar o reflejar el conocimiento adquirido durante la práctica y que se encuentre relacionado con el principal tema de investigación que es Voz sobre IP. Para éste propósito, también se quiso tener en cuenta el proyecto "Herramienta Software para Contact Center", es decir, se hizo necesario diseñar una herramienta que plasmara el conocimiento adquirido de Voz sobre IP, pero que se encontrara orientada hacia los intereses del proyecto, en el cual se enmarca la práctica empresarial. El modelo planteado básicamente es el siguiente:

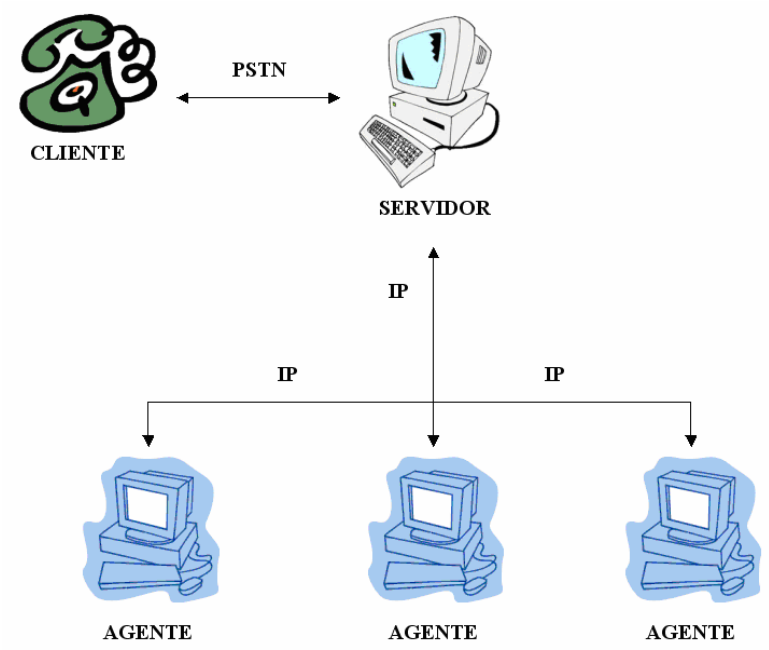

Figura 9. *Esquema básico de funcionamiento del software VoIP.*

Se puede describir el anterior diagrama por medio de los siguientes pasos:

a) Uno de los clientes puede comunicarse por medio de la red de telefonía pública conmutada (PSTN) al número telefónico asociado al servidor de la anterior figura.

b) El servidor procede a enrutar la llamada por medio de IP, a uno de los agentes disponibles. La elección del agente se hace aleatoriamente.

c) Finalmente la comunicación queda directa entre el agente y el cliente.

En este orden de ideas, el anterior software permite hacer la simulación de un gateway, al convertir una señal de voz que viaja por IP hacia PSTN y viceversa. El código que se escribió involucró específicamente la parte a), teniendo en cuenta las partes b) y c) para futuros trabajos muy probablemente al interior de TYT Ltda.

Se desarrolló el software en Microsoft Visual Basic 6.0, debido a que brindaba una mayor flexibilidad en cuanto a la programación orientada a objetos y también por la calidad del soporte que brinda Microsoft en cuanto a la implementación de este tipo de aplicaciones.

3.1.1. Definición de Archivo DLL

[19] DLL es el acrónimo de *Dynamic Linking Library* (Bibliotecas de Enlace Dinámico), término con el que se refiere a los archivos con código ejecutable que se cargan bajo demanda del programa por parte del sistema operativo. Esta denominación se refiere a los sistemas operativos Windows siendo la extensión con la que se identifican los ficheros, aunque el concepto existe en prácticamente todos los sistemas operativos modernos.

Ventajas: Las DLLs pueden verse como la evolución de las bibliotecas estáticas y de forma análoga contienen funcionalidad y/o recursos que utilizan otras aplicaciones. Su uso proporciona algunas ventajas:

- ✓ *Reducen el tamaño de los archivos ejecutables:* Gran parte del código puede estar almacenado en bibliotecas y no en el propio ejecutable lo que redunda en una mejor modularización

- ✓ *Pueden estar compartidas entre varias aplicaciones:* Si el código es suficientemente genérico, puede resultar de utilidad para múltiples aplicaciones (por ejemplo, la MFC es una biblioteca dinámica con clases genéricas que recubren la API gráfica de Windows y que usan gran parte de las aplicaciones).

- ✓ *Facilitan la gestión y aprovechamiento de la memoria del sistema:* La carga dinámica permite al sistema operativo aplicar algoritmos que mejoren el rendimiento del sistema cuando se carguen estas bibliotecas. Además, al estar compartidas, basta con mantener una copia en memoria para todos los programas que la utilicen.

- ✓ *Brindan mayor flexibilidad frente a cambios:* Es posible mejorar el rendimiento o solucionar pequeños errores distribuyendo únicamente una nueva versión de la biblioteca dinámica. Nuevamente, está corrección o mejora será aprovechada por todas las aplicaciones que compartan la biblioteca.

Problemas: Sin embargo, no todo son ventajas. En los sistemas Windows, las DLLs son muy comunes y muchos programas usan las mismas DLLs. Pero debido a la evolución, cada una de las DLLs evoluciona incorporandose mejoras pero modificandolas de tal forma que dejan de ser compatibles. Esto puede producir dos efectos no deseados:

- ✓ La instalación de un programa reemplace una DLL con una nueva versión incompatible.
- ✓ La desinstalación del programa borre una DLL compartida.

En ambos casos, el resultado es que dejan de funcionar los programas que utilizaban la vieja versión. Estos problemas se denominaron *el infierno de las DLLs*. Las versiones más modernas de Windows y los nuevos scripts de instalación MSI (sobre todo su característica de instalaciones reisidentes) abordan este problema. Sin embargo, el problema se mantiene cuando se utilizan otros instaladores (versiones antiguas) o se realizan modificaciones manuales.

3.1.2. Acerca de WaveStream.dll.

[20] WaveStream.dll es uno de los componentes de Microsoft Windows, para soporte de audio. Ella puede grabar la voz y luego comprimirla y enviarla a los sockets TCP conectados. Las funciones pueden ser vistas en el examinador de objetos de Visual Basic 6.0. Desde el lado del receptor, esta descomprime los datos de voz entrante y la reproduce. Las aplicaciones usan las funciones de WaveStream-callback durante la captura de flujos para procesar los buffers de audio. La ventana de captura llama una función WaveStream-callback justo antes de escribir cada buffer de audio al disco.

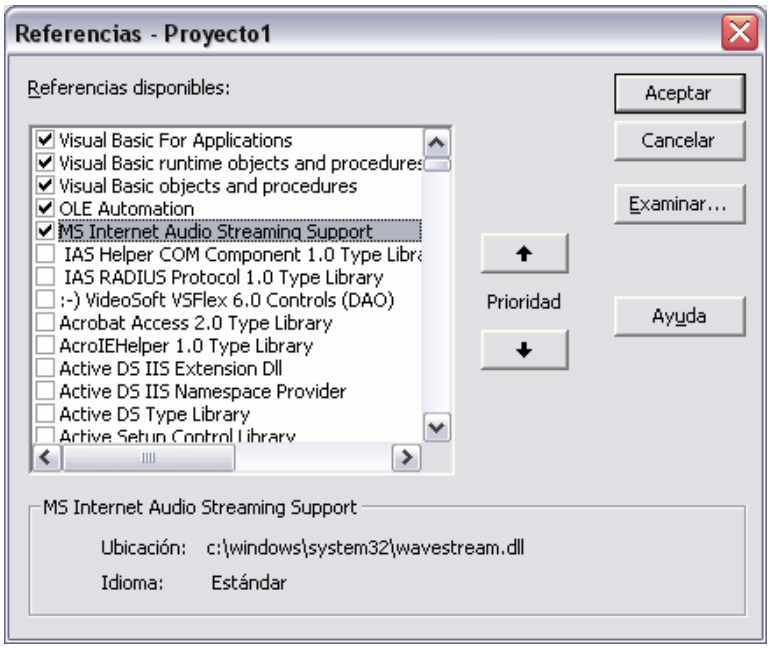

Figura 10. *Soporte para Audio.*

Antes de ejecutar la aplicación Software VoIP, se debe registrar el componente DLL como se muestra a continuación:

a) Clic en Menú Inicio.
b) Clic en Ejecutar.
c) Teclear: *regsvr32 c:\windows\system32\wavestream.dll,* para sistema Operativo Microsoft Windows XP Professional.

El componente WaveStream tiene dos importantes funciones:

a) **RecordWave():** Esta función graba la voz capturandola de los diferentes dispositivos. También obtiene el sonido grabado en los buffers. Este componente entonces comprime el buffer usando un administrador de compresión de Audio y lo envía a todos los sockets conectados. La descripción paso a paso de esta función es:
b) **PlayWave():** Esta función descomprime el buffer entrante y lo reproduce a los dispositivos de salida.

3.1.3. Interfaz de Software VoIP.

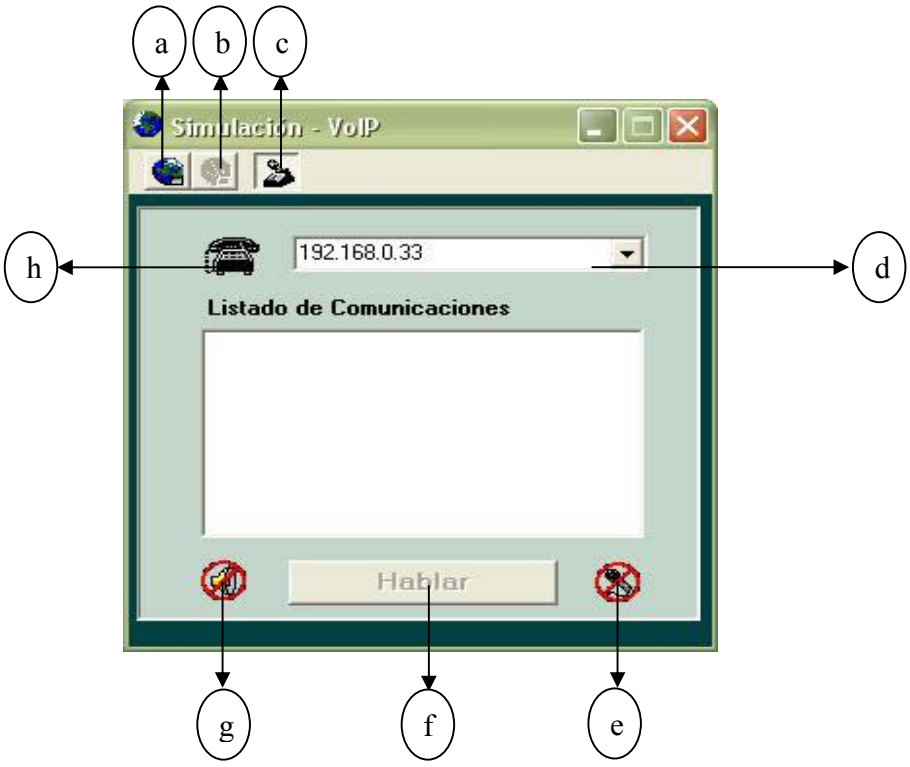

Figura 11. *Interfaz de Software VoIP.*

El Software cuenta con una interfaz sencilla, para que el manejo del mismo sea en lo posible intuitivo. Cada elemento se describe a continuación:

a) Este botón se utiliza para “ejecutar” la llamada a la otra dirección IP. Una vez se halla digitado la dirección en d) se procede a presionar este botón.
b) Este botón se utiliza para “colgar” la llamada, es decir para terminar una conversación de voz entre dos puntos.
c) Este botón se utiliza para poner la aplicación en modo “auto responder”, es decir que cuando se tenga una llamada entrante, el software por si solo responde al llamado.
d) En este combobox, se digita la dirección IP con la cual se quiere “iniciar” una conversación de voz y también se puede seleccionar un IP al cual se ha marcado con anterioridad.
e) Se trata de un indicador, que muestra si localmente se puede “hablar”, o se tienen los mecanismos habilitados para transmitir voz; principalmente cosas como que el micrófono esté funcionando son decisivas en este punto.
f) Con este elemento, se procede a “hablar” con la persona que está del otro lado.
g) Se trata de un indicador, que muestra si hay una llamada en “ejecución”, es decir si dos personas se están comunicando por IP con ésta aplicación, en determinado momento.
h) Se trata de un indicador, que muestra si localmente se puede “escuchar” lo que el usuario conectado del otro lado está tratando de transmitir.

[21] En los anexos, se encuentra el código fuente relacionado con las funciones de Chat y de TCP utilizadas.

3.2. GENERADOR DE CÓDIGO ORACLE.

3.2.1. Introducción a las Herramientas Generadoras de Código.

[22] De entre la variedad de posibilidades que ofrecen los métodos formales para la ayuda en el desarrollo de software, la generación automática de código es probablemente una de las aplicaciones más deseadas. El interés viene de antiguo pues no en vano se consideró que Fortran suponía la “programación automática”. Las teóricas ventajas de la generación de código a partir de especificaciones son bien conocidas:

- ✓ El desarrollador de software permanece a un alto nivel de abstracción más tiempo.
- ✓ El proceso de desarrollo de software es más coherente.
- ✓ El código generado es correcto sin necesidad de pruebas posteriores.
- ✓ Al igual que se genera el código puede generarse la documentación de manera que siempre se tiene una descripción clara y completa de la funcionalidad del sistema.
- ✓ El mantenimiento del software se simplifica al poder realizarse en el mismo contexto.
- ✓ La reutilización se ve favorecida en todas las etapas.

Se considera incluso que puede ser la única vía razonable para lograr código de alta fiabilidad necesario para aplicaciones de alto riesgo. Por el contrario, también existen los detractores de esta idea que encuentran en ella ciertos peligros y problemas:

- ✓ El sistema tendría sus limitaciones lo que podría convertirlo en una herramienta parcial y, por tanto, poco útil.
- ✓ La eficiencia del código generado puede ser un serio problema para muchas aplicaciones de interés.
- ✓ Las especificaciones formales no pueden capturar los requisitos no funcionales del sistema a realizar, lo que, en ciertos casos, penaliza el código obtenido.
- ✓ La generación de código suele necesitar especificaciones completas y no puede manejar especificaciones parciales, lo cual es muy común.

La mayor parte de los sistemas actuales proporcionan un prototipo que, con las limitaciones anteriores, es sumamente útil en varios aspectos:

- ✓ Permite la validación de los requisitos de manera que los malentendidos entre los desarrolladores de software y los usuarios y los requisitos incompletos o inconsistentes pueden identificarse.
- ✓ Permite descubrir requisitos adicionales.
- ✓ Permite evaluar decisiones de diseño e implementación en una etapa muy temprana cuyos cambios serían terriblemente costosos posteriormente.

Desgraciadamente, incluso en el caso de contar con un buen generador de código, lo habitual es que el prototipo obtenido sea “de usar y tirar”. Suele ser necesaria (incluso recomendable) la re-implementación del código con lo que el desarrollador tiene la impresión de malgastar tiempo y esfuerzo. Además, el coste

del desarrollo del prototipo puede representar una fracción inaceptable de los costes generales del sistema, lo que puede hacerlo prohibitivo.

3.2.2. Características de las Herramientas Generadoras de Código.

[21] Se pretende identificar qué herramientas de las propuestas en la literatura han alcanzado un grado de madurez suficiente para que puedan integrarse en un generador de código operativo. Por otra parte, también identificaremos algunas características que a nuestro entender deben aparecer en esta clase de herramientas.

Con respecto a las técnicas de generación de código, es evidente que la obtención de compiladores que obtengan código ejecutable a partir de algoritmos escritos en algún tipo de lenguaje es una tarea bien conocida para la que se conocen técnicas bien establecidas. Dos aspectos merecen más énfasis:

- ✓ *Técnicas de Transformación y Optimización de Programas:* La transformación de programas intenta obtener código más eficiente con la misma funcionalidad partiendo de un programa ya desarrollado (y probablemente validado y/o verificado). En este campo hay multitud de métodos y herramientas que se han revelado efectivos, relativamente fáciles de implementar y de gran aplicabilidad. Entre otros, mencionaremos la eliminación de la recursividad (presente en muchos compiladores), evitar la construcción de estructuras de datos intermedias (la llamada técnica de fusión) o la tabulación.

- ✓ *Técnicas de síntesis de algoritmos a partir de especificaciones:* La síntesis de programas trata de obtener un programa ejecutable a partir de una especificación. Por supuesto no es posible extraer siempre un programa y los métodos de síntesis tienen sus limitaciones. Normalmente se plantean como objetivo obtener un programa bastante abstracto (por ejemplo en un lenguaje declarativo bien funcional o lógico). Las técnicas son aplicables al problema de obtener un algoritmo (aunque sea en un modelo formal similar al de las especificaciones, como se ha propuesto anteriormente) a partir de una especificación de forma que la síntesis se convierte en el complemento de la transformación de programas: A partir de la especificación se obtiene un algoritmo vía el sintetizador y luego éste se transforma para obtener código eficiente.

Ambos métodos pueden ser la base de un buen generador de código una vez que se haya definido un buen modelo formal para expresar especificaciones y algoritmos. Se sugiere que este generador contenga adicionalmente las siguientes características:

- ✓ *Admitir la generación parcial de código:* La herramienta debe asumir como propias las limitaciones de los métodos propuestos. Hay muchas razones por las cuales el sistema puede ser incapaz de generar código completo:

  - Las especificaciones son parciales.
  - Los métodos de síntesis son también parciales, de manera que incluso ante una especificación completa pueden ser incapaces de generar algoritmos para todos los casos.
  - Los requisitos no funcionales pueden no haberse tenido en cuenta.

  Aunque se asuma que el código no cumple toda la funcionalidad propuesta, la herramienta debe ser capaz de generar la mayor información posible, indicando al usuario las limitaciones encontradas para que sea capaz de completarlo con código adicional. Nótese que son muchas las posibilidades de esta estrategia: Los generadores de esqueletos están incluidos en este tipo de herramientas. Por otro lado, incluso si no se es capaz de generar código ejecutable, pueden generarse aserciones ejecutables que ayuden en la depuración del código obtenido por otros métodos.

- ✓ *Generar código modificable:* Si asumimos que el código obtenido puede ser parcial, la conclusión inmediata es que este código debe ser relativamente fácil de completar por otros medios. La solución más sencilla es que el código obtenido sea legible por un programador que pueda completarlo y/o optimizarlo "a mano".

  Esta idea parece más controvertida. Podemos imaginarnos el código generado como un "ensamblador" que no es necesario inspeccionar y en el que "creemos" sin pedir más información. En cualquier caso, ambas ideas no parecen incompatibles: El código generado no tiene por qué ser legible en aquellos casos en que la herramienta generadora pueda asegurar que la funcionalidad se cumple en su totalidad o en el que ciertas herramientas complementarias automaticen el proceso de hacer convivir el código generado con otro producido de otra manera. Esta segunda situación es fácil de imaginar en un sistema donde el generador proporcione todo el código menos, por ejemplo, el de interacción con el usuario. Mientras los puntos de conexión entre una herramienta de generación de interfaces y el generador de código estén bien definidos, ambos pueden complementarse.

3.2.3. Paquetes en Oracle.

[23] Las características más relevantes son:

- ✓ Agrupan código PL/SQL, tipos de datos y subprogamas relacionados.
- ✓ Conformados por: Specification y Body.
- ✓ No pueden ser invocados, parametrizados o anidados.
- ✓ Son cargados en memoria completamente al primer llamado.
- ✓ Un Specification puede existir sin Body, pero un Body no puede existir sin una Specification. Ej: variables globales para la sesión.
- ✓ Se recomienda crearlos en archivos .SQL separados.

Sintaxis

```
CREATE OR REPLACE PACKAGE nombre IS
        -- variables publicas
        -- declaracion de subprogramas
END nombre;
```

Ejemplo:

```
CREATE OR REPLACE PACKAGE LiqNomina IS
        pCedula empleado.cedula%TYPE;

        PROCEDURE seleccionar (pdpto IN NUMBER);
        PROCEDURE liquidar (nit IN NUMBER);
        FUNCTION valorNovedad (nit NUMBER, fecha)
                RETURN NUMBER;
END LiqNomina;
```

Ventajas

- ✓ Encapsulamiento.
- ✓ Facilita el desarrollo.
- ✓ Persistencia en variables globales.
- ✓ Mejora el rendimiento.
- ✓ Una sola copia del código para todos los usuarios.
- ✓ Dependencia entre objetos se simplifica.
- ✓ Overloading.

3.2.4. Acerca de GeneraTYT.

GeneraTYT fue el nombre que se dio a la herramienta implementada. Se trató de dotar a TyT Ltda de una herramienta software que permitiese a los desarrolladores la generación de código de primer nivel, entendiéndose como aquel que implementa sobre una tabla determinada, las rutinas o interfaces comunes

necesarias para su acceso o manipulación. Se construyó una herramienta de software basada en el lenguaje PL/SQL que genera el "Table API" para una tabla específica, la cuál está compuesta de una serie de packages como sigue:

Package	Funcionalidad
<tabla>_tp	Todos los tipos de datos usados en los demás packages y que hacen relación a la tabla.
<tabla>_qp	Rutinas de consulta a la tabla por llave primaria, llaves foráneas o consultas dinámicas.
<tabla>_cp	Rutinas necesarias para la manipulación de registros sobre la tabla

Tabla 2. *Partes de GeneraTYT.*

El software a generar se corre mediante un bloque anónimo PL/SQL y por tanto no requiere de una interface de visualización o pantallas de interacción con el usuario. Toda corrida del generador permitirá que su resultado sea generado en archivos planos en el servidor de base de datos.

Con respecto a la anterior idea planteada, lo que se hizo durante la práctica empresarial fue hacer la construcción de los requerimientos necesarios para sacar adelante el producto, junto con el diseñador de software Ing. Humberto Rueda de TYT Ltda, además de adelantar lo suficiente la implementación del software.

3.2.4.1. Algunos Aspectos Técnicos.

- ✓ Se tienen 5 paquetes en total (3 centrales y 2 de apoyo), con la siguiente información:

Nombre Paquete	***Contenido***
Genera_Dicc	Consultas al Diccionario de Datos.
Genera_Util	Rutinas Utilitarias necesarias en los 3 paquetes centrales, tales como identar, imprimir, entre otras.
Genera_Qp, Genera_Cp y Genera_Tp	Rutinas que generan rutinas de cada paquete específico.

Tabla 3. *Componentes de la arquitectura de GeneraTYT v.1.0.*

- ✓ Se tiene un estándar de codificación para ambos códigos (generado y el generador). Este estándar se basa principalmente en poner las palabras reservadas de PL/SQL en mayúscula (FOR, LOOP, SELECT, etc) y todo lo demás como listados de columnas, nombres de funciones, entre otras, van en minúsculas. También ambos códigos se ciñen a las normas internacionales de indentación, además de que ambos siguen las reglas de las mejores prácticas de programación en PL/SQL.

- ✓ La mayoría de las funciones están acompañadas de otra con el mismo nombre + el sufijo "_Header", lo cual quiere decir que es su cabecera. Esto se hace con efectos de facilitar la construcción del Spec de algún paquete, que solo necesita las cabeceras de las funciones. Por esto la función OneRow tiene en un interior un llamado a la función OneRow_Header, ya que cuando OneRow es invocada debe mostrar tanto el encabezado, como su cuerpo y cierre, mientras que si llama OneRow_Header, solo mostrará lo que va en el Spec, osea la cabecera.

- ✓ Todas las funciones del código generador, cuentan con un vector de cadenas, que inicialmente está vacío y que a medida que avanzan las líneas de la función se van agregando cadenas a este vector, osea lineas del código generado. Por esto es que en todo el código se ven lineas del estilo del siguiente ejemplo:

```
Genera_Util.Ins(V, 'Cadena');
```

  … y de este modo es que todo el código es generado, llenando vectores de cadenas, para ser retornados.

3.2.4.2. Algunos Aspectos Generales para el Usuario.

Como es sabido el software está en capacidad de generar 3 paquetes con distintas funciones y/o procedimientos. Estos resultados pueden ser arrojados en pantalla o en archivo de texto. Para ver el listado de funciones por cada paquete puede remitirse al documento de especificaciones que reposa en la biblioteca de TYT Ltda.

Casos de Generación:

1. Si se quiere generar un paquete completo, basta con correr el siguiente bloque PL/SQL:

```
DECLARE
   V   Genera_Util.T_Vec_Renglon;
BEGIN
   V := Genera_Qp.Completo('GENERA','LINEA_CUENTA');
   Genera_Util.Print(V);
END;
/
```

Obsérvese la 4 línea, en donde se invoca a la función “Completo” del paquete Genera_Qp. De modo que si quiere generar tanto el Spec como el Body de cualquier paquete, basta con invocar a Genera_Qp.Completo, Genera_Cp.Completo ó Genera_Tp.Completo

Se debe tener en cuenta que los parámetros recibidos por cada una de las funciones se mencionan y/o aclaran en el documento de especificaciones, sin embargo, vale la pena mencionar que si usted desea pasar el Owner y el nombre de la tabla (para el caso anteriormente mencionado) en minúsculas, perfectamente puede hacerlo.

2. Si se quiere generar alguna de las funciones de algún paquete, como por ejemplo únicamente el Spec del Query Package, entonces el bloque anónimo de PL, quedaría así:

```
DECLARE
   V   Genera_Util.T_Vec_Renglon;
BEGIN
   V := Genera_Qp.PSpec('GENERA','LINEA_CUENTA');
   Genera_Util.Print(V);
END;
/
```

En la línea 4 nuevamente tenemos al Genera_Qp, pero esta vez invocando a la función PSpec (ó Package Spec). Si se desea invocar alguna otra rutina, basta con conocer su nombre, consultando el documento de especificaciones de GeneraTYT v.1.0.

Finalmente, se presentarán algunos ejemplos de la funcionalidad del software:

Ejemplo 1: Cuál bloque de PL debo usar si quiero generar la función, que me arroja por pantalla el procedimiento Ins?

R/

```
DECLARE
   V   Genera_Util.T_Vec_Renglon;
BEGIN
```

```
   V := Genera_Cp.Ins('GENERA','LINEA_CUENTA');
   Genera_Util.Print(V);
END;
/
```

Ejemplo 2: Cuál bloque de PL debo usar si quiero generar por archivo, todo el Type Package?

R/

```
DECLARE
   V   Genera_Util.T_Vec_Renglon;
BEGIN
   V := Genera_Tp.Completo('GENERA','LINEA_CUENTA');
   Genera_Util.Archivo(V, 'listo.txt');
END;
/
```

Nota: Obsérvese que en la línea 5, se utiliza la función Archivo del paquete Genera_Util. Esta función recibe como primer parámetro un vector de cadenas y luego recibe el nombre del archivo de texto, sin ruta, dado que para eso se creó un objeto directory.

Ejemplo 3: Cuál bloque de PL debo usar si quiero generar por pantalla la función Del_By del Change Package?

R/

```
DECLARE
   V   Genera_Util.T_Vec_Renglon;
BEGIN
   V := Genera_Cp.Del_By('GENERA','LINEA_CUENTA');
   Genera_Util.Print(V);
END;
/
```

Se debe tener en cuenta que el Query Package, el Type Package y el Change Package, tienen en su interior únicamente Funciones. Dentro de ellos no hay Procedure's. De modo que la forma de invocar lo que hay dentro de estos paquetes siempre será (a modo de sugerencia) como los 2 siguientes ejemplos:

- ✓ `V := Genera_Cp.Del_By('GENERA','LINEA_CUENTA');`
- ✓ `Genera_Util.Ins(V, Genera_Cp.DynDelStr('LINEA_CUENTA'));`

4. OTRAS ACTIVIDADES REALIZADAS.

4.1. CAPACITACIÓN ORACLE.

TYT Ltda, decidió a mediados de septiembre del año 2004, invertir en el aprendizaje de sus empleados sobre herramientas Oracle para lo cual dispuso a uno de sus accionistas para transmitir el conocimiento principalmente a los practicantes. Ambos cursos fueron brindados en las instalaciones de TYT Ltda, con un computador personal por asistente.

El docente fue el Ingeniero de Sistemas Humberto Rueda Rivero, quien cuenta con el siguiente Perfil Profesional:

> *Ingeniero con reconocida experiencia en el análisis, diseño e implementación de sistemas de información que enfoca su actividad profesional al ofrecimiento de la mejor solución. Posee gran capacidad investigativa en temas relacionados con la ingeniería del software, las telecomunicaciones y las bases de datos, la cuál es canalizada no sólo optimizando sus labores, sino también compartiendo el conocimiento adquirido con su grupo de trabajo. Está siempre dispuesto a aportar sus ideas y conocimientos en la búsqueda de soluciones a las situaciones que debe analizar. Posee facilidad para coordinar e interactuar en grupos de trabajo en donde sea permanente la búsqueda de metas y resultados.*

Primer Curso

Nombre: Oracle 9i (SQL) Bases para el desarrollo con Oracle.
Intensidad Horaria: 20 horas.
Temas Tratados:

- ✓ Diseño de bases de datos (modelo entidad relacion).
- ✓ Guias de codificación.
- ✓ Select (proyección, selección, join).
- ✓ Where (restricciones).
- ✓ Group by (agrupar datos)
- ✓ Order by (ordenamientos).
- ✓ Having (restricciones de grupo).
- ✓ Funciones.

- ✓ Operaciones con fechas.
- ✓ Subqueries (select aninados).
- ✓ Variables de sustitución.
- ✓ Update.
- ✓ Delete.
- ✓ Merge.
- ✓ Commit and rollback.
- ✓ Creacion de objetos.
- ✓ Constraints.
- ✓ Vistas.
- ✓ Secuencias.
- ✓ Indices.
- ✓ Sinónimos.
- ✓ Privilegios.
- ✓ Roles.
- ✓ Operaciones de conjuntos.
- ✓ Queries involutivas.
- ✓ Tablas externas.

Segundo Curso

Nombre: Oracle PL/SQL, El lenguaje de Oracle 9i.
Intensidad Horaria: 20 horas.
Temas Tratados:

- ✓ Estructura de un bloque PL/SQL.
- ✓ Tipos de bloques.
- ✓ Variables.
- ✓ Escribiendo sentencias.
- ✓ El cursor sql.
- ✓ Estructuras de control.
- ✓ Tipos de datos compuestos.
- ✓ Cursores.
- ✓ Execute inmediate.
- ✓ Manejo de excepciones.
- ✓ Creando procedures.
- ✓ Creando funciones.
- ✓ Privilegios.
- ✓ Packages.
- ✓ Manipulando LOB´s.
- ✓ Triggers.

4.2. INVESTIGACIÓN METODOLOGIAS DE DESARROLLO.

Todo desarrollo de software es riesgoso y difícil de controlar, pero si no se lleva una metodología de por medio, lo que se obtiene son clientes insatisfechos con el resultado y desarrolladores aún más insatisfechos.

La metodología reinante mundialmente en la actualidad es el Proceso Unificado de Desarrollo. Sin embargo una nueva tendencia ha venido emergiendo muy fuertemente en los últimos años y ésta se denomina Programación Extrema. Se pensó en estudiar ambas metodologías con el objeto de extractar lo mejor de las dos, ya sea para adoptar en TYT Ltda alguna en especial o una fusión de las dos como ya lo están haciendo algunas empresas alrededor del mundo. Especificando un poco más sobre los temas de investigación, se muestran a continuación los siguientes items:

4.2.1. Proceso Unificado de Desarrollo de Software (RUP).

[24] RUP es un proceso de desarrollo de software: Una forma disciplinada de asignar tareas y responsabilidades en una empresa de desarrollo (quién hace qué, cuándo y cómo). Tiene como principales objetivos: Asegurar la producción de software de calidad dentro de plazos y presupuestos predecibles. Es dirigido por casos de uso, centrado en la arquitectura, iterativo (mini - proyectos) e incremental (versiones).

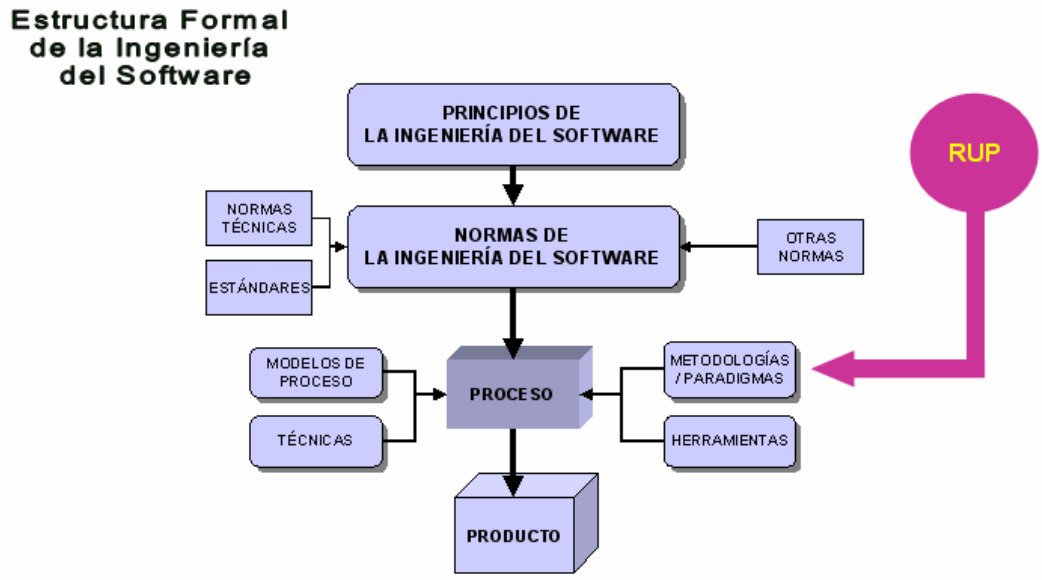

Figura 12. *Campo de Acción o ámbito de RUP.*

Es también un producto desarrollado y mantenido por Rational. Actualizado constantemente para tener en cuenta las mejores prácticas de acuerdo con la experiencia. Se centra en la producción y mantenimiento de modelos del sistema más que en producir documentos. RUP es una guía de cómo usar UML de la forma más efectiva.

Sobre este tema de investigación se generó un documento recopilatorio, con el siguiente contenido:

1. Orígenes.
2. Marco Teórico.
3. Aspectos Generales.
3.1. Mejores Prácticas.
3.2. Desarrollo Iterativo e Incremental.
3.3. Administración de Requerimientos.
3.4. Arquitectura Basada en Componentes.
3.5. Modelado Visual de Software.
3.6. Verificación de la Calidad.
3.7. Control de Cambios.

4. Dimensiones de RUP.
4.1. Dimensión Dinámica.
4.1.1 Primera Fase: Iniciación.
4.1.2. Segunda Fase: Elaboración.
4.1.3. Tercera Fase: Construcción.
4.1.4. Cuarta Fase: Transición.
4.2. Dimensión Estática.
4.2.1. Rol.
4.2.2. Actividad.
4.2.3. Artefacto.
4.2.4. Flujos de Trabajo
5. Otras Generalidades.

4.2.2. Programación Extrema.

[25] XP (eXtreme Programming), es una nueva disciplina para el desarrollo de software, que ha irrumpido recientemente con gran revuelo en el maremágnum de métodos, técnicas y metodologías existentes. Concretando más, se trata de un método «ligero», en contraposición a los métodos «pesados» como Métrica. Antes de continuar, se debe aclarar que en este artículo se hará alusión a él como «método», en contra de la tendencia oficial en Informática a denominar «metodologías» (ciencia de los métodos) a lo que no son más que métodos o incluso meras notaciones gráficas.

Podríamos decir que XP nace «oficialmente» hace cinco años en un proyecto desarrollado por Kent Beck en DaimlerChrysler, después de haber trabajado varios años con Ward Cunningham en busca de una nueva aproximación al problema del desarrollo de software que hiciera las cosas más simples de lo que nos tenían acostumbrados los métodos existentes. Para muchos, XP no es más que sentido común.

La siguiente frase fue extraída del mismo libro de Beck:

«Todo en el software cambia. Los requisitos cambian. El diseño cambia. El negocio cambia. La tecnología cambia. El equipo cambia. Los miembros del equipo cambian. El problema no es el cambio en sí mismo, puesto que sabemos que el cambio va a suceder; el problema es la incapacidad de adaptarnos a dicho cambio cuando éste tiene lugar.»

En cuanto a Extreme Programming, lo que el practicante hizo fue asistir a un ciclo de conferencias brindadas por TYT, lo cual le dio un mayor criterio a la hora de sugerir algunos puntos en la discusión, acerca de cuál metodología se iba a adoptar por la empresa.

4.3. SINTESIS DE LOS INFORMES DE AVANCE

Todas las etapas de la práctica empresarial han sido culminadas. Nunca fue necesario hacer un replanteamiento de objetivos, cronograma o presupuesto, sin embargo, más adelante en este último informe se mencionan algunos otros objetivos también alcanzados durante la práctica empresarial.

Además de los informes de avance mensuales, es preciso mencionar otros puntos de consideración relacionados con la ejecución de la práctica empresarial en TYT Ltda, que entre otras cosas, es de las primeras que se lleva a cabo en la Escuela de Ingeniería de Sistemas de la UIS. Estos se muestran a continuación:

- ✓ Dado que los informes de avance no cuentan con una estructura sugerida oficialmente por EISI – UIS, para la elaboración de los informes de avance, se siguió la estructura plasmada en el texto “El Proceso de la Investigación Científica” del autor Mario Tamayo y Tamayo.

  [26] Tal estructura típica se muestra a continuación:

  i. Nombre del Proyecto.
  ii. Objetivos del Proyecto.
  iii. Programa de Actividades.
  iv. Cronograma propuesto inicialmente.
  v. Actividades desarrolladas y trabajo realizado (compararlo con el cronograma).
  vi. Actividades y trabajo por hacer.
  vii. Replanteamiento de necesidades de tiempo (si fuera preciso).
  viii. Resultados alcanzados.
  ix. Recursos Económicos.
  x. Solicitud de adiciones presupuestales.

También se tuvieron en cuenta las recomendaciones de las normas ICONTEC (Norma Técnica Colombiana NTC 1486 – Quinta Actualización). Se ha hecho un acomodamiento de la anterior estructura a las necesidades y a la realidad presentada en la práctica empresarial en mención. Por esto, algunos de los anteriores ítems no han sido referenciados en los informes de avance. Aquellos ítems seleccionados, han sido fielmente respetados en el transcurso de la práctica empresarial, en cada uno de los informes mensuales.

- ✓ Teniendo acceso a INTERNET de banda ancha (ADSL) es posible conseguir mucha información comercial y menor cantidad de información académica y/o técnica relacionada con Voz sobre IP. Se logró acceder sin embargo, a algunas de las mejores bibliotecas universitarias colombianas. Además de esto, se contactaron profesionales reconocidos en la ciudad de Bucaramanga, relacionados con las áreas en mención durante la práctica empresarial.

- ✓ Para El ECAES (Examen de Estado de Calidad de la Educación Superior), el practicante adquirió conocimiento de valor considerable durante la práctica empresarial, principalmente en el campo de las Bases de Datos y en el campo de las Redes y Comunicaciones, dada la constante investigación y capacitación.

- ✓ [27] De acuerdo con la información brindada por la Universidad Industrial de Santander, los campos de desempeño del ingeniero de sistemas pueden ser los siguientes:

CAMPO DE DESEMPEÑO	FUNCIONES
Administrativo	*Planificación estratégica.* *Administración* *Información* *Personal* *Equipos* *Auditoría* *Gestión empresarial.* *Mercadeo* *Evaluación y ventas de Hardware y Software.*
Técnico	*Diseño y producción de software* *Manejo de redes de comunicación de datos.* *Automatización y control.* *Modelos y simulación.*

Social	*Comunicación y comportamiento Análisis y evaluación de la informática en la realidad social.*

Tabla 4. *Campos de Desempeño del Ingeniero de Sistemas, según la UIS.*

Se puede observar otra visión similar, gracias a La Universidad de los Andes, en donde se encuentra que:

> [28] *El ingeniero de sistemas colombiano se mueve, sobre todo, en 3 grandes áreas de conocimiento:*
>
> - ✓ *Desarrollo de software (entendido como todo el proceso de construcción de una solución informática a un problema bien definido).*
> - ✓ *Diseño y utilización de infraestructura tecnológica y de comunicaciones (entendida como la capacidad de diseñar y proponer una infraestructura de computadores, redes, sistemas operacionales, manejadores de bases de datos, software de comunicaciones, etc. a una organización como respuesta a sus necesidades).*
> - ✓ *Planeación y administración de los componentes informáticos de una organización (entendida como la capacidad de plantear alternativas de uso de la informática en todos los niveles de una organización).*

Esto anterior se ha mencionado con el objeto de mostrar la congruencia existente entre las labores que puede desempeñar un Ingeniero de Sistemas y las labores desempeñadas durante la práctica empresarial por el practicante en mención. Tales actividades o labores desempeñadas atacan principalmente a los siguientes campos:

- ✓ **Investigación:** Se ha hecho un trabajo de recopilación de información, estudio de la misma y exposición, al grupo de trabajo relacionado con la investigación en los temas concernientes a la Herramienta Software para Contact Center. Las principales fuentes de información fueron INTERNET y las Bibliotecas de La Universidad Industrial de Santander, La Universidad Pontificia Javeriana y La Universidad de los Andes. En este proceso investigativo cabe resaltar el contacto obtenido con personal reconocido en la ciudad de Bucaramanga, profesionales expertos en Voz sobre IP.
- ✓ **Diseño:** Algunas herramientas informáticas se hicieron indispensables, como resultado de un proceso investigativo y con miras a tener una ejecución exitosa del proyecto Contact Center.

Además de hacer propiamente un diseño de dos herramientas durante la práctica empresarial, también se hizo investigación sobre las diferentes Metodologías de Desarrollo de Software. Esto último con el propósito de dictaminar o escoger una o varias de ellas y definirla(s) como la Metodología de Desarrollo de TYT Ltda.

- ✓ **Desarrollo:** En cuanto al Desarrollo de Software se trabajaron lenguajes de programación de dos empresas en particular: Microsoft y Oracle. Se trabajaron principalmente los ambientes Visual Studio .NET y Oracle 9i. El software desarrollado en Oracle 9i estaba ajustado a las Mejores Prácticas de Programación. Los módulos desarrollados en Visual Studio .NET fueron hechos contando con una positiva realimentación, dada la constante información adquirida en medios de comunicación virtuales como GenteUIS (Página Oficial de los Estudiantes de EISI - UIS) y el contacto directo con prácticamente la totalidad de los integrantes del grupo CLAP (Computación, Lógica y Algorítmica Pura) de la UIS.

Cabe resaltar que los socios o accionistas de TYT Ltda (Empresa Partner de Oracle Corp.), cuentan con alto conocimiento en cuanto a herramientas Oracle y temas relacionados con Bases de Datos, dado su nivel de estudio y su experiencia de aproximadamente 10 años en una empresa como Telebucaramanga, reconocida en toda Colombia por su trabajo y su excelente servicio. Por ende, la realimentación lograda en cada una de las actividades, con los accionistas de TYT fue fundamental para que la práctica empresarial tuviera un éxito como el alcanzado.

- ✓ Hubo interacción con un equipo de trabajo en los siguientes casos, entre otros:

  - Diseño de software VoIP.
  - Diseño del Generador de Código Oracle.
  - Implementación del Generador de Código Oracle.
  - Capacitación sobre Oracle 9i.
  - Capacitación sobre PL/SQL.
  - Capacitación sobre CMM (Capability Maturity Model).
  - Capacitación sobre XP (Extreme Programming).
  - Unificación de Criterios en las Investigaciones.

- ✓ Con relación a los objetivos específicos presentados en el Plan de Proyecto de Grado, estos fueron cumplidos uno por uno durante la práctica empresarial, de la siguiente forma:

- *Afianzar los conocimientos de UML (Lenguaje Unificado de Modelado) para lograr la integración en cuanto al lenguaje con el grupo de trabajo de TYT Ltda.*

  Una de las primeras fases de de la práctica empresarial, consistió en realizar un estudio o repaso acerca de UML. La idea principal, era reforzar los conocimientos adquiridos en la Universidad, gracias a la lectura, resolución de ejercicios e interacción con el grupo de trabajo. Se hizo esto pensando básicamente en la implementación de la Herramienta Software para Contact Center, ya que era importante encontrar un lenguaje común entre programadores y diseñadores. Se informó sobre dicha actividad en el informe de avance número 1.

- *Trabajar en las tareas de desarrollo asignadas según el cronograma del proyecto de CC de TYT Ltda.*

  De acuerdo con el cronograma de actividades del Proyecto CC de la empresa, lo que se hizo fue suplir la fase de investigación al trabajar temas como Voz IP y metodologías de desarrollo y la implementación del generador de código oracle. Esto último acelera la fase de desarrollo del Proyecto CC. Se informó sobre dicha actividad en los informes de avance número 5 y 6.

- *Llevar a cabo una investigación preliminar de "Voz sobre IP" con el objeto de que los conocimientos adquiridos durante la carrera, se adapten y acoplen a lo que la empresa TYT Ltda necesita de los mismos.*

  En el cronograma de actividades del Plan de Proyecto de Grado, se contemplaban tres meses para investigar sobre las características principales de la VoIP y otros tres meses para trabajar la cuestión de las soluciones y estándares existentes.

  Se ha convertido este trabajo de seis (6) meses a cuatro (4) meses, con el objeto de que se tuviese inicialmente un repositorio de información o una base de conocimiento y que posteriormente en los dos meses finales lo que se hiciese fuera una depuración y mejoramiento de la información ya obtenida. Esta depuración incluye la implementación del software relacionado, con sus respectivas pruebas, así como la actualización de ciertos contenidos en la base de datos. Se informó sobre dicha actividad en los informes de avance número 1, 2, 3 y 4.

- *Participar en el diseño de componentes específicos relacionados con la implementación de software que permita manipular la voz sobre IP*

Este objetivo ha sido cumplido gracias al desarrollo del Software VoIP. La participación fue directa y se hizo tomando ciertos fragmentos de código fuente e ideas de INTERNET. Con este se aplicaron muchos de los conceptos que se investigaron previamente relacionados con la manipulación de VoIP. Se informó sobre dicha actividad en los informes de avance número 5 y 6.

- *Investigar sobre los antecedentes y las tendencias de la voz sobre IP a nivel mundial, con el objeto de alimentar de información a TYT Ltda y de incrementar el conocimiento por parte del practicante.*

  Se obtuvo información de diversas fuentes, que mostrasen una cierta concordancia entre lo que querían expresar y la realidad en general. Esto es útil para que TYT Ltda, comprenda hacia donde va el mercado de VoIP, cuáles son las últimas herramientas que se encuentran en el mundo, cómo ha sido la trayectoria que ha traido este negocio en los últimos años, entre otras ventajas. Se informó sobre dicha actividad en los informes de avance número 1 y 2.

- *Incrementar las bases de datos de TYT Ltda con información adicional acerca de estándares, licenciamientos, plataformas y proveedores, entre otros subtemas que se encuentren relacionados con la voz sobre IP.*

  Se trata de un apoyo adicional, para recopilar mayor información sobre el tema principal de investigación, ya que se necesita tener en cuenta otros puntos, además de las características de la VoIP. Además de los ítems mencionados en este objetivo específico, se obtuvo información sobre otros también de alta relevancia. Se informó sobre dicha actividad en los informes de avance número 3 y 4.

- *Buscar componentes software comercializados por otras casas de software, relacionados con los subtemas anteriormente descritos que sean compatibles con las herramientas de desarrollo del proyecto CC y analizar hasta que punto es más conveniente usarlos en lugar de invertir tiempo en el desarrollo propio por parte de la empresa.*

  En INTERNET se encontró suficiente código fuente gratuito. Existe suficiente código bajo los lenguajes de programación Java y Visual Basic. Para los intereses de TYT éste código fue llevado al repositorio de datos o servidor de información, además de otras herramientas que si bien no se encontraban libres con su respectivo código fuente, si brindaban una mayor visión acerca de lo que podía hacer TYT en el desarrollo del Proyecto CC.

De este modo se tomaron ideas, de ciertas funcionalidades que podía brindar la herramienta software para Contact Center, para posteriores desarrollos, mientras que otras ideas si se iban plasmando directamente en el Software VoIP. Se informó sobre dicha actividad en los informes de avance número 3,4 y 5.

Adicionalmente a los objetivos específicos anteriormente descritos, es importante tener en cuenta que se realizaron algunas tareas o actividades adicionales, las cuales se pueden ver claramente representadas mediante los siguientes objetivos específicos:

- Adquirir experiencia y destreza relacionada con diseño y desarrollo, mediante herramientas de última generación, además de una disciplina de trabajo muy importante a la hora de enfrentar el medio laboral.

- Recibir capacitación constante y altamente calificada acerca de temas relacionados con Bases de Datos y Metodologías para el desarrollo de software, entre otros de actualidad y de interés como profesional y/o estudiante.

- Plasmar en un trabajo continuado, los conocimientos adquiridos durante la carrera de Ingeniería de Sistemas e Informática, a la vez reforzando los mismos mediante el complemento brindado por la práctica empresarial.

5. GLOSARIO

5.1. ACRÓNIMOS

ATM Asynchronous Transfer Mode (Modo de Transferencia Asíncrona).

CTI Computer Telephony Integration (Integración Ordenador- Telefonía).

DiffServ Differentiated Services Internet QoS model (modelo de Calidad de Servicio en Internet basado en Servicios Diferenciados).

DNS Domain Name System (Sistema de Nombres de Dominio).

H.323 Estándar de la ITU-T para voz y videoconferencia interactiva en tiempo real en redes de área local, LAN, e Internet.

IETF Internet Engineering Task Force (Grupo de Trabajo de Ingeniería de Internet).

IP Internet Protocol (Protocolo Internet).

IP Multicast Extensión del Protocolo Internet para dar soporte a comunicaciones multidifusión.

ITU-T International Telecommunications Union - Telecommunications (Unión Internacional de Telecomunicaciones - Telecomunicaciones).

MCU Multipoint Control Unit (Unidad de Control Multipunto).

PBX Private Branch Exchange (Centralita Telefónica Privada).

PPP Point to Point Protocol (Protocolo Punto a Punto).

PSTN Public Switched Telephone Network (Red de Telefonía Conmutada Pública).

QoS Quality of Service (Calidad de Servicio).

RSVP Reservation Protocol (Protocolo de Reserva).

RTCP Real Time Control Protocol (Protocolo de Control de Tiempo Real).

RTP Real Time Protocol (Protocolo de Tiempo Real).

SIP Session Initiation Protocol (Protocolo de Inicio de Sesión).

TCP Transmission Control Protocol (Protocolo de Control de Transmisión).

UDP User Datagram Protocol (Protocolo de Datagramas de Usuario).

VLAN Virtual Local Area Network (Red de Área Local Virtual)

VoIP Voice Over IP.

VPN Virtual Private Network (Red Privada Virtual)

5.2. TÉRMINOS.

Circuit Switching (conmutación de circuitos). Técnica de comunicación en la que se establece un canal (o circuito dedicado) durante toda la duración de la comunicación. La red de conmutación de circuitos más ubicua es la red telefónica, que asigna recursos de comunicaciones (sean segmentos de cable, «ranuras» de tiempo o frecuencias) dedicados para cada llamada telefónica.

Codec (codec). Algoritmo software usado para comprimir/descomprimir señales de voz o audio. Se caracterizan por varios parámetros como la cantidad de bits, el tamaño de la trama (frame), los retardos de proceso, etc. Algunos ejemplos de codecs típicos son G.711, G.723.1, G.729 o G.726.

Extranet (extranet). Red que permite a una empresa compartir información contenida en su Intranet con otras empresas y con sus clientes. Las extranets transmiten información a través de Internet y por ello incorporan mecanismos de seguridad para proteger los datos.

Gatekeeper (portero). Entidad de red H.323 que proporciona traducción de direcciones y controla el acceso a la red de los terminales, pasarelas y MCUs H.323. Puede proporcionar otros servicios como la localización de pasarelas.

Gateway (pasarela). Dispositivo empleado para conectar redes que usan diferentes protocolos de comunicación de forma que la información puede pasar de una a otra. En VoIP existen dos tipos principales de pasarelas: la Pasarela de Medios (Media Gateways), para la conversión de datos (voz), y la Pasarela de Señalización (Signalling Gateway), para convertir información de señalización.

Intranet (intranet). Red propia de una organización, diseñada y desarrollada siguiendo los protocolos propios de Internet, en particular el protocolo TCP/IP. Puede tratarse de una red aislada,

es decir no conectada a Internet.

IP Telephony (Telefonía Internet). Ver «Voice over IP»

Jitter (variación de retardo). Es un término que se refiere al nivel de variación de retado que introduce una red. Una red con variación 0 tarda exactamente lo mismo en transferir cada paquete

de información, mientras que una red con variación de retardo alta tarda mucho más tiempo en entregar algunos paquetes que en entregar otros. La variación de retardo es importante cuando se

envía audio o video, que deben llegar a intervalos regulares si se quieren evitar desajustes o sonidos ininteligibles.

Packet Switching (conmutación de paquetes). Técnica de conmutación en la cual los mensajes se dividen en paquetes antes de su envío. A continuación, cada paquete se transmite de forma individual y puede incluso seguir rutas diferentes hasta su destino. Una vez que los paquetes llegan a éste se agrupan para reconstruir el mensaje original.

Router (encaminador, enrutador). Dispositivo que distribuye tráfico entre redes. La decisión sobre a donde enviar los datos se realiza en base a información de nivel de red y tablas de direccionamiento. Es el nodo básico de una red IP.

VoIP, Voice over IP (Voz sobre IP). Método de envío de voz por redes de conmutación de paquetes utilizando TCP/IP, tales como Internet.

CONCLUSIONES

Indiscutiblemente la Voz sobre IP, se ha convertido en una tecnología que ha tenido mucho auge sobre todo en los países más industrializados y cada vez más se está adoptando incluso en los países tercermundistas. Las empresas de telecomunicaciones colombianas, paulatinamente han comprendido la importancia de llevar VoIP a sus ciudadanos, ya que además de ser un negocio excelente se convierte en un bien o servicio muy codiciado por la sociedad en general.

¿Cuál es la naturaleza de los servicios que presta VoIP? Aquí radica la discusión a nivel nacional, en lo que respecta a regulación y legislación. Lo que menos se quiere es que ésta tecnología tenga tantas "trabas" como para dificultar su normal despliegue sobre un determinado país o región, siendo también indeseado que tenga la suficiente libertad como para que se puedan cometer ilícitos. VoIP no goza del control necesario como si lo tiene PSTN; esta red realmente se encuentra mucho más organizada y mejor desplegada sobre el mundo, teniendo en cuenta también que es una tecnología mucho más antigua.

Alrededor del mundo, una de las formas más novedosas de desplegar VoIP sobre las organizaciones, es la que se observa cuando se hacen implementaciones de Contacts Center. En Colombia, existen varios de ellos, los cuales reunen información de algunas empresas asociadas, para brindar ciertos servicios a un cierto grupo o comunidad interesada.

Esta somera idea, ha tenido mucho auge en el ámbito empresarial y va con los intereses de TYT Ltda generar alguna solución (desde el software) para quienes se encuentren interesados en implementar un Contact Center, no solo en Colombia. Por esto, se inició un trabajo de investigación gracias a los practicantes que dispuso la Universidad Industrial de Santander.

Desde la óptica del practicante en mención en este libro, se adelantaron esfuerzos de investigación y desarrollo, para el momento definitivo de iniciación del proyecto "Herramienta Software para Contact Center" (a trabajar en conjunto con Colciencias). El hecho de haber desarrollado un Generador de Código Oracle y un software que permita comunicar dos PC's por IP, es un resultado claro que muestra los frutos del progreso de la práctica empresarial.

No se trata simplemente de navegar en INTERNET y presumir sobre un número determinado de páginas recopiladas. Para emprender un proyecto de tal magnitud, se requiere también dominar una metodología, lo cual implica que no se debe tener únicamente un pool de programadores, sino también un buen grupo de análisis y diseño, así como también un Gerente de Proyecto. Esta anterior

estructura, descrita muy a groso modo, va ligada a una cierta fusión que TYT hizo, acorde con la investigación llevada a cabo por sus practicantes, entre Rational Unified Process (RUP) y Extreme Programming (XP), las cuales son hoy por hoy las dos más importantes metodologías de desarrollo de software.

Es de resaltar entonces, el hecho de haber investigado sobre estas diferentes metodologías, ya que la empresa con este material recopilado pudo adoptar una postura, en cuanto a sus estándares de trabajo y a su forma de hacer las cosas, no siendo inferior, el aprovechamiento que el estudiante logró con la adquisición de este nuevo conocimiento.

Finalmente y para concluir, es importante dar crédito al mecanismo “práctica empresarial”, ya que realmente fue muy conveniente, tanto para el practicante, como para la universidad, como para las empresas. El hecho de haber interactuado con un grupo de trabajo tan calificado, así como el de inmiscuirse aún más en el medio laboral, entre muchas otras ventajas, hacen que ésta experiencia sea tan grata como aprovechable.

BIBLIOGRAFIA

CAPITULO 2.

[1] Serie de cuadernos de Política No 2. VoIP. Borrador para discusión. BOGOTA, Junio de 2004. Ministerio de Comunicaciones. http://www.mincomunicaciones.gov.co/Archivos/Documentos/vozipjun2004.pdf.

[2] Netvoice de AVANTEL. AVANTEL la compañía IP. http://www.netvoice.com.mx/index.html

[3] VoIP Brujula.Net. Telefonía Gratis por Banda Ancha. http://voip.brujula.net/

[4] Website Skype Technologies SA. 2002 – 2004. http://www.skype.com/

[5] Website Nortel Networks. Historias de éxito. http://www.nortelnetworks.com/solutions/pt/es/success.htm

[6] Website British Telecommunications. Perspectivas. Voz sobre IP. http://www.btglobalservices.com/es/perspectives/voip.html

[7] Cisco Systems Latinoamérica. Redacción Virtual. Estudio de IDC confirma adopción acelerada de Telefonía IP en Latinoamérica. Miami, USA. Junio 23 de 2004. http://www.redaccionvirtual.com/redaccion/articulodestacado/noticia4.asp?Id=506

[8] Monografías.com. Descripción Técnica Detallada sobre Voz sobre IP. Documento hecho por YMDG. http://www.monografias.com/trabajos11/descripip/descripip.shtml

[9] Recursos VoIP. Telefonía – Netmeeting. http://www.recursosvoip.com/intro/index.php

[10] Website Cisco Systems Latinoamérica. Telefonía IP. Historias de éxito. Pontificia Universidad Católica. Perú. Julio 2004. http://www.cisco.com/global/LA/LATAM/sne/pc/tecnologia/telefoniaip/ucatol.shtml

[11] Website Intel Corporation. IP Telephony. http://www.intel.com/network/csp/solutions/iptelephony/index.htm

[12] Website Netmedia SA. Artículos. Diciembre 14/2001.
http://www.netmedia.info/netmedia/articulos.php?id_sec=30&id_art=1357

[13] Microsoft Corporation. Microsoft Windows Netmeeting. Netmeeting Resource Kit. Deployment Chapter 11. Understanding the H.323 Standard.
http://www.microsoft.com/windows/NetMeeting/Corp/reskit/Chapter11/default.asp

[14] GONCALVEZ Marcus. Ed. McGraw-Hill. Agosto 31 de 1998.
ISBN 0079137830.

[15] Monografías.com. Telefonía IP – Protocolos de Señalización. Documento hecho por Roberto Ares.
http://www.monografias.com/trabajos16/telefonia-senalizacion/telefonia-senalizacion.shtml#protocolo

[16] IEEE Communications Surveys. JOSEF GLASMANN, WOLFGANG KELLERER, HARALD MÜLLER. Service Architectures in H.323 and SIP: A Comparison.
http://www.comsoc.org/livepubs/surveys/public/2003/oct/glasmann.html

[17] Website Terra. Sección Tecnología. Junio 29 de 2004. Primer Paso para Regular VoIP en Colombia.
http://enter.terra.com.co/ente_secc/ente_actu/noticias/ARTICULO-WEB-1001940-1720671.html

[18] SISTEMA DE CONTACT CENTER BASADO EN VOZ SOBRE IP (INTERNET PROTOCOL). Gustavo González Sánchez, Javier Mauricio Vásquez Gómez. Director Homero Ortega Boada. Tesis Posgrado. Escuela de Ingeniería Eléctrica, Electrónica y Telecomunicaciones. Universidad Industrial de Santander.

CAPITULO 3.

[19] Website de Wikipedia. Wikipedia la Enciclopedia Libre. DLL.
http://es.wikipedia.org/wiki/DLL

[20] TSENG, Ching-Yuan. IMPLEMENTATION OF VOICE OVER IP IN PEER TO PEER ENVIRONMENT.Thesis for the Degree of Master of Computer and Information Science. Knowledge Systems Institute. Skokie, Illinois, USA. December 2004. Approved By Dr. C.Y. Hsieh.

[21] MUGMAN21. Interner Voice Chat. Diciembre 21 de 2003.
http://www.planet-source-code.com/vb/scripts/ShowCode.asp?txtCodeId=50550

[22] MORENO, Juan José. ¿Qué se puede pedir hoy a un Generador Automático de Código? Universidad Politécnica de Madrid. http://www.willydev.net/descargas/GeneraCode.pdf

[23] RUEDA, Humberto. Curso de Oracle PL/SQL, El lenguaje de Oracle 9i. TYT Ltda.

CAPITULO 4.

[24] JACOBSON Ivar, BOOCH Grady, RUMBAUGH James. EL PROCESO UNIFICADO DE DESARROLLO DE SOFTWARE. Ed. Pearson Educación S.A., Madrid, 2000. ISBN: 84-7829-036-2.

[25] César F. Acebal, Juan M. Cueva Lovelle. E*xtreme Programming* (XP): Un nuevo método de desarrollo de software. Universidad de Oviedo. http://www.willydev.net/descargas/Articulos/General/xp.pdf

[26] TAMAYO, Mario. EL PROCESO DE LA INVESTIGACIÓN CIENTÍFICA. 4a. ed., Ed LIMUSA S.A. ISBN 968-18-5872-7. México. 2002.

[27] Website Universidad Industrial de Santander. Información Académica. Programas Académicos. Programa de Ingeniería de Sistemas e Informática. http://dodo.uis.edu.co/site/info_academica/prog_academicos/progs.jsp?cual3=16&codigo=undefined

[28] Website Universidad de los Andes. Programas Académicos. Departamento de Ingeniería de Sistemas. Programa de Pregrado. http://sistemas.uniandes.edu.co/manager.php?id=24

ANEXOS

ANEXO A. Funciones de TCP y Chat utilizadas en Sofware VoIP.

Funciones de Chat

```
Option Explicit

Public Sub InitServerList(ServerList As ComboBox)
   ServerList.AddItem "192.168.0.33"
   ServerList.AddItem "192.168.0.34"
End Sub

Public Sub DebugSocket(TCPSocket As Winsock)
   Debug.Print "TCPSocket.RemoteHost", TCPSocket.RemoteHost
   Debug.Print "TCPSocket.RemoteHostIP", TCPSocket.RemoteHostIP
   Debug.Print "TCPSocket.RemotePort", TCPSocket.RemotePort
   Debug.Print "TCPSocket.LocalHostName", TCPSocket.LocalHostName
   Debug.Print "TCPSocket.LocalIP", TCPSocket.LocalIP
   Debug.Print "TCPSocket.LocalPort", TCPSocket.LocalPort
   Debug.Print "TCPSocket.State", TCPSocket.State
   Debug.Print "=================================================="
End Sub

Public Sub ResPlaySound(ResourceId As Long)
   Dim sndBuff As String
   sndBuff = StrConv(LoadResData(ResourceId, "WAVE"), vbUnicode)
   Call sndPlaySound(sndBuff, SND_SYNC Or SND_MEMORY)
End Sub

Public Sub AddConnectionToList(Socket As Winsock, ConnList As ListBox)
   Dim MemberID As String

   MemberID = Socket.RemoteHostIP & "  [" & _
           Format(Socket.RemotePort, "0") & "] - [" & _
           Format(Socket.LocalPort, "0") & "]"

   ConnList.AddItem MemberID
   ConnList.ItemData(ConnList.NewIndex) = Socket.Index
End Sub

Public Sub RemoveConnectionFromList(Socket As Winsock, ConnList As ListBox)
   Dim Conn As Long
   Dim MemberID As String

   MemberID = Socket.RemoteHostIP & "  [" & _
           Format(Socket.RemotePort, "0") & "] - [" & _
           Format(Socket.LocalPort, "0") & "]"
```

```
    For Conn = 0 To ConnList.ListCount - 1
        If (ConnList.List(Conn) = MemberID) Then
            ConnList.RemoveItem Conn
        End If
    Next
End Sub

Public Sub GetIdxFromMemberID(Sockets As Variant, MemberID As String, Index As Long)
    Dim Idx As Long
    Dim LocPortID As Long
    Dim RemPortID As Long
    Dim RemoteIP As String
    Dim sStart As Long
    Dim sEnd As Long
    Dim Socket As Winsock

    sStart = 1
    sEnd = InStr(1, MemberID, " ") - 1
    If (sEnd > 1) Then
        RemoteIP = Mid(MemberID, sStart, sEnd)
        sStart = InStr(sEnd, MemberID, "[") + 1
        If (sStart > 1) Then
            sEnd = InStr(sStart, MemberID, "]") - 1
            If (sEnd > 2) Then
                RemPortID = Val(Mid(MemberID, sStart, sEnd))
                sStart = InStr(sEnd, MemberID, "[") + 1
                If (sStart > 1) Then
                    sEnd = InStr(sStart, MemberID, "]") - 1
                    If (sEnd > 2) Then
                        LocPortID = Val(Mid(MemberID, sStart, sEnd))
                        For Each Socket In Sockets
                            If ((Socket.RemoteHostIP = RemoteIP) And _
                                (Socket.RemotePort = RemPortID) And _
                                (Socket.LocalPort = LocPortID) And _
                                (Socket.Index > 0)) Then
                                Index = Socket.Index
                                Exit Sub
                            End If
                        Next
                    End If
                End If
            End If
        End If
    End If
End Sub
```

Funciones de TCP

```
Option Explicit

Public Function InstanceTCP(TCPArray As Variant) As Long
   Dim Ind As Long

   InstanceTCP = -1
   On Error GoTo InitControl
   For Ind = Mintcp To Maxtcp
      If (TCPArray(Ind).Name = "") Then
      End If
   Next
InitControl:
   On Error GoTo ErrorHandler

   If ((Ind >= Mintcp) And (Ind <= Maxtcp)) Then
      Load TCPArray(Ind)
      InstanceTCP = Ind
   End If

   Exit Function
ErrorHandler:
   Debug.Print Err.Number, Err.Description
   Resume Next
End Function

Public Function Connect(Socket As Winsock, RemHost As String, RemPort As Long) As Boolean
   Connect = False
   Call CloseListen(Socket)
   Socket.LocalPort = 0
   Call Socket.Connect(RemHost, RemPort)

   Do While ((Socket.State = sckConnecting) Or _
         (Socket.State = sckConnectionPending) Or _
         (Socket.State = sckResolvingHost) Or _
         (Socket.State = sckHostResolved) Or _
         (Socket.State = sckOpen))
      DoEvents
   Loop

   Connect = (Socket.State = sckConnected)
End Function

Public Function Listen(Socket As Winsock) As Long
   If (Socket.State <> sckListening) Then
      If (Socket.LocalPort = 0) Then
         Socket.LocalPort = Voiceport
      End If
      Call Socket.Listen
   End If
End Function
```

```
Public Function CloseListen(Socket As Winsock) As Long
   If (Socket.State = sckListening) Then
      Socket.Close
   End If
End Function

Public Sub Disconnect(Socket As Winsock)
   If (Socket.State <> sckClosed) Then
      Socket.Close
   End If
End Sub
```

ANEXO B. Specification del Query Package de GeneraTYT v.1.0.

```
PACKAGE GENERA_QP IS
-- INICIO DEL SPECIFICATION

-- PROPOSITO:
-- Este paquete contiene los procedimientos y funciones que permiten generar
-- el Query Package.

-- HISTORIAL DE MODIFICACIONES:
-- Humberto Rueda       23/02/2005      Creación del Paquete
-- César Ruiz²          25/02/2005      Organización Sintáxis y e inicio implementación

--------------------------------------------------------------------------------
-- Retorna el Cierre de una Determinada Rutina, en un Vector
-- Parámetros:  Cadena:     Nombre de la Rutina
FUNCTION Cierre(Cadena IN VARCHAR2)
RETURN Genera_Util.T_Vec_Renglon;
--------------------------------------------------------------------------------
-- Retorna Todo el Query Package, en un Vector
-- Parámetros:  P_Owner:    Nombre del Owner
--              P_Tabla:    Nombre de la Tabla
FUNCTION Completo(  P_Owner IN VARCHAR2,
                    P_Tabla IN VARCHAR2)
RETURN Genera_Util.T_Vec_Renglon;
--------------------------------------------------------------------------------
-- Retorna el Cuerpo del Specification del QueryPackage, en un Vector
-- Parámetros:  P_Owner:    Nombre del Owner
--              P_Tabla:    Nombre de la Tabla
FUNCTION CuerpoSpec(P_Owner IN VARCHAR2,
                    P_Tabla IN VARCHAR2)
RETURN Genera_Util.T_Vec_Renglon;
--------------------------------------------------------------------------------
-- Retorna el Spec del QueryPackage, en un Vector
-- Parámetros:  P_Owner:    Nombre del Owner
--              P_Tabla:    Nombre de la Tabla
FUNCTION PSpec( P_Owner IN VARCHAR2,
                P_Tabla IN VARCHAR2)
RETURN Genera_Util.T_Vec_Renglon;
--------------------------------------------------------------------------------
-- Retorna el Cuerpo del Body del QueryPackage, en un Vector
-- Parámetros:  P_Owner:    Nombre del Owner
--              P_Tabla:    Nombre de la Tabla
FUNCTION CuerpoBody(P_Owner IN VARCHAR2,
                    P_Tabla IN VARCHAR2)
RETURN Genera_Util.T_Vec_Renglon;
--------------------------------------------------------------------------------
-- Retorna el Package Body del QueryPackage, en un Vector
-- Parámetros:  P_Owner:    Nombre del Owner
--              P_Tabla:    Nombre de la Tabla
FUNCTION PBody( P_Owner IN VARCHAR2,
                P_Tabla IN VARCHAR2)
RETURN Genera_Util.T_Vec_Renglon;
--------------------------------------------------------------------------------
-- Retorna El Encabezado que va en el Spec del Query Package, en un Vector
-- Parámetros:  P_Owner:    Esquema
--              P_Tabla:    Tabla
FUNCTION EncabezaSpec(P_Owner IN VARCHAR2,
                      P_Tabla IN VARCHAR2)
RETURN Genera_Util.T_Vec_Renglon;
--------------------------------------------------------------------------------
-- Retorna El Encabezado que va en el Body del Paquete, en un Vector
-- Parámetros:  P_Owner:    Nombre del Owner
--              P_Tabla:    Nombre de la Tabla
```

```
FUNCTION EncabezaBody(  P_Owner IN VARCHAR2,
                        P_Tabla IN VARCHAR2)
RETURN Genera_Util.T_Vec_Renglon;
-------------------------------------------------------------------------------
FUNCTION OneRow_Header (P_Owner IN VARCHAR2,
                        P_Tabla IN VARCHAR2)
RETURN Genera_Util.T_Vec_Renglon;
-------------------------------------------------------------------------------
-- Retorna el Encabezado de la Función Columna en un Vector
-- Parámetros:  P_Owner:    Esquema
--              P_Tabla:    Tabla
--              P_ColName:  Columna
FUNCTION FColumna_Header (  P_Owner     IN VARCHAR2,
                            P_Tabla     IN VARCHAR2,
                            P_ColName   IN VARCHAR2)
RETURN Genera_Util.T_Vec_Renglon;
-------------------------------------------------------------------------------
-- Retorna el Encabezado de la Función OneRow_CV en un Vector
-- Parámetros:  P_Owner:    Esquema
--              P_Tabla:    Tabla
FUNCTION OneRow_CV_Header ( P_Owner IN VARCHAR2,
                            P_Tabla IN VARCHAR2)
RETURN Genera_Util.T_Vec_Renglon;
-------------------------------------------------------------------------------
-- Retorna el Encabezado de la Función AllRows en un Vector
-- Parámetros:  P_Owner:    Esquema
--              P_Tabla:    Tabla
FUNCTION AllRows_Header (   P_Owner IN VARCHAR2,
                            P_Tabla IN VARCHAR2)
RETURN Genera_Util.T_Vec_Renglon;
-------------------------------------------------------------------------------
-- Retorna el Encabezado de la Función AllRows_CV en un Vector
-- Parámetros:  P_Owner:    Esquema
--              P_Tabla:    Tabla
FUNCTION AllRows_CV_Header (    P_Owner IN VARCHAR2,
                                P_Tabla IN VARCHAR2)
RETURN Genera_Util.T_Vec_Renglon;
-------------------------------------------------------------------------------
-- Retorna el Encabezado de la Función AllRows_By en un Vector
-- Parámetros:  P_Owner:    Esquema
--              P_Tabla:    Tabla
FUNCTION AllRows_By_Header (    P_Owner IN VARCHAR2,
                                P_Tabla IN VARCHAR2)
RETURN Genera_Util.T_Vec_Renglon;
-------------------------------------------------------------------------------
-- Retorna el Encabezado de la Función AllRows_By_CV en un Vector
-- Parámetros:  P_Owner:    Esquema
--              P_Tabla:    Tabla
FUNCTION AllRows_By_CV_Header ( P_Owner IN VARCHAR2,
                                P_Tabla IN VARCHAR2)
RETURN Genera_Util.T_Vec_Renglon;
-------------------------------------------------------------------------------
-- Retorna el Encabezado de la Función Row_Exists en un Vector
-- Parámetros:  P_Owner:    Esquema
--              P_Tabla:    Tabla
FUNCTION Row_Exists_Header (   P_Owner IN VARCHAR2,
                        P_Tabla IN VARCHAR2)
RETURN Genera_Util.T_Vec_Renglon;
-------------------------------------------------------------------------------
-- Retorna el Encabezado de la Función Existe en un Vector
-- Parámetros:  P_Owner:    Esquema
--              P_Tabla:    Tabla
FUNCTION Existe_Header (    P_Owner IN VARCHAR2,
                            P_Tabla IN VARCHAR2)
RETURN Genera_Util.T_Vec_Renglon;
-------------------------------------------------------------------------------
-- Retorna el Encabezado de la Función Cnt en un Vector
-- Parámetros:  P_Owner:    Esquema
```

```
--              P_Tabla:    Tabla
FUNCTION Cnt_Header (   P_Owner IN VARCHAR2,
                        P_Tabla IN VARCHAR2)
RETURN Genera_Util.T_Vec_Renglon;
-------------------------------------------------------------------------------
-- Retorna la Función OneRow en un Vector
-- Parámetros:  P_Owner:    Esquema
--              P_Tabla:    Tabla
FUNCTION OneRow (   P_Owner IN VARCHAR2,
                    P_Tabla IN VARCHAR2)
RETURN Genera_Util.T_Vec_Renglon;
-------------------------------------------------------------------------------
-- Retorna la Función Columna en un Vector
-- Parámetros:  P_Owner:    Esquema
--              P_Tabla:    Tabla
--              P_ColName:  Columna
FUNCTION FColumna ( P_Owner     IN VARCHAR2,
                    P_Tabla     IN VARCHAR2,
                    P_ColName   IN VARCHAR2)
RETURN Genera_Util.T_Vec_Renglon;
-------------------------------------------------------------------------------
-- Retorna la Función OneRow_CV en un Vector
-- Parámetros:  P_Owner:    Esquema
--              P_Tabla:    Tabla
FUNCTION OneRow_CV (    P_Owner IN VARCHAR2,
                        P_Tabla IN VARCHAR2)
RETURN Genera_Util.T_Vec_Renglon;
-------------------------------------------------------------------------------
-- Retorna la Función AllRows en un Vector
-- Parámetros:  P_Owner:    Esquema
--              P_Tabla:    Tabla
FUNCTION AllRows(   P_Owner IN VARCHAR2,
                    P_Tabla IN VARCHAR2)
RETURN Genera_Util.T_Vec_Renglon;
-------------------------------------------------------------------------------
-- Retorna la Función AllRows_CV en un Vector
-- Parámetros:  P_Owner:    Esquema
--              P_Tabla:    Tabla
FUNCTION AllRows_CV (   P_Owner IN VARCHAR2,
                        P_Tabla IN VARCHAR2)
RETURN Genera_Util.T_Vec_Renglon;
-------------------------------------------------------------------------------
-- Retorna la Función AllRows_By en un Vector
-- Parámetros:  P_Owner:    Esquema
--              P_Tabla:    Tabla
FUNCTION AllRows_By (   P_Owner IN VARCHAR2,
                        P_Tabla IN VARCHAR2)
RETURN Genera_Util.T_Vec_Renglon;
-------------------------------------------------------------------------------
-- Retorna la Función Row_Exists en un Vector
-- Parámetros:  P_Owner:    Esquema
--              P_Tabla:    Tabla
FUNCTION Row_Exists (   P_Owner IN VARCHAR2,
                        P_Tabla IN VARCHAR2)
RETURN Genera_Util.T_Vec_Renglon;
-------------------------------------------------------------------------------
-- Retorna la Función AllRows_By_CV en un Vector
-- Parámetros:  P_Owner:    Esquema
--              P_Tabla:    Tabla
FUNCTION AllRows_By_CV (    P_Owner IN VARCHAR2,
                            P_Tabla IN VARCHAR2)
RETURN Genera_Util.T_Vec_Renglon;
-------------------------------------------------------------------------------
-- Retorna la Función Cnt en un Vector
-- Parámetros:  P_Owner:    Esquema
--              P_Tabla:    Tabla
FUNCTION Cnt (  P_Owner IN VARCHAR2,
                P_Tabla IN VARCHAR2)
```

```
RETURN Genera_Util.T_Vec_Renglon;
-------------------------------------------------------------------------------
-- Retorna la Función Existe en un Vector
-- Parámetros:  P_Owner:    Esquema
--              P_Tabla:    Tabla
FUNCTION Existe (   P_Owner IN VARCHAR2,
                    P_Tabla IN VARCHAR2)
RETURN Genera_Util.T_Vec_Renglon;
-------------------------------------------------------------------------------
-- Retorna las Cabeceras de las Funciones <fk> Generadas dinámicamente en un Vector
-- Parámetros:  P_Owner:    Esquema
--              P_Tabla:    Tabla
FUNCTION Foraneas(  P_Owner IN VARCHAR2,
                    P_Tabla IN VARCHAR2,
                    Ind     IN INTEGER)
RETURN Genera_Util.T_Vec_Renglon;
-------------------------------------------------------------------------------
-- Retorna las Cabeceras de las Funciones <fk>_EX Generadas dinámicamente en un Vector
-- Parámetros:  P_Owner:    Esquema
--              P_Tabla:    Tabla
FUNCTION ForaneasBool(  P_Owner IN VARCHAR2,
                        P_Tabla IN VARCHAR2,
                        Ind     IN INTEGER)
RETURN Genera_Util.T_Vec_Renglon;
-------------------------------------------------------------------------------
-- Retorna las Cabeceras de las Funciones <fk>_CV Generadas dinámicamente en un Vector
-- Parámetros:  P_Owner:    Esquema
--              P_Tabla:    Tabla
FUNCTION ForaneasCur(   P_Owner IN VARCHAR2,
                        P_Tabla IN VARCHAR2,
                        Ind     IN INTEGER)
RETURN Genera_Util.T_Vec_Renglon;
-------------------------------------------------------------------------------
-- Retorna las Cabeceras de las Funciones <fk>_Cnt Generadas dinámicamente en un Vector
-- Parámetros:  P_Owner:    Esquema
--              P_Tabla:    Tabla
FUNCTION ForaneasEnt(   P_Owner IN VARCHAR2,
                        P_Tabla IN VARCHAR2,
                        Ind     IN INTEGER)
RETURN Genera_Util.T_Vec_Renglon;
-------------------------------------------------------------------------------

-- FIN DEL SPEC
END Genera_Qp;
```