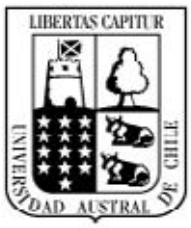


Universidad Austral de Chile

Facultad de Ciencias de la Ingeniería

Escuela de Ingeniería Civil en Informática

"Diseño y desarrollo de prototipo de sistema de traducción instantánea de habla y transmisión en tiempo real, sobre el protocolo RTP utilizando tecnologías de reconocimiento de voz"

Tesis para optar al título de:
Ingeniero Civil en Informática

PROFESOR PATROCINANTE:
Sr. Erick Araya Araya.
Ingeniero Ejecución Electrónico.
Magíster en Ingeniería Electrónica.

PROFESOR COPATROCINANTE:
Sr. Luis Vidal Vidal.
Ingeniero Civil en Informática.

PROFESOR INFORMANTE:
Sra. Gladys Mansilla Gómez.
Ingeniero Matemático.
Analista de Sistemas.
Magíster en Estadística.

Richard Nolberto Rojas Bello
Valdivia - Chile
2005

COMUNICACIÓN INTERNA N° 191/05 REF. CALIFICACIÓN PROYECTO DE TÍTULO.

VALDIVIA, 09 de Noviembre de 2005

DE : ERICK ARAYA A.

A : DIRECTORA ESCUELA INGENIERÍA CIVIL EN INFORMÁTICA

MOTIVO:

INFORME TRABAJO DE TITULACIÓN

Nombre Trabajo de Titulación: DISEÑO Y DESARROLLO DE PROTOTIPO DE SISTEMA DE TRADUCCIÓN INSTANTÁNEA DE HABLA Y TRANSMISIÓN EN TIEMPO REAL, SOBRE EL PROTOCOLO RTP UTILIZANDO TECNOLOGÍAS DE RECONOCIMIENTO DE VOZ

Nombre del Alumno: RICHARD NOLBERTO ROJAS BELLO

Nota: 7,0 (en números) SIETE COMA CERO (en letras)

FUNDAMENTO DE LA NOTA:
Excelente trabajo, donde se abordan, analizan e integran con criterios cuidadosamente escogidos nuevas tecnologías, generando un prototipo de grandes proyecciones.

Considerar : Cumplimiento del objetivo propuesto
Satisfacción de alguna necesidad
Aplicación del método científico
Interpretación de los datos y obtención de conclusiones
Originalidad
Aplicación de criterios de análisis y diseño Perspectivas del trabajo Coherencia y rigurosidad lógica Precisión del lenguaje técnico en la exposición, composición, redacción e ilustración.

Atentamente.

ERICK ARAYA A

Valdivia, 9 de noviembre de 2005.

De : Luis Hernán Vidal Vidal.
A : Sra. Miguelina Vega R.
Directora de Escuela de Ingeniería Civil en Informática.
Ref.: Informe Calificación Trabajo de Titulación.

MOTIVO: Informar revisión y calificación del Proyecto de Título "Diseño y desarrollo de prototipo de sistema de traducción instantánea de habla y transmisión en tiempo real, sobre el protocolo RTP utilizando tecnologías de reconocimiento de voz", presentado por el alumno Richard Nolberto Rojas Bello, que refleja lo siguiente:

Se logró el diseño y desarrollo de un prototipo de traducción instantánea de habla y transmisión en tiempo real sobre el protocolo RTP/RTCP utilizando tecnologías de reconocimiento de voz.

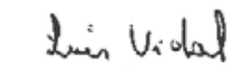
El trabajo es muy interesante desde el punto de vista de I+D.

A continuación se detalla la evaluación de su trabajo de tesis, desde mi perspectiva:

Cumplimiento del objetivo Propuesto.	7,0
Satisfacción de Alguna Necesidad.	7,0
Aplicación del Método Científico.	7,0
Interpretación de los datos y obtención de conclusiones.	7,0
Originalidad.	7,0
Aplicación de criterios de análisis y diseño.	7,0
Perspectivas del trabajo.	7,0
Coherencia y rigurosidad lógica.	7,0
Precisión del lenguaje técnico en la exposición, composición, redacción e ilustración.	7,0
Nota Final.	**7,0**

Por todo lo anterior expuesto califico el trabajo de titulación del señor Richard Nolberto Rojas Bello con nota siete coma cero (7,0).

Sin otro particular, se despide atentamente.



Luis Hernán Vidal Vidal.
Ingeniero Civil en Informática.
Profesor Instituto de Informática.
Universidad Austral de Chile.

VALDIVIA, 10 de noviembre del 2005

DE: GLADYS MANSILLA GÓMEZ
A : DIRECTORA DE ESCUELA INGENIERÍA CIVIL EN INFORMÁTICA

MOTIVO

INFORME TRABAJO DE TITULACIÓN

Nombre Trabajo de Titulación: "DISEÑO Y DESARRROLLO DE PROTOTIPO DE SISTEMA DE TRADUCCIÓN INSTANTÁNEA DE HABLA Y TRANSMISIÓN EN TIEMPO REAL, SOBRE EL PROTOCOLO RTP UTILIZANDO TECNOLOGÍAS DE RECONOCIMIENTO DE VOZ"

Nombre del alumno: SR. RICHARD NOLBERTO ROJAS BELLO.

Nota: 7.0 (en números) siete (en palabras)

Fundamento de la nota:

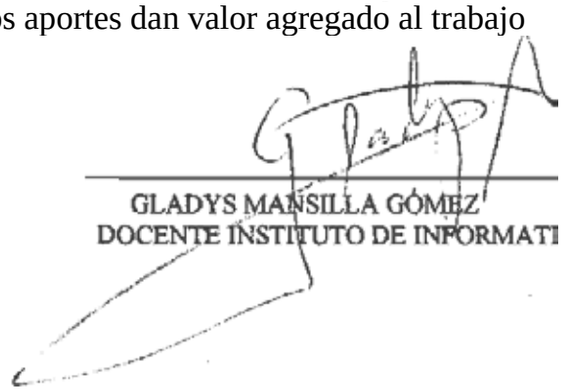
-Este trabajo de tesis constituye una aplicación de mucha utilidad para todos quienes desean comunicarse con anglo-parlantes sin la dificultad que constituye la diferencia idiomática.

- En la realización de este trabajo de titulación se alcanzan plenamente los objetivos planteados al inicio.

- La presentación y redacción del informe están bien elaboradas, abarcando tópicos que inciden directamente en esta tesis y expresado en un lenguaje formal apropiado.

- El alumno ha sabido introducirse en un tema nuevo, estudiarlo e implementar una aplicación.

- También se desataca el hecho que como producto de este trabajo, el alumno haya establecido contacto con académicos extranjeros de alto nivel, especialistas en el área, cuyos aportes dan valor agregado al trabajo



GLADYS MANSILLA GÓMEZ
DOCENTE INSTITUTO DE INFORMATI

AGRADECIMIENTOS

En primer lugar agradezco a Dios por permitirme culminar satisfactoriamente mi Tesis de grado, y por escucharme en mis momentos difíciles a lo largo de mis estudios.

Agradezco a mi mamita Ida, que está con Dios, por dejarme como legado los valores necesarios para forjar mi vida de estudiante y persona, valores que sigo poniendo en práctica y espero hacerlo siempre.

A mi madre Miriam que me enseñó a afrontar con firmeza y fe las decisiones tomadas desde el inicio de mis estudios. A mi papá Nolberto, por regalarme parte de su vida y de su trabajo para poder alcanzar esta meta.

A mis tíos, pilares fundamentales en mi formación profesional. En especial a Hardy y Alejandro quienes siempre se han preocupado de apoyarme al tomar decisiones y emprender nuevos proyectos.

Agradezco también a mi profesora de enseñanza básica Elena Soto, quien me entregó las bases y herramientas para un desempeño exitoso como estudiante.

A mi profesor patrocinante Erick Araya por su motivación y fuerte compromiso con el desarrollo de este proyecto.

Finalmente agradezco a todas las personas que alguna vez me entregaron sinceras palabras de aliento, me tendieron una mano cuando lo necesité, y que de alguna forma también son parte de esto.

ÍNDICE

RESUMEN

El fuerte desarrollo de Internet, la continua optimización de las técnicas de transmisión de voz sobre redes de datos, y el perfeccionamiento de los sistemas reconocedores de voz traen a nuestras vidas realidades que antiguamente eran sólo fantasías y elementos de películas de ciencia ficción.

Las distancias muchas veces ya no son un impedimento para intercambiar información. Sin embargo no todo está solucionado, la diferencia de idiomas es una traba que persiste y sobre la cual se está trabajando fuertemente para encontrar una forma eficiente de eliminarla. Esfuerzos de universidades y empresas privadas han dado significativos resultados en las áreas de transmisión de voz, sobre todo en el área de reconocimiento automático del habla.

El presente documento, expone el desarrollo de un prototipo de sistema de traducción; el cual consiste básicamente en la captura del flujo de voz del emisor integrando tecnologías de reconocimiento de voz avanzadas, traducción instantánea, y comunicación sobre el protocolo Internet RTP (Real time Transport Protocol) para enviar en tiempo real la información al receptor. Este prototipo no transmite imagen, sólo aborda la etapa de audio.

Finalmente, el proyecto además de abarcar un problema de comunicaciones personales, pretende aportar al desarrollo de actividades relacionadas con el reconocimiento de voz, motivando nuevas investigaciones y avances.

SUMMARY

The strong Internet development, the continuous optimization of the techniques of voice transmission over data nets, and the improvement of the voice recognizers systems brings to our lives realities that formerly were only fantasies and elements of science fiction movies.

The distances, many times are no longer an obstacle to interchange information. However not everything is solved, the difference of languages is an obstacle that persists and about which is working strongly to find an efficient form of eliminating it. Efforts of universities and private companies have given significant results in the areas of voice transmission, mainly in the area of automatic recognition of the speech.

The present document exposes the development of a prototype of translation system which consists basically on the capture of the flow of voice of the emitter, integrating advanced technologies of voice recognition, instantaneous translation and communication over the Internet protocol RTP/RTCP (Real time Transport Protocol) to send in real time the information to the receiver. This prototype doesn't transmit image, it only boards the audio stage.

Finally, the project besides embracing a problem of personal communications, tries to contribute to the development of activities related with the voice recognition, motivating new investigations and advances.

CAPÍTULO 1
INTRODUCCIÓN

1.1 Antecedentes generales

Actualmente Internet brinda distintas y eficientes formas de comunicarnos casi instantáneamente y sin importar que tan lejanas se encuentren las personas. La tecnología actual permite el acceso a correos electrónicos, servicios de news[1], servicios de mensajería instantánea (por ejemplo *MSN Messenger*[2]) y aplicaciones para video conferencia. No obstante, en el tema de video conferencia y específicamente en conversaciones por voz, todavía quedan obstáculos que dificultan una plena comunicación; uno de ellos es la diferencia de idiomas o lenguas. Este es el punto sobre el cual se enfoca la solución propuesta en la tesis; solución que aborda el problema integrando tecnologías de reconocimiento y síntesis de voz, junto a tecnologías de transmisión de voz sobre redes IP (VoIP).

El reconocimiento de voz ha evolucionado considerablemente y se presenta como una interfaz confiable y efectiva entre el usuario y un computador. Ya es posible encontrar en el mercado sofisticadas aplicaciones orientadas al uso en oficinas, comercio electrónico, medicina para rehabilitación y telefonía entre otras.

Por otra parte, la transmisión de voz sobre redes IP ya dejó de emplearse solamente en aplicaciones de videoconferencia; hoy en día es un elemento indispensable de comunicación en algunas empresas comerciales, y no sólo por su flexibilidad y

[1] Servicio Internet en el cual los usuarios publican mensajes que quieran compartir con otros subscritos al servicio, como si se tratara de un mural virtual.
[2] Aplicación de mensajería instantánea desarrollada por Microsoft.

confiabilidad, sino que también por los costos reducidos que conlleva. La telefonía IP es una prueba concreta.

1.2 Objetivos a lograr

Los objetivos a lograr se dividen en dos puntos. El primero de ellos es el objetivo general, donde se indica el propósito final de la tesis. En tanto, el segundo punto detalla los ítems necesarios para alcanzar el objetivo general.

1.2.1 Objetivo general

Diseñar y desarrollar un prototipo de sistema de traducción instantánea de habla y transmisión en tiempo real sobre el protocolo RTP/RTCP utilizando tecnologías de reconocimiento de voz.

1.2.2 Objetivos específicos

1. Investigar sobre el estado actual de las técnicas de reconocimiento, síntesis de voz, y herramientas disponibles para desarrollo.
2. Diseñar sistema traductor de lenguaje hablado, empleando reconocimiento y síntesis de voz.
3. Investigar sobre el estado actual de VoIP y herramientas disponibles para desarrollo.
4. Diseñar sistema de transmisión, recepción y reproducción de audio digital utilizando RTP/RTCP.
5. Implementación de prototipo basado en las etapas anteriores.

1.3 Organización de la tesis

Los primeros capítulos acercarán al lector a un conocimiento más acabado sobre la tecnología empleada para construir el sistema. Pasarán por métodos, investigaciones y desarrollos. Finalmente, se explica el funcionamiento y módulos que conforman el prototipo.

Los capítulos del documento son:

Capítulo 2, Fundamentos sobre reconocimiento y síntesis de voz: Descripción de esta tecnología desde sus inicios en 1930, hasta las últimas técnicas y softwares empleados.

Capítulo 3, Fundamentos sobre VoIP: Descripción de los estándares de mayor desarrollo e influencia en el área de transmisión de voz sobre redes IP.

Capítulo 4, Diseño y desarrollo del prototipo: Herramientas, módulos, flujo de información

Capítulo 5, Validación del prototipo: Aporte de las herramientas usadas y pruebas de rendimiento.

Capítulo 6, Conclusiones.

Capítulo 7, Bibliografía: Referencias electrónicas y textos utilizados.

Capítulo 8, Anexos: Detalles sobre tecnologías citadas, fragmentos de código fuente y guía de usuario.

CAPÍTULO 2
FUNDAMENTOS SOBRE RECONOCIMIENTO Y SÍNTESIS DE VOZ

2.1 Antecedentes generales

La capacidad auditiva de un ser humano se caracteriza por percibir audio en un rango de 16Hz a 16Khz [MÖS04], y por diferenciar y comprender fácilmente distintos tipos de fuentes sonoras. Sin embargo, para que las máquinas logren tener esta última habilidad, se está trabajando durante muchos años. Dichos esfuerzos, han culminado en resultados que ya están presentes en el mercado mundial, y que se integran poco a poco al diario vivir.

Básicamente el proceso de reconocimiento de voz se puede explicar en dos pasos.

1. Extracción de fonemas: Los fonemas son unidades lingüísticas, sonidos que al agruparlos forman palabras. Son la unidad fonológica más pequeña en que puede dividirse un conjunto fónico; por ejemplo la palabra /páso/ “paso”, está formada por una serie de cuatro fonemas, ya que el máximo de unidades mínimas en que puede ser dividida es /p/+/a/+/s/+/o/ [QUI68].

   Para extraer los fonemas de la voz de entrada, la señal se analiza espectralmente vía transformadas de Fourier. El espectro de la palabra “AUDIO” se ve en la figura 2.1-1:

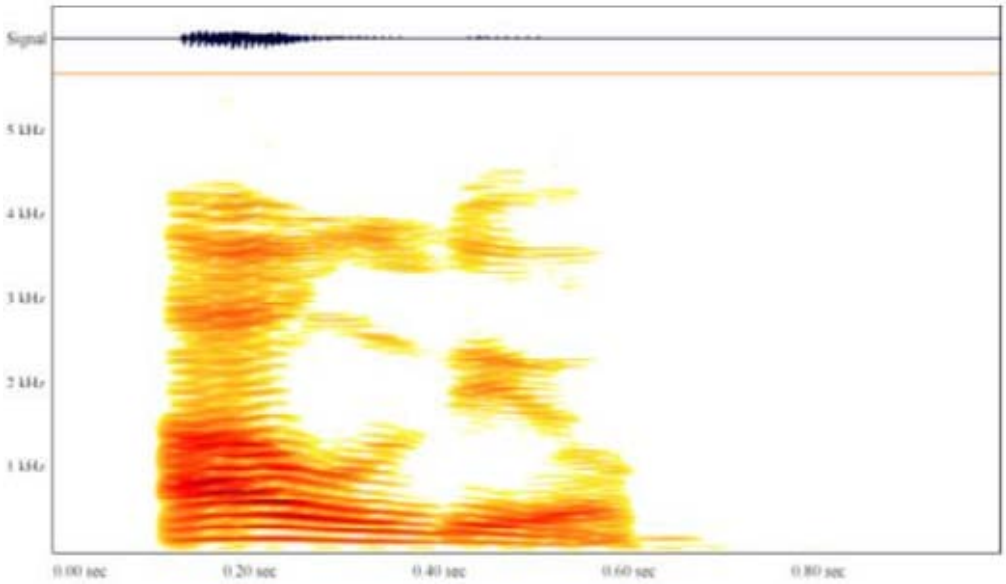


Figura 2.1-1: *Espectro de la palabra "AUDIO".*

2. Conversión de los fonemas en palabras identificables: Este proceso se puede realizar con ayuda de métodos topológicos, probabilísticos y de redes neuronales. Cada uno de ellos se detallarán en el punto 2.4 "Métodos de reconocimiento de voz".

Pero el proceso anteriormente descrito no siempre fue así. Desde hace mucho tiempo se viene trabajando para desarrollar sistemas de reconocimiento de voz 100% confiables.

En el año 1930, el científico húngaro, Tihamér Nemes (fig. 2.1-2) quiso patentar el desarrollo de una máquina de transcripción automática de voz, lamentablemente su iniciativa fue considerada como poco realista y no progresó [KOH88].

Figura 2.1-2: *Tihamér Nemes. Científico (1895 - 1960)* [JÓZ05]

En 1936, Bell Laboratories creó el primer analizador de voz y el primer sintetizador de voz; los llamaron: *Vocoder* y *Voder* respectivamente. Homer Dudley (fig. 2.1-3), su creador, reconoció la naturaleza de la portadora de la voz. Observó que la señal de voz se forma modulando la forma espectral del sonido producido por la fuente vocal. Las fuentes de energía vocal pueden ser periódicas (producidas por vibración de las cuerdas vocales), o aperiódicas (producidas por turbulencias del flujo de aire en una constricción).

Figura 2.1-3: *Homer Dudley. Físico investigador* [BEL00].

Las modulaciones en la forma del espectro de voz, pudieron por lo tanto ser medidas en términos de energía en sucesivos filtros de bandas; y las fuentes periódicas y aperiódicas pudieron representarse por un medidor de frecuencia.

Voder, realiza el proceso inverso. Toma los datos del análisis, los entrega a una serie de filtros excitados por una señal de pulsos periódicos o una fuente de ruido y crea finalmente una señal audible.

Voder era operado por sólo una persona (fig. 2.1-3). Un operador entrenado podía hacer que el sistema "hable" con una pronunciación razonablemente entendible. El sistema tenía la habilidad de asombrar y entretener mientras se demostraban sus principios científicos [BEL00].

Figura 2.1-3: *Sistema Voder* [WOL98].

Luego, en 1952, los investigadores de Bell Laboratories: K. H. Davis, R. Biddulph and S. Balashek, construyeron un sistema dependiente del locutor y capaz de reconocer dígitos de 0 a 9 basándose en las características del espectro de cada número [RES04].

En 1953, Walter Lawrence creó el primer sintetizador de voz basado en frecuencias formantes, el que denominó PAT (*Parametric Artificial Talker*).

En 1956, en los laboratorios RCA, los investigadores Olson y Belar intentaron reconocer 10 sílabas distintas con un método dependiente del locutor. El sistema se basó nuevamente en el análisis espectral.

George Rosen, creó en 1958 el primer sintetizador articulatorio: DAVO (*Dynamic Analog VOcal tract*). DAVO era controlado por una grabación de señales de control hechas manualmente. El mismo año Peterson, Wang y Silverman propusieron el primer sistema de síntesis por concatenación de difonemas; estas unidades fonológicas eran segmentos pregrabados que incluían información parcial de dos fonemas consecutivos.

Más tarde, en 1959, University College London con sus investigadores Fry y Denes, crearon un sistema capaz de reconocer cuatro vocales y nueve consonantes. Emplearon el análisis espectral y comparación de patrones, aunque en realidad el aspecto innovador fue el uso de información estadística, con ello determinaron secuencias posibles de fonemas en inglés [RES04] [AHU99].

Durante la década de 1960 las técnicas de reconocimiento de voz dieron un nuevo paso evolutivo. Los investigadores abordaron el problema empleando vocabularios pequeños, dependientes del locutor y con un flujo discreto[3]. En esta década irrumpió la tecnología digital.

[3] Pausas al terminar palabras y frases.

A principios de los 60 (1962), el físico Lawrence Kersta de los laboratorios Bell, realizó el primer gran paso en la identificación de locutores al introducir el término *Voiceprint* para un espectrograma generado por un complejo dispositivo electro-mecánico [GRA03].

Paralelamente, IBM y Carnegie Mellon University, investigaban en reconocimiento de voz continuo [AHU99].

Los años 70 fueron testigos de esfuerzos por mejorar los sistemas dependientes del locutor con entrada de voz discreta y vocabularios reducidos. ARPA[4] comienza a interesarse en el reconocimiento de voz e inicia sus propias investigaciones. Nacieron técnicas como "Time warping", "Modelado probabilístico" (Aplicación de los Modelos ocultos de Markov), y el "Algoritmo de Retropropagación" [ABR03].

En 1980, los costos reducidos de las aplicaciones, el fuerte comienzo de desarrollo de los PC y el apoyo de ARPA beneficiaron el desarrollo del reconocimiento de habla. Se trabajó en el tamaño del vocabulario (llegando algunos casos hasta 20.000 palabras) y se cambió el enfoque trasladándose desde las técnicas de reconocimiento según patrones a técnicas probabilísticas como son las cadenas de Markov.

Durante los años 90 se siguió trabajando con vocabularios amplios. Los costos siguieron disminuyendo y se hicieron más comunes las aplicaciones independientes del locutor y flujo continuo [AHU99].

[4]Advanced Research Projects Agency. Sección de defensa estadounidense.

Grabar la voz humana para reconocimiento de locutores requiere que una persona emita información hablada, o en otras palabras, muestre cuál es el comportamiento de su voz.

La señal de voz es una función que proviene de la persona que habla y del entorno que la rodea; dicha función puede ser capturada a través de un micrófono o de un teléfono común, y el desempeño aumenta con una mejora de la calidad en los dispositivos. El costo de hardware es bajo ya que la mayoría de los PC incluyen un micrófono o fácilmente se les puede conectar uno. Sin embargo, el reconocimiento de voz presenta problemas con las voces de personas que son demasiado "roncas" o imitan otra voz. Son casos donde el usuario podría no ser reconocido por el sistema. Además, la posibilidad de que el sistema reconozca eficazmente la voz, disminuye si en la ubicación física del hablante existe ruido.

A diferencia del reconocimiento biométrico tradicional - como puede ser el de huella digital - el reconocimiento de voz no es fijo ni estático; en el reconocimiento de voz sólo hay información dependiente del acto.

El estado del arte de la verificación automática del habla (en inglés ASV), es construir un modelo estocástico del hablante basado en sus propias características y extraídas de los entrenamientos realizados.

Bergdata Biometrics GMBH, por ejemplo, diferencia entre información de alto y bajo nivel para el reconocimiento de voz [GRA03]. Dentro de la información de alto nivel se encuentra el dialecto, acento, el estilo de habla y el contexto; características que actualmente sólo son reconocidas por los humanos. La información de bajo nivel contempla ritmo, tono, magnitud espectral, frecuencias y ancho de banda de la voz del

individuo; características que actualmente están siendo usadas en los sistemas de reconocimiento.

Plantean finalmente que el reconocimiento de voz será complementario a las técnicas biométricas. No obstante posee una alta tasa de error, y por consiguiente no es usado para procesos de identificación. Se requieren métodos y modelos adaptativos ya que el habla es variante en el tiempo.

2.2 Ramas del reconocimiento de voz

Las técnicas de reconocimiento de voz se dividen en tres ramas principales [HER01]:

1. Reconocimiento de voz o Reconocimiento del habla: proceso que consiste en convertir un mensaje hablado en texto. Es la rama que más ha crecido en los últimos años.

2. Conversión texto-a-voz: generación de audio que emule la voz humana (síntesis de voz, TTS del inglés *Text-To-Speech*) a partir de información en formato texto digital.

3. Reconocimiento de Locutores: identificación o verificación de la persona que le habla a un sistema; su uso se proyecta como parte de medidas de seguridad.

La codificación de voz, también se postula como una rama del reconocimiento de voz; aunque pudiese considerarse un tema complementario al estar más relacionado con los canales de comunicación y el aprovechamiento del ancho de banda.

En [GON01] se hace una clasificación de las aplicaciones de reconocimiento de voz en tres grupos distintos, que son:

- Aplicaciones locales: aplicaciones desarrolladas para personas con discapacidad motriz, o softwares para trabajo en oficinas.
- Respuesta vocal interactiva: aplicaciones de difusión de información, captura de información, y mensajería vocal.
- Automatización de sistemas telefónicos: marcación vocal, cobro alternativo automático.

Como ya se mencionó anteriormente, otra área importante donde se aplica esta tecnología es el control Biométrico. Estudios descritos en [VIV03] revelan el interés de crear repositorios de datos suficientemente robustos, que sirvan para ajustar sistemas de reconocimiento, incluyendo el de reconocimiento del habla. Detallan técnicas, métodos, dispositivos, resultados y conclusiones.

2.3 Arquitectura Básica de un Sistema de Lenguaje Hablado

El procesamiento de lenguaje hablado se refiere a las tecnologías relacionadas con el reconocimiento de voz, texto-a-voz, y comprensión del lenguaje hablado. Un sistema de lenguaje hablado tiene al menos uno de los siguientes tres subsistemas:

1. Sistema de reconocimiento de habla que convierta el diálogo en palabras.
2. Sistema texto a voz que transmita la información hablada.
3. Sistema de comprensión del lenguaje hablado que traduzca las palabras en

acciones y que planifique acciones del sistema.

El modelo matemático fuente-canal - que habitualmente se usa para formular los problemas de reconocimiento de habla - se muestra en la figura 2.3-1; en él, el locutor decide la secuencia de palabras a decir, **W,** que dirá y entregará a través de su generador de texto. La señal origen se entrega por medio de un canal de comunicación con ruido, que consiste en el sistema vocal del mismo locutor y el módulo de procesamiento de señal de habla del reconocedor. Finalmente el decodificador convierte la señal acústica **X** en una secuencia de palabras $\hat{\mathbf{W}}$, una cercana aproximación de la secuencia original **W**.

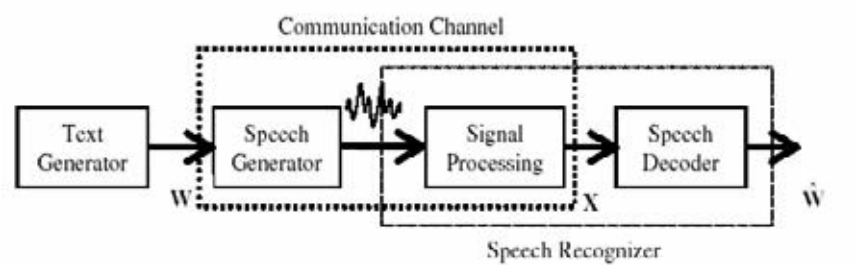


Figura 2.3-1: *Modelo fuente-canal para sistemas de reconocimiento de habla* [XUE01].

Un sistema típico de reconocimiento de habla (fig. 2.3-2) consta de cinco componentes básicos:

1. Modelos acústicos: incluyen representación de conocimiento sobre acústica, fonética, variabilidad del entorno y micrófono, género, y diferencias de dialecto entre los hablantes.
2. Modelos de idioma: conocimiento del sistema acerca de qué puede constituir una posible palabra, qué palabras son probablemente co-ocurrencias y cuál es la secuencia.

3. Módulo de procesamiento de señal: obtiene los vectores característicos para el decodificador.
4. Decodificador: usa los modelos acústicos y de lenguaje para generar la secuencia de palabras que tiene la máxima probabilidad posterior[5] para los vectores característicos de entrada. También puede entregar información al módulo de adaptación.
5. Módulo de adaptación: Modifica los modelos acústicos y de lenguaje para mejorar así el desempeño a obtener.

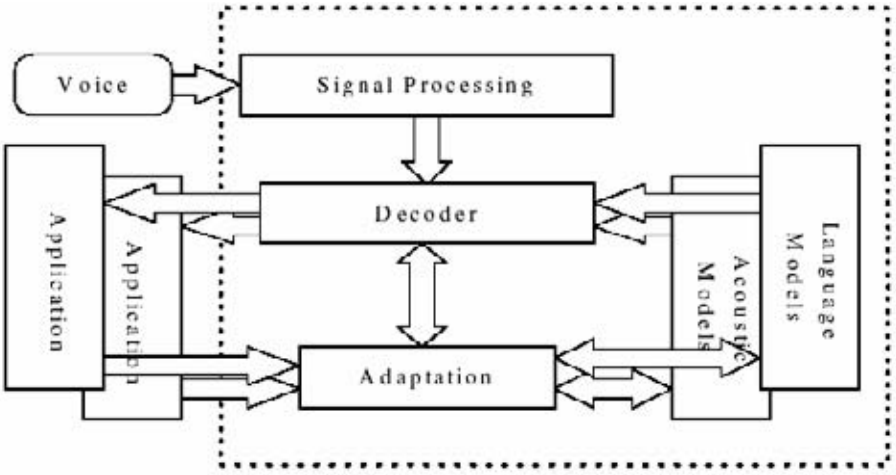


Figura 2.3-2: *Arquitectura básica de un sistema de reconocimiento de habla* [XUE01].

Pero actualmente no solo se plantean sistemas de este tipo, sino que se pretende ir más allá y llegar a completos sistemas de diálogo.

Un modelo genérico que describe estos sistemas es el de la figura 2.3-3.

[5] Dado que se generó el vector característico **X**, que probabilidad es que la palabra siguiente sea $\hat{\mathbf{W}}$.

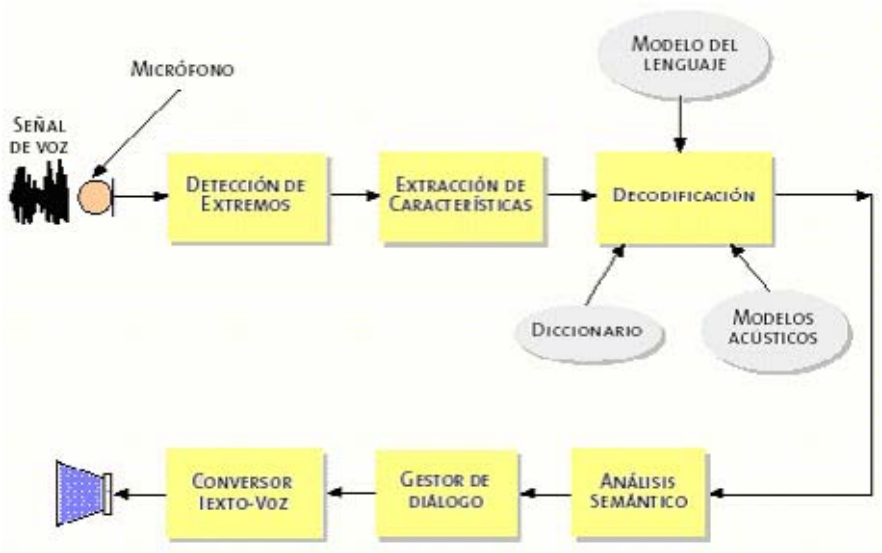


Figura 2.3-3: *Modelo genérico de un sistema de diálogo* [ROD01].

En la figura se aprecia que la señal de voz es capturada por un micrófono y digitalizada, para luego pasar al "Detector de extremos" el cual detectará la presencia de voz y la entregará al próximo bloque.

La "Extracción de características" consiste en dividir la señal en varios segmentos, obteniendo así una representación de las características acústicas más distintivas de cada segmento. A partir de estas características se construyen vectores que son la entrada del siguiente módulo.

En el "Decodificador" se genera la frase reconocida haciendo uso de modelos acústicos, modelos de lenguaje y del "Diccionario".

Una segunda etapa comienza con el módulo "Análisis semántico". Este módulo se encarga de extraer un significado coherente de la frase, su salida la entrega al "Gestor de diálogo" que hará una predicción de la próxima interacción con el usuario en función del estado actual del diálogo.

El último bloque es el "Conversor Texto-Voz". Su función es recibir las frases del "Gestor de diálogo" y convertirlas nuevamente en audio. La figura 2.3-4 describe la arquitectura genérica de un sistema texto a voz.

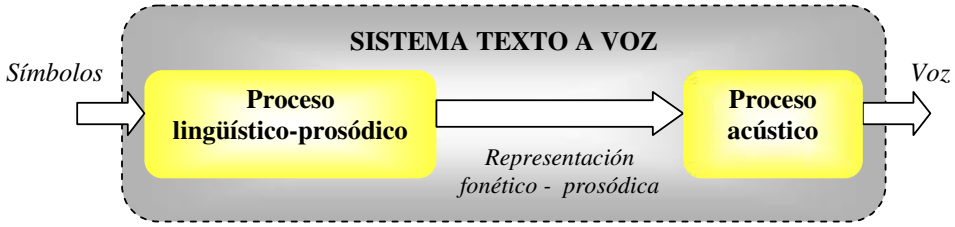


Figura 2.3-4: *Arquitectura de un sistema texto a voz* [HER01].

Los sistemas de texto a voz están compuestos de dos módulos o procesos:

1. Lingüístico-prosódico: determina dos tipos de información necesarios para generar voz en el proceso acústico: información "segmental" e información "suprasegmental". La información segmental está asociada a la cadena de sonidos que componen el mensaje, es decir, a los fonemas. La información suprasegmental, en cambio, está ligada a la prosodia, la cual relaciona elementos lingüísticos (ortografía y gramática) con elementos no lingüísticos (características personales del locutor o estados de ánimo) [HER01].

2. Acústico: convierte los fonemas en sonidos, los concatena considerando la información prosódica, y genera la frase hablada [HER01]. El proceso acústico puede implementarse haciendo uso de alguno de los tres métodos descritos en el capítulo 2.5.

En términos generales, la información necesaria en los sistemas reconocedores

de voz proviene de cuatro fuentes:

- Modelos acústicos: proporcionan información sobre características y propiedades de los sonidos identificados.

- Diccionarios: indican el conjunto de sonidos que forman las palabras del vocabulario.

- Modelos del lenguaje: contienen información sobre cómo deben combinarse las palabras para formar frases.

- Sistemas de diálogo o conversacionales: almacenan predicciones sobre el contenido de la siguiente frase que pronunciará el locutor.

2.4 Métodos de reconocimiento de voz

Tres son los métodos que han marcado la historia del reconocimiento de voz; ellos son: “Alineamiento temporal dinámico”, “Modelos ocultos de Markov”, y “Redes neuronales”. Cada uno de estos métodos se aborda en los puntos siguientes.

2.4.1 Alineamiento temporal dinámico

El concepto de “Alineamiento temporal dinámico” (conocido como DTW, del inglés *Dynamic Time Warping)* se ha empleado para obtener la distorsión o diferencia entre dos palabras. Muchas veces una palabra puede no pronunciarse siempre a la

misma velocidad o bajo las mismas condiciones del ambiente o del mismo locutor, es necesario entonces, ajustarla a un patrón para interpretar correctamente la información.

DTW está basado en la comparación todas las plantillas referencia - resultado de anteriores entrenamientos - contra plantillas compuestas de vectores de parámetros, calculados a partir de los distintos segmentos en que fue dividida la señal de entrada.

Para hacer la comparación se calcula la distancia mínima entre la referencia y la entrada, y finalmente se escoge la plantilla que entregue la menor distancia.

Los reconocedores de habla basados en DTW son fáciles de implementar y muy efectivos para vocabularios pequeños [XUE01].

2.4.2 Modelos ocultos de Markov

Los sistemas basados en cadenas de Markov modelan procesos aleatorios, requiriendo menos memoria que los basados en DTW.

Un "Modelo oculto de Markov" (HMM, del inglés *Hidden Markov Models*) se puede ver como una máquina de estados finitos en la que el estado siguiente depende únicamente del estado actual, y asociado a cada transición entre estados se produce un vector de parámetros.

Un modelo de Markov lleva asociados dos procesos [POZ04]:

- Un proceso oculto (no observable directamente) correspondiente a las transiciones entre estados.
- Un proceso observable (y directamente relacionado con el primero). En él se generan, desde cada estado, vectores de parámetros que forman la plantilla a reconocer.

En reconocimiento de voz, las cadenas de Markov se encargan de ajustar los distintos fonemas captados a fonemas de palabras completas previamente establecidas, adquiridos por entrenamiento. Se asume que la cantidad de estados posibles para cada fonema es finita, por lo tanto el número de estados en la cadena también lo es.

Un tipo de HMM especialmente apropiado para reconocimiento de voz son los modelos "de izquierda a derecha" (fig. 2.4.2-1); modelos en los que una vez abandonado un estado ya no se puede volver a él. Su plantilla se conforma de los vectores que se obtienen en cada uno de los nodos recorridos; cada nodo visitado genera un vector [POZ04].

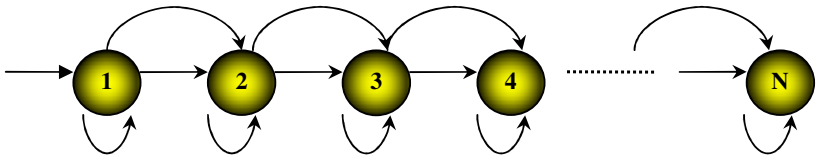


Figura 2.4.2-1: *Modelo de izquierda a derecha simplificado.*

Las cadenas de Markov no sólo son aplicables para extraer fonemas de la señal de voz, sino que también aplican para unir los fonemas y convertirlos en palabras, y luego tomar estas palabras y transformarlas finalmente en frases.

2.4.3 Redes neuronales

El empleo de redes neuronales en el reconocimiento de voz se justifica debido a que estas redes intentan emular complejos procesamientos cerebrales, y uno de ellos es precisamente el reconocimiento del habla. Además, su gran capacidad de resolver problemas que con otros métodos requieren mucha carga para los computadores, como son: el reconocimiento de patrones, evaluación de hipótesis y predicción.

Las redes neuronales organizan sus neuronas en capas (fig. 2.4.3-1). Existe una capa de entrada y una de salida. La capa de entrada procesa directamente los vectores o plantillas, si el resultado de la operación de cada neurona supera un umbral predefinido la neurona realiza sinapsis con sus neuronas post-sinápticas. De esta forma, el resultado de la aplicación de una función de transformación no lineal a la combinación lineal de todos los puntos de la plantilla de entrada se traspasa a las neuronas siguientes.

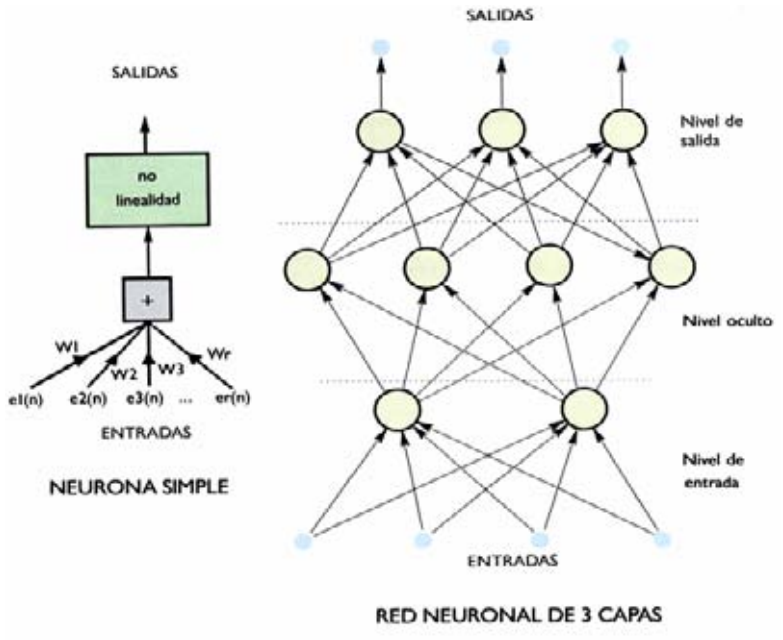


Figura 2.4.3-1: *Capas en una red neuronal* [POZ04].

Así entonces, las redes neuronales pueden inferir sobre cuál es la palabra que viene en la señal de entrada. Sin embargo su eficiencia dependerá del entrenamiento al que haya sido sometida. Un excesivo entrenamiento de la red puede hacer que pierda su efectividad, llevándola a "exigir" demasiado a las plantillas de entrada para ajustarlas a sus modelos internos.

La señal de voz requiere de métodos con capacidad de proceso en dos dimensiones: espacio y tiempo. Las redes neuronales por sí solas sólo tienen capacidad de procesado espacial. Ello nos obliga a combinarlas con técnicas de Programación Dinámica como HMM; consiguiendo con ello modelar la variable tiempo, clasificaciones muy acertadas de las entradas de la red, y segmentación de la señal de entrada [COL01].

2.5 Métodos de Síntesis de voz

La síntesis de voz puede clasificarse dentro de tres tipos, según el modelo usado para generar la voz. Dichos modelos son: síntesis por formantes, síntesis articulatoria y síntesis por concatenación.

2.5.1 Sintetizadores por formantes

Las formantes son las resonancias características de cada articulador del tracto vocal (fig. 2.5.1-1) [XUE01]. Determinan el timbre particular de cada vocal y definen las características individuales de las voces. Cada palabra emitida puede definirse en términos de las frecuencias formantes propias de cada individuo.

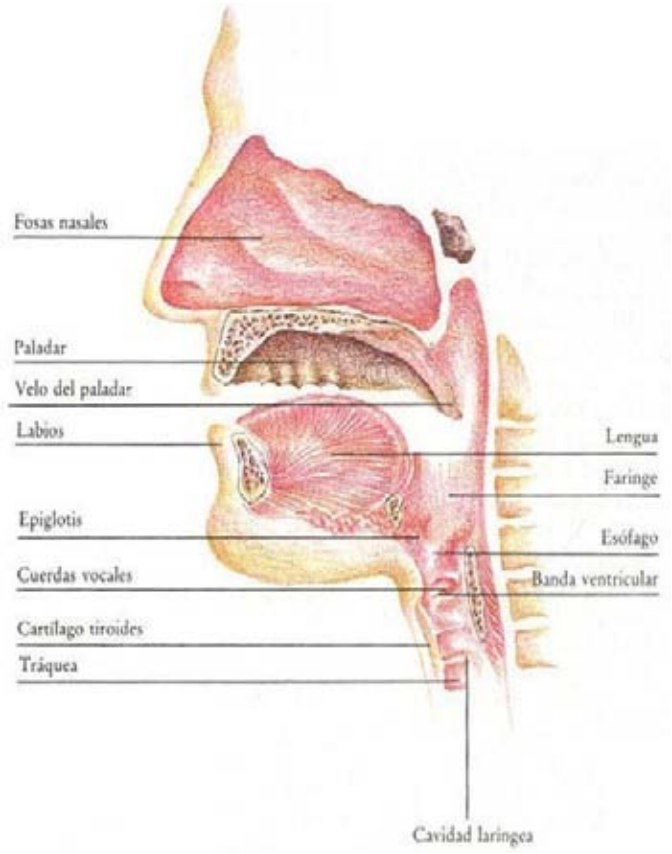


Figura 2.5.1-1: *Órganos de generación de sonido* [ENC96]

Basándose en estos principios, los sintetizadores por formantes, modelan la resonancia del tracto vocal aplicando filtros para generar cada formante.

Los filtros son ajustables y poseen parámetros definibles mediante reglas. Éstas indican cómo modificar los parámetros entre un sonido y otro sin perder la continuidad presente en los sistemas de generación de voz físicos [XUE01].

Los sintetizadores por formantes involucran un procedimiento manipulable y flexible, son capaces de generar diversas voces modificando parámetros de sus filtros. Sin embargo, en la síntesis automática se necesita un número enorme de reglas, lo que

requiere compiladores cada vez más sofisticados, capaces de integrar todo el conocimiento que se adquiere a base de experimentar con el sistema [HER01].

2.5.2 Sintetizadores articulatorios

Los sintetizadores articulatorios usan un modelo físico de la producción de la voz, simulando la propagación de las ondas acústicas [HER01]. Emplean parámetros - definidos por reglas, al igual que los sintetizadores por formantes - que modelan los movimientos mecánicos del aparato fonador, y de las distribuciones resultantes de volumen y presión de aire en pulmones, laringe, tracto vocal y nasal [XUE01].

Los parámetros pueden obtenerse desde la voz real a través de rayos X y resonancias magnéticas, aunque posicionar los sensores en tracto vocal altera la forma en que se produce el habla e impide completamente sonidos naturales.

Esta técnica de síntesis no es capaz de generar voz con una calidad comparable a la síntesis por formantes y concatenación [XUE01] e implica altos costos en investigación sobre el sistema humano de generación de voz.

2.5.3 Sintetizadores por concatenación

El estado del arte de los sintetizadores por reglas ofrece sonidos inteligibles y poco naturales. La dificultad para poder mejorarlos está en cómo capturar todos los matices del habla natural en un pequeño conjunto de reglas deducidas manualmente.

En la síntesis por concatenación, un segmento de voz se sintetiza simplemente reproduciendo la onda sonora con el fonema respectivo. Un discurso completo se sintetiza entonces concatenando varios fragmentos de voz [XUE01].

Al contrario de los sintetizadores citados anteriormente, no necesita reglas ni ajustes manuales. Cada segmento es completamente natural, con lo que se puede esperar una salida del mismo tipo [XUE01].

El problema que se presenta con los sintetizadores por concatenación es que si se unen dos segmentos de voz no adyacentes entre sí, pueden generarse discontinuidades espectrales o prosódicas. Las discontinuidades espectrales ocurren cuando las formantes no coinciden en el punto de concatenación; las discontinuidades prosódicas, cuando los tonos no coinciden en el punto de concatenación. La discontinuidad puede llevar a clasificar un sistema de síntesis como deficiente, aunque esté basado en segmentos totalmente naturales.

Para enfrentar el problema de la discontinuidad se definieron “unidades”, que son representaciones abstractas de segmentos de voz. Las unidades pueden ir desde fonemas hasta sentencias completas. Mientras mayor tamaño tenga la unidad, mejor será la calidad de la síntesis, pero la cantidad de unidades a almacenar podría extenderse ilimitadamente. El gráfico G2.5.3-1 es un ejemplo que expresa el número de unidades necesarias para abarcar una cantidad N de los apellidos ingleses más frecuentes en EEUU.

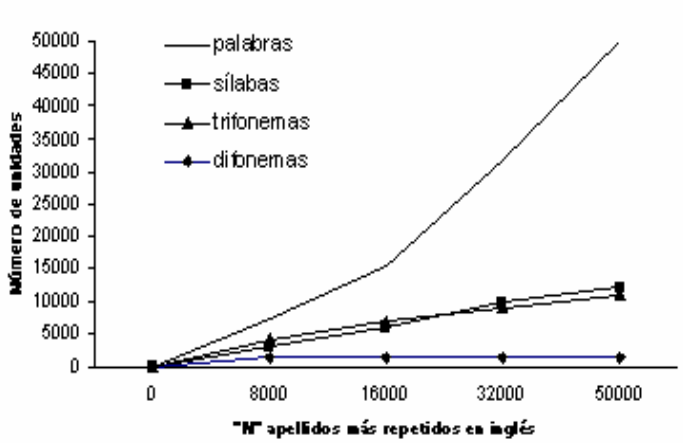


Gráfico G2.5.3-1: *Unidades requeridas, de diferentes tipos, para formar N apellidos* [XUE01] (Adaptación).

Lo que se plantea actualmente es emplear híbridos que combinen unidades grandes (como palabras) con otras más pequeñas (fonemas), manteniendo el realismo y dando flexibilidad a la pronunciación.

2.6 Investigación y desarrollo

El 23 de diciembre de 1999, Science Daily, hizo pública la información de que científicos de Carnegie Mellon University y sus colegas del C-STAR[6] (Consortium for Speech Translation Advanced Research) conducirían una video conferencia internacional. Probaron, en tal evento, un sistema web de planificación de viajes. El sistema empleó traducción habla a habla, interpretando 6 idiomas distintos en 6 ubicaciones distintas al rededor del mundo. La demostración fue exitosa; sin embargo, se encontraron con problemas propios del habla espontánea: interrupciones, vacilaciones y tartamudeos [SCI99].

[6] http://www.c-star.org

El software para comunicarse y hacer referencias a documentos web fue *JANUS*[7].

JANUS, se desarrolló en Carnegie Mellon University y Universität Karlsruhe a finales de los años ochenta y principios de los noventa en colaboración con ATR (Japón) y Siemens AG (Alemania) [QUA00].

El primer prototipo, *JANUS-I*, tradujo de inglés a japonés y alemán [OST92]. Procesaba sólo el habla sintácticamente correcta (lectura) sobre 500 palabras. Su sucesor, *JANUS-II*, funcionó con diálogos interpersonales, espontáneos, en dominios limitados, y con terminologías de alrededor de 3000 palabras.

Alex Waibel[8] comenta en una publicación reciente, que el éxito de *JANUS* se ha extendido hasta alcanzar su nueva versión *JANUS-III*, qué ahora maneja diálogos hablados espontáneos, de conversación, y con un vocabulario abierto en varios dominios de discurso [WAI97].

JANUS consta de tres componentes principales: reconocimiento de voz, una máquina de traducción, y un módulo de síntesis de voz.

[7] http://www.is.cs.cmu.edu/mie/janus.html
[8] Doctor, principal investigador de la escuela de ciencias de la computacion de Carnegie Mellon University y profesor de Universität Karlsruhe, Alemania. Uno de los fundadores de C-STAR.

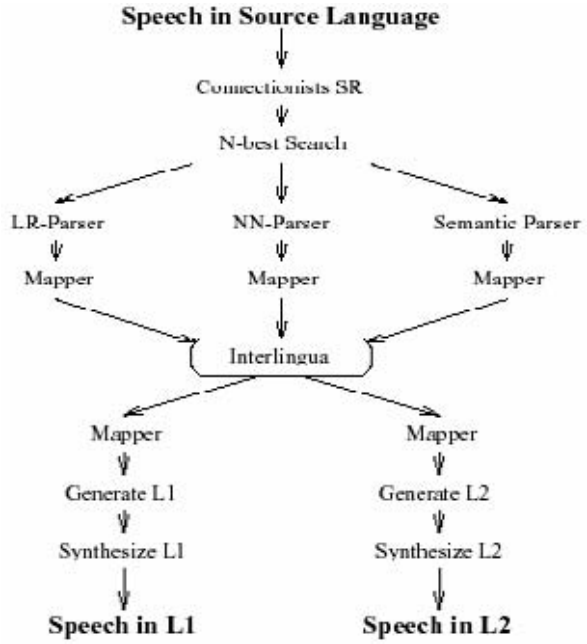


Figura 2.6-1: *Arquitectura del sistema JANUS* [WOZ96]

Luego de la captura de la señal de voz, *JANUS* aplica el algoritmo "N-Best" [9] (fig. 2.6-1), lo que reduce el tiempo de ejecución y la lista de hipótesis que se ajustan a la frase original.

Tres procesos paralelos se encargan de dar una correcta interpretación: "LR-Parser" (ajuste sintáctico), "Semantic Parser" (ajuste semántico) y "NN-Parser" (ajuste del árbol de salida).

Un módulo muy importante dentro de *JANUS* es "Interlingua". Este módulo, se encarga de elaborar una representación del significado independiente del lenguaje y para los distintos idiomas destino.

Finalmente el resultado de *Interlingua* se entrega a los generadores y sintetizadores de los lenguajes destino [WOZ96].

[9] Algoritmo que otorga un número "n" de hipótesis que se ajustan a la frase hablada.

Un proyecto similar, donde también forma parte Carnegie Mellon University, es *NESPOLE*[10]. *NESPOLE* es un sistema aplicado directamente al mundo del comercio electrónico; permite a usuarios de Internet visitar sitios web y conectarse transparentemente con un agente humano de la empresa proveedora del servicio [CAT02]. De esta forma se elimina la restricción de que el agente no comprenda el idioma del cliente, lo que en la práctica es una pérdida de ingresos directa.

NESPOLE es una colaboración de:

- 3 grupos de investigación europea:
  - IRST[11], Italia.
  - ISL[12]de la Universität Karlsruhe (TH), Alemania.
  - CLIPS[13] de la Université Joseph Fourier, Francia.
- 1 grupo de EEUU:
  - ISL de Carnegie Mellon University.
- 2 socios comerciales:
  - APT[14], Italia.
  - AETHRA[15], Italia.

Actualmente soporta inglés, francés, alemán e italiano y se usa como complemento de *Microsoft NetMeeting*®[16].

[10] http://nespole.itc.it/. NEgotiating through SPOken Language in E-commerce.
[11] http://www.itc.it/irst/. Centro per la ricerca scientifica e tecnologica.
[12] http://www.is.cs.cmu.edu/js/. Interactive Systems Laboratorios.
[13] http://www-clips.imag.fr/. Communication Langagière et Interaction Personne-Système.
[14] http://www.apt.trento.it/. Trentino provincial tourism borrad.
[15] http://www.aethra.it/. Empresa de telecomunicaciones.
[16] Aplicación para videoconferencia y aplicaciones compartidas desarrollada por Microsoft.

El 13 de septiembre de 2004, Carnegie Mellon University publicó en su sitio (www.cmu.edu) que investigadores de su centro están trabajando en la creación de un chip silicio capaz de reconocer habla. El profesor de ingeniería eléctrica y computación de la misma universidad, Rob A. Rutenbar, trabajando en conjunto con investigadores de University of California de Berkeley, recibieron US$1.000.000 por parte de la National Science Foundation[17] de EEUU. El objetivo, trasladar el reconocimiento automático del habla desde software hasta hardware.

La meta es crear una nueva y eficiente arquitectura de chip de silicio que solamente reconozca habla, pero que lo haga de 100 a 1.000 veces más eficiente que una computadora convencional integrándose primero en celulares y PDAs. Según Rutenbar la razón por la cual fueron escogidos para recibir el premio, fue que ellos apuntan el resultado de su trabajo hacia la seguridad de su nación, y plantea la siguiente situación: "*El personal que asiste a una emergencia necesita consultar una base de datos en línea solo con la voz, en un ambiente con ruido y peligroso y sin regresar a su vehículo. Las posibilidades son infinitas*" [CAR04].

Pero en lo que más se puede ver el avance de este tipo de tecnología es en las aplicaciones que están ya en el mercado, a la venta como un producto o como parte de un servicio público.

El reconocimiento de habla se proyecta como ayuda a personas con capacidades físicas distintas – consideradas muchas veces como discapacitadas – y que tienen dificultades motrices o intelectuales. Las tecnologías de reconocimiento de voz ayudarían a hacerles más fácil su relación con el entorno a través de aplicaciones de

[17] http://www.nsf.gov/

domótica[18] que les permitan controlar persianas, luces y electrodomésticos; o aplicaciones de PC tales como software de dictado de habla continua, de enseñanza de lectura, correcciones de sintaxis para personas con problemas de dislexia; e incluso artículos de oficina: fax, fotocopiadoras, impresoras.

IBM, creador del sistema *WebSphere Voice Response* – destinado para centros de asistencia a clientes, asistencia en caminos y reservaciones en hoteles y restaurantes – adaptó su producto para que sea usado en automóviles como parte de un proyecto de ATX. El proyecto consistió en desarrollar un sistema asistido por GPSs, capaz de dar información completa acerca del tráfico vial más próximo, y atender cada una de las consultas de voz hechas por el conductor. No existiría la necesidad de dejar de atender el objetivo principal: conducir con el máximo de atención en el camino y sin desviar la mirada hacia focos distractores. El producto final adoptó el nombre de *WebSphere Voice Server* y se ha implementado en vehículos de Mercedes Benz, BMW, Jaguar y Lincoln Mercury [IBM02].

A modo de síntesis, el reconocimiento de voz – apoyado por una fuerte base matemática y análisis de la estructura y comportamiento de la voz – es una técnica comúnmente usada en:

- Aplicaciones con fines pedagógicos.
- Técnicas de ayuda a personas con problemas físicos.
- Medidas de seguridad por control biométrico[19].
- Aplicaciones ofimáticas.

[18] Técnica de integración y aplicación de elementos electrónicos e informáticos con el hogar.
[19] Herramientas que autentican a los usuarios en base a características biológicas suyas como: olor corporal, estructura facial, huellas dactilares, patrones de retina o iris, trazado de las venas y voz.

- Interfaces telefónicas cliente-central.

Por otra parte, la síntesis de voz además de ser un complemento en las soluciones anteriormente descritas se ha incorporado en:

- Sistemas IVR (*Interactive Voice Request*).
- Portales de voz.
- Dispositivos convertidores de divisas.
- PDAs intérpretes de lenguas.
- Traductores de textos.
- Aplicaciones de asistencia médica.

2.7 Bibliotecas disponibles

En los puntos que siguen, se describen las principales características de una serie de herramientas para el desarrollo de aplicaciones de reconocimiento y síntesis de voz.

2.7.1 AT&T Natural Voices TTS Engine 1.4[20]

AT&T Natural Voices TTS Engine es un motor de texto a voz que incluye un kit para desarrollo de aplicaciones. Sus principales características son:

- Retorno de voz sintetizada a 8Khz ley-μ[21]/ley-A[22], y archivos WAV en respuesta al texto de entrada.

[20] http://www.naturalvoices.att.com
[21] Sistema de codificación de voz usado en gran parte del mundo, excepto en América del Norte y Japón.

- SNMP[23].
- VoiceXML[24] y SSML[25].
- Velocidad, volumen y tonos de voz modificables.
- Diccionarios de palabras múltiples.
- Diccionarios modificables.
- Soporte para inglés británico, inglés norteamericano, español latinoamericano, alemán y francés. Todos con una voz masculina y femenina.
- Compatible con SAPI 4, SAPI 5.1 y *Java Speech API*.
- Desarrollo sobre Java y C++.
- Microsoft Windows 98, ME, NT, 2000 y XP.
- Computadores mono y multiprocesador.

AT&T Natural Voices TTS Engine está disponible en el mercado en dos versiones: Server Edition y Desktop Edition.

2.7.2 Fonix DECtalk 4.6.3

Fonix Dectalk es un sintetizador TTS. Incorpora un modelo del tracto vocal para simular voces que se asemejen lo más posible a la voz real.

Considera entre sus características:

- Diccionarios de pronunciación programables.
- Control de prosodia, velocidad, tonos y pausas.

[22] Sistema de codificación de voz usado en América del Norte y Japón.
[23] Simple Network Management Protocol. Protocolo simple de administración de redes.
[24] Voice eXtensible Markup Language. Lenguaje de voz de marcas extensibles.
[25] Speech Synthesis Markup Language. Lenguaje de marcas de síntesis de voz.

- Cuatro voces masculinas, cuatro femeninas y una de niño.
- Idiomas inglés estadounidense y británico, francés europeo, castellano y español latinoamericano.
- Soporte para SAPI 3,4 y 5.
- Soporte para Visual Basic y C.
- Soporte para Microsoft Windows en sus versiones CE, 98, 2000 y XP; además de soporte para Linux[26], Solaris, MAC OS, Palm OS y Symbian.

2.7.3 Fonix VoiceIn 4.0.1[27]

El reconocedor de voz automático *Fonix VoiceIn* fue desarrollado en base a redes neuronales, y a diferencia de otros sistemas no requiere entrenamiento por parte del usuario.

Entre sus características destacan:

- Soporte para caracteres Unicode.
- Tolerante a ruido y a ambientes diversos.
- Independiente del locutor.
- Gramáticas dinámicas[28].
- Soporte para Visual Basic, C# y Java.
- Soporte para Microsoft Windows en sus versiones CE, 98, 2000 y XP; .además de Linux, Solaris, MAC OS, Palm OS y Symbian.

[26] Sistema operativo de código abierto desarrollado bajo licencia GNU (General Public License).
[27] http://www.fonix.com
[28] Gramáticas modificables por la aplicación.

2.7.4 Lernout & Hauspie TTS3000

TTS3000 es un motor TTS gratuito; fue la última entrega de los laboratorios Lernout & Hauspie antes de pasar a formar parte de ScanSoft[29]. Sus características son:

- Soporte para ingles norteamericano, alemán, francés, español, italiano y coreano.
- Control de volumen, tono y velocidad de la voz.
- Soporte de múltiples idiomas durante la reproducción.
- Retorno de voz sintetizada PCM a 8 bits ley-A, 8 bits ley-μ y 16 bits codificación lineal, y frecuencia de muestreo de 20Khz.
- Soporte para Microsoft Windows 95, NT y posteriores.

2.7.5 Microsoft Speech Application Program Interface (SAPI) 5.1

Microsoft SAPI es un conjunto de herramientas para desarrollo de aplicaciones que reconozcan y sinteticen voz. Es una capa entre el hardware (captura y reproducción) y los motores de reconocimiento y síntesis instalados en el sistema operativo.

Estas bibliotecas, gratuitas y descargables directamente desde el sitio de Microsoft, están implementadas en C++. Sus objetos permiten crear aplicaciones de voz dependientes del locutor con una interpretación correcta de hasta un 95% según el entrenamiento [MIC03].

[29] http://www.scansoft.com/

Microsoft SAPI 4.0 no incluye por defecto modos de reconocimiento y síntesis de voz en español.

La arquitectura SAPI está compuesta de 8 componentes básicos que se detallan en el ANEXO I.

SAPI incluye tres tipos de gramáticas:

1. Gramáticas libres de contexto: tienen reglas que predicen las próximas palabras que se hablarán.
2. Gramáticas de dictado: definen un contexto para el locutor considerando el tema en cuestión, idioma y dictados satisfactorios anteriores.
3. Gramáticas de dominio limitado: no proveen una estructura sintáctica rígida, pero si provee una amplia gama de palabras disponibles.

Las principales características de SAPI son:

- Tiempo de proceso de reconocimiento ajustable, esto se denomina como *Complete-Phrase Value*.
- Niveles de aceptación y rechazo ajustables.
- Síntesis de voz para los idiomas inglés norteamericano, y español.
- Generación de archivos WAV de 16 bits a 22Khz monoaural.
- Velocidad, volumen y tono de reproducción configurables.
- Soporte XML, con lo que se logra reutilizar componentes, crear listas de frases, y ajustar un correcto sentido semántico a los distintos niveles de la gramática.

- Soporte Unicode.
- Compatible con Microsoft Visual C++ y Microsoft Visual Basic.
- Dependiente del locutor.
- Reconocimiento de voz para los idiomas inglés norteamericano, chino y japonés.
- Compatible con Microsoft Windows 95, 98, NT, 2000, XP y 2003.

2.7.6 Nuance Speech Recognition System 8.5

Nuance [NUA01] es un sistema de reconocimiento de lenguaje natural. Está destinado para el desarrollo de sistemas en empresas comerciales y de telecomunicaciones.

Nuance se emplea en:

- Aplicaciones para Call center.
- Portales de voz.
- Aplicaciones para transacciones comerciales.

Sus características son:

- Arquitectura basada en cliente-servidor (ver ANEXO II).
- Reconocimiento independiente del locutor con un 97% de efectividad.
- Verificación biométrica del hablante.
- Gramáticas dinámicas.
- Procesamiento N-Best.

- Asignación de probabilidades (según el criterio del usuario) a cada frase o palabra.
- Mecanismo de reducción de ruido.
- Funciones telefónicas básicas[30]:
    - Ubicación del teléfono destino.
    - Contestación automática.
    - Detección de cuelgue.
    - DTMF[31].
    - Desvío de llamada.
    - Soporte SNMP.
    - Registro de información relevante para una llamada.

Nuance soporta 28 idiomas y trabaja sobre los sistemas operativos: Microsoft Windows 2000, Sun SPARC Solaris 2.8, X86 Solaris 2.8, IBM-AIX.

2.7.7 Nuance Vocalizer 4.0

Nuance Vocalizer es un sistema TTS orientado hacia asistencias telefónicas, marcación por voz, rescate de información bancaria y financiera, y lectura de correo electrónico.

Vocalizer incorpora soporte para:

- Voice XML y SSML.
- Modificación de prosodia, velocidad, volumen y tono.

[30] Empleando tarjetas telefónicas Dialogic, Natural MicroSystems, o Aculab.
[31] Dual Tone Multifrecuency. Multifrecuencia de tono dual.

- 2 idiomas en una misma aplicación y 18 individualmente.

2.7.8 Philips Speech SDK 4.0[32]

Las bibliotecas *Philips Speech SDK* están diseñadas para aplicaciones computacionales tradicionales y de telefonía.

Speech SDK destaca entre sus adelantos:

- Arquitectura de almacenamiento de datos abierta. El desarrollador puede indicar de donde debe extraer datos el motor de reconocimiento, desde archivos o bases de datos SQL.
- Aprendizaje automático en tiempo de ejecución.
- Soporte para Voice XML.
- Soporte para C, C++ y .NET.
- Compresión de audio.
- Manejador de titubeos de usuario o problemas de puntuación o formatos de texto.
- Soporte para 23 idiomas.

2.7.9 Sakrament ASR Engine 1.0[33]

Sakrament ASR Engine es un motor de reconocimiento de voz creado por la compañía Sakrament.

[32] http://www.speechrecognition.philips.com
[33] http://www.sakrament-speech.com/products/asr/asr2/

Entre sus características destacan:

- Independencia del locutor, por lo que no necesita entrenamiento.
- Independencia del idioma. Es posible desarrollar directamente el soporte para una lengua.
- Gramática abierta. No rechaza palabras desconocidas.
- Tamaño de la gramática no limitado.
- Compatible con SAPI 4, SAPI 5 y TAPI 3[34].
- Soporte para desarrollo en: C, C++, Delphi y Java.
- Compatibilidad con Microsoft Windows 98, ME, NT, 2000 y XP.

2.7.10 Sakrament Text-to-Speech Engine 2.0[35]

Sakrament TTS Engine es también es producto de la compañía Sakrament. Es un motor de texto a voz cuyas características principales son:

- Salida de flujo a dispositivos multimedia, archivos, buffer de memoria, línea telefónica.
- Audio en formato PCM 22Khz 16 bits.
- Entonación, volumen y velocidad del texto leído modificables.
- Idiomas modificables mediante etiquetas XML.
- Idiomas ruso, bielorruso, ucraniano e inglés.
- Compatible con SAPI 5.
- Soporte para desarrollo en: C, C++, Visual Basic, Delphi, Java.

[34] *Telephony Application Programming Interfaces 3*. Interfaces para programación de aplicaciones que soporten telefonía tradicional y telefonía IP. Desarrolladas por Microsoft.
[35] http://www.sakrament-speech.com/products/tts/tts2/

- Compatibilidad con Microsoft Windows 98, ME, 2000 y XP.

2.7.11 Dragon NaturallySpeaking 8 Preferred Edition Spanish[36]

NaturallySpeaking 8 Preferred Edition Spanish es una herramienta, desarrollada por Scansoft® para reconocimiento y síntesis de voz. Se integra a las aplicaciones de los sistemas operativos Microsoft Windows XP y Microsoft Windows 2000. No es un kit de desarrollo, pero sus motores son compatibles con Microsoft SAPI.

Sus características principales son:

- Reconocimiento con 99% de precisión.
- Reconocimiento de 160 palabras/minuto.
- Creación de comandos personalizados.
- Vocabulario expansible de 300.000 palabras.
- Edición y formato de cualquier texto por comandos de voz.
- Compatible con sistemas Pocket PC.
- Inicio y control de cualquier aplicación Microsoft Windows como Microsoft Word, Microsoft Excel y Microsoft Internet Explorer.
- Soporte para reconocimiento en inglés y español.

[36] http://www.scansoft.com/naturallyspeaking/preferred/

CAPÍTULO 3
FUNDAMENTOS SOBRE VoIP

3.1 Antecedentes Generales

La tecnología de VoIP consiste básicamente en la transmisión de voz sobre redes IP.

Se originó a partir de distintos factores que entre sí se potencian. El crecimiento de Internet y el desarrollo de métodos de compresión de voz, transmisión en tiempo real, y principalmente la necesidad de estar siempre comunicados son la base y antesala a la VoIP.

El creciente desempeño del protocolo IP y de las redes Ethernet, y la administración del ancho de banda, permiten aplicaciones como distribución automática de llamada, trabajo a distancia y mensajería instantánea; aplicaciones que se apoyan en estándares de constante evolución.

Dentro de los estándares más empleados para establecer sesiones multimedia se encuentran H.323 [URL1] y SIP [URL17]. Ambos marcan una fuerte presencia en Internet. Sus características son descritas a partir del punto 3.2.

3.2 El estándar H.323

H.323 fue publicado en 1996 por la ITU-T[37] y representa las bases para comunicación de datos, voz y video vía IP, basadas en redes LAN[38] e Internet. H.323 hace referencia a otros estándares de ITU-T, como por ejemplo los de señalización H.225 [URL2], H.245 [URL3] y Q.931 [URL4] [WYN01].

3.2.1 Topología

La topología de H.323 comprende cuatro componentes principales y especifica su funcionalidad obligada y opcional.

Terminales, gateways, gatekeepers y unidades de control multipunto (UCM) (fig. 3.2.1-1). Una zona H.323 consta de muchos de estos componentes dentro de un segmento LAN, y varios segmentos pueden unirse por medio de routers. Por cada zona H.323 debe haber un gatekeeper. Los demás componentes pueden existir en cantidad aleatoria [WYN01].

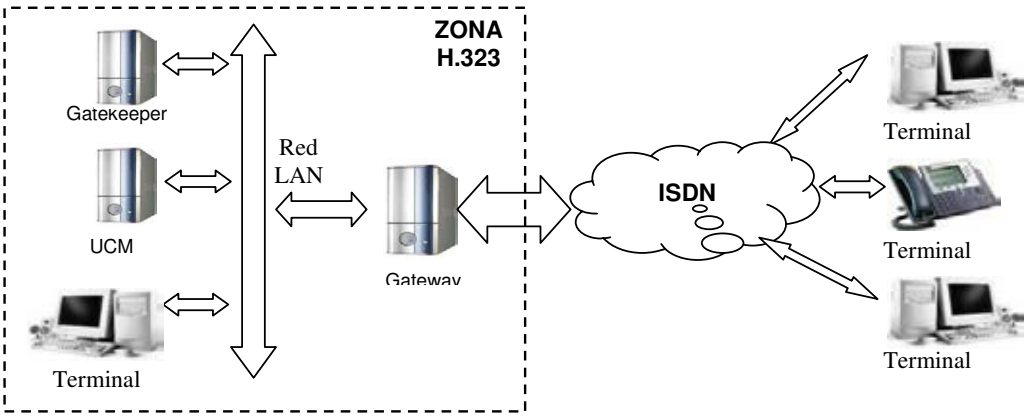


Figura 3.2.1-1: *Componentes en una zona H.323.*

[37] International Telecommunication Union - Telecom Standardization Sector.
[38] *Local Area Network*. Red de area local.

1. Terminales: representan los puntos finales de cada conexión H.323 y pueden implementarse en hardware o software. Dentro de un segmento LAN los terminales pueden intercomunicarse directamente. Para conexiones en otras redes o en otras LAN se necesita un gateway.

2. Gateways: se utilizan para establecer conexión con una red telefónica o vía PBX[39]. Su función es convertir distintos formatos de datos en transporte, control de proceso y proceso de audio y video. La comunicación de los gateways con los terminales es por H.245 y H.225.

3. Gatekeepers: tienen funciones de control y administración dentro de una zona H.323 y sus servicios son usados por los terminales. Se permite solo un gatekeeper por zona.

   Las dos tareas principales de un gatekeeper son la conversión de direcciones y administración del ancho de banda.

   El gatekeeper también puede jugar un papel en la recepción y asignación de ruta de canales H.245 para conexiones entre dos usuarios. Si la conferencia se extiende a tres o más usuarios, el gatekeeper dirige el flujo H.245 a una unidad de control multipunto que toma la tarea de controlar la conferencia.

4. Unidades de control multipunto (UCM): se emplean en conferencias con más de dos usuarios; ello asegura que las conexiones están configuradas y liberadas adecuadamente; el audio y video son mezclados, y los datos son distribuidos entre la conferencia. Un UCM consiste en un controlador multipunto y procesadores

[39] *Private Branch eXchange*. Sistema telefónico empleado en organizaciones privadas para administrar llamadas internas y externas.

multipunto. El controlador multipunto cubre H.245, negocia funciones generales para procesamiento de audio y video, y controla los recursos determinando qué flujos de datos serán transmitidos por los procesadores multipunto.

Los procesadores multipunto reciben flujos de datos de los participantes de una conferencia, los procesan y los distribuyen a los terminales.

3.2.2 Procesamiento de audio

La transmisión de audio en los terminales H.323 se realiza por el codec G.711 [URL5], el cual es una recomendación de codificación PCM. G.711 se diseñó originalmente para las redes ISDN[40] con tasas de transmisión estables y con un rendimiento de 64 kbit/s. Aunque es factible en la mayoría los entornos LAN, G.711 no puede emplearse en enlaces de bajo ancho de banda; por lo tanto, ITU-T especificó G.723 [URL6], codec capaz de comprimir voz a 5.3 y 6.3 Kbit/s [WYN01].

3.2.3 Procesamiento de video

La transmisión de video es una función opcional de los terminales H.323. Puede ser controlada por los estándares multimedia ITU-T H.261 [URL7] y H.263 [URL8].

H.261 provee una tasa de transmisión de *n x 64 kBit/s (n=1,2,..,30)* y puede por lo tanto usar varios canales ISDN. Además usa *intra* e *inter frame*[41] similar a MPEG[42] [URL26].

---

[40] *Integrated Services Digital Network*. Red telefónica digital de servicios integrados, diseñada para voz, datos y señalización.

[41] Intraframe: tratamiento de cada fotograma como una foto independiente.
Interframe: extracción y almacenamiento de las diferencias entre dos frames consecutivos.

El estándar más reciente es H.263, tiene una tasa de transmisión de 64 Kbit/seg, y es compatible con H.261 (interactuando vía QCIF[43]) [WYN01]. Posee mejor calidad de imagen como resultado del algoritmo de *Estimación de Movimiento de 1/2 Píxel* [CHE98] y *Frames Predecidos*. Es además apropiado para tasas de transmisión más bajas.

3.2.4 Conferencias de datos

Para una transmisión de datos entre terminales, el estándar H.323 hace referencia al estándar ITU-T T.120 [URL9] que puede usarse para variadas aplicaciones en el campo del trabajo colaborativo, tales como pizarras, aplicaciones compartidas y administración de documentos; como lo hace *Microsoft NetMeeting® y MSN Messenger*.

T.120 es independiente del sistema operativo y del protocolo de transporte, y es soportado por más de 100 compañías, entre ellas Apple, AT&T, British Telecom, Cisco Systems, Intel, MCI, PictureTel y Microsoft [DAT98].

Características de T.120 son:

- Conferencias de datos multipunto.
- Transmisión con corrección de errores y reconocimiento de recepción, control de secuencias seguras de paquetes en la estación receptora.

[42] *Moving Picture Experts Group.* Grupo de trabajo en representación digital de audio y video. Miembro de ISO/IEC (International organization for standaritation / International Electrotechnical Commission).
[43] *Quarter Common Intermediate Format*. Formato estándar de vídeo que ofrece un tamaño de imagen de 176 x 144 píxeles.

- Independencia de redes, sean (ISDN, CSDN[44], LAN, etc.).
- Interoperabilidad e independencia de la plataforma.
- Soporte de topologías heterogéneas.

3.3 El estándar SIP

SIP (*Session Initiation Protocol*) fue aceptado y presentado oficialmente como un estándar IETF en 1999.

SIP es un protocolo de control de la capa de aplicación en el modelo ISO/OSI. Fue diseñado para iniciar, modificar y terminar sesiones multimedia con uno o más participantes. Las sesiones pueden ser: llamadas telefónicas sobre Internet, distribución de contenido multimedia y video conferencias [URL17].

La forma en que interactúan dos dispositivos SIP es través de mensajes de señalización. Estos cumplen los propósitos básicos [DAN02] de:

1. Registrar un usuario y sistema.
2. Invitar a unirse a una sesión.
3. Negociar términos y condiciones de una sesión.
4. Establecer un stream[45] entre dos o más puntos finales.
5. Finalizar una sesión.

[44] *Circuit Switched Data Network*. Red de circuitos conmutados.
[45] Flujo multimedia proveniente del servidor que contiene el recurso. La entrega de audio o video a través del flujo se denomina *streaming*.

3.3.1 Topología

La topología de SIP involucra dos componentes principales: "agentes de usuario" (User Agents, UA) y servidores.

Agente de usuario: Representa un sistema final; contiene un agente de usuario cliente (UAC) que genera peticiones, y un agente de usuario servidor (UAS) que responde a ellas. Los UAS pueden implementarse en hardware como teléfonos IP fijos y *gateways*, o en software como los *softphones*[46]. Independiente de los protocolos que se empleen para transportar contenido multimedia, SIP puede trabajar junto a ellos gracias los UA.

Servidores: SIP contempla cuatro tipos de servidores (fig. 3.3.1-1):

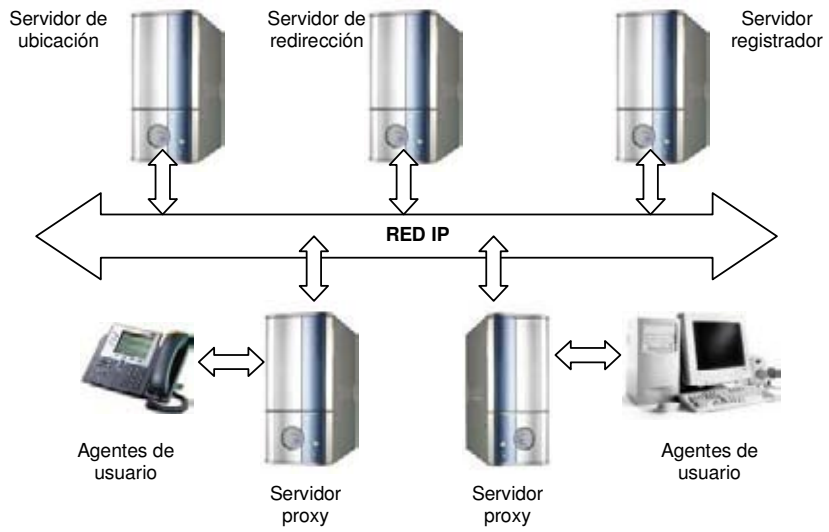


Figura 3.3.1-1: *Servidores y clientes SIP.*

1. Servidor de ubicación: maneja información sobre las posibles ubicaciones del terminal que debe recibir la llamada.

[46] Software diseñado para operar como teléfono IP en un computador personal.

2. Servidor proxy: recibe y resuelve peticiones de clientes. Si no puede responder, hace nuevas peticiones hacia otros servidores en nombre de los clientes.

3. Servidor de redirección: retorna a los clientes la dirección del servidor que desea contactar. No acepta llamadas como sucede con un proxy, pero puede generar respuestas que indican el UAC para ubicar la entidad SIP.

4. Servidor registrador: acepta peticiones de tipo REGISTER [ANEXO III]. Maneja información acerca de los terminales que se están comunicando una sesión. Habitualmente se ubica junto a servidores proxy y de redirección.

3.3.2 Sesiones en SIP

Una sesión se define como un conjunto de entidades que envían datos, entidades que reciben datos, y streams fluyendo entre ellos [URL17].

La figura 3.3.2-1 muestra un diagrama de secuencia de una sesión SIP. Aquí el usuario del teléfono IP A llama al usuario del teléfono IP B.

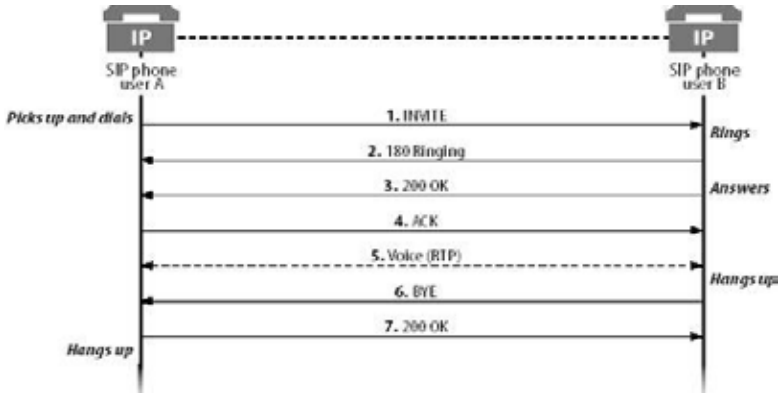


Figura 3.3.2-1: *Llamada básica con protocolo SIP* [DAN02].

Los mensajes de la secuencia son:

1. INVITE: el usuario descuelga el teléfono A y marca el teléfono B. Se envía el mensaje INVITE que contiene un SDP[47] [URL18] que indica dónde el teléfono A recibirá el flujo RTP y cuáles son los codecs ofrecidos para establecer el flujo.

2. 180 Ringing: el teléfono B recibe el mensaje, activa su timbre, y envía el mensaje de respuesta 180 Ringing al teléfono A. Este mensaje contiene un SDP que indica cuál de los codecs ofrecidos fue usado y dónde el teléfono B recibirá el flujo RTP.

3. 200 OK: el usuario del teléfono B contesta el teléfono. El teléfono B envía un mensaje 200 OK al teléfono A.

4. ACK: el teléfono A recibe los mensajes 180 Ringing y 200 OK y responde con un mensaje ACK.

5. Voice (RTP): en cuanto el usuario del teléfono B contesta se establece un canal de voz entre ambos teléfonos. Para crear el canal se usa el protocolo RTP.

6. BYE: el usuario del teléfono B cuelga. El teléfono B envía una petición BYE al teléfono A.

7. 200 OK: el teléfono A recibe la petición y responde con un mensaje 200 OK. La llamada termina y el canal de voz se cierra. Luego, el usuario cuelga.

[47] *Session Description Protocol*. Usado para describir los componentes del canal de comunicación que están bajo negociación.

Una lista con mayor información sobre las peticiones y respuestas se pueden ver en el ANEXO III.

SIP no especifica los codecs que han de emplearse para transmitir información de audio o video; no obstante es frecuente encontrar aplicaciones con soporte SIP que integran codecs tales como G.711, G.723, G.729 [URL10] y GSM[48].

Finalmente se puede decir que SIP, además de ser más liviano que H.323, es un protocolo más extensible dado que al estar basado en texto, como los protocolos HTTP y SMTP, hace más sencilla la tarea de añadir nuevas funciones y controlar las ya existentes.

3.4 VoIP y Telefonía

El ámbito en el cual la tecnología de VoIP ha causado gran impacto y empuje es el de la Telefonía IP. Este tipo de telefonía permite llamadas desde PC a PC, PC a teléfono IP o tradicional, o viceversa; y por último desde teléfono IP a teléfono IP.

Los costos de una llamada IP son inferiores a los de una llamada tradicional, esto pasa por la cantidad de recursos físicos que se requieren para efectuar cada una de ellas. En una llamada telefónica normal se establece una conexión permanente entre los participantes de la llamada, la cual se hace a través de una centralita[49]. En cambio en una llamada telefónica por IP, la voz se envía comprimida y digitalizada en paquetes

[48] *Global System for Mobile communication*. Sistema global de comunicaciones móviles.
[49] Central a la que llegan líneas telefónicas para conectarse al resto de la red.

sucesivos al usuario remoto, donde serán descomprimidos y nuevamente convertidos a señal de voz.

La telefonía celular no está ajena a la llegada de VoIP. Ericsson - una de las compañías más grandes en el rubro de las comunicaciones - propone en una publicación del año 2000 [ERI00] el modelo que representa lo que debería alcanzar la investigación y desarrollo de VoIP sin hilos; es decir, VoIP inalámbrica (VoIPoW).

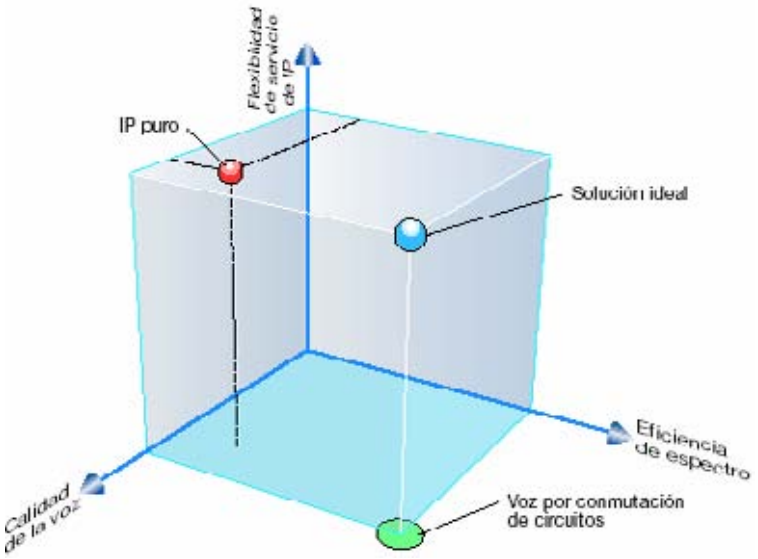


Figura 3.4-1: *Estado y expectativas de VoIP* [ERI00]

Señalan en su publicación que todos los sistemas oferentes de servicios de voz se han optimizado en un espacio bidimensional, donde los ejes X e Y representan la calidad de voz y la eficiencia de espectro respectivamente. Destacan además que la principal ventaja de utilizar el IP por todo el interfaz aéreo es la flexibilidad de servicio, lo que añade un tercer parámetro que es la demanda de flexibilidad de servicio IP. Para servicios de VoIP sin hilos el reto principal, y en el cual se trabaja, es conseguir calidad de la voz y eficiencia de espectro [ERI00].

3.5 Los protocolos RTP y RTCP

Para transmisión de datos en tiempo real como audio o video se introdujeron en SIP y H.323 mecanismos adicionales para garantizar una comunicación exitosa. En H.323 el protocolo H.225 hace referencia al protocolo de transmisión RTP estandarizado por la IETF.

RTP al ser un protocolo de tiempo real, realiza sus operaciones manteniendo un comportamiento temporal estricto. Privilegia que las acciones se realicen intervalos de tiempo fijos, en lugar de ofrecer un desempeño a la velocidad más rápida.

RTP puede usarse para "media-on-demand"[50] y para servicios interactivos. Su estructura básica está definida por dos partes: una relacionada con los datos a transmitir (sincronización, detección de pérdidas, seguridad e identificación de contenido) y otra dedicada a las funciones de control (identificación de fuente, soporte para gateways como puentes de audio y video, traductores de multicast a unicast).

RTP es independiente del protocolo de transporte; aunque fue desarrollado con el objetivo de residir sobre UDP, esfuerzos adicionales lograron usarlo sobre protocolos como IPX[51] y CLNP[52], y experimentalmente sobre AAL5/ATM[53].

[50] Servicio asíncrono de entrega de información. La información se transmite en el momento del pedido.
[51] *Internetworking Packet eXchange*. Protocolo del sistema operativo NetWare usado para transferir datos desde servidores a clientes.
[52] *Connection-Less Network Protocol*. Protocolo ISO que proporciona un servicio de datagramas.
[53] Capa 5 de adaptación, de ATM. Verificación de paquetes.

3.5.1 RTP

RTP se implementa sobre UDP[54], no aportando fiabilidad adicional ni reservas de recursos u otras garantías. Incluye información sobre los orígenes del tráfico y marcas de tiempo específicas para la sincronización de cada medio transportado [GRU01].

La cabecera de un paquete RTP tiene la siguiente estructura:

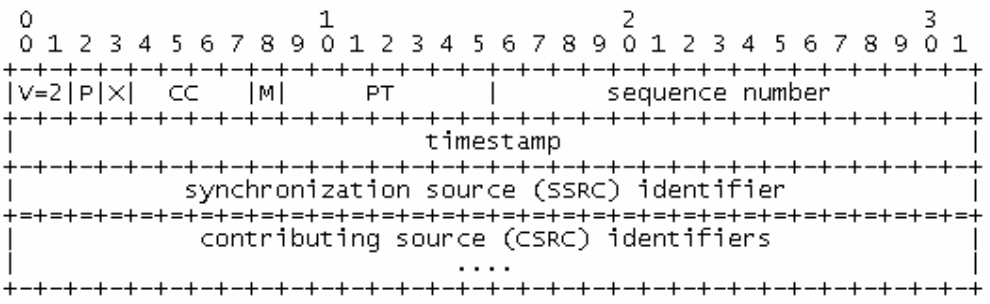


Figura 3.5.1-1: *Estructura de cabecera de un paquete RTP* [URL19]

Los doce primeros octetos están presentes en todos los paquetes RTP, mientras que la lista de los identificadores está presente solo cuando se inserta un "mezclador". El significado de cada campo se detalla en el ANEXO IV.

Entre emisores y receptores puede haber 2 tipos de nodos: mezcladores y receptores [URL19].

Mezclador

Recibe varios paquetes RTP, los combina, y envía otro nuevo.

[54] *User Datagram Protocol*. Protocolo de datagrama de usuario.

No siempre es conveniente que todos los participantes de una sesión reciban los datos en el mismo formato. Por ejemplo, en el caso donde alguno de los participantes pertenece a un área con bajo ancho de banda y los demás participantes son de un área con un ancho de banda privilegiado. En lugar de forzar a todos a usar una baja calidad en la comunicación se utiliza un "*mixer*" o mezclador, el que:

1. Resincroniza los paquetes entrantes para reconstruir la constante de 20ms (milisegundos) de espacio generado por el emisor.
2. Mezcla el audio en un solo flujo.
3. Recodifica el audio para enviarlo por bajos ancho de banda. Lo transmite a sus participantes en un paquete con un nuevo SSRC.

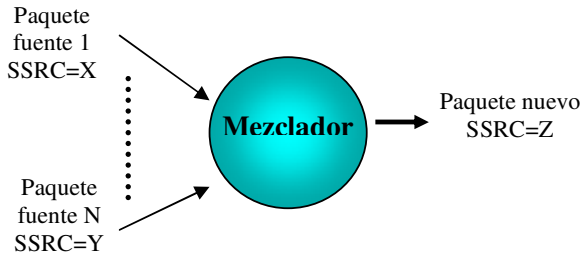


Figura 3.5.1-2: *Mezclador de paquetes RTP*

Las identificaciones de las fuentes que contribuyen al paquete mixto se registran en el campo CSRC, a fin de que los receptores puedan tener referencia correcta de los emisores (fig. 3.5.1-2).

Traductor

Al igual que el mezclador, es un sistema intermedio. Ejemplos de traductor son:

- Conversores de codificación.
- Replicadores de multicast a unicast.
- Filtros a nivel de aplicación en cortafuegos.

El traductor reenvía paquetes tras modificarlos pero sin cambiar su identificador SSRC (fig. 3.5.1-3). Posibilita que los receptores identifiquen fuentes individualmente, aunque todos los paquetes pasen a través del mismo traductor y lleven la dirección fuente de la red del traductor.

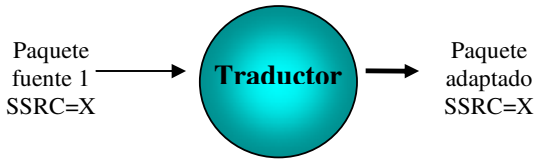


Figura 3.5.1-3: *Traductor de paquetes RTP*

Los receptores no pueden detectar la presencia de un traductor, a menos que conozcan por algún otro recurso el tipo de carga o dirección de transporte utilizada por la fuente original.

Algunos traductores pasan los datos sin tocarlos, pero otros pueden cambiar su codificación, y en consecuencia el tipo de dato que va como carga y su "timestamp" [ANEXO IV]. Si varios paquetes se recodifican en uno solo, o viceversa, el traductor debe asignar nuevos números de secuencia a los paquetes que van saliendo.

Pero RTP solo se encarga de encapsular tráfico en tiempo real. El protocolo de reserva y garantía de calidad de servicio a determinados flujos se conoce como RTCP (*Real Time Control Protocol*). Cada participante envía un paquete RTCP para que se sepa quienes están escuchando [SAL02].

3.5.2 RTCP

RTCP se basa en la transmisión periódica de paquetes de control a todos los participantes de una sesión, y usando el mismo mecanismo de distribución que los paquetes de datos.

RTCP tiene cuatro funciones [URL19]:

1. Proveer una realimentación de la calidad de la distribución de datos como función primaria.

2. Llevar un identificador de nivel de transporte para una fuente RTP, CNAME (*Canonical Name*). Es importante para la sincronización de datos de los participantes de una sesión.

3. Las dos primeras funciones requieren que todos los participantes envíen paquetes RTCP, por lo tanto la tasa debe ser controlada para mantener la escalabilidad de un gran número de participantes. Una vez que cada participante haya enviado sus paquetes de control para los restantes, cada uno puede observar independientemente el número de participantes. Este número es usado para calcular la tasa en que los paquetes deben ser enviados.

4. Transportar información mínima de control de la sesión; por ejemplo, la identificación de los participantes para desplegarla en la interfaz de usuario. Esta función es opcional, pero puede ser útil en sesiones de muy bajo control, donde los participantes entran y salen sin el control de los miembros o parámetros de negociación.

Las funciones uno, dos y tres son obligatorias cuando se usa RTP en entornos IP multicast y se recomiendan para los demás.

Se definen distintos tipos de paquetes RTCP que transportan variada información de control.

Los tipos pueden ser:

- SR (*Sender Report*): para estadísticas de transmisión y recepción de los participantes que son emisores activos.
- RR (*Receiver Report*): para estadísticas de recepción de los participantes que no son emisores activos.
- SDES (*Source Description Items*): ítems de descripción de fuente, incluye CNAME.
- BYE: Indica el fin de la participación en una sesión.
- APP: funciones específicas de la aplicación.

Cada paquete RTCP comienza con una parte fija, similar a la de los paquetes de datos RTP. Le siguen elementos estructurados que pueden ser de largo variable dependiendo del tipo de paquete, pero siempre con un límite de 32 bits. Múltiples paquetes RTCP pueden ser concatenados para formar un nuevo paquete compuesto RTCP, que se envía por un protocolo de capa baja como UDP.

Los paquetes que conforman al paquete compuesto pueden procesarse independientemente sin requerimientos de orden o combinación de los mismos.

3.5.3 Investigación y desarrollo en RTP

Entre las agrupaciones dedicadas al estudio del protocolo RTP, la de mayor relevancia es el Departamento de Ciencias de la computación de Columbia University. Este departamento, entre otras actividades, investiga y recopila información sobre avances y desarrollos de nuevas aplicaciones RTP. En su sitio web [SHU03] presenta el estado actual de las comunicaciones RTP/RTCP, esfuerzos que han concluido en aplicaciones software, publicaciones y bibliotecas para desarrollo VoIP entre otros.

En el ámbito de las bibliotecas al servicio de RTP, Columbia University presenta en su sitio a 10 de ellas, que si bien no necesariamente son las únicas, son las más connotadas en el mundo Internet.

En el capítulo siguiente se da una reseña acerca de las bibliotecas disponibles en la red para desarrollar aplicaciones.

3.5.4 Bibliotecas disponibles

En los puntos que continúan, se detalla una serie de herramientas para desarrollo de aplicaciones VoIP.

3.5.4.1 Common Multimedia Library 1.2.14[55]

Common Multimedia Library (*CML*) es un desarrollo del Grupo de investigación de redes multimedia de University College London cuyos principales esfuerzos están destinados a las áreas de:

- Audio en redes de trabajo.
- Coordinación y control de conferencias multicast.
- Video bajo demanda y servidores multimedia.
- Codificación y enrutamiento de streams.
- Infraestructura y sistemas de conferencia seguros.
- Calidad en audio y video.
- Evaluación de conferencias multicast.
- IPv6[56].

CML soporta los protocolos: IPv6, RTP, MBUS[57], SAP[58] [URL25] y SDP; y los sistemas operativos Linux y Microsoft Windows.

3.5.4.2 GNU ccRTP 1.0.2[59]

GNU ccRTP son bibliotecas c++ bajo licencia LGPL[60]. Proveen interfaces de programación para RTP y RTCP.

[55] http://www-mice.cs.ucl.ac.uk/multimedia/overview/
[56] Nueva versión de 128 bits del Protocolo Internet.
[57] Protocolo empleado en teléfonos móviles de Nokia.
[58] Protocolo de anuncio de sesión.
[59] http://www.gnu.org/software/ccrtp/
[60] Licencia GNU para el uso de bibliotecas en programas propietarios.

GNU ccRTP soporta:

- Datos de audio y video.
- Unicast, multicast.
- Múltiple sincronización de fuentes.
- Múltiples sesiones RTP (espacios SSRC).
- Múltiples aplicaciones RTP (espacios CNAME).
- Sistemas operativos Linux y Microsoft Windows.

Y sus funciones principales son:

- Mezclar distintos tipos de carga en el stream.
- Filtrar paquetes RTP y RTCP.
- Implementar aplicaciones que cumplan con las especificaciones RFC2833 [URL24].

Soporta los codecs:

- Audio: G.711 ley-μ, G.726 [URL15], GSM, G.723, IMA[61], LPC[62], MPEG, G.728 [URL12], DVI, 6.729, NV[63].
- Video: JPEG[64] [URL16], H.261, MPEG, MPEG-2 [URL27], H.263.

[61] http://www.apple.com
[62] *Linear Predictive Coding*. Codificación por predicción lineal.
[63] Códec empleado en MBone.
[64] *Join Photographic Experts Group*. Estándar de compresión de imágenes.

3.5.4.3 Java Media Framework 2.1.1[65]

Java Media Framework (JMF), desarrollado por Sun Microsystems, es un entorno de trabajo para el desarrollo de aplicaciones java, las cuales permiten presentar recursos multimedia.

Las APIs RTP de *JMF* hacen posible:

- Abrir sesiones RTP.
- Recibir y enviar streams y crear reproductores *JMF*.
- Recibir estadísticas de streams y monitorear sesiones usando información de los paquetes RTCP.
- Transmitir distintos tipos de carga RTP durante una sesión.
- Crear sesiones unicast, multicast y broadcast.
- Mantener independencia del protocolo de red.

JMF está disponible para los sistemas operativos: Microsoft Windows, Solaris y Linux.

Los formatos soportados por *JMF* se pueden ver en el ANEXO V.

[65] http://java.sun.com/products/java-media/jmf/index.jsp

3.5.4.4 JVOLIB 1.2.0[66]

JVOLIB son bibliotecas VoIP de código abierto con licencia LGPL, orientadas a objetos e implementadas en C++. Son el resultado de una tesis universitaria sobre RTP y del trabajo cooperativo entre Limburgs Universitair Centrum y Universiteit Maastricht.

Sus características más destacas son:

- Sesiones configurables antes y durante la conexión; se puede modificar parámetros como la frecuencia de muestreo y el tipo de compresión.
- Escalabilidad. Al ser bibliotecas orientadas a objeto se facilita la adición de nuevos componentes.
- Soporte para efectos de sonido 3D.
- Compresión DPCM[67], codificación por ley-μ, compresión GSM, compresión LPC de 5.4 Kbps.
- Un mezclador de voz.
- Soporta Linux y Microsoft Windows.

3.5.4.5 LIVE.COM Streaming Media 2003.11.25[68]

Es un conjunto de bibliotecas C++ para streaming multimedia. Con *LIVE.COM* se pueden construir clientes y servidores que procesen:

[66] http://research.edm.luc.ac.be/jori/jvoiplib/jvoiplib.html
[67] *Differential pulse code modulation*. Modulación por código de pulso diferencial.
[68] http://www.live.com/liveMedia

- Codecs MPEG, MPEG-2, MPEG-4 [URL28], H.261, H.263.
- Audio en formato MP3 [URL26] y audio PCM.

Soporta los protocolos:

- RTP/RTCP.
- RTSP (*Real Time Streaming Protocol*) [URL20].
- SIP.

Son compatibles con los sistemas Unix, Microsoft Windows y QNX[69].

El código de *LIVE.COM Streaming Media* es abierto, con licencia LGPL.

3.5.4.6 RADvision RTP/RTCP[70]

RADVISION RTP/RTCP es un conjunto de herramientas, desarrolladas por la empresa de comunicaciones IP RADVISION. Están escritas en ANSI C++ con API's orientadas a objetos e independientes de la plataforma.

Responde a los siguientes estándares:

- IETF RFC1889 [URL19].
- IETF RFC1890 [URL21].
- IETF RFC2032 [URL22].
- IETF RFC2190 [URL23].

[69] http://www.qnx.com. Sistema operativo de tiempo real.
[70] http://www.radvision.com/corporate/radvision.

- IETF RFC2833.
- ITU-T H.235 [URL11].

Y sus características más importantes son soporte de:

- IPv4 e IPv6.
- Perfiles de seguridad.
- Algoritmos de encriptación: DES[71], 3DES[72] y AES[73].
- Unicast, multicast y broadcast.
- Operaciones bloqueantes y no bloqueantes.
- Codecs de voz: G.711 ley-A/ley-μ, G.722.x [URL13], G.723, G.728, G.729.
- Codecs de video: H.261, H.263, MPEG-4, H.264 [URL14], GSM-AMR [URL29].
- Traductores y mezcladores RTP.
- Monitores RTCP.
- Detecciones de loop y colisiones SSRC.
- Ejecución de hilos segura (los hilos no interfieren en ejecuciones que no le corresponden y no modifican los datos de otros).
- IP TOS[74]/Diffserv[75]

Sistemas operativos soportados por *RADVISION*:

[71] Algoritmo de encriptación en bloques de 64 bits empleando una clave de 56 bits.
[72] Variante de DES. Ejecuta tres veces el algoritmo DES.
[73] Algoritmo de encriptación simétrica de 128 bits.
[74] Type Of Service. Método de marcado de paquetes para asignar prioridad en una red que proveE QoS.
[75] Modelo de calidad de servicio de la IETF.

- Microsoft Windows.
- Microsoft Windows CE.
- Monta Vista Linux[76].
- Symbian[77].
- Linux.
- VxWorks[78].
- pSOS[79].
- Nucleus[80].
- Solaris.
- Tru64[81].
- OSE[82].
- INTEGRITY[83].
- UnixWare[84].

3.5.4.7 RTP Library 1.0b2[85]

RTP Library es una interfaz, basada en la especificación RFC1889, para el desarrollo de aplicaciones sobre RTP/RTCP. Fue desarrollada por Lucent Technologies en cooperación con Columbia University, University of Massachusetts Amherst y Bell Laboratories.

[76] http://www.mvista.com/
[77] http://www.symbian.com/
[78] http://www.windriver.com/products/device_technologies/os/vxworks5/
[79] http://syscon.web.arizona.edu/
[80] http://www.acceleratedtechnology.com/embedded/nucleus.html
[81] http://www.tru64.org/
[82] http://www.ose.com/
[83] http://www.ghs.com/products/rtos/integrity.html
[84] http://www.caldera.com/
[85] http://www-out.bell-labs.com/project/RTPlib.

Sus principales características son:

- Soporte para tipos de carga no definidos en IETF RFC1890.
- Encriptación de paquetes RTP y RTCP.

3.5.4.8 Vovida RTP Library 1.5.0[86]

Vovida.org es un sitio Internet que provee un foro de comentarios dedicado a desarrollos de software de código abierto que se use en entornos de comunicaciones de datos y telecomunicaciones.

Uno de sus software publicado es *Vovida RTP Library*. Sus bibliotecas incluyen:

- Soporte para transmisión RTP.
- Transporte de audio.
- Servicio unicast.
- Envío de paquetes RTCP y reportes de calidad.
- Envío de información RTCP SDES.
- Detección de paquetes perdidos, jitter y paquetes fuera de secuencia.
- DTMF.
- Soporte para el codec G.711.

[86] http://www.vovida.org/.

3.5.4.9 WINRTP: Audio RTP Library for Windows 2.1.0[87]

WinRTP fue desarrollado como parte del producto *Cisco IP SoftPhone* y pertenece a Vovida Networks.

Cisco IP SoftPhone es un teléfono IP que trabaja con el software *Cisco CallManager*, y se integra con la solución de negocios *Cisco AVVID* (*Architecture for Voice, Video and Integrated Data*) [CIS04].

WinRTP es una arquitectura de flujo para audio. También se define como un conjunto de bloques prefabricados que permiten que un usuario pueda escribir aplicaciones streaming de audio. Es un componente COM[88] implementado en C++.

WinRTP se compone de dos partes independientes:

1. La que tiene la capacidad de capturar la voz del usuario, codificarla y enviarla como un stream RTP.

2. La que escucha los streams RTP que vienen desde la red, extrae el audio y lo reproduce por los parlantes del PC.

Características de funcionamiento:

- Soporta un stream de entrada y uno de salida.

[87] http://www.vovida.org/
[88] Component Object Model. Modelo diseñado por Microsoft para reutilizar y comunicar distintos componentes software.

- La versión de código abierto soporta el codec G.711 64 Kbps bajo la ley-A y la ley-μ. La versión utilizada para *Cisco IP Softphone* soporta G.723 y G.729.
- Soporta mezcla de audio, enviando y recibiendo archivos WAV al mismo tiempo que se procesan los streams de entrada y salida por defecto.
- Control de volumen de micrófono, parlantes y archivos.
- Supresión de silencio (Voice Activity Detection, VAD), con lo que consigue reducir el uso del ancho de banda al no enviar paquetes RTP cuando no hay señal de entrada.
- Soporta QoS[89] con DiffServ.
- Los extremos del transmisor y receptor son independientes el uno del otro; pueden usarse diferentes codecs para transmitir y recibir, y escoger qué operación hacer (transmitir, recibir o ambas).
- Su buffer de jitter[90] es configurable.
- Soporta todas las versiones de Microsoft Windows posteriores a Microsoft Windows 95.
- Compatible con lenguajes de programación que soporten objetos COM.

[89] *Quality of service*. Calidad de servicio.
[90] Variación en el tiempo promedio entre llegadas de paquetes RTP.

CAPÍTULO 4
DISEÑO Y DESARROLLO DEL PROTOTIPO

4.1 Arquitectura

La figura 4.1-1 representa el prototipo de sistema de reconocimiento, traducción y transporte de VoIP propuesto como solución. El diagrama muestra flujo enviado por un usuario A de lengua española hacia un usuario B de habla inglesa. El usuario B ejecuta una instancia del prototipo remotamente.

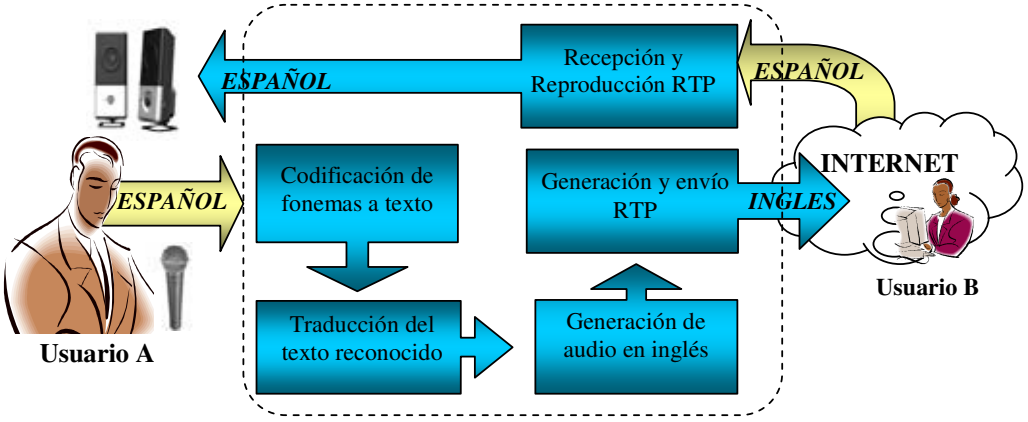


Figura 4.1-1: *Prototipo de sistema de traducción.*

Las bibliotecas empleadas para desarrollar la aplicación de reconocimiento y síntesis de voz fueron *Microsoft SAPI SDK* (capítulo 2.7.5) versión 4.0[91]. Estas bibliotecas, añaden la ventaja de que la aplicación no solamente use los motores incluidos en SAPI, sino que motores de otros desarrolladores también pueden ser reconocidos por la aplicación si son compatibles con SAPI.

[91] La versión 4.0 de SAPI incluye soporte para incorporar diversos modos de síntesis de voz en español. Característica que no comparte con la versión 5.1.

Para habilitar el reconocimiento de voz en español se empleó el motor *Dragon Spanish NaturallySpeaking* (capítulo 2.7.11). El reconocimiento en inglés se logró de dos formas distintas, con *English Continuous* de *Microsoft SAPI SDK* (incluido en las bibliotecas) y con *Dragon English NaturallySpeaking*.

La síntesis en español se implementó haciendo uso de las voces masculina y femenina de *TTS3000* de Lernout & Hauspie (capítulo 2.7.4); la síntesis en inglés con las voces de *Microsoft SAPI SDK* 4.0.

Se investigó sobre motores traductores de texto controlables mediante lenguajes de programación. Los resultados obtenidos arrojaron dos traductores en línea, *AltaVista BabelFish*[92] y *Google Language Tools*[93].

Para acceder a *Altavista BabelFish* es necesario un cliente SOAP[94] que enlace el prototipo con el servicio de *Altavista*. En la práctica, al ejecutar su servicio de traducción, la conexión a *BabelFish* se comporta de manera inestable, y pierde reiteradamente la conexión con el servicio. Una causa posible, es una modificación de acceso al servicio que el cliente no es capaz de resolver. El ANEXO VI describe cómo hacer una conexión SOAP a *AltaVista BabelFish* con *VC++ .NET*.

El acceso a *Google Language Tools* - a diferencia de *AltaVista BabelFish* – se puede obtener directamente de la página de traducción empleando la *Clase C#*

[92] http://babelfish.altavista.com/
[93] http://www.google.cl/language_tools?hl=es
[94] *Simple Object Access Protocol*. Estándar de World Wide Web Consortium (W3C). Protocolo ligero basado en XML, fundado para el intercambio de información en un ambiente descentralizado y distribuido.

WebWagon[95], la cual permite cargar el código completo de paginas HTML. Sin embargo, la sintaxis de *VC#* no es compatible con *VC++* y por lo tanto no es directamente integrable al proyecto de tesis. Por esta razón, como solución a la incompatibilidad de la clase *WebWagon*, se exportó su código a una biblioteca TLB[96]. Así, puede invocarse desde otros desarrollos. Mayor detalle de este proceso se encuentra en el ANEXO VII.

Por otra parte, para realizar la transmisión de la voz se empleó el protocolo RTP. Como ya se describió en el capítulo 3.3, RTP es un protocolo ligero que proporciona soporte a aplicaciones de audio y/o video, o a aquellas que requieren transmisión en tiempo real. Además es un protocolo común a los estándares H.323 y SIP.

Las bibliotecas RTP que constituyen la base de la transmisión y recepción del audio son *WinRTP* (capítulo 3.5.4.9).

La elección de *WinRTP* pasa principalmente por los siguientes puntos:

1. Compatibilidad plena con el entorno de desarrollo.
2. Existe dentro de tecnologías presentes en el mercado (soluciones *Cisco AVVID*).
3. Soporte para envío de archivos de audio.
4. Código abierto y sin costos asociados.

[95] http://www.vsj.co.uk/articles/display.asp?id=389. Clase implementada por Jon Vote, desarrollador certificado de aplicaciones Microsoft, graduado en Ciencias de la computación en Southern Oregon University.
[96] *Type library*. Biblioteca de tipos de datos. Almacena información necesaria para que entornos de programación puedan crear objetos COM y usar sus interfaces.

4.1.1 Reconocimiento, traducción y síntesis

El reconocimiento, traducción y síntesis se lleva a cabo en tres etapas. La primera de ellas consiste la emisión de un mensaje hablado en inglés. El mensaje se captura mediante APIs de reconocimiento de voz (SAPI SR). La figura 4.1.1-1 representa un modelo simplificado de la etapa dentro del sistema.

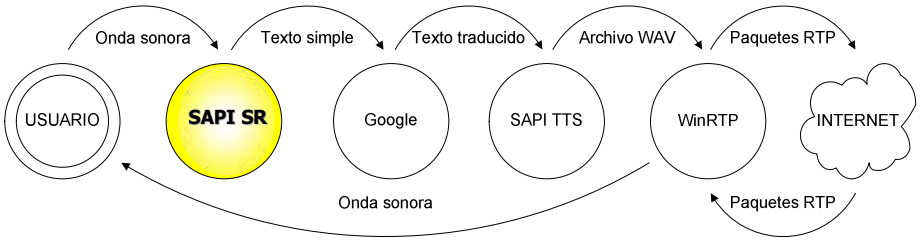


Figura 4.1.1-1: *Etapa de conversión fonemas-texto.*

El reconocimiento se divide en dos funciones. La primera, inicia las APIs y se ejecuta sólo al comenzar la aplicación. La segunda función - residente en memoria – "escucha" la voz del usuario, la convierte en texto y la envía al traductor.

El pseudo-código de su implementación es:

1ª función:

```
INICIO SAPI_SR_INI
      Crear interfaz de origen de audio;
      Obtener los modos SR;
      Seleccionar el modo SR;
      Crear interfaz de gramática;
FIN SAPI_SR_INI
```

2ª función:

```
INICIO SAPI_SR
      MIENTRAS fin de sesión <> VERDADERO
```

```
            SI el usuario habla
                  TEXTO = Extraer texto de la voz;
                  Enviar TEXTO a TRADUCTOR;
            FIN SI
      FIN MIENTRAS;
FIN SAPI_SR
```

Luego, el texto es traducido empleando el servicio web de traducción en línea "*Herramientas del idioma*" de Google (fig. 4.1.1-2).

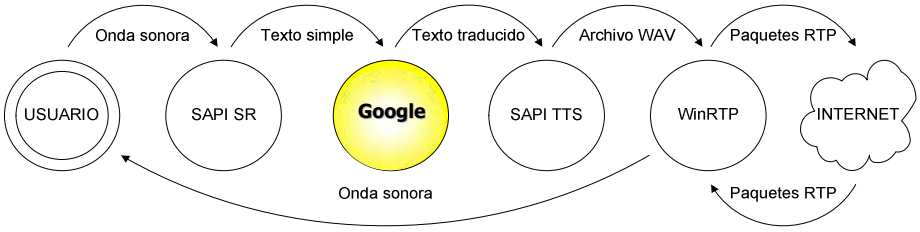


Figura 4.1.1-2: *Etapa de traducción.*

Su pseudo-código es:

```
INICIO TRADUCTOR (FRASE)
      Cargar PAGINA Google;
      Si MI_IDIOMA = ESPAÑOL
            MODO = ESPAÑOL-INGLES
      Si no
            MODO = INGLES-ESPAÑOL;
      TRADUCCION = consulta a PAGINA (FRASE, MODO);
      Enviar TRADUCCION a SAPI_TTS;
FIN TRADUCTOR
```

Cada vez que se genera un texto traducido las *SAPI TTS* se encargan de reproducirlo dentro de un archivo de audio WAV de 16 bits, con frecuencia de muestreo de 22Khz. Cumpliendo la función de buffer temporal.

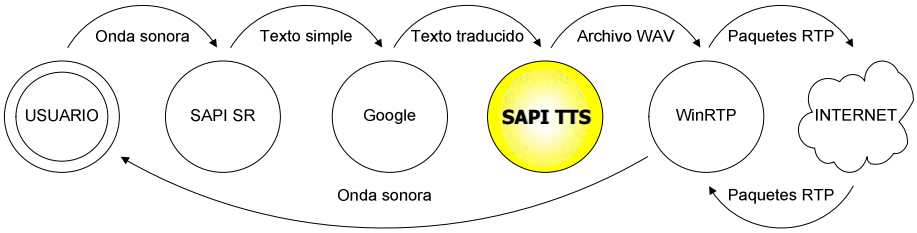


Figura 4.1.1-3: *Etapa de síntesis de voz.*

Al igual que en el nodo de reconocimiento, el proceso de síntesis está dividido en dos funciones. La primera, inicia las APIs creando las interfaces. La segunda función recibe el texto del nodo "Traductor", reproduce el texto en un archivo WAV y lo envía al nodo *WinRTP*.

En pseudo-código:

1ª función:

```
INICIO SAPI_TTS_INI
      Obtener los modos TTS;
      Crear interfaz de destino de audio;
      Seleccionar el modo TTS;
      Crear interfaz de notificación de eventos TTS;
FIN SAPI_TTS_INI
```

2ª función:

```
INICIO SAPI_TTS (FRASE)
      Crear y abrir WAV;
      Reproducir TRADUCCION en WAV;
      Cerrar WAV;
      Enviar WAV a WinRTP;
FIN SAPI_TTS
```

Una vez que la reproducción termina, el archivo se cierra. Termina el reconocimiento, traducción y síntesis.

4.1.2 Transmisión, recepción y reproducción

Los procesos de transmisión, recepción y reproducción de audio son realizados por las bibliotecas *WinRTP* (fig. 4.1.2-1).

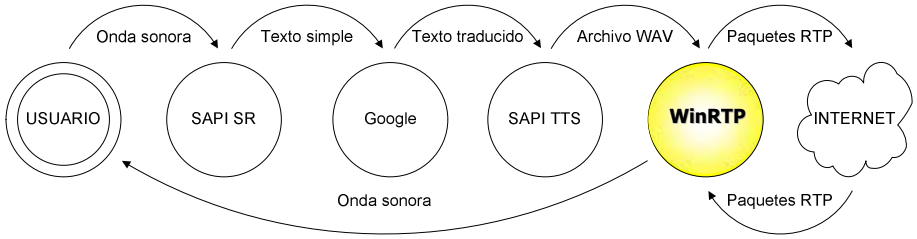


Figura 4.1.2-1: *Etapa de transmisión, recepción y reproducción.*

A un nivel más detallado, lo que se tiene en la etapa *WinRTP* son dos nodos independientes que trabajan al mismo tiempo: "Tx" y "Rx" (fig. 4.1.2-2). "Tx" se encarga de transmitir paquetes RTP, "Rx" recibe los paquetes del participante remoto y los reproduce.

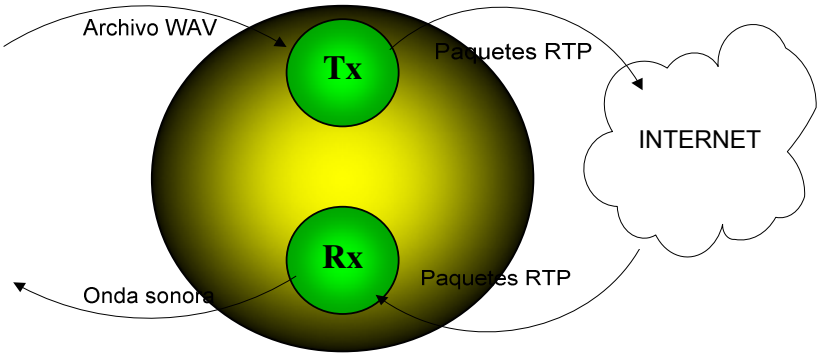


Figura 4.1.2-1: *Nodos de la etapa WinRTP.*

Cada vez que llega un archivo WAV desde el nodo "TTS", "Tx" lo envía directamente al usuario remoto en sucesivos paquetes RTP. La inicialización del

módulo se ejecuta sólo una vez y al comienzo de la sesión. Por lo tanto se tienen dos funciones: la primera de inicialización, y la segunda de envío del archivo con la traducción.

En pseudo-código:

1ª función:

```
INICIO WINRTP_ENVIO_INI
      Fijar codec de transmisión;
      Fijar IP y puertos destino;
FIN WINRTP_ENVIO_INI
```

2ª función:

```
INICIO WINRTP_ENVIO_WAV (WAV)
      Iniciar el flujo de audio;
      Enviar WAV en paquetes RTP;
      Detener el flujo de audio;
FIN WINRTP_ENVIO_WAV
```

El nodo de recepción "Rx" (fig. 4.1.2-1), "espera" constantemente paquetes desde la red hasta que el usuario decida terminar la sesión. Envía el contenido de los paquetes RTP a la interfaz de sonido para reproducirlos por los altavoces del PC.

"Rx" está compuesto por una función de inicialización y un módulo para reproducción continua.

La recepción en pseudo-código es:

1ª función:

```
INICIO WINRTP_RECEPCION_INI
      Fijar codec de recepción;
      Fijar puerto para recibir el flujo;
      Fijar volumen de los altavoces;
FIN WINRTP_RECEPCION_INI
```

2ª función:

```
INICIO WINRTP_RECEPCION_WAV (RTP)
      MIENTRAS fin de sesión <> VERDADERO
            Recibir paquetes RTP;
            Reproducir flujo entrante en altavoces;
      FIN MIENTRAS;
FIN WINRTP_RECEPCION_WAV
```

4.2 Implementación

El desarrollo del software fue por prototipado. Para implementarlo se buscó un lenguaje que además de ser compatible con las bibliotecas *SAPI* y *WinRTP*, fuese de un nivel lo más cercano posible al sistema operativo para obtener un mejor rendimiento.

Por estas razones, se optó por implementar el prototipo en Microsoft Visual C++ .NET accediendo directamente a las APIs de Windows. Cabe destacar que no se emplearon funciones propias de .NET tales como MFC (*Microsoft Foundation Classes*) y herramientas para servicios web.

El nombre dado al prototipo es "*Intérprete*". Para tener información en detalle sobre el proceso para una instalación y entrenamiento adecuado de motores vea el ANEXO VIII.

La aplicación comienza con el cuadro de diálogo de la figura 4.2-1. El usuario debe ingresar un nombre y elegir el modo de reconocimiento de voz que necesite.

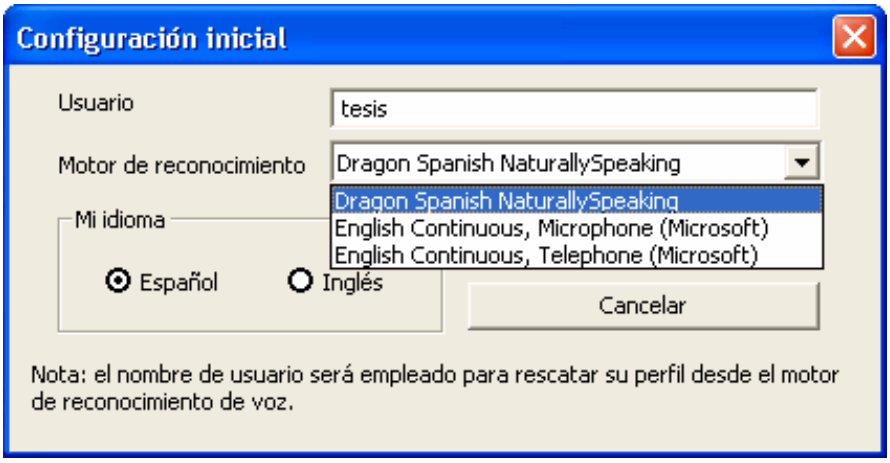


Figura 4.2-1: *Diálogo de configuración inicial.*

El nombre de usuario es necesario para asociar al locutor con el modo de reconocimiento escogido; de esta forma, mientras más sesiones de entrenamiento ejecute el locutor, mejor será el reconocimiento. Toda la información extraída acerca de su voz se guarda en un perfil creado con su nombre de usuario, que a la vez, es su identificador principal. Al hacer clic en "*Aceptar*" un mensaje informará si el perfil existe en el sistema o si ha creado uno nuevo (fig. 4.2-2).

Figura 4.2-2: *Mensajes de registro de usuarios.*

El recuadro "*Mi idioma*" (fig. 4.2-3) es el menú de las dos opciones de traducción. Si por ejemplo el locutor escoge "Inglés", entonces todo el texto reconocido se traducirá de inglés a español.

Figura 4.2-3: *Selección de idioma del usuario.*

Luego, en el segundo cuadro de diálogo (fig. 4.2-4) el usuario debe elegir una voz que lo represente. La voz escogida será la que escuchará el segundo participante de la sesión. Cada vez que el usuario seleccione un tipo de voz del menú, el personaje seleccionado se presentará al usuario, comunicándole que desde ése momento él será su voz sintetizada.

Figura 4.2-4: *Diálogo principal de la aplicación.*

Es necesario para el locutor, conocer el número IP del computador de la segunda persona que ejecuta una instancia de la aplicación (fig. 4.2-5). La comunicación para este prototipo es unicast y se inicia al presionar el botón "Transmitir". Este botón tiene un doble propósito: iniciar, y terminar la transmisión y recepción.

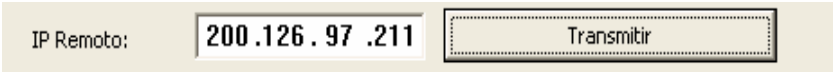


Figura 4.2-5: *Transmisión hacia IP remoto.*

Dos cuadros de texto muestran por separado el texto reconocido (fig. 4.2-6) y la traducción obtenida desde *Google Language Tools* (fig. 4.2-7).

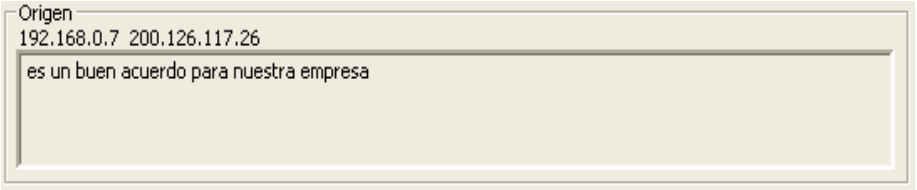


Figura 4.2-6: *Cuadro de texto para voz reconocida.*

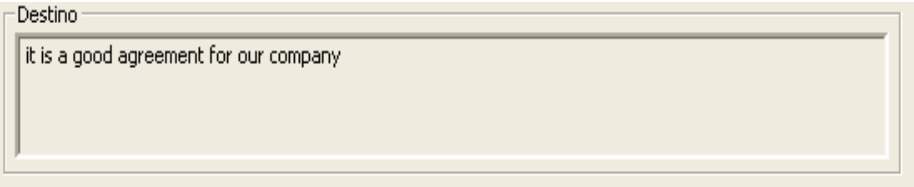


Figura 4.2-7: *Cuadro de texto para traducción recibida.*

En la parte inferior de la figura 4.2-4 se observan tres botones, "Entrenar", "Sensibilidad" y "Audio". Sus funciones son:

- Entrenar: asistente para entrenar el motor de reconocimiento de voz (fig. 4.2-8). Este asistente es propio de los motores Microsoft SAPI, no obstante, es probable que motores ajenos a Microsoft – como *Dragon NaturallySpeaking* - no permitan el entrenamiento por esta vía; en tal caso, dicho motor debe ser entrenado directamente con el software con que fue adquirido. Para mayor información de cómo entrenar los motores de *Dragon NaturallySpeaking* vea el ANEXO VIII.

Figura 4.2-8: *Asistente para entrenamiento de Microsoft SAPI.*

- Sensibilidad: configuración de precisión del reconocimiento, tiempo de respuesta, rechazos y reconocimientos exitosos (fig. 4.2-9 y fig. 10). Las opciones varían entre distintos motores.

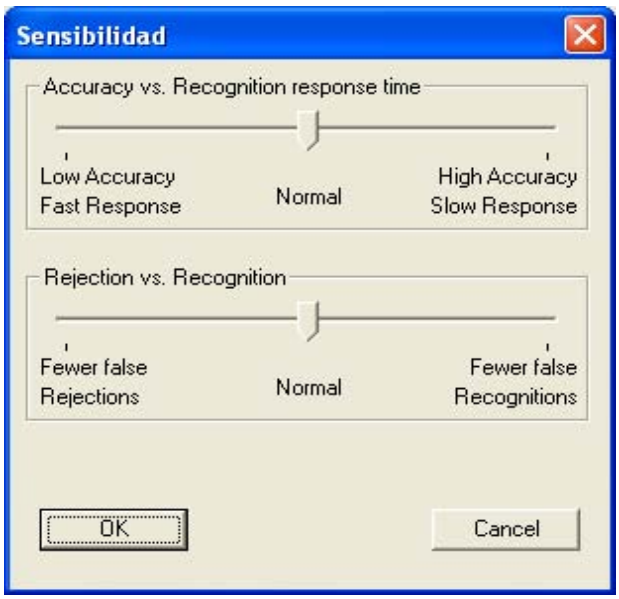


Figura 4.2-9: *Configuración de sensibilidad en Microsoft SAPI.*

Figura 4.2-10: *Configuración de sensibilidad en Dragon NaturallySpeaking*

- Audio: asistente para configuración de los niveles de audio en micrófono y parlantes del PC (fig. 4.2-11).

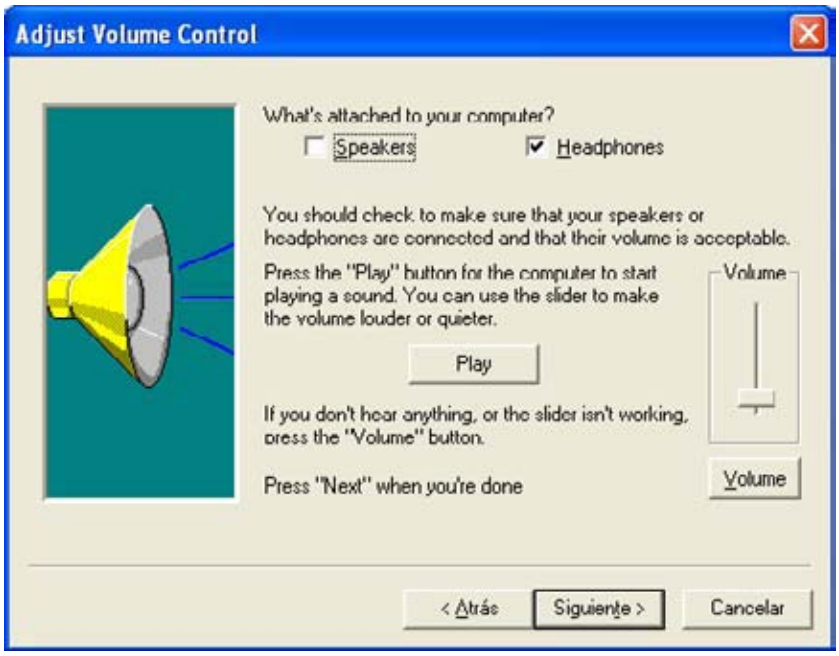


Figura 4.2-11: *Configuración de micrófono y audífonos.*

Para asistir al usuario durante la ejecución del prototipo, el prototipo incluye un archivo de ayuda en formato HLP (fig. 4.2.12). Informa sobre los nombres de usuarios, motores, idiomas y ejecución de sesiones. Se abre al hacer clic en el botón  .

El archivo HLP fue generado con la herramienta *Shalom Help Maker Freeware version 0.5.2*[97].

[97] *http://www.danish-shareware.dk/soft/shelpm/index.html*. Consta de diversas opciones para dar formato a textos y hacer enlaces a páginas web o a aplicaciones externas.

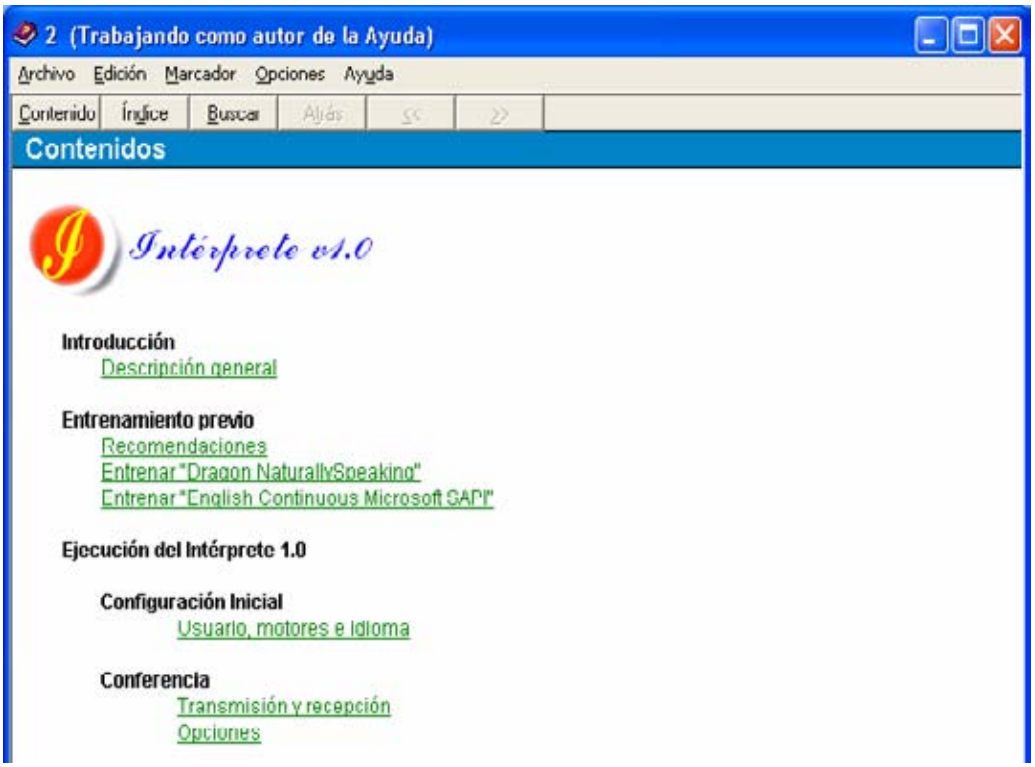


Figura 4.2-12: *Ayuda para usuarios.*

El botón  despliega un cuadro de diálogo con los datos básicos del software.

Figura 4.2-13: *Acerca de Intérprete*

La instalación del programa "Intérprete 1.0" se implemento con el software gratuito *Inno Setup Compiler 5.0.8*[98]

[98] *http://www.jrsoftware.org/. Inno Setup* genera archivos de instalación a partir de un script que define los archivos a instalar y sus destinos en máquinas clientes. Puede invocar variables de sistema, ejecuta aplicaciones externas, valida instalaciones con contraseñas.

CAPÍTULO 5
VALIDACIÓN DEL PROTOTIPO

El mejor rendimiento obtenido por el prototipo fue usando *Dragon NaturallySpeaking Spanish*. Alcanzó una precisión superior a un 96% con sólo 5 sesiones de entrenamiento.

La incorporación de *Google Language Tools* se ajusta a las necesidades de traducción del prototipo; considerando la estabilidad del servicio, rapidez y la adecuada semántica de las traducciones.

La variedad de voces disponibles para síntesis de voz pone a disposición del usuario distintas formas de identificarse. El prototipo cuenta con voces nítidas masculinas, femeninas, con variados tonos y en ambientes diversos.

Se analizó también el comportamiento del ancho de banda (AB) al ejecutar una instancia del prototipo. Las pruebas se hicieron en dos conexiones distintas, y se observó el tráfico de entrada/salida en intervalos de 1 segundo en los nodos de menor velocidad. El software empleado para observar el tráfico fue *PRTG Traffic Grapher V4.3.0.470 Freeware Edition*[99].

Para una conexión entre un nodo de 320/128 Kbps y otro de 512/256 Kbps, se obtuvo el siguiente gráfico:

[99] *http://www.paessler.com/*. PRTG Traffic Grapher destaca entre otros softwares por la versatilidad de sus gráficos, capacidad para analizar AB de distintas interfaces de red y soporte para publicación de datos por HTML. Actualmente lo emplean instituciones educacionales, químicas, financieras, informáticas, de telecomunicaciones y transporte.

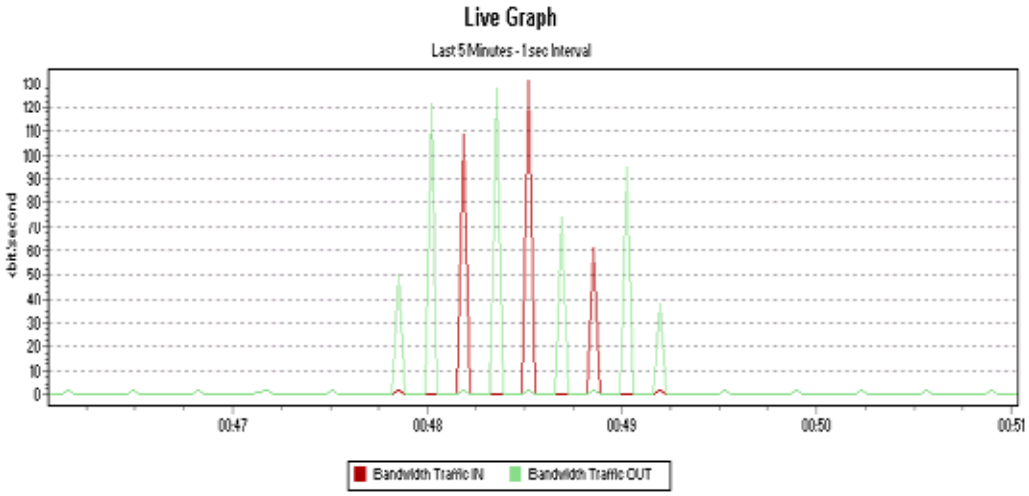


Gráfico G5-1: *Consumo de ancho de banda en conexión de 320/128 Kbps.*

Del gráfico se desprende la tabla T5-1, que refleja que el consumo de AB de salida en el nodo 320/128 Kbps no superó los 128 Kbps, y el flujo de entrada alcanzó como máximo los 131 Kbps. Las frases transmitidas se recibieron con un retardo aproximado de 1 seg. y sin jitter perceptible.

Frase sintetizada	Kbps
buenos días	50
me gustaría acordar una fecha para la reunión	122
we will travel Monday of the next week	109
entonces los esperamos en el aeropuerto	128
our airplane will arrive at the 11 hours	131
perfecto ahí estaremos	74
we see Monday	62
buen viaje y nos vemos pronto	95
Adiós	39

Tabla T5-1: *Consumo de ancho de banda en conexión de 56/48 Kbps.*

En una sesión entre un nodo con Internet por RTC[100] 56/48 Kbps y un nodo ADSL 320/128 Kbps, el resultado de *PRTG Traffic Grapher* es el gráfico G5-2.

[100] RTC. *Red Telefónica Conmutada* .

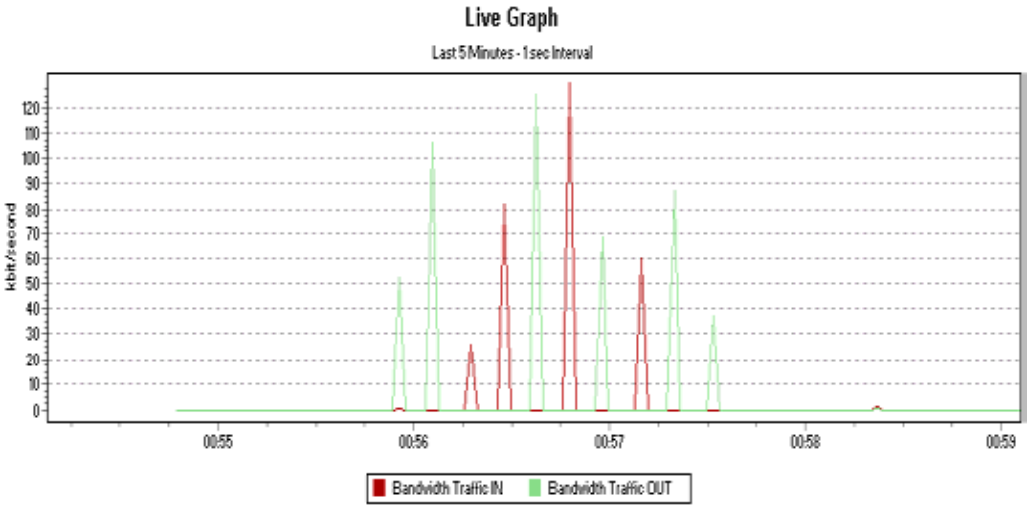


Gráfico G5-2: *Consumo de ancho de banda en conexión de 56/48 Kbps.*

Como se ve en la tabla T5-2, en esta conexión el consumo de AB de salida y entrada sigue el mismo comportamiento de consumo que en la prueba anterior. No obstante, las frases traducidas se escuchan interrumpidas por silencios, lo que significa una disminución en la calidad del audio. Además, el retardo aumentó a 3 segs. aproximadamente.

Frase sintetizada	Kbps
buenos días	52
me gustaría acordar una fecha para la reunión	108
we will travel Monday of the next week	82
entonces los esperamos en el aeropuerto	125
our airplane will arrive at the 11 hours	128
perfecto ahí estaremos	70
we see Monday	60
buen viaje y nos vemos pronto	88
adiós	38

Tabla T5-2: *Consumo de ancho de banda en conexión de 56/48 Kbps.*

Para comprobar cómo se comporta el consumo de AB con frases más extensas, desde el nodo 320/128 se realizó la transmisión de frases con distintos tamaños. En la tabla T5-3 se observan los tamaños (en palabras y caracteres) de 11 frases. Cada frase

tiene asociado el tiempo necesario para sintetizarla, y el consumo máximo de AB de salida medido en intervalos de 1 seg.

Frase	Caracteres	Palabras	Kbps	Segundos
1	500	77	400	35
2	450	66	400	31
3	400	60	400	27
4	350	50	400	25
5	300	46	385	21
6	250	38	380	18
7	200	32	370	14
8	150	26	350	10
9	100	19	230	7
10	50	9	140	4
11	25	5	75	2

Tabla T5-1: *Rendimiento por caracteres en conexión 320/128 Kbps.*

Al observar los datos de la tabla en el gráfico G5-3, se distingue un consumo ascendente de AB para frases de hasta 150 caracteres. Luego, el consumo de AB se estabilizó en los 400 Kbps.

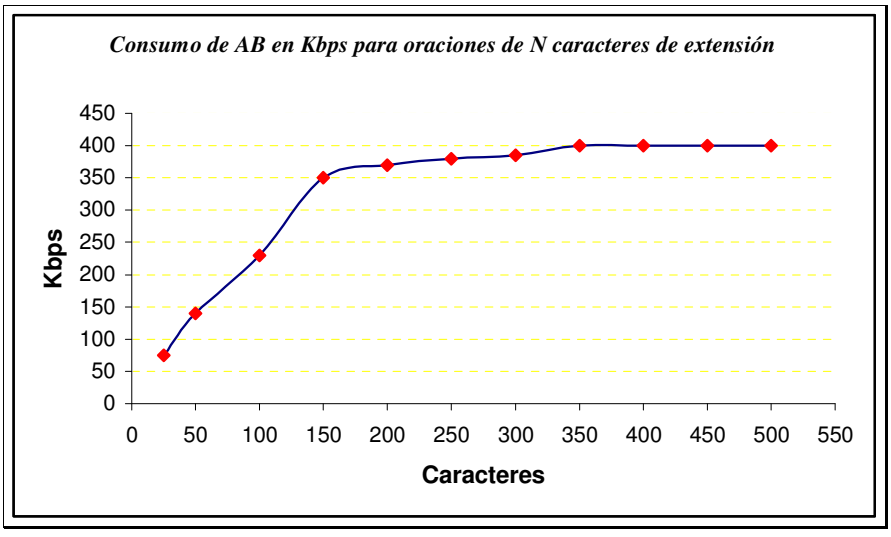


Gráfico G5-3: *Consumo de ancho de banda en Kbps para frases de N caracteres de extensión.*

La calidad del audio recibido en el nodo 512/256 Kbps, no se percibió con jitter hasta la recepción de frases de 100 caracteres. La utilización de AB registró 230 Kbps en el nodo 320/128.

El tiempo necesario para generar el buffer de voz sintetizada se comportó de forma lineal (gráf. G5-4). La extensión de la frase es proporcional al tiempo empleado para sintetizarla.

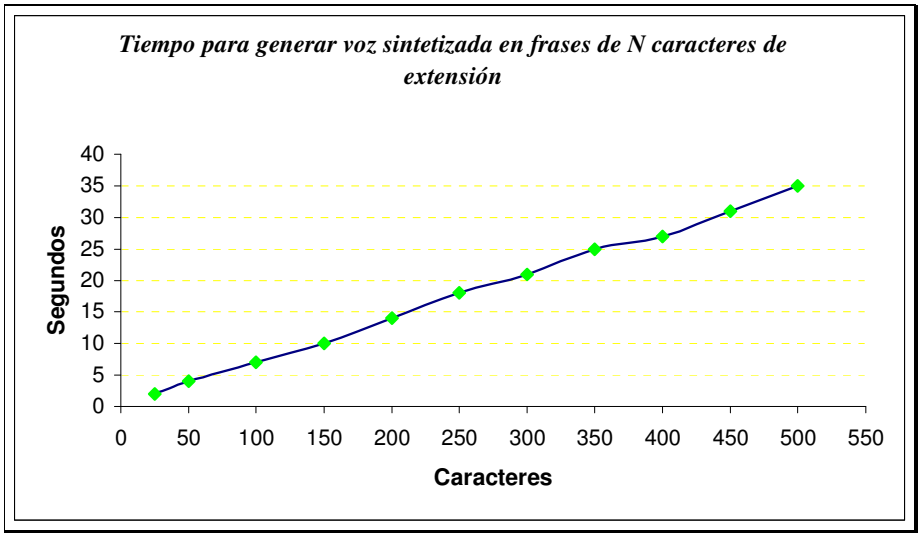


Gráfico G5-4: *Tiempo para generar voz sintetizada en frases de N caracteres de extensión.*

Requerimientos funcionales

- *Microsoft SAPI SDK 4.0* (incluye motores de reconocimiento y síntesis en inglés).
- Motores:
    - *Lernout & Hauspie TTS3000* (síntesis masculina y femenina en español).
    - *Dragon NaturallySpeaking 8 Preferred Edition Spanish* (reconocimiento y síntesis en español e inglés)[101].
- Microsoft .NET Framework runtime files 1.1.
- Hardware mínimo requerido:
    - Procesador 500 MHz o superior.
    - 256 MB RAM.
    - 630 MB de espacio libre en disco.
    - Tarjeta de sonido *Creative SoundBlaster 16* o equivalente.
    - Micrófono y audífonos.

[101] Recomendado para reconocimiento en español, aunque en su lugar puede emplearse otro compatible con SAPI. No se incluye en el CD de la tesis por ser un software comercial.

CAPÍTULO 6
CONCLUSIONES

Sobre los objetivos de investigación…

La investigación sobre reconocimiento de voz confirmó la diferencia entre información de alto y bajo nivel (propuesta hecha por Bergdata Biometrics). Todas las herramientas para desarrollo descritas en la tesis abordan la información de bajo nivel (ritmo, tono, magnitud espectral, frecuencias) eficientemente. Estos esfuerzos, son el primer paso para enfocar la nueva generación de motores hacia el reconocimiento de dialecto, acento, estilo de habla, contexto y emociones.

El hardware dejará de ser una limitante para los algoritmos de voz y se convertirá en una extensión de ellos. Técnicas como "síntesis por formantes" y "síntesis por concatenación de grandes unidades", cuestionadas por consumir demasiada memoria o procesador, podrían revitalizarse a la luz del explosivo desarrollo del hardware. Igualmente, las iniciativas por traspasar el software a chips de silicio fomentarían la masificación de las tecnologías de voz, sobre todo en el sector de la domótica y aplicaciones de asistencia personal.

La investigación sobre VoIP reveló a SIP como uno de los estándares multimedia con más proyección. Ser liviano, flexible y extensible, lo hacen destacar sobre el estándar H.323.

RTP se ha consolidado como el protocolo para transmisión de datos multimedia en tiempo real. La independencia del protocolo de transporte, el soporte para funciones

de transmisión y control, y la importante presencia en los estándares SIP y H.323, propios de RTP, corroboran la afirmación.

Sobre los objetivos de diseño e implementación…

Esta tesis se enfocó como un trabajo de ingeniería que busca solución a una limitante social y de comunicaciones, la diferencia de idiomas. Se enfrentaron procesos de apertura de paquetes tecnológicos e integración de reconocimiento de voz, síntesis de voz, VoIP y las funciones de un servicio web.

El diseño de solución propuesto, otorga al proyecto un potencial equiparable con complejas propuestas universitarias. Un ejemplo es el sistema JANUS, que ha sido empleado para tomar decisiones entre personas que no comparten el mismo idioma, y que necesitan una traducción inmediata.

Se logró la implementación de un prototipo que hace converger plenamente las tres tecnologías estudiadas: reconocimiento de voz, síntesis de voz y VoIP.

El software diseñado sintoniza con las necesidades del usuario. La elección de bibliotecas compatibles con soluciones ajenas a Microsoft, hizo que el usuario final tenga en sus manos la posibilidad de adquirir motores (de reconocimiento y/o síntesis), que estén a su alcance económico y se ajusten a sus requerimientos específicos.

El módulo de traducción implementado cumple con la funcionalidad requerida para el prototipo. El enlace a *Google Language Tools* proporcionó frases traducidas

exitosamente en ambos sentidos (español/inglés, inglés/español) y con una correcta sintaxis y semántica.

El diseño del prototipo propuesto posee una conectividad extensible. Incorporar un módulo SIP para abordar la telefonía IP y tradicional, no representaría un cambio radical en la arquitectura propuesta. Además, SIP emplea RTP para transmitir información en tiempo real.

Sobre la validación del prototipo…

Las conexiones ADSL soportan el flujo de entrada/salida generado por las traducciones, mejorando la comprensión del mensaje hablado. Por lo tanto, basándose en los resultados del gráfico G5-3 - donde el consumo de AB se estabiliza en 400Kbps - se recomienda este tipo de conexiones para un mejor desempeño.

La información del gráfico G5-4, indica que el óptimo sería sintetizar traducciones menores de 50 caracteres (10 palabras aprox.). Una extensión de 50 caracteres no supera los 4 segundos en sintetizar. La implementación de un algoritmo que divida frases extensas, en nuevas de menor tamaño, podría reducir aún más el tiempo de síntesis.

La utilización de motores de reconocimiento con porcentajes altos de precisión - cómo es el caso de *Dragon NaturallySpeaking* - reduce la ocurrencia de reconocimientos erróneos. Esto beneficia directamente la eficiencia del módulo de reconocimiento y del sistema en general.

Comentarios finales…

En el marco de la investigación sobre transmisión en tiempo real, se estableció contacto con una de las personas que ha hecho posible la VoIP, el profesor Henning Schulzrinne. Este docente - miembro del departamento de ciencias de la computación de Columbia University e ingeniero eléctrico participante en el desarrollo de RTP, RTSP e impulsor principal de SIP - accedió (vía correo electrónico) a orientar la búsqueda de material para el proyecto, específicamente en la investigación sobre protocolos de tiempo real y RTP. El interés demostrado por Schulzrinne en la tesis, y los esfuerzos comunes de universidades extranjeras comprometidas con la investigación, son muestras de la gran importancia dada al trabajo en conjunto, y al intercambio de conocimientos para el desarrollo informático. Es una forma de crear ciencia, la cual debería imitarse instaurando nexos de cooperación intra e inter-universitarios.

El sistema de traducción de voz diseñado puede complementarse con trabajos afines dentro de la universidad. Paralelamente a la tesis, se visualizó la posibilidad de crear un desarrollo en conjunto con alumnos de la escuela ingeniería civil acústica UACh. Ellos ya desarrollaron un motor de reconocimiento voz, y actualmente se encuentran trabajando en un motor de síntesis de voz. Son esfuerzos que, unidos al sistema de traducción de voz, podrían fortalecerse entre sí, apuntando a resolver problemáticas como: titubeos, cambios de tonos, disfonías, cambios de velocidad al hablar, ruidos propios del ambiente, y definición de las gramáticas.

CAPÍTULO 7
BIBLIOGRAFÍA

[MÖS04] Möser, M.; Barros, J.. "*Ingeniería Acústica Teoría y Aplicaciones*". Universidad Austral de Chile 2004. Capítulo 1, p 1.

[QUI68] Quilis, A.; Fernández, J.. "*Curso de fonética y fonología españolas*". Instituto Miguel de Cervantes 1968. Capítulo 1, p 9.

[XUE01] Huang, X.; Acero, A.. "*Spoken Language Processing*". Prentice Hall 2001. Capítulo 1 p. 4 y 5. Capítulo 2 p. 27. Capítulo 16 p. 793, 796, 803-805, 807, 809.

[QUI68] Quilis, A.; Fernández, J.. "*Curso de fonética y fonología españolas*". Instituto Miguel de Cervantes 1968. Capítulo 1, p 9.

[KOH88] Kohonen, T.. "*The "Neural" Phonetic Typewriter*" [en línea]. Helsinki University of Technology.
<http://csdl.computer.org/comp/mags/co/1988/03/r3011abs.htm>
[Consulta: 30 nov 2004]

[JÓZ05]József, N. "*Tudományos-technikai eredményeink és az európaiság*". Technikai örökségünk az interneten.
<http://www.scitech.mtesz.hu/>
[Consulta: 09 may 2005]

[BEL00] Multimedia Communications Research Laboratory. "*Invention of Vocoder*" [en línea]. Lucent Technologies.
<http://www.bell-labs.com/org/1133/Heritage/Vocoder/>
[Consulta: 30 nov 2004]

[WOL98] Traunmüller, H.. "*Wolfgang von Kempelen's speaking machine and its successors*" [en línea]. Inst. för lingvistik Stockholms universitet.
<http://www.ling.su.se/staff/hartmut/kemplne.htm>
[Consulta: 30 nov 2004]

[RES04] Research-Lab Inc.. "*Introduction to Speech Recognition*" [en línea]. Research-Lab Inc..
<http://www.research-lab.com/docintro.htm>
[Consulta: 15 dic 2004]

[AHU99] Ahuactzin, A.. "*Diccionario español/inglés para el aprendizaje de vocabulario utilizando una interfaz de voz*" [en línea]. Universidad de las Américas.
<http://140.148.3.250/u_dl_a/servlet/mx.udlap.ict.tales.html.Block?Thesis=15&Type=O>
[Consulta: 29 nov 2004]

[GRA03] Graevenitz, G.. "*About Speaker Recognition Technology*" [en línea]. Bergdata Biometrics GmbH.
<http://www.bergdata.com/downloads/Introduction%20to%20Speaker%20Recognition%20Technology.pdf>
[Consulta: 25 oct 2004]

[ABR03] De Abreu, F.; Lucena, L.. "*Identificación de sistemas estáticos y dinámicos a través de redes autoasociativas*" [en línea]. Universidad Simón Bolivar.
< http://iq.coord.usb.ve/pdf/miniproyecto/abr_jul2002/deabreu_luce.pdf>
[Consulta: 30 nov 2004]

[HER01] Hernández, L.; Caminero F. J.. "*Estado del arte en Tecnología del Habla*" [en línea]. Universidad Politécnica de Madrid, Telefónica investigación y desarrollo.
<http://www.tid.es/presencia/publicaciones/docs_comtid/numero10.pdf>
[Consulta: 19 jun 2003]

[GON01] González, E.; Calero, J.. "*Aplicaciones de la Tecnología del Habla*" [en línea]. Telefónica investigación y desarrollo, Telefónica de España.
<http://www.tid.es/presencia/publicaciones/docs_comtid/numero10.pdf>
[Consulta: 19 jun 2003]

[VIV03] Vivaracho, C.; Moro, I.. "*Creación de una base de datos para reconocimiento de personas mediante multimodalidad biométrica*" [en línea]. Universidad de Valladolid, Universidad del País Vasco.
<http://www.infor.uva.es/biometria/Documentos/Articulos/Biometria_SA.pdf>
[Consulta: 28 may 2003]

[XUE01] Huang, X.. "*Spoken Language Processing*". Prentice Hall 2001. Capítulo 1, p. 4 y 5.

[ROD01] Rodríguez, M.; Cortázar, I.. "*Estado del arte en tecnologías de voz*" [en línea]. Telefónica Investigación y Desarrollo.
<http://www.tid.es/presencia/publicaciones/comsid/esp/20/8XX.PDF>
[Consulta: 28 may 2004]

[POZ04] Poza, M.; Villarrubia, L.. "*Teoría y aplicaciones del reconocimiento automático del habla*" [en línea]. Telefónica Investigación y Desarrollo.
<www.tid.es/presencia/publicaciones/docs_comtid/numero3.pdf>
[Consulta: 6 dic 2004]

[ENC96] Enciclopedia Británica Publisher, Inc. “Enciclopedia Hispánica Macropedia vol.6”. Editorial Ercilla Galicia 1996.

[COL01] Colás, J.; “*Estrategias de incorporación de conocimiento sintáctico y semántico en sistemas de comprensión de habla continua en español*” [en línea]. Escuela Técnica Superior de Ingenieros de Telecomunicación, Madrid (España).
<http://elies.rediris.es/elies12/>
[Consulta: 6 dic 2004]

[SCI99] ScienceDaily. “*Carnegie Mellon Scientists To Demonstrate Spontaneous Speech-To-Speech Translation In Six Languages*” [en línea]. Carnegie Mellon University.
<http://www.sciencedaily.com/releases/1999/07/990723084011.htm>
[Consulta: 13 dic 2004]

[QUA00] Waibel, A.. “*La traducción interactiva del habla*” [en línea]. Carnegie Mellon University.
<http://www.imim.es/quark/19/default.htm>
[Consulta: 20 feb 2003]

[OST92] Osterholtz, L.; Augustine, C.. “*Testing generality in JANUS: A multi-lingual speech translation system*” [en línea]. Carnegie Mellon University, University of Pennsylvania, Universität Karlsruhe, Keyko University.
<http://ieeexplore.ieee.org/>
[Consulta: 18 dic 04]

[WAI97] Waibel, A.. “*Interactive Systems Laboratories*” [en línea]. Carnegie Mellon University, Universität Karlsruhe.
<http://www.ifp.uiuc.edu/nsfhcs/abstracts/waibel.txt>
[Consulta: 18 dic 2004]

[WOZ96] Wozczyna, M.; Coccaro, N.. "*Recent advances in JANUS: A speech translation system*" [en línea]. Carnegie Mellon University, Universität Karlsruhe.
<http://ieeexplore.ieee.org/>
[Consulta: 18 dic 2004]

[CAT02] Cattoni, R.; Lazzari, G.. "*Not only Translation Quality: Evaluating the NESPOLE! Speech-to-Speech Translation System along other Viewpoints*" [en línea]. ITC-irst, Carnegie Mellon University, Universität Karlsruhe, CLIPS, University of Trieste, AETHRA.
<http://www-clips.imag.fr/geta/herve.blanchon/Docs/nespo-acl02.pdf>

[CAR04] Carnegie Mellon University. "*Carnegie Mellon Engineering Researchers To Create Speech Recognition in Silicon*" [en línea]. Carnegie Mellon University.
<http://www.cmu.edu/PR/releases04/040913_speech.html>
[Consulta: 27 nov 2004]

[IBM02] IBM Corporation 2002. "*ATX Technologies uses voice recognition for hands-free traffic reporting*" [en línea]. IBM Corporation.
<http://www-
1.ibm.com/industries/wireless/doc/content/casestudy/1153037104.html>
[Consulta: 25 oct 2004]

[NUA01] Nuance Communications, Inc.. "*Nuance Speech Recognition System Version 7.0 Introduction to the Nuance System*" [en línea]. Nuance Communications, Inc..
<http://www.nuance.com>
[Consulta: 9 nov 2002]

[WYN01] Wyne, M.. "*Open and standardized – the world of IP Protocols. H.323: The Leading Standard in Voice over IP*" [en línea]. H.323 Forum.
<http://www.h323forum.org/papers/h.323_white_paper.pdf>

[Consulta: 02 sep 2004]

[CHE98] Chen, Y.; Wang, Y.. "*A New Fast Motion Estimation Algorithm*" [en línea]. University of Texas.
<http://www.ece.utexas.edu/~bevans/courses/ee381k/projects/fall98/zhou/literatureSurvey.pdf>
[Consulta: 22 dic 2004]

[DAT98] DataBeam Corporation. "*A Primer on the T.120 Series Standard*". DataBeam Corporation.
<http://www.packetizer.com/conf/t120/primer/t120_primer.pdf>
[Consulta: 22 dic 2004]

[DAN02] Dang, L.; Cullen, J.. "*Practical VoIP Using Vocal*" [en línea]. Editorial O'Reilly. Capítulo 7 p. 126.
<www.oreilly.com/catalog/voip/chapter/ch07.pdf>
[Consulta: 15 dic 2004]

[ERI00] Eriksson, G.; Olin, B.. "*Los retos de la voz por IP sin hilos*" [en línea]. Ericsson Review n° 1, 2000.
<http://www.ericsson.com/about/publications/review/2000_01/files/es2000013.pdf>
[Consulta: 16 ago 2004]

[GRU01] Grupo de Sistemas y Comunicaciones. "*Protocolos de transporte con entrega en tiempo real*" [en línea]. Universidad Rey Juan Carlos.
<http://gsyc.escet.urjc.es/docencia/cursos/fse-mbone/transpas/node9.html>
[Consulta: 22 abr 2003]

[SAL02] Salvachúa J.. "*Realtime Transport Protocol RTP*". Departamento de Ingeniería de Sistemas Telemáticos. Universidad Politécnica de Madrid.

<http://www.lab.dit.upm.es/~labscom/almacen/sld/rtp.pdf>
[Consulta: 22 abr 2003]

[SHU03] Schulzrinne, H.. "*RTP: About RTP and the Audio-Video Transport Working Group*" [en línea]. Columbia University Department of Computer Science.
<http://www.cs.columbia.edu/~hgs/rtp/>
[Consulta: 22 abr 2003]

[CIS04] Cisco Systems, Inc.. "*Cisco IP Softphone*", "*Cisco CallManager*", "*Cisco AVVID Network Infrastructure*". Cisco Systems, Inc..
<http://www.cisco.com/en/US/products/sw/voicesw/>
[Consulta: 22 abr 2003]

[SUN04] Sun Microsystems. "*Java Media Framework API (JMF)*" [en línea]. Sun Microsystems.
<http://java.sun.com/products/java-media/jmf/reference/docs/index.html>
[Consulta: 09 nov 2003]

[MIC03] Microsoft SAPI. "*Microsoft SAPI 4.0a*" [en línea]. Microsoft.
<http://www.microsoft.com/speech/download/old/sdk40a.asp>
[Consulta: 09 nov 2003]

[LIN01] Anónimo. "*Linux Máxima Seguridad*", Edición Especial. Editorial Prentice Hall. Capítulo 2 p. 36.

[IAN04] IANA Internet Assigned Numbers Authority. "*Session Initiation Protocol (SIP) Parameters*" [en línea]. The Internet Corporation for Assigned Names and Numbers.
<http://www.iana.org/assignments/sip-parameters>
[Consulta: 29 mar 2005]

Estándares

[URL1] ITU-T Telecom Standardization Sector. "*H.323 : Sistemas de comunicación multimedios basados en paquetes*". International Telecommunication Union.
<http://www.itu.int/rec/recommendation.asp?lang=s&type=folders&parent=T-REC-H.323>
[Consulta: 02 sep 2004]

[URL2] ITU-T Telecom Standardization Sector. "*H.225.0: Protocolos de señalización de llamada y paquetización de trenes de medios para sistemas de comunicación multimedios por paquetes*". International Telecommunication Union.
<http://www.itu.int/rec/recommendation.asp?type=folders&lang=s&parent=T-REC-H.225.0>
[Consulta: 02 sep 2004]

[URL3] ITU-T Telecom Standardization Sector. "*H.245 : Protocolo de control para comunicación multimedios*". International Telecommunication Union.
<http://www.itu.int/rec/recommendation.asp?type=folders&lang=s&parent=T-REC-H.245>
[Consulta: 02 sep 2004]

[URL4] ITU-T Telecom Standardization Sector. "*Q.931 : Especificación de la capa 3 de la interfaz usuario-red de la red digital de servicios integrados para el control de la llamada básica*". International Telecommunication Union.
<http://www.itu.int/rec/recommendation.asp?type=folders&lang=s&parent=T-REC-Q.931>
[Consulta: 02 sep 2004]

[URL5] ITU-T Telecom Standardization Sector. "*G.711 : Modulación por impulsos codificados (MIC) de frecuencias vocales*". International Telecommunication Union.
<http://www.itu.int/rec/recommendation.asp?type=folders&lang=s&parent=T-REC-G.711>
[Consulta: 02 sep 2004]

[URL6] ITU-T Telecom Standardization Sector. "*G.723 : Extensiones de la modulación por impulsos codificados diferencial adaptativa de la Recomendación G.721 a 24 y 40 kbit/s para aplicaciones en equipos de multiplicación de circuitos digitales*". International Telecommunication Union.
<http://www.itu.int/rec/recommendation.asp?type=folders&lang=s&parent=T-REC-G.723>
[Consulta: 02 sep 2004]

[URL7] ITU-T Telecom Standardization Sector. "*H.261 : Códec vídeo para servicios audiovisuales a p x 64 kbit/s*". International Telecommunication Union.
<http://www.itu.int/rec/recommendation.asp?type=folders&lang=s&parent=T-REC-H.261>
[Consulta: 02 sep 2004]

[URL8] ITU-T Telecom Standardization Sector. "*H.263: Codificación de vídeo para comunicación a baja velocidad binaria*". International Telecommunication Union.
<http://www.itu.int/rec/recommendation.asp?type=folders&lang=s&parent=T-REC-H.263>
[Consulta: 02 sep 2004]

[URL9] ITU-T Telecom Standardization Sector. "*T.120: Protocolo de datos para conferencias multimedios*". International Telecommunication Union.
<http://www.itu.int/rec/recommendation.asp?type=folders&lang=s&parent=T-REC-T.120>
[Consulta: 02 sep 2004]

[URL10] ITU-T Telecom Standardization Sector. "*G.729: Codificación de la voz a 8 kbit/s mediante predicción lineal con excitación por código algebraico de estructura conjugada*". International Telecommunication Union.
<http://www.itu.int/rec/recommendation.asp?type=folders&lang=s&parent=T-REC-G.729>
[Consulta: 22 abr 2003]

[URL11] ITU-T Telecom Standardization Sector. "*H.235 : Seguridad y criptado para terminales multimedios de la serie H (basados en las Recomendaciones UIT-T H.323 y H.245)*". International Telecommunication Union.
<http://www.itu.int/rec/recommendation.asp?type=folders&lang=s&parent=T-REC-H.235>
[Consulta: 22 abr 2003]

[URL12] ITU-T Telecom Standardization Sector. "*G.728 : Codificación de señales vocales a 16 kbit/s utilizando predicción lineal con excitación por código de bajo retardo*". International Telecommunication Union.
<http://www.itu.int/rec/recommendation.asp?type=folders&lang=s&parent=T-REC-G.728>
[Consulta: 22 abr 2003]

[URL13] ITU-T Telecom Standardization Sector. "*G.722 : 7 Khz audio-coding within 64 kbit/s*". International Telecommunication Union.

<http://www.itu.int/rec/recommendation.asp?type=folders&lang=s&parent=T-REC-G.722>
[Consulta: 22 abr 2003]

[URL14] ITU-T Telecom Standardization Sector. "*H.264 : Codificación de vídeo avanzada para los servicios audiovisuales genéricos*". International Telecommunication Union.
<http://www.itu.int/rec/recommendation.asp?type=folders&lang=s&parent=T-REC-H.264>
[Consulta: 22 abr 2003]

[URL15] ITU-T Telecom Standardization Sector. "*G.726 : Modulación por impulsos codificados diferencial adaptativa (MICDA) a 40, 32, 24, 16 kbit/s*". International Telecommunication Union.
<http://www.itu.int/rec/recommendation.asp?type=folders&lang=s&parent=T-REC-G.726>
[Consulta: 22 abr 2003]

[URL16] ITU-T Telecom Standardization Sector. "*Information technology - Digital compression and coding of continuous-tone still images: Registration of JPEG Profiles, SPIFF Profiles, SPIFF Tags, SPIFF colour Spaces, APPn Markers, SPIFF Compression types and Registration Authorities (REGAUT)*". International Telecommunication Union.
<http://www.itu.int/rec/recommendation.asp?type=items&lang=E&parent=T-REC-T.86-199806-I>
[Consulta: 22 abr 2003]

[URL17] Network Working Group. *Request for Comments: 3261 "SIP: Session Initiation Protocol*" [en línea]. The Internet Engineering Task Force.
<ftp://ftp.rfc-editor.org/in-notes/rfc3261.txt>
[Consulta: 29 mar 2005]

[URL18] Network Working Group. *Request for Comments: 2327* "SDP: Session Description Protocol" [en línea]. The Internet Engineering Task Force.
<ftp://ftp.rfc-editor.org/in-notes/rfc2327.txt>
[Consulta: 29 mar 2005]

[URL19] Network Working Group. *Request for Comments: 1889* "*RTP: A Transport Protocol for Real-Time Applications*" [en línea]. The Internet Engineering Task Force.
<ftp://ftp.rfc-editor.org/in-notes/rfc1889.txt>
[Consulta: 22 abr 2003]

[URL20] Network Working Group. *Request for Comments: 2326* "*Real Time Streaming Protocol (RTSP)*" [en línea]. The Internet Engineering Task Force.
<ftp://ftp.rfc-editor.org/in-notes/rfc2326.txt>
[Consulta: 22 abr 2003]

[URL21] Network Working Group. *Request for Comments: 1890* "*RTP Profile for Audio and Video Conferences with Minimal Control*" [en línea]. The Internet Engineering Task Force.
<ftp://ftp.rfc-editor.org/in-notes/rfc1890.txt>
[Consulta: 22 abr 2003]

[URL22] Network Working Group. *Request for Comments: 2032* "*RTP Payload Format for H.261 Video Streams*" [en línea]. The Internet Engineering Task Force.
<ftp://ftp.rfc-editor.org/in-notes/rfc2032.txt>
[Consulta: 22 abr 2003]

[URL23] Network Working Group. *Request for Comments: 2190* "*RTP Payload Format for H.263 Video Streams*" [en línea]. The Internet Engineering Task Force.

<ftp://ftp.rfc-editor.org/in-notes/rfc2190.txt>
[Consulta: 22 abr 2003]

[URL24] Network Working Group. *Request for Comments: 2833* "*RTP Payload for DTMF Digits, Telephony Tones and Telephony Signals*" [en línea]. The Internet Engineering Task Force.
<ftp://ftp.rfc-editor.org/in-notes/rfc2833.txt>
[Consulta: 22 abr 2003]

[URL25] Network Working Group. *Request for Comments: 2974* "*Session Announcement Protocol*" [en línea]. The Internet Engineering Task Force.
<ftp://ftp.rfc-editor.org/in-notes/rfc2974.txt>
[Consulta: 22 abr 2003]

[URL26] Moving Picture Experts Group. "*Information technology -- Coding of moving pictures and associated audio for digital storage media at up to about 1,5 Mbit/s*". ISO/IEC, International organization for standaritation / International Electrotechnical Commission.
<http://www.iso.org/iso/en/CombinedQueryResult.CombinedQueryResult?queryString=CODING+OF+MOVING+PICTURES+AND+AUDIO>
[Consulta: 22 dic 2004]

[URL27] Moving Picture Experts Group. "*Information technology -- Generic coding of moving pictures and associated audio information*". ISO/IEC, International organization for standaritation / International Electrotechnical Commission.
<http://www.iso.org/iso/en/CombinedQueryResult.CombinedQueryResult?queryString=Generic+coding+of+moving+pictures+and+associated+audio+information>
[Consulta: 22 dic 2004]

[URL28] Moving Picture Experts Group. "*Information technology --*

Coding of audio-visual objects". ISO/IEC, International organization for standaritation / International Electrotechnical Commission.
<http://www.iso.org/iso/en/CombinedQueryResult.CombinedQueryResult?queryString=MPEG-4>
[Consulta: 22 dic 2004]

[URL29] The 3rd Generation Partnership Project. "*AMR speech Codec; General description*". The 3rd Generation Partnership Project.
<http://www.3gpp.org/specs/htmlinfo/26071.htm>
[Consulta: 22 abr 2003]

CAPÍTULO 8

ANEXOS

ANEXO I: ARQUITECTURA SAPI

El sistema SAPI está compuesto por los 8 componentes que se ven en la figura II-1.

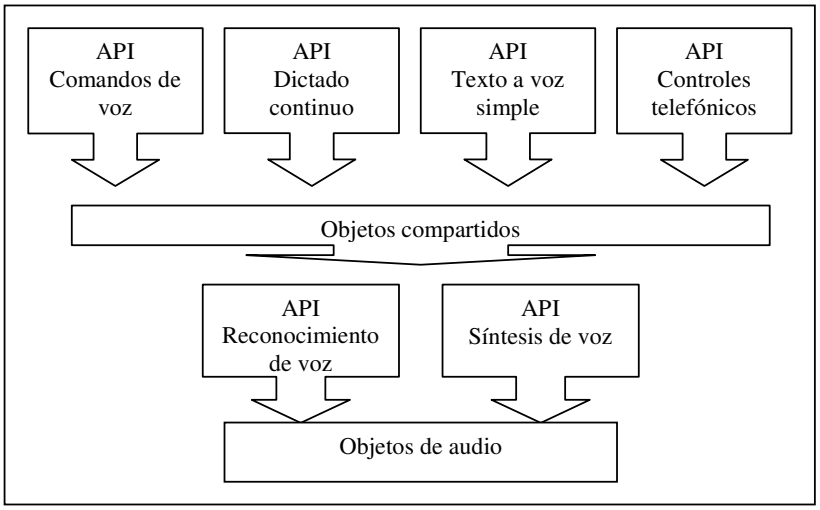


Figura I-1: *Arquitectura SAPI.*

1. Comandos de voz (alto-nivel): comandos precisos e identificables por la aplicación como una orden o respuesta a un evento.

2. Dictado continuo (alto-nivel): componente que se encarga de escribir, lo hablado por el locutor, en cuadros de texto. Da formato a lo reconocido.

3. Texto a voz simple (alto-nivel): funciones básicas para reproducir texto, de una aplicación basada en Windows, hacia un dispositivo de audio.

4. Controles telefónicos (alto-nivel): componentes gráficos que contienen subrutinas aplicables a sistemas de síntesis y reconocimiento telefónico.

5. Objetos compartidos: permite usar los motores de síntesis y reconocimiento a los objetos de alto nivel.

6. Reconocimiento de voz directo (bajo-nivel): componentes que tienen acceso directo a los motores de reconocimiento, proveen mayor control.

7. Síntesis de voz directa (bajo-nivel): componentes con acceso directo a los motores de síntesis de voz.

8. Objetos destinos y objetos fuentes de audio (bajo-nivel): objetos que permiten a los desarrolladores acceder a los motores y utilizarlos en sus aplicaciones.

ANEXO II: ARQUITECTURA NUANCE

El sistema *Nuance* consta de 9 componentes principales (fig. I-1):

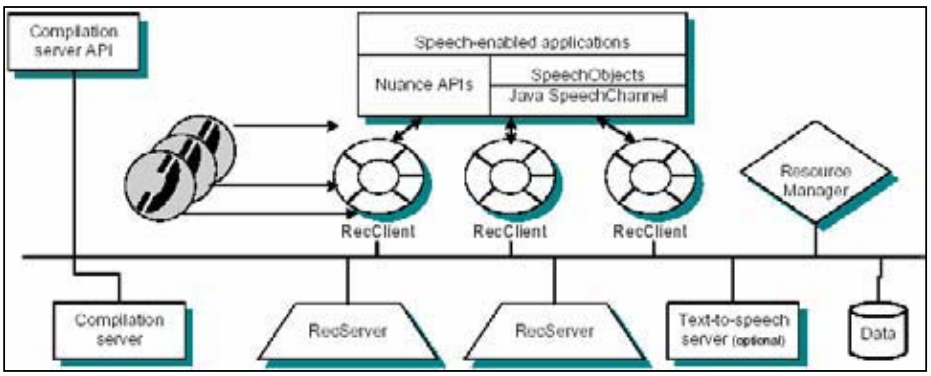


Figura II-1: *Arquitectura Nuance* [NUA01]

1. APIs Nuance: APIs que permiten a los desarrolladores crear aplicaciones que interactúen con los demás componentes del sistema.

2. SpeechObjects y Java SpeechChannel: clases de Java para reconocimiento y funciones de audio.

3. RecClient (Recognition Client): maneja la interacción entre una aplicación y el sistema *Nuance*. Controla la entrada y salida de audio además de soportar parte del control telefónico.

4. RecServer (Recognition Server): reconoce la voz y el lenguaje natural que proviene de los componentes RecClient. Emplea modelos acústicos y gramáticas para identificar correctamente lo que se habló. Usado junto a Nuance Verifier puede identificar locutores.

5. Administrador de recursos: provee balanceo de carga, en tiempo real, para cada RecServer existente en el sistema.

6. Bases de datos: almacenan gramáticas dinámicas y perfiles de usuario. *Nuance* soporta bases de datos *Oracle* y *ODBC*.

7. Servidor de compilación: compila gramáticas dinámicas.

8. Administrador de licencia: controla las máquinas que están bajo una misma licencia.

9. Servidor TTS: enlace entre los componentes RecClient y software TTS externos al sistema *Nuance*.

ANEXO III: CÓDIGOS PARA PETICIONES Y RESPUESTAS EN SESIONES SIP

Métodos para peticiones SIP

ACK: confirma que el cliente ha recibido la respuesta final para la solicitud INVITE [URL17].

BYE: indica que el usuario terminará la sesión [URL17].

CANCEL: cancela una petición previa [URL17].

INFO: envía información de la capa de aplicación opcional, por ejemplo datos que no pertenezcan a streams [RFC2976]

INVITE: el usuario o servicio está siendo invitado para participar en una sesión [URL17].

MESSAGE: extensión de SIP para transportar mensajes instantáneos [RFC3428].

NOTIFY: notificación de eventos relacionados al estado de un recurso [RFC3265]

OPTIONS: rescata información para los UA acerca de otros UA o servidores proxy [URL17].

PRACK: desempeña el mismo papel que ACK pero para proveer información sobre el progreso de una solicitud [RFC3262].

PUBLISH: publica estados de eventos [RFC3903].

REFER: transfiere hacia la ubicación real del recurso [RFC3515].

REGISTER: registra la dirección del servidor SIP que está en el campo de cabecera "To" [URL17].

SUBSCRIBE: solicita el estado y actualizaciones de estado de nodos remotos [RFC3265].

UPDATE: provee información actualizada de la sesión [RFC3311].

Códigos de respuesta

La categoría de los mensajes de respuesta está dada por su primer dígito. Las categorías son:

1xx: provisional, petición recibida, se procesará.

2xx: satisfactoria, la acción fue recibida satisfactoriamente, comprendida y aceptada.

3xx: redirección, se necesita otra acción para llevar a cabo la petición.

4xx: error de cliente, la solicitud tiene una mala sintaxis o no puede ser cumplida en el servidor.

5xx: error de servidor, falla en el servidor para completar una solicitud aparentemente válida.

6xx: falla global, la solicitud no puede ser cumplida en ningún servidor.

ANEXO IV: CAMPOS DE UN PAQUETE RTP

Versión (V): 2 bits

Este campo indica la versión de RTP. La versión definida por esta especificación es dos (2). El valor 1 se usó en la primera versión de RTP y el valor 0 se usó inicialmente en el protocolo de la herramienta de audio "vat".

Padding (P): 1 bit

Si este bit está fijado, el paquete contiene uno o más octetos adicionales de padding al final y el cual no es parte de la carga. El último objeto padding contiene la cantidad de octetos que deben ignorarse. El padding puede necesitarse por algunos algoritmos de encriptación o para transportar varios paquetes RTP en un protocolo de capa más baja.

Extension (X): 1 bit

Si se fija este bit, la cabecera es seguida por una extensión de cabecera.

CRC count (CC): 4 bits

Contiene el número de identificadores CSRC que siguen a la cabecera.

Marker (M): 1 bit

Su interpretación está definida por un perfil que está pensado para permitir eventos significativos como límites de frames que son marcados en una cadena de paquetes. Un perfil puede definir bits de marca adicionales o especificar, si no lo son, cambiando el numero de bits en el campo payload type.

Payload Type (PT): 7 bits

Especifica el formato de la carga y determina su interpretación para la aplicación. Códigos adicionales de tipo de carga pueden definirse automáticamente por medio de recursos que no sean RTP.

Secuence number: 16 bits

Secuence number se incrementa en uno por cada paquete de datos RTP enviado, puede usarce por el receptor para detetctar paquetes perdidos y reparar la secuencia. El valor inicial del Secuence number es aleatorio para dificultar los ataques al encriptado que aunque la fuente no lo hace, porque los paquetes pueden pasar por un repetidor que si lo haga.

Timestamp: 32 bits

Refleja el instante de muestreo del primer byte del paquete de datos RTP. El instante de muestreo debe derivarse de un reloj que incrementa monótona y linealmente para permitir sincronización y calculos de jitter. La resolución del reloj debe ser suficiente para la precisión de sincronización deseada y para medir jitter de los paquetes que llegan**.** La frecuencia del reloj depende del formato de los datos transportados como carga y está especificado en el perfil o en la especificación del formato de carga, o puede ser especificado dinámicamente para formatos de carga que no son RTP definidos por recursos que no sean RTP.

El valor inicial del timestamp es aleatorio. Varios paquetes RTP consecutivos pueden tener timestamps iguales si son generados al mismo tiempo, lo cual puede suceder si por ejemplo pertenecen al mismo frame de video. Paquetes RTP consecutivos pueden contener timestamps que no son monótonos si los datos no son transmitidos en el orden que se extrajeron las muestras, como es en el caso de frames de video MPEG interpolados.

SSRC: 32 bits

Identifica la fuente de sincronización, este identificador se escoge aleatoriamente con el propósito de que no hayan dos fuentes de sincronización que tengan el mismo identificador SSRC dentro de la misma sesión RTP.

CSRC list: 0-15 items cada uno de 32 bits

Identifica las fuentes contribuyentes para la carga contenida en el paquete. El número de identificadores está dado por el campo CC. Los identificadores CSRC son insertados por mixers usando los identificadores SSRC de las fuentes contribuyentes.

ANEXO V: FORMATOS SOPORTADOS POR JMF v 2.1.1 [SUN04]

JMF puede transmitir y recibir distintos formatos de audio y video sobre RTP.

Donde:

- **R:** recibe datos en el formato indicado.
- **T:** transmite datos en el formato indicado.

FORMATO	RTP Payload	JMF 2.1.1 Cross Platform Versión	JMF 2.1.1 Solaris/Linux Performance Pack	JMF 2.1.1 Windows Performance Pack
Audio: G.711 (ley-μ) 8 Khz	0	R,T	R,T	R,T
Audio: GSM mono	3	R,T	R,T	R,T
Audio: G.723 mono	4	R	R,T	R,T
Audio: 4-bit mono DVI 8 Khz	5	R,T	R,T	R,T
Audio: 4-bit mono DVI 11.025 Khz	16	R,T	R,T	R,T
Audio: 4-bit mono DVI 22.05 Khz.	17	R,T	R,T	R,T
Audio: MPEG Layer I, II	14	R,T	R,T	R,T
Video: JPEG (420, 422, 444)*	26	R	R,T	R,T
Video: H.261	31	-	R	R
Video: H.263	34	Modo A	R,T	R,T
Video: MPEG-I	32	T	R,T	R,T

* JPEG/RTP solo puede transmitirse en dimensiones de video múltiplos de 8 píxeles.

** H.263/RTP solo puede ser transmitido en tres dimensiones diferentes: SQCIF (128x96), QCIF (176x144), CIF (352x288).

*** MPEG/RTP solo puede ser transmitido desde contenido MPEG pre-codificado, es decir un archivo MPEG, o una fuente que capture en formato MPEG. La codificación de MPEG por software en tiempo real no es posible para transmisión RTP.

ANEXO VI: CONEXIÓN A SERVICIOS WEB CON CLIENTES SOAP

Para crear un cliente SOAP que se conecte a un servicio web (como el servicio de traducción *BabelFish*) es necesario añadir una referencia web en el proyecto *VC++.NET*. La figura VI-1 muestra el menú y la opción correspondiente.

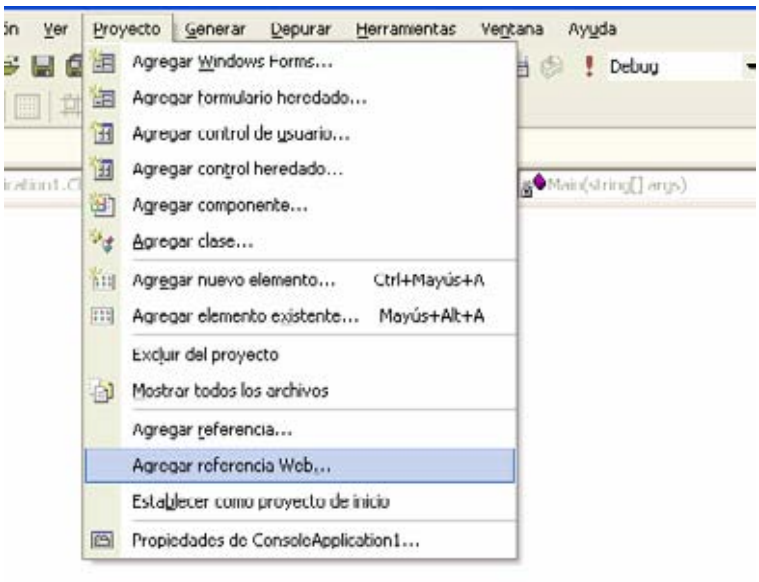


Figura VI-1: *Opción "Agregar referencia Web".*

Debe indicarse la URL del servicio web (fig. VI-2). En este caso, es la ruta completa al archivo que contiene la descripción y métodos de acceso al servicio *BabelFish*.

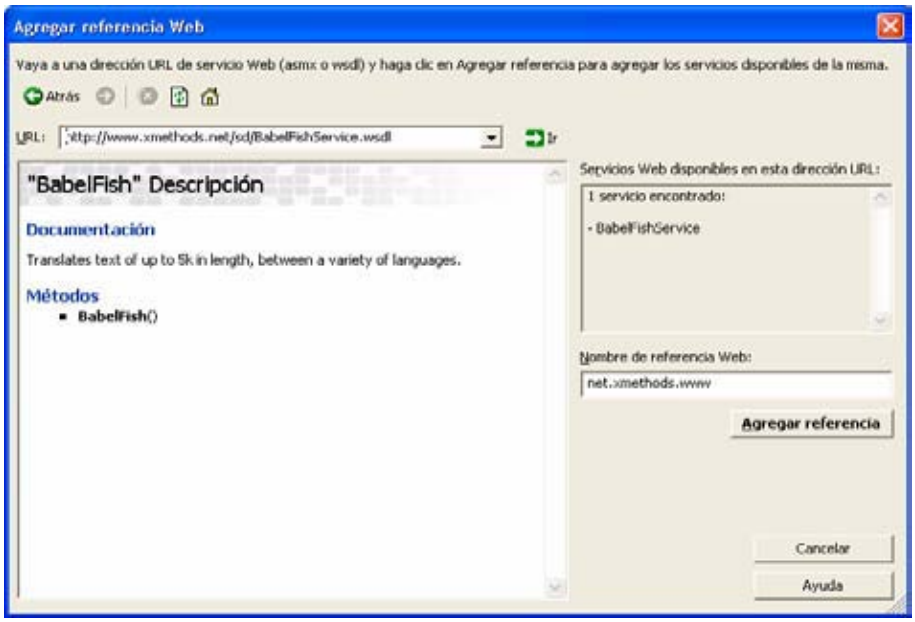


Figura VI-2: *Opción "Agregar referencia Web".*

Una vez agregada la referencia, el desarrollador puede crear el espacio de nombres y emplear los métodos que ofrece el servicio:

```
using namespace BabelFish;
CoInitialize(NULL);
CBabelFish *fish= new CBabelFish();
HRESULT hr = fish->BabelFish(bstrMode,bstrSource,&bstrResult);
```

ANEXO VII: INVOCAR CÓDIGO VC# EN VC++ .NET

Para usar la clase *WebWagon* C# en *VC++ .NET*, el código se adaptó y exportó como una biblioteca de tipos. El ejemplo siguiente, describe el formato de una aplicación exportable.

Considere el código *VC#*:

```
// Clases fundamentales
using System;
// Interoperabilidad COM
using System.Runtime.InteropServices;

// Tipo global
namespace Ejemplo
{
      // Expone la interfáz para uso COM
      [InterfaceType(ComInterfaceType.InterfaceIsIDispatch)]
      public interface miInterfaz
      {
            // Métodos expuestos
            [DispId(1)]
            int metodo1();
            [DispId(2)]
            int metodo2();
            .
            .
            .
            [DispId(N)]
            string metodoN();
      }
      // Habilita el acceso a la clase solamente por la intefáz
      [ClassInterface(ClassInterfaceType.None)]
      public class miEjemplo : miInterfaz
      {
            // Implementación de los métodos
            public miEjemplo() {}
            public int metodo1
            {
                  // Implementación
            }
            public int metodo2
            {
                  // Implementación
            }
            .
            .
            .
            public string metodoN
            {
                  // Implementación
            }
      }
}
```

Luego, hay que habilitar la interoperabilidad COM a través de las propiedades del proyecto (fig. VII-1)

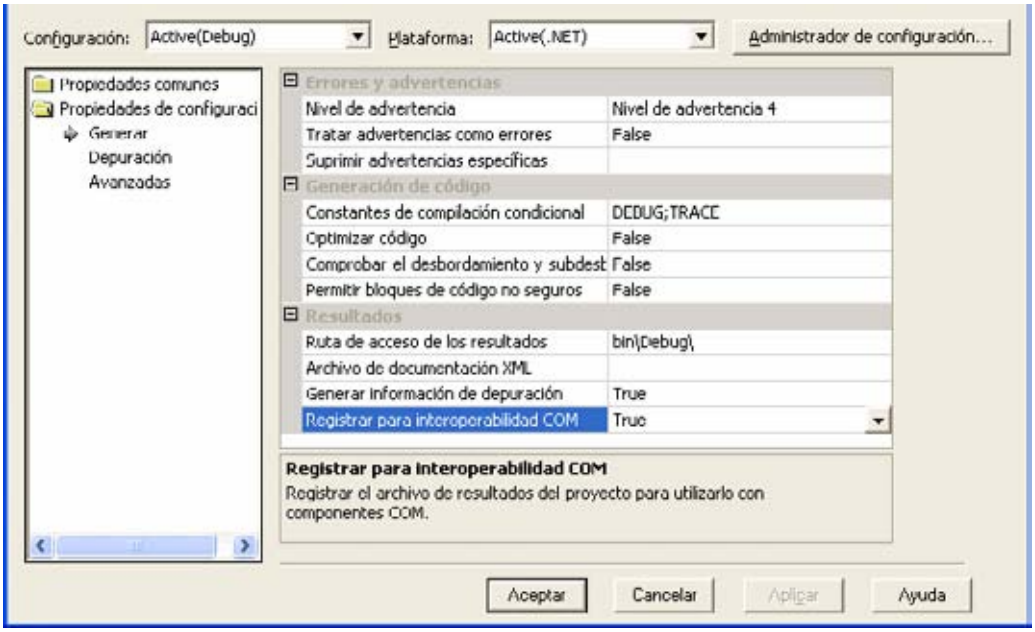


Figura VII-1: *Interoperabilidad COM en VC#.*

Finalmente, al compilar la solución, se genera un archivo de tipos que conserva el nombre del proyecto, *Ejemplo.tlb*. Este archivo contiene la definición completa del código *VC#*, el cual se invoca desde *VC++ .NET* como un componente COM:

```
#import "Ejemplo.tlb"
using namespace Ejemplo;

//Interfáz COM a "Ejemplo"
miInterfazPtr ifzEjemplo(__uuidof(miEjemplo));
```

ANEXO VIII: CÓMO USAR EL PROTOTIPO EFICIENTEMENTE

A continuación se describen los pasos previos para ejecutar eficientemente una instancia del prototipo desde la instalación, y haciendo uso del motor de reconocimiento en español *Dragon NaturallySpeaking Spanish* y el motor de reconocimiento en inglés *English Continuous Microsoft SAPI SDK* .

Instalación del prototipo

1. Inserte el CD de la tesis en el lector del equipo. Asumiendo que su unidad de CD tiene asignada la letra "**D**", acceda al directorio **D:\Software\Requisitos**. Encontrará los siguientes directorios, que corresponden a los requerimientos funcionales del prototipo:

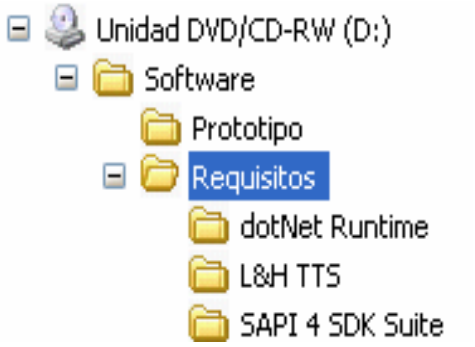


Figura VIII-1: *Directorio de requerimientos funcionales.*

2. Instale primero las bibliotecas para ejecución de aplicaciones desarrolladas en .Net. Para ello, ingrese a la carpeta **\dotNet Runtime** y ejecute el archivo "**dotnetfx.exe**".

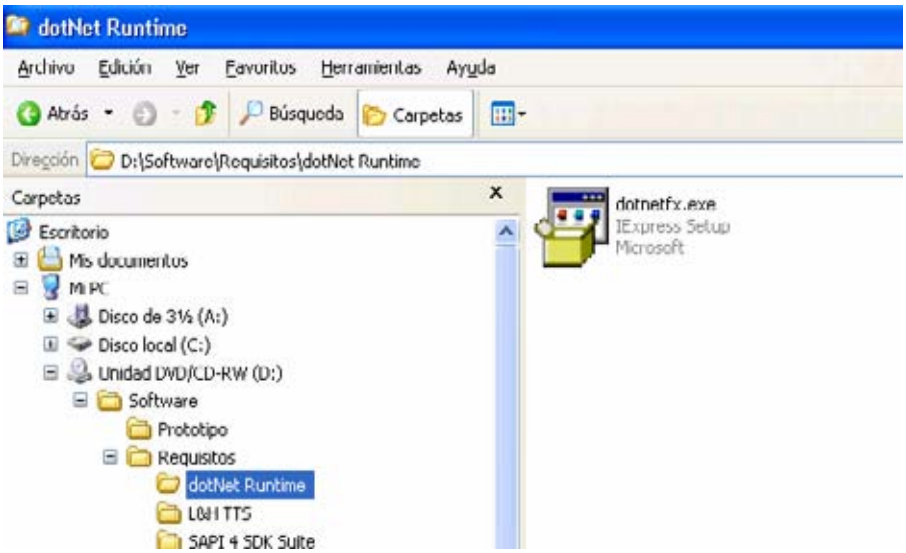


Figura VIII-2: *Bibliotecas para aplicaciones .Net*

3. Ahora instalará las bibliotecas para ejecución de aplicaciones SAPI y los motores Microsoft para reconocimiento y síntesis en inglés. Acceda "**\SAPI 4 SDK Suite**". Ejecute (doble clic) el programa "**SAPI4SDKSUITE.exe**".

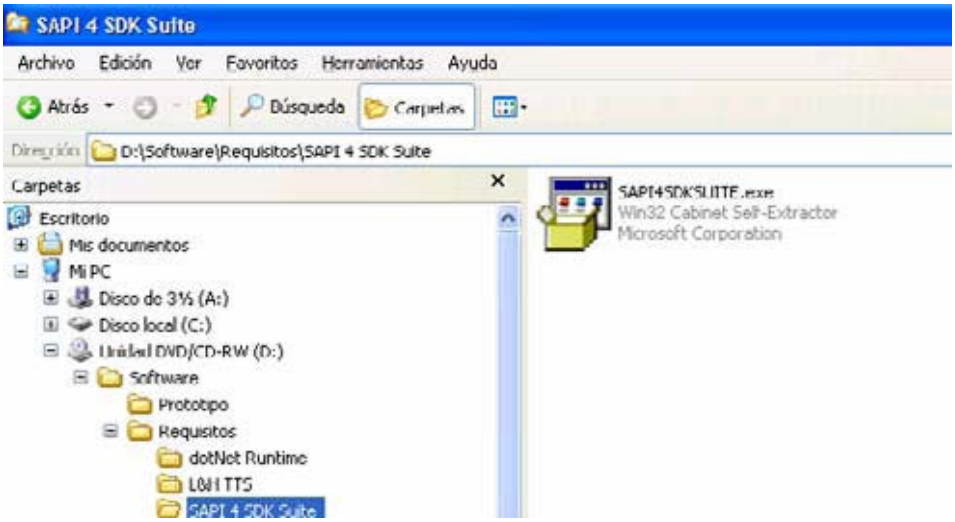


Figura VIII-3: *Bibliotecas y motores SAPI*

4. Proceda a instalar los motores Lernout&Hauspie para síntesis en español. Ingrese al directorio "**\L&H TTS**" y ejecute el programa "**lhttsspe(español).exe**".

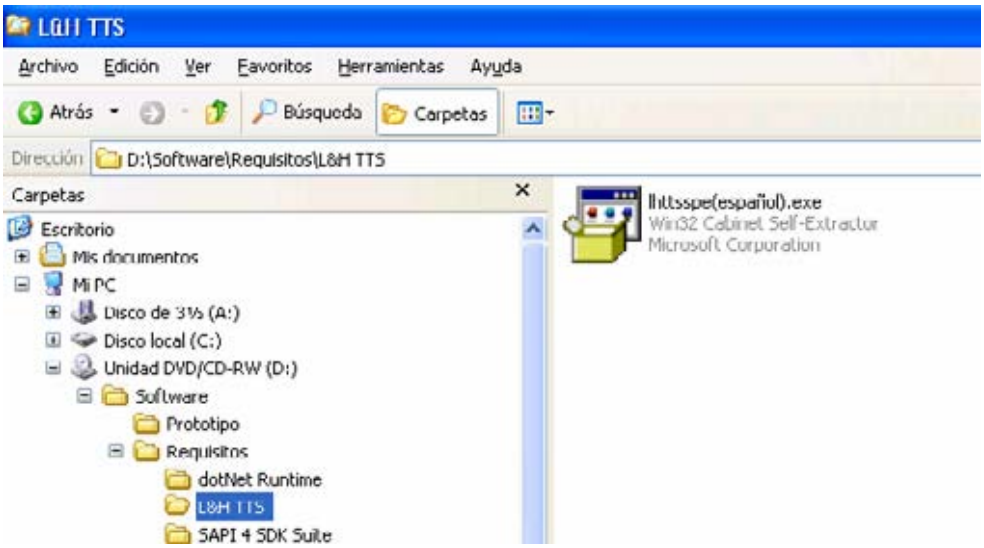


Figura VIII-4: *Motor Lernout&Hauspie para síntesis en español.*

5. Finalmente, ejecute la instalación del prototipo ubicada en "**D:\Software\Prototipo**". Ejecute "**Setup Intérprete 1.0.exe**", un asistente lo guiará durante el proceso de instalación.

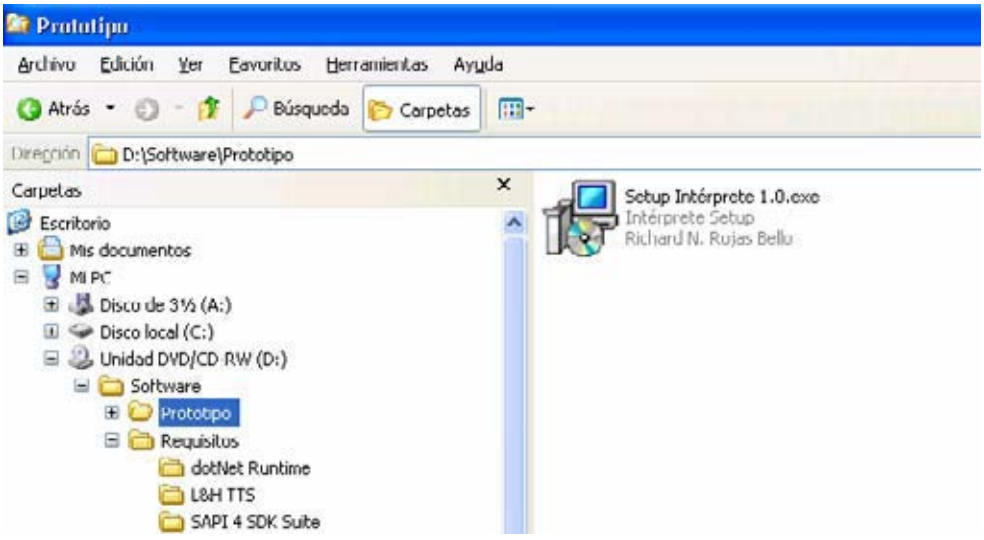


Figura VIII-5: *Archivo de instalación del prototipo.*

Figura VIII-6: *Asistente de instalación.*

La parte final del asistente le indicará que la instalación ha finalizado y que las bibliotecas para transmisión en tiempo real (WinRTP) se han registrado en el sistema operativo.

Figura VIII-7: *Registro de bibliotecas WinRTP.*

Puede iniciar el prototipo desde el mismo cuadro de diálogo, o desde la ubicación indicada por usted en el transcurso de la instalación. Sin embargo, se recomienda entrenar previamente los motores de reconocimiento instalados en el sistema operativo.

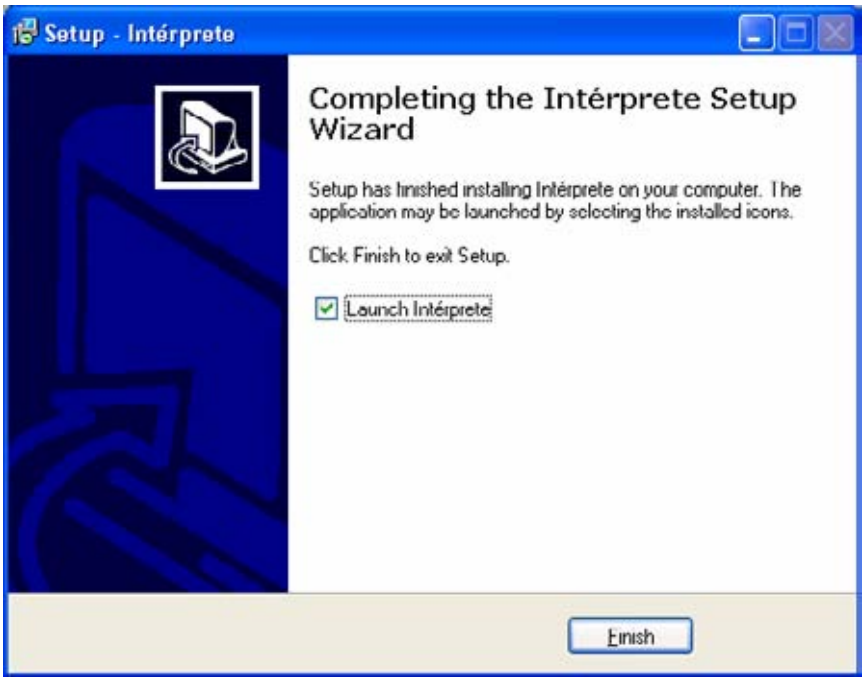


Figura VIII-8: *Fin de la instalación.*

Entrenamiento de los motores

Procure que su micrófono sea de buena calidad y que el entorno en que se encuentra tenga el menor ruido ambiente posible, así maximizará la precisión del reconocimiento.

Mientras más sesiones de entrenamiento se ejecuten, más fiel será el proceso de captura de la voz.

Si tiene alguna instancia del prototipo en ejecución, ciérrela.

“Dragon NaturallySpeaking”

1. Inicie *Dragon NaturallySpeaking Spanish* desde ***Menú Inicio->Todos los programas->Dragon NaturallySpeaking***. Si ya ha creado un usuario y desea solamente entrenarlo, puede continuar esta guía desde el punto 9.

2. En la ventana principal “*Administrar usuarios*” cree un nuevo usuario pinchando en el botón “*Nuevo…*”. Un asistente lo guiará en el proceso de creación.

Figura VIII-9: *Administración de usuarios.*

3. Indique su nombre de usuario (por ejemplo *“jperez*”), idioma (inglés o español) y fuente de dictado. Luego haga clic en el botón “*Siguiente*”. Los puntos que siguen asumen la elección de idioma español.

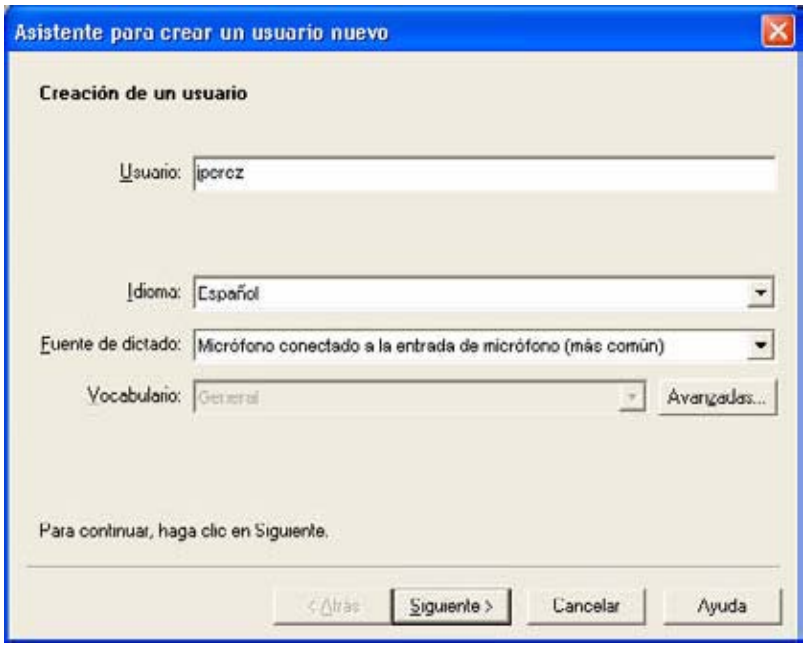


Figura VIII-10: *Datos del nuevo usuario.*

4. Encienda o habilite su micrófono. Lea el texto en voz alta luego de presionar el botón "*Iniciar prueba del volumen*".

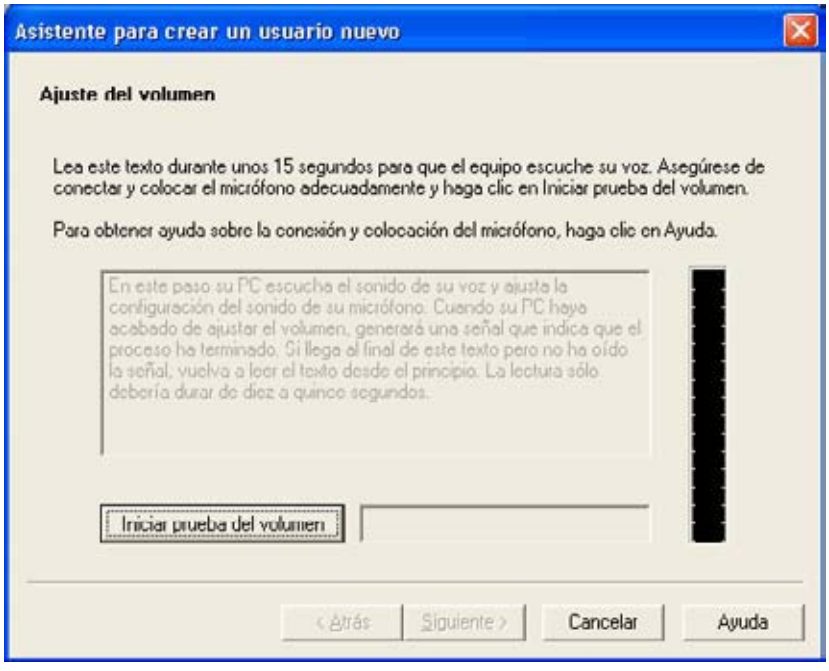


Figura VIII-11: *Ajuste de volumen.*

Terminado el ajuste, haga clic en el botón "*Siguiente*".

5. Una nueva prueba de audio verificará la calidad del micrófono y el nivel de ruido en el ambiente. Siga las instrucciones del asistente y al finalizar haga clic en "*Siguiente*".

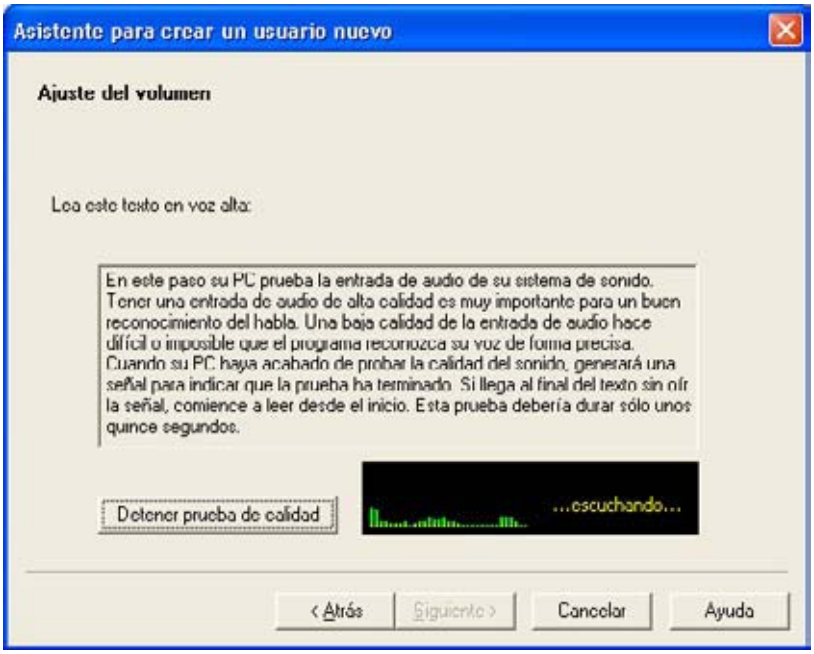


Figura VIII-12: *Detección de ruido.*

6. Empiece el entrenamiento del usuario. Haga clic en "*Comenzar*" y lea naturalmente y en voz alta el texto presentado.

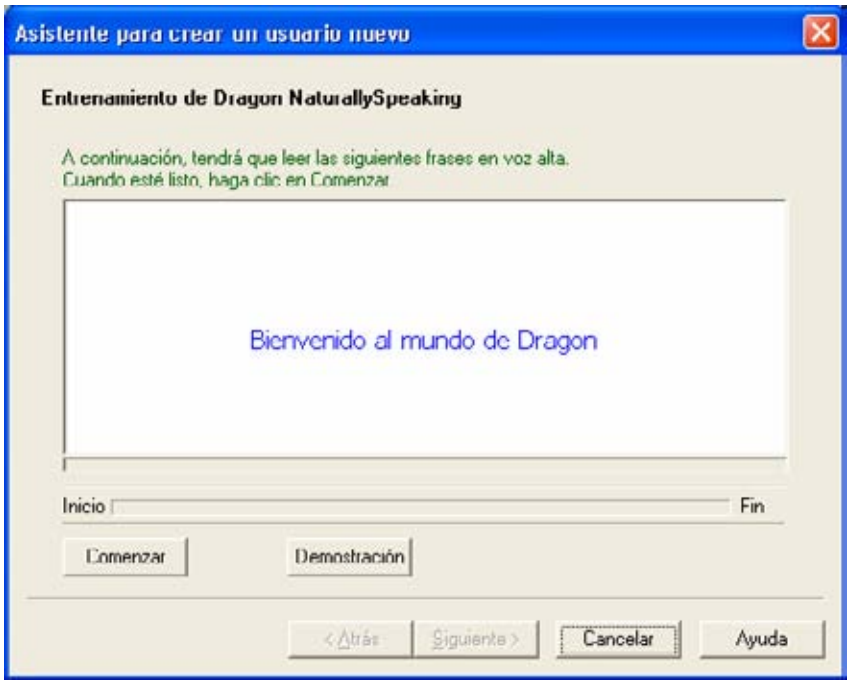


Figura VIII-13: *Etapa de entrenamiento.*

7. Cuando termine de leer el texto, deberá escoger entre distintos fragmentos para entrenamiento y continuar la lectura en voz alta.

Figura VIII-14: *Textos disponibles para entrenamiento.*

8. Al finalizar, ***Dragon NaturallySpeaking*** almacenará las características de su forma de hablar en un perfil con su nombre de usuario.

Figura VIII-15: *Fin del entrenamiento.*

9. Ahora ya existe un usuario *"jperez*" registrado en *Dragon NaturallySpeaking*. Seleccione el usuario y haga clic en "*Abrir*".

Figura VIII-16: *Selección de un usuario.*

10. Puede iniciar nuevamente el asistente haciendo clic en ***NaturallySpeaking->Avanzadas->Entrenar usuario…***

Si por alguna razón debe cambiar su ubicación física, es recomendable que entrene nuevamente el motor o acceda al menú ***NaturallySpeaking->Avanzadas->Comprobar calidad de sonido…*** para medir el nuevo nivel de ruido.

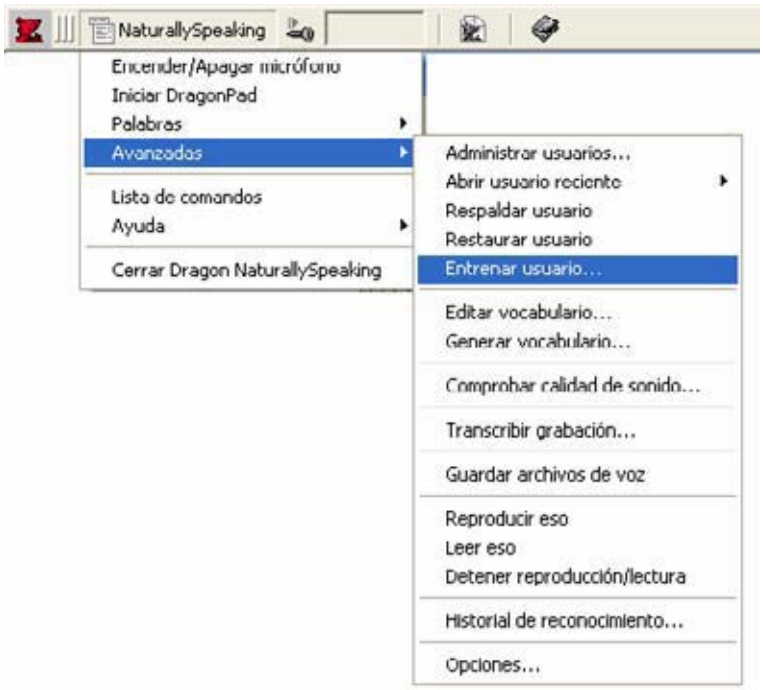


Figura VIII-17: *Reiniciar asistente de entrenamiento.*

11. Al finalizar el entrenamiento, cierre *Dragon NaturallySpeaking* y ejecute el prototipo de la tesis desde el menú *Inicio->Todos los programas->Intérprete 1.0* o desde el menú que haya indicado al instalarlo.

12. Como nombre de usuario, ingrese el nombre de usuario creado en *NaturallySpeaking*, motor de reconocimiento *Dragon Spanish NaturallySpeaking*, e idioma *Español*. Luego haga clic en "*Aceptar*".

Figura VIII-18: *Acceso del nuevo usuario.*

Un mensaje corroborará que su nombre de usuario fue registrado y ya tiene un perfil creado.

Figura VIII-19: *Mensaje de perfil existente.*

13. Continúe con la ejecución del prototipo en forma normal.

"English Continuous Microsoft SAPI SDK"

Con *English Continuous Microsoft SAPI SDK* los usuario se crean y se entrenan por medio de la aplicación que invoca al motor, en este caso es el prototipo *Intérprete 1.0.exe.*

1. Ejecute el prototipo desde el menú *Inicio->Todos los programas->Intérprete 1.0* o desde el menú que haya indicado al instalarlo.

2. Ingrese un nombre de usuario (por ejemplo "*tesis*", que es el nombre por defecto), seleccione el motor de reconocimiento *English Continuous Microphone (Microsoft)*, e idioma *Inglés*. Luego haga clic en "*Aceptar*".

Figura VIII-20: *Creación de un usuario nuevo.*

Un mensaje le informará que su nombre se registró exitosamente y desde ahora cuenta con un perfil creado.

Figura VIII-21: *Mensaje de usuario nuevo creado satisfactoriamente*

Haga clic en aceptar.

3. En la parte inferior del nuevo cuadro de diálogo, haga clic en "*Entrenar*".

Figura VIII-22: *Opciones del prototipo*

4. Escoja uno de los distintos fragmentos disponibles, haga clic en "*Siguiente*".

Figura VIII-23: *Textos disponibles en English Continuous*

5. Un asiste lo guiará a lo largo del entrenamiento. Siga sus instrucciones y haga clic en "*Siguiente*".

Figura VIII-24: *Asistente de entrenamiento English Continuous.*

6. Ahora comienza la etapa de entrenamiento. Lea en voz alta y naturalmente el texto presentado.

Figura VIII-25: ***Entrenamiento*** *English Continuous.*

7. Cuando el texto finalice, tendrá la opción de entrenar nuevamente el motor o terminar con el entrenamiento y volver al prototipo.

Figura VIII-26: *Repetición de entrenamiento.*