

UNIVERSIDAD AUTONOMA DE BAJA CALIFORNIA

FACULTAD DE CIENCIAS



**DISEÑO E IMPLEMENTACION DEL SERVIDOR DE AUDIO
DE LA SALA ELECTRONICA DE REUNIONES DEL CICESE.**

TESIS

Que para obtener el título de

Licenciado en Ciencias Computacionales

presenta:

JOSE MANUEL ALBA ALVAREZ

Ensenada, Baja California. Enero del 2000.

UNIVERSIDAD AUTONOMA DE BAJA CALIFORNIA

FACULTAD DE CIENCIAS

DISEÑO E IMPLEMENTACION DEL SERVIDOR DE AUDIO DE LA SALA
ELECTRONICA DE REUNIONES DEL CICESE.

TESIS PROFESIONAL

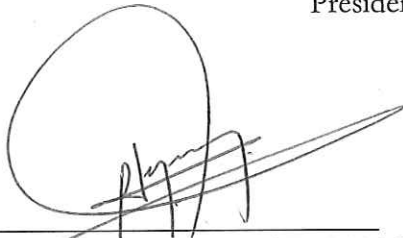
QUE PRESENTA

JOSE MANUEL ALBA ALVAREZ

APROBADO POR:



DR. JESUS FAVELA VARA
Presidente del jurado



M.C. ALBERTO LEOPOLDO MORAN Y SOLARES
SECRETARIO



M.C. MARIA VICTORIA MEZA KUBO
1er. VOCAL

Dedicatoria.

A mis padres y mi hermana: José Manuel, Anita y Leslie.

A mis abuelos.

Agradecimientos.

A mi director de tesis, Dr. Jesús Favela Vara, por la oportunidad y el apoyo para la realización de este trabajo.

A mi familia por mostrarme el camino y apoyarme en todo momento.

A los maestros y personal de la Universidad Autónoma de Baja California, especialmente a los que laboran en la Facultad de Ciencias, por haberme dado la oportunidad de formarme profesionalmente.

A mis compañeros de generación, matemáticos incluidos, con quienes compartí una de las etapas más felices y educativas de mi vida, gracias por haber estado ahí conmigo. Gracias gente Reltop, gracias hermanos.

Al CICESE y sus estudiantes de la Maestría en Ciencias de la Computación con los que participe en la elaboración de este trabajo, Marcela Rodríguez y Francisco Garcilazo, gracias por su ayuda. A los estudiantes del CICESE que me brindaron su amistad, especialmente al grupo “The Web Ones”.

Al creador del universo, cualquiera que sea su forma.

RESUMEN de la Tesis de José Manuel Alba Alvarez presentada como requisito parcial para la obtención de la Licenciatura en Ciencias Computacionales. Ensenada, Baja California, México. Enero del 2000.

DISEÑO E IMPLEMENTACION DEL SERVIDOR DE AUDIO DE LA SALA ELECTRONICA DE REUNIONES DEL CICESE.

Resumen aprobado:


DR. JESUS FAVELA VARA
Director de tesis

El trabajo en grupo ha sido siempre una actividad muy importante. Con el avance en el área de las telecomunicaciones y la computación ha surgido el Trabajo Cooperativo Asistido por Computadora (CSCW), el área que estudia como utilizar sistemas computacionales para facilitar la colaboración entre los miembros de un grupo de trabajo. Las herramientas que sirven de apoyo al CSCW han propiciado además la formación de grupos de trabajo distribuidos en el tiempo y en el espacio. Una de estas herramientas es la Sala Electrónica de Reuniones, una sala equipada con computadoras y otros dispositivos a través de los cuales un grupo de personas puede trabajar de manera colaborativa.

Debido a las características y necesidades del CICESE, una Sala Electrónica fue construida dentro del área de Ciencias de la Computación, donde el grupo de trabajo en CSCW decidió construir el software de control para la sala. Uno de los módulos del software de control es un servidor de audio. En este trabajo se documenta el desarrollo del diseño y la implementación de un servidor de audio, con su respectiva aplicación cliente, para la Sala de Reuniones Electrónicas del CICESE. El objetivo de este módulo del software de control de la sala es registrar las participaciones de los asistentes a una reunión y notificar al resto de los módulos cual es el participante que se encuentra hablando, para que sea enfocado de manera automática por una de las cámaras de la sala. En este trabajo se describen además los componentes de la Sala Electrónica, el formato de audio utilizado para la aplicación, se muestran los resultados obtenidos al utilizar dos protocolos diferentes para la transmisión de datos de audio. Además, se exploran las ventajas de la utilización de un protocolo social, en vez de un protocolo tecnológico, como mecanismo de control de piso.

Contenido.

	Página
Capítulo 1. Introducción.	1
1.1 Antecedentes.	1
1.2 Planteamiento del problema.	5
1.3 Objetivo.	6
1.4 Organización de la tesis.	6
Capítulo 2. Metodología utilizada.	8
Capítulo 3. Audio en Sistemas Cooperativos.	11
3.1 Trabajo Cooperativo Asistido por Computadora.	11
3.2 Sistemas Cooperativos, Groupware.	13
3.3 Reuniones Electrónicas.	15
3.3.1 Sistemas Electrónicos de Reuniones.	16
3.3.2 Salas de Reuniones Electrónicas.	18
3.3.2.1 Pantalla Pública	19
3.3.2.2 Configuración de los lugares de trabajo.	20
3.3.2.3 Iluminación y ruido.	21

3.3.2.4 Software para las Salas de Reuniones Electrónicas.	21
3.3.2.5 Otras consideraciones.	23
3.4 Utilización y rol del audio en aplicaciones para CSCW.	24
3.5 Registro de las sesiones de audio en las Salas de Reuniones Electrónicas, Memoria Organizacional.	25
Capítulo 4. La Sala de Reuniones Electrónicas del CICESE.	27
4.1 Escenarios de utilización.	27
4.2 Equipo de la Sala de Reuniones Electrónicas del CICESE.	29
4.3 Software de control para la Sala de Reuniones.	32
4.3.1 Aplicación Servidor.	33
4.3.2 Aplicación Cliente.	35
4.4 Servidor de audio de la Sala de Reuniones Electrónicas del CICESE.	36
Capítulo 5. Diseño e implementación del servidor de audio.	39
5.1 Diseño de la aplicación para el manejo de audio.	39
5.2 Implementación.	43
5.3 Aplicación servidor.	44
5.3.1 Control de piso.	46

5.3.2 Formato de los datos de audio.	47
5.4 Aplicación cliente.	49
5.4.1 El Medical Collaboration Project de la Universidad de Michigan	50
5.4.2 La clase SoundBite.	51
5.4.3 SoundBiteSenderUDP y WriterThreadUDP.	51
5.4.4 Detección de voz.	52
5.5 Comunicación Cliente-Servidor.	54
5.5.1 Conexión.	55
5.5.2 Transmisión de datos.	56
5.6 Seguridad.	60
5.7 Configuración.	60
5.8 Utilización.	61
Capítulo 6. Pruebas y resultados.	62
6.1 Pruebas realizadas al servidor y al cliente de audio.	62
6.1.1 Pruebas al software de control de la Sala de Reuniones Electrónicas.	66
6.2 Calidad del audio registrado.	66
6.2.1 Valores de silencio absoluto.	67

6.2.2 Valores umbral de detección de voz.	67
6.3 Longitud de los archivos de audio.	68
Capítulo 7. Discusión.	69
7.1 Funcionamiento.	69
7.2 Protocolo de transporte de datos.	70
7.3 Manejo de control de piso.	71
7.4 Reutilización del módulo de manejo de audio de la Sala de Reuniones Electrónicas.	71
7.5 Otras consideraciones.	72
7.6 Trabajo futuro.	73
Capítulo 6. Conclusiones.	74
Literatura Citada.	77
Apéndice A. Tabla comparativa de algunos formatos utilizados en la compresión de audio.	80

Figuras.

	Página
Figura 1. Clasificación de aplicaciones Groupware.	14
Figura 2. Distribución y conexiones del equipo utilizado en la Sala de Reuniones Electrónicas del CICESE.	32
Figura 3. Ejemplo de una sesión de control de equipo con el software de control de la Sala de Reuniones Electrónicas.	34
Figura 4. Configuración del conmutador de video.	35
Figura 5. Ventana desplegada para incorporarse a una sesión de trabajo.	36
Figura 6. Diagrama de clases UML del servidor de audio y su relación con el software de control de la Sala de Reuniones Electrónicas.	40
Figura 7. Diagrama de clases de la aplicación cliente del software de control de la sala.	41
Figura 8. Arquitectura del módulo de audio del software de control de la Sala de Reuniones Electrónicas.	43
Figura 9. Encabezado de los datos de audio manejados por el servidor de audio.	48
Figura 10. Los datos de audio dentro de la zona gris son considerados como silencio.	53
Figura 11. Interfaz de la aplicación para probar de manera aislada el sistema de audio (servidor).	63

Capítulo 1. Introducción.

1.1 Antecedentes.

Durante miles de años el ser humano se comunicó únicamente a través de señas y sonidos. Esos sonidos se fueron convirtiendo, de manera gradual, en lo que hoy conocemos como el habla. A pesar de lo efectivo del habla en comparación a los primeros métodos de expresión, para establecer una comunicación entre dos seres humanos era necesario que los participantes de la conversación estuvieran a un mismo tiempo en un lugar común para que el intercambio de información se diera de manera efectiva.

Las limitantes de espacio y tiempo para el intercambio de experiencias fueron después atenuadas. Conforme las primeras civilizaciones se fueron desarrollando, medios alternativos como la pintura y la escritura permitieron ampliar el alcance de las comunicaciones. Era ya posible plasmar en piedra, y después en pergaminos, documentos oficiales, enviar mensajes, así como escribir poemas y leyendas.

Hoy en día los medios de comunicación siguen siendo utilizados para la diseminación de la información, para el envío de mensajes y como medio de expresión cultural. Sin embargo, en vez de seguir dependiendo de rocas y pergaminos, son utilizadas herramientas más efectivas como la televisión, el teléfono y las redes de computadoras, tecnologías con las cuales es posible comunicarnos a cualquier parte del mundo, siempre y

cuando existan dispositivos disponibles en el origen y destino deseados, casi de manera instantánea. Además de transmitir sonido, es posible transmitir video y datos.

Gran parte del trabajo de una organización actual es hecho por personas trabajando de manera individual. Sin embargo, muchos de los problemas que surgen hoy en día están lejos de ser tareas rutinarias, no pueden ser resueltos por expertos individualistas ya que no hay persona que tenga toda la visión, toda la experiencia, toda la información o toda la inspiración para cumplir con semejante tarea. Es así como nacen los equipos de trabajo. Estos equipos han alcanzado grandes alturas, sorteando problemas insalvables para individuos trabajando por su propia cuenta. Como resultado de trabajar en proyectos comunes, los participantes del grupo suelen reunirse para realizar evaluaciones, intercambiar información, tomar o no tomar decisiones, socializar, etc. (Nunamaker et al., 1995).

A pesar de todas las ventajas que el trabajo en equipo presenta, este está dotado con su propia dosis de problemas (Grudin, 1989). Bass (1980) subraya que en grupos de trabajo muy grandes los patrones de interacción se tornan sumamente complejos. Debido a esto el añadir miembros a un equipo de trabajo no necesariamente implica el aumentar el poder de acción del grupo (Brooks, 1975). Estas situaciones suelen verse acentuadas sobre todo cuando la composición de la agrupación es diferente a la de los grupos de trabajo tradicionales.

El avance en el área de las telecomunicaciones ha hecho cada vez más frecuente el que personas que se encuentran localizadas en lugares geográficamente distantes puedan, además de compartir información, colaborar en proyectos conjuntos dando lugar a los equipos de trabajo distribuidos, en los cuales uno o más miembros se encuentran separados geográficamente del resto del equipo (Haywood, 1998). La separación de los integrantes del grupo de trabajo no está restringido únicamente a los aspectos del tipo geográfico: cuando los horarios de trabajo de los diferentes participantes del equipo son incompatibles, poder reunirse cara-a-cara puede resultar muy difícil, aún cuando todas las personas involucradas radiquen en la misma ciudad. Esto trae como consecuencia que tales grupos de trabajo interactúen de manera diferente a la tradicional. Dicho cambio en la forma de trabajar no es exclusiva de los grupos distribuidos o con problemas de horario, también los grupos de trabajo tradicionales se ven impactados por el uso de procesadores de texto, bases de datos, correo electrónico y otros medios de comunicación electrónicos.

El espacio donde este tipo de grupos se desenvuelve es conocido como el lugar electrónico de trabajo (Electronic Workspace) – el sistema que se extiende a lo largo y ancho de una organización que integra procesamiento de información y actividades de comunicación. El estudio de tales sistemas es parte del campo multidisciplinario del Trabajo Cooperativo Asistido por Computadora (CSCW: Computer Supported Cooperative Work) . Tomando como base la experiencia y colaboración de muchos especialistas, incluyendo sociólogos y científicos de las Ciencias de la Computación, el CSCW estudia como funcionan los grupos de trabajo y busca la forma en que la tecnología (especialmente la relacionada con las computadoras) puede ayudarlos a trabajar (Ellis et al., 1991).

La familia de productos de software utilizados para el soporte de CSCW es comúnmente llamada Groupware (Ellis et al., 1991). Y aunque otros autores definen al Groupware como el software de soporte para la colaboración de grupos pequeños o con un enfoque muy estrecho (Grudin, 1988), Ellis sugiere que el groupware sea visto como la clase de aplicaciones, orientadas tanto a pequeños grupos como a organizaciones completas, que surge de la unión del uso de computadoras, grandes bancos de información y tecnología de comunicaciones.

Uno de los retos que el Groupware impone, de manera más frecuente, a la tecnología de las comunicaciones es como hacer que las interacciones distribuidas sean tan efectivas como la interacción cara-a-cara. Tal vez la perspectiva correcta de este reto sea ver a la interacción remota, tomando como base el uso de tecnología apropiada, como un medio alternativo. Aún cuando este medio alternativo no puede remplazar la comunicación cara-a-cara, este nuevo método puede ser preferido en algunas situaciones para cierto tipo de grupos, ya que con esto es posible minimizar algunas dificultades inherentes al trabajo en equipo. Por ejemplo, las interacciones distribuidas permiten a los participantes acceder otra información relevante, ya sea a través de una terminal o de un libro en un librero, sin interrumpir el flujo de la interacción. Esto es análogo a los descubrimientos en el uso del teléfono, el correo electrónico y otras tecnologías, sin remplazar la interacción cara-a-cara, cada una de estas herramientas ocupa un nicho donde se desenvuelve como un modo de comunicación útil y único. El reto es, entonces, aplicar la combinación tecnológica adecuada

a los diferentes tipos de interacciones que se beneficiarán con el uso del nuevo medio (Ellis et al., 1991).

1.2 Planteamiento del problema.

Una de las combinaciones tecnológicas utilizada como herramienta de soporte para el trabajo en equipo, es el concepto de Sistema Electrónico de Reuniones. Este sistema consiste en la integración de equipo electrónico (Hardware) y herramientas de software que permitan el desarrollo de actividades de grupo (Groupware) (Nunamaker, 1991). Los dos principales componentes de este tipo de sistemas son:

1. Una Sala de Reuniones Electrónicas. Una sala de reuniones donde cada uno de los participantes cuenta con una computadora personal (PC), las cuales se encuentran conectadas a una computadora que funciona como servidor. Este tipo de configuración permite a los asistentes a la reunión intercambiar y procesar ideas de manera interactiva y simultánea (Rodríguez, 1998).
2. Software de control para la Sala de Reuniones Electrónicas. Este software está compuesto por un conjunto de herramientas de comunicación que permiten intercambiar ideas que serán compartidas instantáneamente por el grupo. Con el uso de estas herramientas los usuarios realizarán sus actividades colaborativas libre y honestamente, productiva y decisivamente, obteniendo compromisos a largo plazo y

resultados documentados. Algunas de las herramientas proporcionadas por este tipo de software permiten realizar lluvia de ideas, votaciones, dar prioridad y organizar comentarios, así como herramientas que producen cierto tipo de documentación (Lowe, 1995).

Es precisamente en el área del software de control en que se ubica el problema presentado en este documento. Dicho problema surge de la necesidad de la instalación de una Sala de Reuniones Electrónicas en el Centro de Investigación Científica y de Educación Superior de Ensenada (CICESE). Era necesario contar en la sala con un componente que pudiera controlar y registrar las interacciones de los grupos de personas que hicieran uso de ella.

1.3 Objetivo.

El objetivo de este trabajo fue documentar el proceso de desarrollo (diseño e implementación) de un servidor de audio y de una aplicación cliente, para la Sala de Reuniones Electrónicas del CICESE. Esta aplicación del tipo Cliente-Servidor es un auxiliar en el control de los recursos disponibles en la Sala Electrónica, además de permitir grabar las sesiones de audio generadas durante las reuniones realizadas dentro de la sala.

1.4 Organización de la tesis.

En el siguiente capítulo se presenta la metodología utilizada para el desarrollo de este trabajo. En el capítulo 3 se describen de manera más detallada los elementos que conforman el contexto en el cual se sitúa este trabajo, al profundizar en conceptos tales como: Trabajo Cooperativo Asistido por Computadora, las aplicaciones Groupware como soporte a esta actividad, las Salas de Reuniones Electrónicas y la Memoria Organizacional. La relación existente entre tales entidades y la utilización del audio como medio de comunicación también es presentada en el capítulo 3. En el capítulo 4 se da una descripción de la Sala de Reuniones Electrónicas del CICESE al enlistar sus objetivos y explicar su funcionamiento, mostrar el equipo utilizado en su construcción y el software de control necesario para su operación. El capítulo 5 da una descripción del diseño del servidor de audio para la Sala de Reuniones Electrónicas del CICESE, se documentan las decisiones tomadas durante el proceso de diseño e implementación y se ubica la aplicación dentro del contexto del software de control de los recursos disponibles en la sala. Las pruebas realizadas y los resultados obtenidos son mostrados en el capítulo 6. En el capítulo 7 se presenta una discusión acerca del desarrollo y la implementación del servidor de audio en relación con los resultados obtenidos y en el capítulo 8 se presentan las conclusiones de este trabajo de tesis.

Capítulo 2. Metodología utilizada.

El primer paso para el desarrollo de este trabajo fue familiarizarse con el contexto en el cual se ubica la aplicación desarrollada: el área del Trabajo Cooperativo Asistido por Computadora. También fue necesario estudiar en detalle los problemas frecuentes y las características de los Sistemas Electrónicos de Reuniones, haciendo énfasis en las Salas de Reuniones Electrónicas.

Una vez familiarizado con estos conceptos, con la bibliografía relacionada, y tomando como base los documentos “Análisis de Requerimientos” y “Especificación de Análisis” de la Sala de Reuniones Electrónicas del CICESE, (Rodríguez, 1998) se realizó el análisis y el diseño de la aplicación para el manejo de audio dentro de la sala, utilizando el lenguaje de modelado UML para la realización de los diagramas de eventos, diagramas de estados, modelo funcional y diagramas de clases. El análisis y diseño del módulo de audio se realizó con la asesoría de la persona encargada del análisis, diseño y desarrollo del software de control de la Sala de Reuniones Electrónicas, Marcela Rodríguez Urrea, facilitando la integración del módulo de manejo de audio con el resto del software de control.

Cuando se terminó el diseño del módulo de audio, se exploraron algunas herramientas para el registro de sonido, construyendo pequeños programas prototipo. Se realizaron primero algunos prototipos utilizando programación de bajo nivel, accediendo directamente las direcciones de los registros para manipular tarjetas compatibles con el

modelo SoundBlaster. Estos prototipos se construyeron en lenguaje C. Sin embargo, debido a que estos programas sólo podían ser utilizados en tarjetas compatibles con el modelo SoundBlaster, estos primeros prototipos fueron abandonados al encontrar la biblioteca SoundBite, desarrollada en Java por la Universidad de Michigan. Se exploró además la posibilidad de utilizar las bibliotecas Java Media Framework (JMF), también desarrolladas en Java, pero la versión disponible en ese momento no proporcionaba la funcionalidad necesaria para el desarrollo de este trabajo.

Después de seleccionar las herramientas para construir el módulo, se procedió a implementar una primera versión del servidor y de la aplicación cliente para el manejo de audio. Estas aplicaciones fueron mejorándose de manera gradual hasta cumplir con las especificaciones planteadas en las etapas de análisis y diseño del software de control de la sala y del módulo de audio. Cada una de las funciones requeridas fueron probadas de manera independiente en la versión final del módulo para el manejo de audio. Para el desarrollo de estas aplicaciones se utilizó el Java Development Kit, (JDK) versión 1.1.4 para el sistema operativo Windows95 y para llevar un control de versiones de las aplicaciones en desarrollo se utilizó el software RCS, (Revision Control System) versión 5.7 para Windows95.

Después de probar el módulo de audio de manera independiente, se procedió a realizar su integración con el resto del software de control de la Sala Electrónica de Reuniones. Una vez hecha la integración, el software de control de la sala fue probado en su totalidad. Después de realizar estas pruebas fue necesario realizar algunos ajustes al módulo

de audio, eliminando el despliegado de información utilizada al probar el módulo de manera independiente, información innecesaria al hacer la integración con el resto del software de control.

Finalmente, también fue necesario hacer modificaciones a los datos de los programas cliente debido a la diferencia en la calidad de los micrófonos instalados en la Sala de Reuniones Electrónicas del CICESE.

Capítulo 3. Audio en Sistemas Cooperativos.

En este capítulo se describen de manera más detallada las definiciones, los componentes y los elementos relacionados con el Trabajo Cooperativo Asistido por Computadora (CSCW), conceptos como el de Sistema Electrónico de Reuniones y el de Sistemas Cooperativos, es decir, el Groupware. Se describe también la importancia de la inclusión de canales de audio como forma de comunicación dentro de las aplicaciones utilizadas para realizar trabajo en grupo.

3.1 Trabajo Cooperativo Asistido por Computadora.

En la actualidad poca gente trabaja de manera aislada, las personas pertenecen, la mayoría de las veces, cuando menos a un grupo. Los miembros de un grupo se comunican, comparten información, generan y organizan ideas, políticas y procedimientos, colaboran en la elaboración de reportes, forman consensos, comparten una visión, toman decisiones etc. (Nunamaker et al., 1991). Sin embargo, la mayoría de los sistemas de software proveen interacción únicamente entre un usuario y el sistema mismo. Cuando se redacta un documento, se consulta una base de datos o cuando se juega un videojuego, el usuario interactúa solamente con la computadora. Incluso muchos sistemas diseñados para usuarios múltiples, como algunos Sistemas de Información para oficina, proveen un soporte mínimo para la interacción usuario-a-usuario. Debido a que la mayor parte de las actividades de una

persona se realizan como parte de un grupo, este tipo de soporte es claramente necesario (Ellis et al., 1991).

Además de establecer canales de comunicación entre los miembros de un grupo, la colaboración entre éstos es también muy importante en la actividad grupal. Para que la colaboración sea efectiva es necesario que las personas intercambien información de manera eficiente. Desgraciadamente muchos de los Sistemas de Información actuales, particularmente los sistemas de bases de datos, aíslan a los usuarios del resto del grupo del que son miembros. Lo que se requiere en este tipo de aplicaciones son ambientes de trabajo compartidos que ofrezcan contextos de grupo actualizados y notificaciones explícitas de las acciones realizadas por cada participante, cuando esto sea necesario (Ellis et al., 1991).

El concepto de Trabajo Cooperativo Asistido por Computadora (CSCW: Computer Supported Cooperative Work), introducido en 1984 por Irene Greif y Paul Cashman, trabajadores del Massachusetts Institute of Technology (MIT) y Digital Equipment Corporation (DEC) respectivamente. De acuerdo a Bannon y Schmidt, Greif y Cashman eligieron esa frase como una manera corta de referirse a un conjunto de aspectos acerca del soporte de múltiples individuos trabajando de manera conjunta con sistemas computacionales. El CSCW debe ser concebido, de acuerdo a Bannon y Schmidt, como la empresa a través de la cual se pretende comprender la naturaleza y características del trabajo cooperativo con el objetivo de diseñar tecnologías apropiadas basadas en el uso de computadoras (Bannon y Schmidt, 1993).

3.2 Sistemas Cooperativos, Groupware.

El objetivo del Groupware es asistir a los grupos de trabajo en su comunicación, en su colaboración y en la coordinación de sus actividades. Específicamente, Ellis define al Groupware como:

“Sistemas basados en computadoras que soportan grupos de personas involucradas en una tarea u objetivo común y que provee una interfaz a un ambiente compartido” (Ellis et al., 1991).

De acuerdo a Ellis las nociones de tarea común y ambiente compartido son cruciales en esta definición, con esto se excluyen a los sistemas multiusuario, como los sistemas de tiempo compartido, donde los usuarios no comparten una tarea común. Esta definición tampoco especifica que los usuarios se encuentren activos de manera simultánea.

Los sistemas Groupware pueden ser concebidos como para dar soporte a una reunión cara-a-cara o para que sea utilizado por un grupo de personas distribuidas en diferentes lugares. También se puede pensar en aplicaciones Groupware que faciliten la comunicación y colaboración entre participantes de manera síncrona (en tiempo real) o de manera asíncrona (en diferentes horarios). Estas consideraciones espacio-tiempo dan origen

a la clasificación de sistemas Groupware proporcionada por Robert Johansen (Ellis et al., 1991). Dicha clasificación se presenta en la Figura 1.

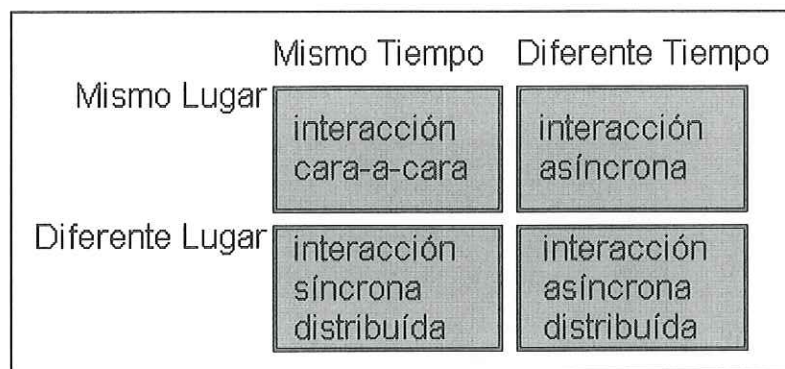


Figura 1. Clasificación de aplicaciones Groupware.

Existen otras clasificaciones de las aplicaciones Groupware como la propuesta por Esther Dyson, la cual toma como base el enfoque del control de la aplicación (control centrado en el usuario, centrado en un trabajo u objeto o centrado en un proceso) (Coleman, 1997a). Otro ejemplo es la clasificación basada en la funcionalidad de las aplicaciones Groupware. Dentro de esta clasificación Ellis menciona que los Sistemas de Soporte para la toma de Decisiones proveen herramientas basadas en computadoras para la exploración de problemas no estructurados en una configuración grupal, también menciona que estos sistemas son implementados como Salas de Reuniones Electrónicas que contienen varias computadoras conectadas a una red, grandes pantallas públicas controladas por computadora y equipo de audio y video (Ellis et al., 1991). De esta forma se ubica a los Sistemas Electrónicos de Reuniones, y a las Salas de Reuniones Electrónicas, como una parte integrante de los Sistemas Cooperativos, es decir, de las aplicaciones Groupware. Si

bien la utilización de Salas Electrónicas de Reuniones puede combinarse con tecnologías como videoconferencia o redes de datos para realizar sesiones de trabajo con personas o grupos localizados en otros lugares o con mecanismos para realizar reuniones en los que los usuarios participan en diferentes horarios (Soriano y Favela, 1997) ubican a este tipo de salas, de manera independiente, dentro de las aplicaciones Groupware en donde los usuarios se encuentran trabajando en un mismo lugar, al mismo tiempo.

3.3 Reuniones Electrónicas.

Es muy común que cuando un grupo de personas se encuentran trabajando en un mismo proyecto se realicen reuniones para tomar decisiones, compartir información, realizar evaluaciones, etc. Sin embargo estas reuniones pueden ser poco productivas por múltiples razones. Los participantes pueden no tener un objetivo bien definido o pueden seguir agendas ocultas, contrarias a la propuesta originalmente. Algunas personas pueden temer hablar ante sus compañeros, mientras que otros pudieran tender a dominar las discusiones. Es posible que surjan malentendidos cuando algunos participantes utilicen las mismas palabras para expresar diferentes ideas y diferentes palabras para describir el mismo concepto. Puede ser que asista a la reunión el personal equivocado o que no se tenga a la mano la información necesaria. Aún con estas dificultades las reuniones siguen siendo esenciales en la realización de trabajo en grupo, y lo seguirán siendo como lo han sido por miles de años. Sin embargo, la manera en que las reuniones eran realizadas anteriormente había cambiado muy poco hasta el día de hoy, pero con la utilización de Sistemas

Electrónicos de Reuniones se está modificando la forma en que se trabaja y se conducen las reuniones de trabajo. Incluso el concepto mismo de reunión está evolucionando con la utilización de estas nuevas herramientas (Nunamaker et al., 1995).

Una Reunión Electrónica es una forma de reunión donde los participantes intercambian información, platican, ya sea cara-a-cara o de manera remota, utilizando una computadora o algún otro medio electrónico a través del cual se accesa o se transmite información (Coleman, 1997). Existen diversos tipos de Reuniones Electrónicas, estas abarcan desde los pizarrones electrónicos (BBS: Bulletin Board Systems) y las listas de correo electrónico, hasta los sistemas de videoconferencia y las Salas de Reuniones Electrónicas descritas posteriormente. La mayoría de las Reuniones Electrónicas son llevadas a cabo utilizando un Sistema Electrónico de Reuniones donde, frecuentemente, cada usuario dispone de una computadora conectada a una red de datos. Cada uno de los participantes utiliza las estaciones de trabajo para trabajar ya sea de manera distribuida o cara-a-cara. Las sesiones distribuidas se llevan a cabo cuando los participantes se encuentran separados geográfica o espacialmente. Las reuniones cara-a-cara ocurren frecuentemente dentro de Salas de Reuniones Electrónicas.

3.3.1 Sistemas Electrónicos de Reuniones.

Un Sistema Electrónico de Reuniones es un nuevo ambiente de reuniones cuyo objetivo es hacer las sesiones de trabajo en grupo más productivas aplicando tecnologías de

la información. Este tipo de sistemas está diseñado para impactar y cambiar la conducta de los grupos de trabajo aumentando su efectividad, su eficiencia y su armonía (Nunamaker et al., 1991). Nunamaker define una reunión como un concepto bastante amplio, incluyendo en él cualquier actividad donde las personas se reúnen, ya sea en el mismo lugar, en un mismo tiempo o en diferentes lugares a diferentes horarios.

En un Sistema Electrónico de Reuniones siempre existe la oportunidad de expresar una idea, todos pueden “hablar al mismo tiempo” utilizando el teclado de su computadora. El sistema que maneja este tipo de reuniones hace públicas las contribuciones de los demás participantes casi inmediatamente, lo que significa que las personas pueden mantener la secuencia de sus propias ideas mientras siguen pendientes de lo que los demás aportan. Además, todas las contribuciones son anexadas a las anotaciones de la reunión y todos los participantes tienen el mismo derecho a contribuir (Nunamaker et al., 1995).

Los Sistemas Electrónicos de Reuniones, S. E. R., permiten incrementar dramáticamente el número de personas que pueden participar en una reunión. Muchos estudios han probado que cuando una reunión tradicional incluye a más de cinco ó seis personas la productividad decrece, el incluir más personas a la reunión reduce la productividad cada vez más (Nunamaker et al., 1995). Sin embargo, el utilizar S. E. R.s permite trabajar juntas a docenas de personas de manera efectiva. Estudios en grupos que utilizan este tipo de tecnología han detectado incrementos en productividad desde reuniones con tres participantes hasta reuniones donde se encuentran trabajando sesenta

personas. Aún cuando una reunión utilizando un S. E. R. tiene típicamente de diez a cuarenta participantes, algunos investigadores han reportado haber realizado de manera exitosa reuniones con más de doscientas personas (Nunamaker et al., 1995).

Debido a que un S. E. R. permite la realización de reuniones con un número mayor de personas, es posible que todos los interesados participen en la misma discusión en vez de que cada uno, o un grupo de ellos, desarrollen su propia visión de la discusión a través de una serie fragmentada de pequeñas reuniones. El grupo puede así evolucionar rápidamente hacia una visión compartida que cumple con las necesidades de la mayoría de sus miembros (Nunamaker et al., 1995).

La utilización de los S. E. R.s permiten además formar equipos de trabajo distribuidos, en los cuales uno o más miembros se encuentran geográficamente separados del resto del equipo. Diversos estudios en equipos de trabajo distribuidos han encontrado una lista importante de ventajas de éstos sobre los equipos de trabajo tradicionales, entre las que se encuentran: el incremento en la productividad de los miembros, una mayor capacidad en la recuperación de desastres, una mayor retención y satisfacción de clientes, la reducción en los requerimientos de espacios de trabajo, mayor flexibilidad, un mayor acceso a técnicos expertos y un mayor rango de trabajadores potenciales (Haywood, 1998).

3.3.2 Salas de Reuniones Electrónicas.

Como se mencionó en el capítulo 1, un S. E. R. está compuesto principalmente por una Sala de Reuniones Electrónicas y el software de control para administrar los recursos de hardware disponibles en la sala. Una Sala de Reuniones Electrónicas típica es una sala que cuenta con el equipo electrónico necesario para facilitar una sesión de trabajo de forma colaborativa y distribuida entre varios usuarios. Aunque diferentes diseños de salas pueden ser implementados, ésta debe contar con una configuración mínima que consiste de varias computadoras conectadas a una red local de datos, una para cada participante, y otra que sirva como la consola del facilitador de la reunión. También, es necesario contar con dispositivos de apoyo audio-visual, como los pizarrones electrónicos, cámaras de video, y proyectores (Rodríguez, 1998). A continuación se mencionan algunos de los componentes más importantes de una sala de este tipo, así como ciertos aspectos a tomar en cuenta al realizar su planeación.

3.3.2.1 Pantalla Pública.

La mayoría de las Salas de Reuniones Electrónicas incluyen una o más pantallas públicas. La interacción entre los participantes se da tanto con el intercambio de mensajes por medio de las herramientas electrónicas, así como a través de discusiones orales realizadas cara-a-cara. La utilización de una o varias pantallas públicas es una forma de ofrecer al grupo un punto de discusión común, así como compartir información importante para todos los asistentes. Aún cuando una pantalla es suficiente para la mayoría de las actividades de grupo, es posible utilizar pantallas adicionales para mostrar presentaciones

electrónicas, para desplegar una vista grupal de una pantalla de uno de los usuarios o para presentar una vista pública de un documento externo (Nunamaker et al., 1995).

3.3.2.2 Configuración de los lugares de trabajo.

Las primeras Salas de Reuniones Electrónicas fueron configuradas con las estaciones de trabajo de los participantes distribuidas en forma de herradura. Esto permite a los asistentes tener una buena visión del resto de los participantes y de la pantalla pública, la cual es colocada en el extremo abierto de la herradura. Esto también permite al coordinador de la reunión situarse en medio de la herradura para obtener la atención del grupo. Otras configuraciones han sido implementadas obteniendo diversos resultados. Se han distribuido a los participantes alrededor de una mesa situando la pantalla pública en uno de sus extremos. Esto permite enfocar la atención del grupo muy bien, pero no es posible realizar reuniones con muchos participantes. Otras veces se ha sentado a los participantes en mesas redondas, pero esto provoca que algunas personas se encuentren de espaldas a la pantalla pública.

Todas las configuraciones tienen ventajas y desventajas sobre otras, por lo que es fundamental considerar los objetivos principales de la sala para decidir que tan importante es la atención del grupo, el acceso a las pantallas públicas y el soporte para grupos grandes de personas (Nunamaker et al., 1995).

3.3.2.3 Iluminación y ruido.

La iluminación es una consideración extremadamente importante al diseñar este tipo de salas, esta debe ser apropiada para evitar causar molestias al visualizar tanto la información desplegada en la pantalla pública como en los monitores de las computadoras. La iluminación también puede ser utilizada para dirigir la atención o participación del grupo de trabajo. Por ejemplo, durante sesiones de intercambio de mensajes electrónicos la iluminación puede ser reducida a pequeños espacios al lado de las estaciones de trabajo de cada participante. Si se realiza una discusión general se puede iluminar todo el cuarto y durante alguna presentación se puede iluminar solamente la parte central y frontal de la sala.

El ruido generado por los aparatos de una Sala de Reuniones Electrónicas puede ser una fuente de distracción o molestia durante las sesiones de trabajo. Las principal causa del ruido son los abanicos de las computadoras y los proyectores de la sala. Al minimizar el ruido se aumenta la utilidad de la sala, esto puede lograrse colocando los dispositivos dentro de gabinetes cerrados, pero manteniéndolos con una buena ventilación para evitar sobrecalentamiento.

3.3.2.4 Software para las Salas de Reuniones Electrónicas.

El software de una Sala de Reuniones Electrónicas no es monolítico, consiste mas bien en un conjunto de módulos a través de los cuales se pueden expresar, evaluar y

organizar ideas con el fin de lograr que el grupo de trabajo tome las decisiones más apropiadas de una manera eficiente. Para lograr esto, el software utilizado debe proveer principalmente mecanismos para la comunicación entre los participantes, herramientas para generación, disseminación y organización de ideas, módulos que permitan el acceso a información y el soporte para fusionar objetivos individuales con el esfuerzo grupal (Nunamaker et al., 1995). Un ejemplo de un software para la generación y disseminación de ideas sería una herramienta para realizar una “lluvia de ideas”, actividad en la cual se motiva al grupo a generar tantas ideas como sea posible. Inicialmente todos los usuarios aportan sus ideas escribiéndolas en su computadora de manera simultánea, posteriormente el sistema se encarga de pasar esas ideas de manera aleatoria a otro participante. Los participantes pueden entonces argumentar o expandir las ideas que reciben, las cuales pueden a la vez guiarlos hacia nuevos pensamientos. Por otro lado, una herramienta de organización de ideas guía al grupo a converger rápidamente hacia cuestiones específicas para que sean exploradas de manera más profunda.

En una Sala de Reuniones Electrónicas se puede proveer el acceso a la información de varias maneras. Muchas herramientas, por ejemplo, utilizan palabras y frases clave alimentadas por los usuarios para realizar un filtrado de fuentes de información externas y evitar una sobrecarga de información. Además, debido a que la utilización de los Sistemas Electrónicos de Reuniones permite que se realicen sesiones de trabajo con un número mayor de personas, la base de información contenida en las mentes de los participantes es más grande, lo que provoca un incremento en la productividad del grupo. Una ventaja más de las interacciones electrónicas es la posible generación automática del historial de la

reunión. Este historial puede ser utilizado para consultas posteriores y se convierte en información muy valiosa para el grupo (Nunamaker et al., 1995).

Este software no está formado solamente de herramientas para realizar “lluvia de ideas” y votaciones, cada uno de sus módulos debe incluir los procesos y métodos relacionados con el uso de las herramientas y el tipo de ambiente en el cual son utilizadas. Cuando estos elementos son utilizados adecuadamente, la productividad grupal se ve incrementada significativamente.

De manera adicional, el software de la Sala de Reuniones Electrónicas puede ser utilizado para controlar los componentes de hardware disponibles en la sala y coordinar su funcionamiento con las herramientas de comunicación y colaboración, logrando así una integración total del ambiente de trabajo.

3.3.2.5 Otras consideraciones.

Existen muchos otros aspectos a tomar en cuenta al construir una Sala de Reuniones Electrónicas, como la inclusión de espacios para descansar o socializar, el no bloquear la visión de los usuarios, el soporte para comunicación verbal y no verbal, etc. Además todos estos recursos y requerimientos deben ser controlados de manera fácil y eficiente. Este control puede lograrse utilizando software que permita al coordinador de la reunión dirigir al grupo para llegar a su meta.

3.4 Utilización y rol del audio en aplicaciones para CSCW.

La gente pasa gran parte de su tiempo de trabajo hablando, de acuerdo a estudios mencionados por Hindus y Schmandt, las llamadas por teléfono consumen el 20% del horario de trabajo diario mientras que las reuniones cara-a-cara consumen de un 25 a un 50% adicional. Sin embargo, este tiempo que se utiliza para la comunicación utilizando la voz no ha sido tomado en cuenta como se debiera por la tecnología de las computadoras. Esta pérdida de información verbal es sumamente negativa dada la influencia del audio en el resultado de una comunicación, independientemente de la presencia o carencia de medios de comunicación visual (Hindus et al., 1992).

Al añadir soporte para la comunicación por medio de audio a los ambientes de CSCW, de manera que los usuarios se puedan comunicar verbalmente acerca de la tarea en la que se trabaja, además de utilizar las formas de comunicación textuales y gráficas, se enriquece la calidad total y la efectividad de la colaboración (Mathur et al., 1993). Debido a que eventualmente el solo soportar conversaciones síncronas entre los participantes de una sesión de trabajo será insuficiente, se debe realizar más trabajo en el almacenamiento y la recuperación del contenido de las interacciones (Hindus et al., 1992).

Actualmente el principal uso del habla como tipo de dato dentro de las aplicaciones Groupware es el de grabar pequeños segmentos de voz. Estos segmentos son utilizados en

sistemas de mensajes como correo de voz y en sistemas de edición colaborativa. La voz puede también ser utilizada para realizar anotaciones sobre segmentos de texto o almacenar notas en un calendario o en celdas de una hoja de cálculo. Los segmentos de habla son tratados como entidades atómicas y las aplicaciones que los utilizan simplemente mantienen una referencia al sonido en forma de un número de mensaje, una fecha en el calendario o una posición entre un párrafo o en una tabla de información. Esta información de audio puede combinarse con anotaciones de los usuarios y descriptores situacionales para crear audio semi-estructurado. La utilización del audio semi-estructurado provee un marco de trabajo en el cual los contenidos de las interacciones pueden ser manipulados y organizados.

3.5 Registro de las sesiones de audio en las Salas Electrónicas de Reuniones, Memoria Organizacional.

Al capturar el contenido hablado de las sesiones de trabajo realizadas en una Sala Electrónica de Reuniones se pueden generar minutas de la misma de manera automática, pero debido a los aspectos tecnológicos de esta tarea, el estructurar estas grabaciones es muy difícil. Uno de los primeros problemas a enfrentar es la obtención de las participaciones de los asistentes con un nivel alto de calidad, es crítico para esto proporcionar un micrófono individual a cada una de las personas. Utilizar un sólo micrófono de amplio alcance es más simple, pero deteriora la calidad del sonido. Otro aspecto importante es el de asociar cada participación con el participante que la generó. Idealmente, cada usuario debiera de contar con un canal de audio que registre sus participaciones.

El utilizar audio semi-estructurado para el registro de las participaciones permite la flexibilidad suficiente para tener acceso posterior al contenido de las intervenciones sin depender demasiado en la estructuración explícita de las mismas por parte del emisor o el receptor del mensaje. Los usuarios pueden cambiar la estructura de la información de la manera más apropiada, pero no tienen que estructurarla para que su contenido sea accesible y manejable.

Este tipo de información puede registrarse dentro de la Memoria Organizacional de la entidad donde se genera. La Memoria Organizacional incluye los registros de los proyectos realizados en la organización, las decisiones que se tomaron durante la ejecución de éstos, y la experiencia obtenida mientras se trabajaban en ellos. Tradicionalmente, las organizaciones han preservado esta información implícitamente en los individuos que trabajan en éstas y en los procesos que crean para coordinar su trabajo, y explícitamente en los manuales de procedimientos y en la documentación de los proyectos. El uso del término “Memoria Organizacional” se deriva del punto de vista del procesamiento de información de lo que una organización es y hace. En esta metáfora, una organización es vista como un sistema de información involucrada en la adquisición, procesamiento y recuperación de la información (Rodríguez, 1998).

Capítulo 4. La Sala de Reuniones Electrónicas del CICESE.

Las actividades de formación e investigación que se desarrollan en el CICESE generan muchas situaciones en las cuales puede ser utilizado un Sistema Electrónico de Reuniones. Investigadores y alumnos de la institución trabajan a menudo con personas de otros centros de investigación geográficamente distantes. Es frecuente también que se realicen en CICESE proyectos que involucran grupos de trabajo, los cuales pueden beneficiarse con la utilización de un Sistema Electrónico de Reuniones.

Dados los beneficios de la utilización de un sistema de este tipo y al ver el amplio número de situaciones en las cuales puede ser utilizado dentro de la institución, se decidió diseñar y desarrollar un Sistema Electrónico de Reuniones que sirviera como soporte a las actividades realizadas en grupo dentro del CICESE. El sistema en cuestión debía estar compuesto por una Sala de Reuniones Electrónicas y un software para Reuniones Electrónicas. La Sala de Reuniones se encuentra actualmente en operación y es utilizada frecuentemente para impartición de clases, presentaciones y reuniones de trabajo.

4.1 Escenarios de utilización.

De acuerdo a las necesidades del CICESE, al realizar el diseño de la Sala de Reuniones Electrónicas, se planteó que la sala fuera utilizable tomando en cuenta los siguientes escenarios:

- **Reunión colocalizada para la toma de decisiones.** Los participantes de la reunión se reúnen en el mismo lugar y al mismo tiempo con el objetivo de tomar decisiones.
- **Reunión distribuida para la toma de decisiones.** Los participantes de la reunión localizados en diferentes áreas geográficas tienen el objetivo de tomar acuerdos o decisiones.
- **Clase colocalizada.** El uso de la sala como un salón de clases electrónico para impartir cursos en forma local, ofrece a los instructores y alumnos una nueva forma de interactuar con la tecnología. Ya que la tecnología proporciona nuevas formas de comunicación y a la vez mejora la calidad de instrucción con respecto a un aula de clase tradicional.
- **Clase distribuida en CICESE y otros.** La sala de reuniones puede ser usada como aula electrónica con capacidad de diez personas colocalizadas y otros grupos remotos de varias personas que tengan acceso a la clase desde sus propios espacios de trabajo.
- **Sala para revisiones técnicas distribuida y colocalizada.** La sala puede ser utilizada en proyectos de ingeniería de software, donde los usuarios que participen en una revisión técnica pueden hacer uso del equipo para intercambiar ideas y opiniones. Los revisores pudieran estar colocalizados o distribuidos.

- **Sala de presentaciones para audiencias remotas.** El instructor utiliza la sala para impartir una clase o presentación y las demás personas accesan remotamente la presentación. Esto significa que en este escenario no existe una interacción con la audiencia como en una clase, la información fluye solamente del instructor hacia quienes siguen la presentación (Rodríguez, 1998).

4.2 Equipo y distribución del mismo dentro de la Sala de Reuniones Electrónicas del CICESE.

La lista del equipo utilizado y la funcionalidad que proporciona dentro de la Sala Electrónica de Reuniones del CICESE es la siguiente:

- **11 Computadoras personales.** Estas son las estaciones donde los usuarios trabajan al realizar una reunión, cada una cuenta con un micrófono y acceso a la red local de datos. Las computadoras utilizan Windows95 como sistema operativo. Una de estas computadoras se encuentra reservada para la persona que dirige la presentación y se encuentra localizada al lado del área de proyección.
- **1 Computadora servidor.** Es la computadora desde donde se controla el equipo de la sala y la sesión de trabajo. Es generalmente utilizada por la persona a cargo de la reunión.

- **3 Cámaras de video Sony EVI-D30.** Una cámara de video es utilizada para enfocar al presentador. Las otras dos cámaras enfocan a la audiencia localizada en la sala de reuniones. Cada una de estas cámaras se encarga de enfocar a cinco de los participantes.
- **1 Cámara de documentos VideloLabs FlexCam Pro.** Su trabajo consiste en captar la imagen de un documento impreso y transmitirla a través de la computadora a la cual está conectada. El contenido del documento puede ser en forma de fotografías, diagramas o texto.
- **1 Pizarrón Electrónico Ibid.** El pizarrón se utiliza como un área de trabajo global, la que puede ser pensada como un bloque para almacenar conceptos relevantes durante una sesión. Este dispositivo hace posible que los participantes de la reunión trabajen en forma interactiva, generen, capturen y comuniquen sus ideas. Es posible con este pizarrón capturar los resultados de la reunión y guardar esta información directamente a la computadora.
- **1 Proyector Multimedia 3M MP8030.** Es utilizado en la sala de reuniones para proyectar presentaciones e información que requieren ser difundidas a los participantes.

- **1 Videocasetera VHS.** Permite grabar, reproducir y editar el video generado durante una reunión.
- **1 Conmutador (switch) de video.** Proporciona un mecanismo manual para seleccionar la salida de video deseada (puede también funcionar de manera independiente, sin conexión a una PC), cuenta con 6 entradas y 2 salidas. El equipo que se conecta como entrada al conmutador es el siguiente: cámaras de video EVI-D30, cámara de documentos y videocasetera, mientras que, el proyector de video y una computadora están conectadas a la salida (Rodríguez, 1998).

La Sala de Reuniones Electrónicas cuenta con un área de control en donde se encuentran localizadas la computadora servidor, la videocasetera y el conmutador de video, en la Figura 2 se muestra la distribución de los dispositivos dentro de la sala, así como las conexiones que existen entre ellos. La videocasetera no se tomó en cuenta dentro del diagrama ya que es un componente que no se encuentra conectado de manera permanente y puede ser cambiado de lugar con facilidad.

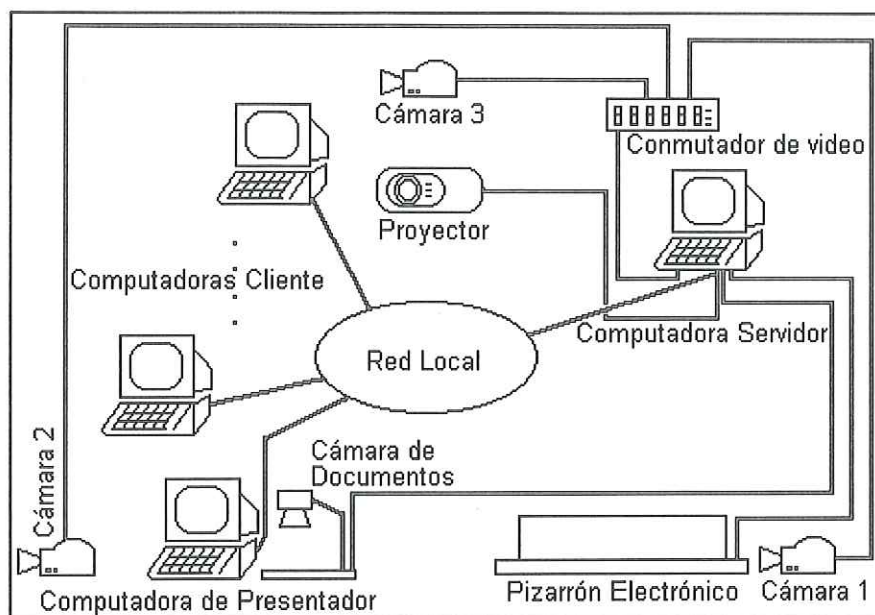


Figura 2. Distribución y conexiones del equipo utilizado en la Sala de Reuniones Electrónicas del CICESE.

La configuración de los asientos es en forma de herradura y en la parte frontal se cuenta con un área en donde puede presentarse el desplegado del proyector de video. Además, las paredes de la sala fueron acondicionadas para evitar fenómenos auditivos negativos como el eco o una mala distribución del sonido y se utilizaron lámparas especiales para la iluminación con el objetivo de evitar reflejos en los monitores de las computadoras o en la pantalla de presentación.

4.3 Software de control para la Sala de Reuniones.

Este software es el que se encarga de controlar los dispositivos disponibles en la sala y el desarrollo de las reuniones de trabajo que en ella se efectúan. Este es un sistema del tipo Cliente-Servidor en donde las porciones cliente son ejecutadas en las diez estaciones de

trabajo y en la computadora del presentador, mientras que la aplicación servidor radica en la máquina servidor. Este sistema fue desarrollado principalmente por Marcela D. Rodríguez, como parte de su trabajo de Tesis para obtener el título de Maestría en Ciencias de la Computación (Rodríguez, 1998). Tanto la aplicación servidor como la cliente fueron desarrolladas utilizando el lenguaje de programación Java.

4.3.1 Aplicación Servidor.

Esta aplicación se encuentra instalada en la máquina que cumple la función de servidor. Esta computadora es la que controla el equipo que se encuentra disponible en la sala y transmitirá el audio y video generado durante la reunión, por lo que es necesario que tenga instaladas tarjetas de audio y video, así como software (e.g. NetMeeting) para establecer conexión con usuarios remotos (Rodríguez, 1998).

Para realizar una sesión de trabajo bajo uno de los escenarios descritos al inicio de este capítulo, es necesario ejecutar primero la aplicación servidor y después las aplicaciones cliente que sean requeridas de acuerdo al número de participantes. Una vez que la aplicación servidor es ejecutada, la persona a cargo de la reunión debe elegir cual es el escenario bajo el cual se llevará a cabo la sesión, de forma que el sistema active y administre los recursos necesarios de acuerdo a la situación. Cuando uno de los escenarios de trabajo es elegido, se presenta al usuario una pantalla con los dispositivos a utilizar (ver Figura 3). Desde esta pantalla los dispositivos pueden ser configurados (ver Figura 4) y administrados.

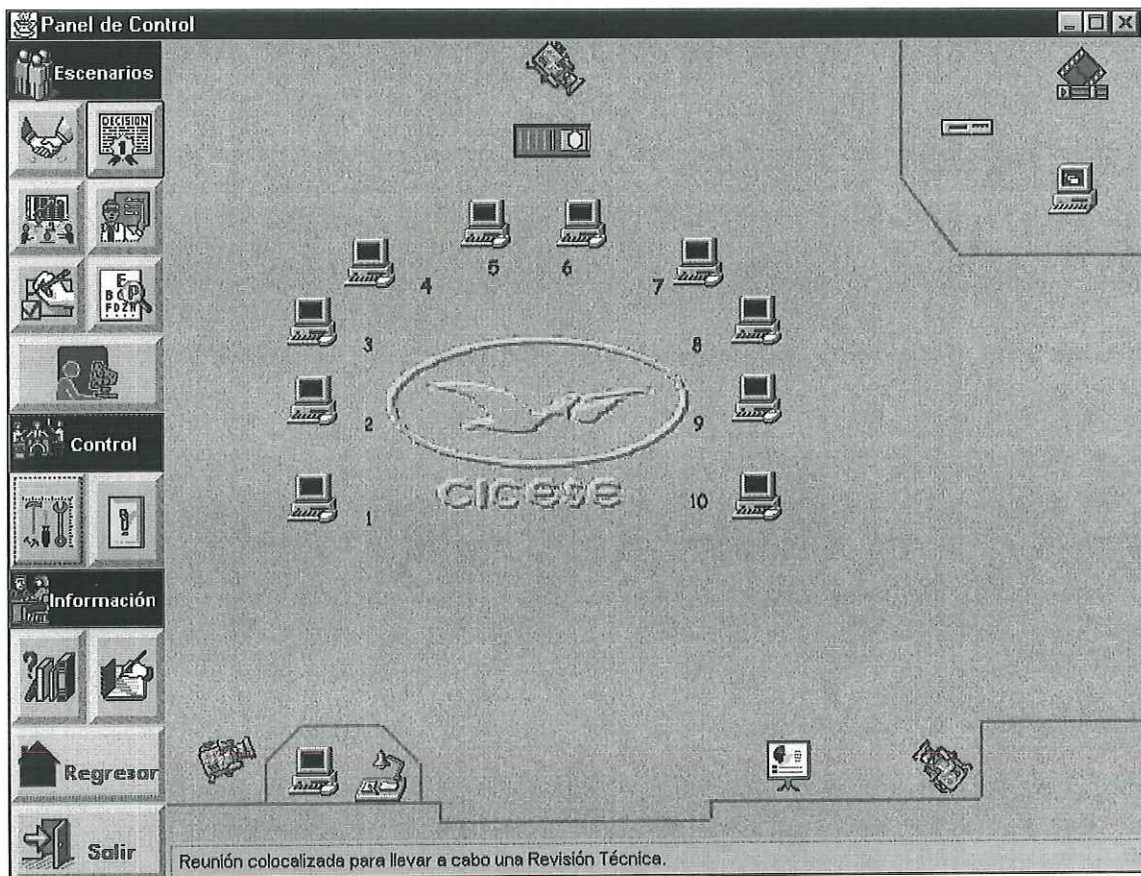


Figura 3. Ejemplo de una sesión de control de equipo con el software de control de la Sala de Reuniones Electrónicas.

La configuración de los dispositivos se realiza seleccionando con el apuntador (ratón) aquel que se quiere modificar. Cuando uno de los recursos es seleccionado una pequeña ventana de configuración es mostrada al usuario para que realice las modificaciones necesarias, tal configuración comprende desde las direcciones de red de las máquinas cliente hasta la especificación de entradas y salidas en el conmutador de video (ver Figura 4).

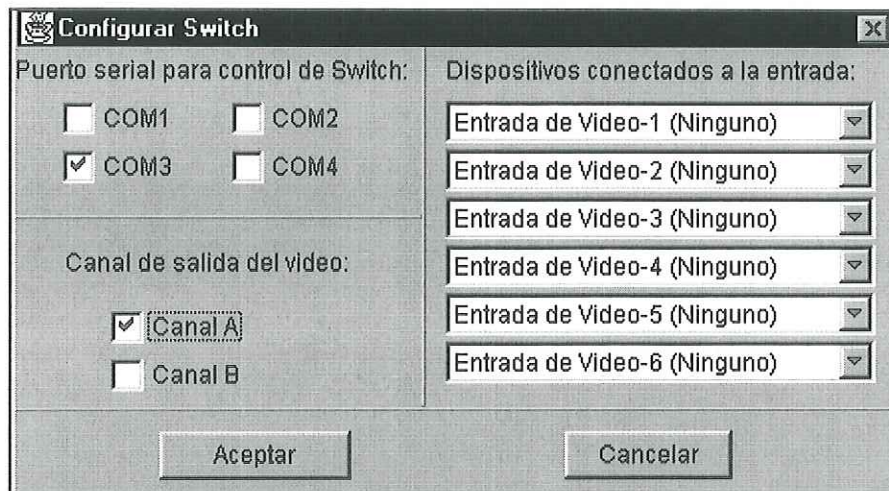


Figura 4. Configuración del conmutador de video.

La aplicación servidor permite también realizar movimientos explícitos de las cámaras de video y visualizar una ayuda en línea acerca del funcionamiento del software de control y de la Sala de Reuniones Electrónicas en general.

4.3.2 Aplicación Cliente.

Esta parte del sistema se instala en cada una de las diez computadoras utilizadas durante una sesión. Tiene la función de mandar señales al servidor que indiquen si se está generando audio, ésto con el fin de que el servidor reconozca quien es el usuario que está participando en ese momento y proceda a enfocarlo con alguna de las dos cámaras dedicadas a la audiencia. Es por esto que se requiere que las computadoras cliente tengan tarjeta de audio y un micrófono instalados (Rodríguez, 1998).

Para incorporarse a un sesión de trabajo desde una de las computadoras cliente simplemente se selecciona el ícono que indica “Sala Electrónica de Reuniones” que se encuentra en la ventana principal de las computadoras en donde la aplicación cliente ha sido instalada. La ventana que se muestra una vez iniciado el sistema es la que puede observarse en la Figura 5.

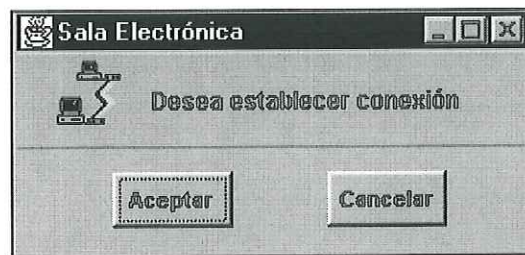


Figura 5. Ventana desplegada para incorporarse a una sesión de trabajo.

Una vez que la conexión ha sido establecida, el usuario es notificado, en caso contrario, se muestra un mensaje de error.

4.4 Servidor de audio de la Sala de Reuniones Electrónicas del CICESE.

Al planear la forma en que los participantes de una reunión dentro de la sala serían enfocados por alguna de las cámaras al realizar una intervención, se propuso utilizar la propia voz de los usuarios, es decir, que el control del movimiento de las cámaras se llevara a cabo detectando el momento en el que uno de los asistentes comience a hablar, instante en el cual una de las cámaras lo enfocará, de acuerdo a la posición de cámara pre-

programada para el lugar de la computadora correspondiente. Esto debe realizarse censando de manera constante el nivel de audio presente en los micrófonos de las estaciones. Otra de las opciones para el manejo de las participaciones era utilizar el método de “presione para hablar” (push to talk) presente en algunas herramientas de comunicación. Sin embargo, este esquema requiere un esfuerzo adicional por parte del usuario quien tiene que recordar presionar un botón para que su voz sea escuchada. Además, el tener que posicionar el apuntador en una región específica de la pantalla puede también interrumpir alguna actividad que el participante haya estado realizando, por ejemplo, algún trazo o dibujo (Watson, 1996).

Dentro del software de control de la sala, era necesario también contar con una herramienta que permitiera registrar las intervenciones de los participantes. De esta forma, el audio de las estaciones de trabajo podría ser almacenado para transmisión, procesamiento posterior o consulta como parte de la memoria de la reunión.

Para lograr lo expuesto en los dos párrafos anteriores se desarrolló un servidor de audio y su correspondiente aplicación cliente, ambas aplicaciones fueron incrustadas dentro del software de control de la Sala de Reuniones Electrónicas del CICESE. El servidor de audio es parte de la aplicación servidor del software de control de la sala y se encuentra en ejecución ininterrumpida siempre que la porción servidor del software de control esté activa. El servidor de audio tiene una lista con las direcciones de red de las máquinas de donde puede recibir señales de voz, pero mantiene abierto únicamente un solo canal de audio por donde recibe los datos de voz de la máquina donde se encuentra quien está

hablando en ese momento. Cada una de las participaciones enviadas por los clientes son almacenadas en archivos de audio por el servidor, para poder utilizar su contenido posteriormente. El servidor de audio notifica también al software de control cual es la máquina donde se encuentra quien tiene la palabra para que sea enfocado por una de las cámaras de la sala.

El cliente de audio se encuentra censando de manera permanente el nivel de sonido que detecta el micrófono de la máquina en la cual está en ejecución. Una vez que el cliente detecta que el nivel de sonido ha sobrepasado en forma continua un umbral predefinido por un cierto período de tiempo, deduce que el usuario de la estación se encuentra hablando y notifica inmediatamente al servidor para que la persona sea enfocada por una de las cámaras y que su intervención sea registrada dentro de las participaciones de la reunión.

El diseño y el desarrollo del servidor y el cliente de audio son documentados a detalle en el capítulo siguiente.

Capítulo 5. Diseño e implementación del servidor de audio.

En este capítulo se describe y documenta el diseño y desarrollo del servidor de audio que se utiliza dentro del software de control de la Sala de Reuniones Electrónicas del CICESE, así como también, su interacción con los demás componentes del software de control, su funcionamiento y su configuración.

5.1 Diseño de la aplicación para el manejo de audio.

Como se ha mencionado anteriormente la aplicación desarrollada para el manejo de audio dentro de la Sala de Reuniones Electrónicas en cuestión es del tipo Cliente-Servidor. Bajo esta arquitectura la aplicación cliente hace una petición de servicio a un servidor, el cual ofrece dicho servicio a los clientes. El programa servidor puede estar en ejecución en la misma computadora o en una diferente, cuando esto último sucede, la comunicación se efectúa generalmente utilizando una red de datos (Robbins et. al., 1996).

La arquitectura Cliente-Servidor utiliza la dirección en la cual se inicia el intercambio de datos para clasificar a una aplicación como cliente o como servidor. De manera general, el programa que empieza la comunicación es designado como el cliente. El software de tipo cliente es invocado usualmente al utilizar servicios de red como navegadores de Internet o programas de acceso remoto como TELNET. De esta forma, se considera como aplicación servidor a todo programa que espera por peticiones de comunicación generadas por una aplicación cliente (Comer et. al., 1996). La configuración de la Sala de Reuniones

Electrónicas del CICESE, así como el tipo de control y de comunicación que se requería realizar dentro de la misma, proporcionaron un ambiente ideal para desarrollar la aplicación para el manejo de audio bajo el modelo Cliente-Servidor.

En el diagrama de clases ilustrado por la Figura 6 se puede observar el diseño y la estructura del servidor de audio, la clase `AudioServerUDP`, así como su relación con las demás clases que forman parte del software de control de la sala. Para simplificar el diagrama, los atributos y métodos de las clases `SalaReunion`, `MultiServer` y `java.lang.Thread` fueron omitidos. Los diagramas de clases utilizados en las Figuras 6 y 7 están basados en el lenguaje de modelado UML (Unified Modeling Language), una notación principalmente gráfica para expresar diseños (Fowler et. al., 1997).

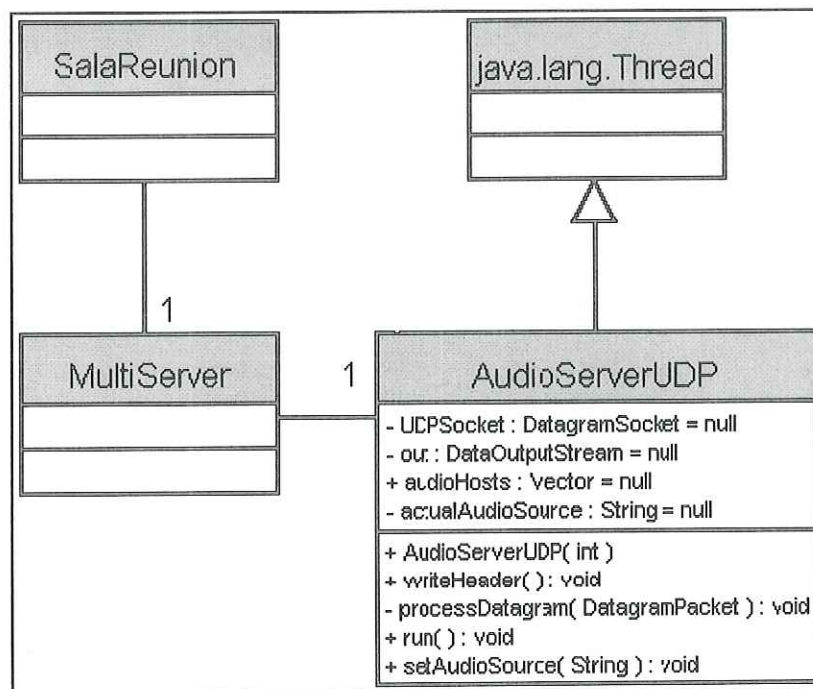


Figura 6. Diagrama de clases UML del servidor de audio y su relación con el software de control de la Sala de Reuniones Electrónicas.

La clase SalaReunion es el componente principal del software de control de la sala. Es a través de la interfaz de esta clase que el operador puede configurar los dispositivos disponibles, así como inicializar y detener las sesiones de trabajo. Como se puede observar en el diagrama, la clase SalaReunion cuenta con una clase llamada MultiServer. La clase MultiServer es la encargada de controlar las cámaras y de decidir cual es el usuario que está participando en un momento dado, una vez determinado esto, MultiServer interactúa con la clase AudioServerUDP, la cual recibe la información de audio de las computadoras de la sala en las que se encuentra ejecutando la aplicación cliente del software de control.

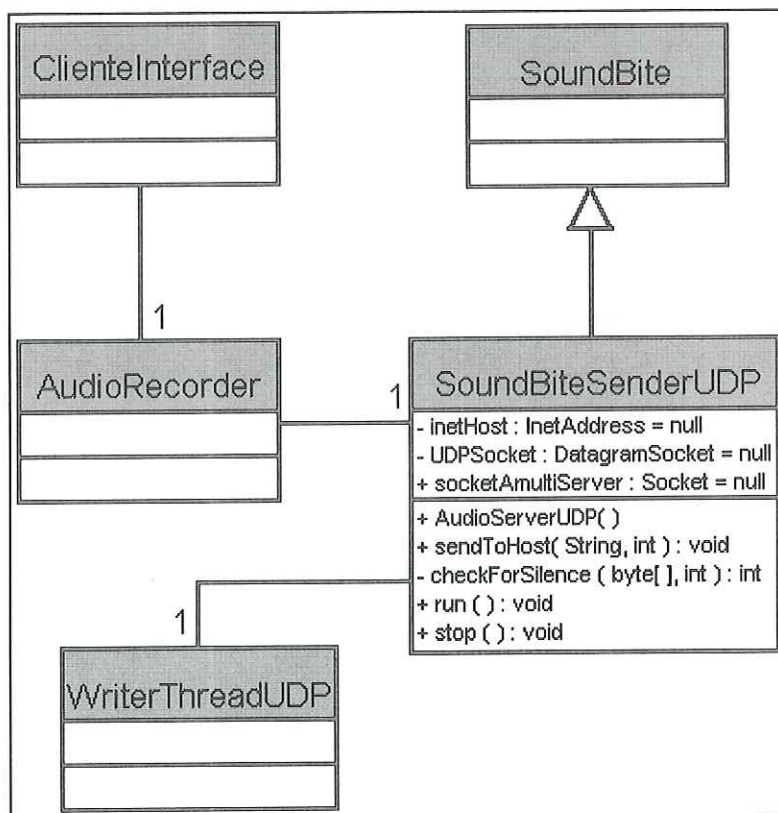


Figura 7. Diagrama de clases de la aplicación cliente del software de control de la sala.

En la parte cliente del software de control la sesión es inicializada a través de la clase `ClienteInterface` (ver Figura 7). `ClienteInterface` activa la clase `AudioRecorder` que contiene a su vez un objeto del tipo `SoundBiteSenderUDP`, al que activa pasándole la dirección de red de la computadora y el número de puerto en donde se encuentra ejecutándose la aplicación servidor del software de control de la Sala de Reuniones Electrónicas. La clase `SoundBiteSenderUDP` mantiene abierto un canal de comunicación con la aplicación servidor a través del objeto `socketAmultiServer`. Por esta vía, informa al servidor cuando existe información de audio que debe ser tomada en cuenta como una participación del usuario de la computadora en la que se ejecuta la aplicación cliente y le notifica también cuando la participación termina. El envío de la información de sonido al servidor de audio la realiza `WriterThreadUDP`, el cual es un componente de `SoundBiteSenderUDP`, a través de otro canal de comunicación. La existencia de este canal adicional es necesaria ya que las características de los datos de audio son diferentes a las de los datos de control en la comunicación entre el cliente y el servidor. Es decir, `MultiServer` maneja solamente la información de control proveniente de los clientes, estableciendo una conexión con cada uno de ellos, mientras que `AudioServerUDP` maneja exclusivamente la información del audio generado en las reuniones. Esta situación se ilustra en el diagrama de la arquitectura de la aplicación para el manejo de audio (ver Figura 8).

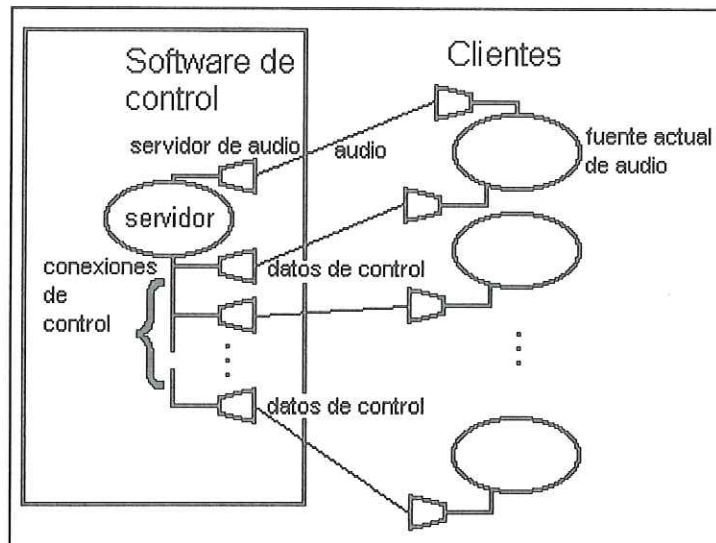


Figura 8. Arquitectura del módulo de audio del software de control de la Sala de Reuniones Electrónicas.

5.2 Implementación.

La implementación de esta aplicación se realizó utilizando el lenguaje de programación Java. Este lenguaje de programación, introducido a finales de 1995, tiene muchas características deseables, las cuales lo han colocado rápidamente como uno de los lenguajes preferidos por un gran número de desarrolladores de software. En este caso, Java fue elegido por sobre otras opciones principalmente debido a sus características de portabilidad y su amplio soporte para acceder los recursos de comunicación de redes de datos. Debido a que los programas escritos en Java son compilados a un formato neutral a la arquitectura (byte-code), una aplicación Java puede ser ejecutada en cualquier sistema, siempre y cuando ese sistema cuente con una implementación de la Máquina Virtual de Java

(Flanagan, 1997). Las bibliotecas **java.net** proporcionan una fuerte y flexible infraestructura para transmitir y recibir información a través de una red de datos que permiten realizar desde conexiones a bases de datos o servidores de documentos HTML hasta operaciones de transporte de datos de bajo nivel.

Además, Java es un lenguaje de programación orientado a objetos. En este tipo de lenguajes el programador puede concentrarse más en los datos de la aplicación y en los métodos que los operan, en vez de pensar estrictamente en términos de procedimientos o funciones. En un sistema como este, la colección de datos y métodos que operan sobre ellos son encapsulados en una “clase”. Una vez juntos, los datos y los métodos, describen el estado y el comportamiento de un “objeto”. Las clases son organizadas en una jerarquía, de tal manera que una subclase puede heredar datos y funcionalidad de su clase padre (Flanagan, 1997).

A pesar del número de clases incluídas en la versión 1.1 de Java, y la gran funcionalidad con que estas cuentan, en esa versión no hay forma de grabar información de audio utilizando alguna de las clases que forman parte del lenguaje. En este caso, para poder registrar audio con Java fue utilizada una biblioteca escrita en lenguaje C++, integrándola a la aplicación principal gracias a la posibilidad de Java de ejecutar porciones de código escritas en otros lenguajes utilizando JNI (Java Native Interface).

5.3 Aplicación servidor.

La porción servidor de la aplicación para el manejo de audio se implementó construyendo la clase `AudioServerUDP`. Como se muestra en la Figura 6, esta clase extiende la clase `java.lang.Thread`, clase que se encuentra dentro del paquete de clases `java.lang`, el cual contiene algunas de las clases más importantes del lenguaje Java (Flanagan, 1997). Un Thread es un hilo de ejecución de un programa, ya que la máquina virtual de Java permite a sus aplicaciones tener múltiples Threads ejecutándose de manera concurrente (Sun Microsystems, 1998). El método `run()` de la clase Thread es el cuerpo del hilo de ejecución y empieza a ejecutarse cuando el método `start()` del objeto Thread es invocado. La ejecución del Thread termina cuando el método `run()` llega al final o cuando el método `stop()` es llamado (Flanagan, 1997).

Cuando el operador de la Sala de Reuniones Electrónicas comienza una sesión en la que es necesario administrar los dispositivos de la sala, la clase `MultiServer` inicializa la ejecución del servidor de audio `AudioServerUDP` como un hilo de ejecución independiente. Este hilo de ejecución se encuentra activo hasta que la sesión de trabajo termina. Mientras el servidor `AudioServerUDP` está activo, se encuentra escuchando ininterrumpidamente la información de sonido que los programas cliente le envían. Cada información de audio que el servidor recibe contiene la dirección de la computadora en donde éste se registró, de esta forma, el servidor decide que hacer con los datos de acuerdo a la dirección de la computadora en que fueron originados. En el caso de la Sala de Reuniones, el servidor cuenta con una tabla donde se encuentran registradas las direcciones

de las computadoras de las cuales es válido recibir datos, cualquier información originada en otra computadora es desechada. Las direcciones de red de las computadoras son comunicadas a AudioServerUDP por la clase SalaReunion, la clase principal del software de control de la sala. AudioServerUDP cuenta también con una variable para llevar el registro de cual es la computadora cliente de la que se recibe información en un momento dado, es decir, cual es la fuente de audio en ese momento. Se decidió tener abierto sólo un canal de comunicación a la vez debido a que, por una parte, es difícil mantener sincronizadas múltiples señales de audio generadas en computadoras distintas que pudieran utilizarse como registro de las participaciones en las reuniones de trabajo y, por otro lado, es difícil entender lo que dos o más personas dicen cuando se encuentran hablando al mismo tiempo.

5.3.1 Control de piso.

Cuando la información recibida corresponda a una dirección válida, y cuando esta dirección es igual a la que se tiene registrada como fuente de audio en ese instante, ésta es escrita directamente a un objeto del tipo `DataOutputStream`. Este objeto puede ser direccionado a un archivo o a un dispositivo que decodifique los datos mientras son recibidos.

Si la información recibida proviene de un cliente válido, pero la dirección es diferente a la de la fuente de audio seleccionada en ese momento, la fuente de audio es

cambiada a la dirección contenida en los datos que acaban de ser recibidos. Esto se realiza como un mecanismo de control de piso, es decir, ceder la palabra al participante que se encuentre hablando, sin que el operador tenga que realizar alguna actividad especial para lograr tal objetivo. El software de control puede también ser notificado por los clientes que la dirección de origen de audio debe ser actualizada, en ese caso, el software de control notifica a AudioServerUDP para que realice los ajustes necesarios. Como parte del mecanismo de control de piso, cuando una de las computadoras cliente es seleccionada como la fuente de audio, el participante que la utiliza es además enfocado con una de las cámaras de video. Esto permite registrar información visual acerca de la intervención realizada que puede ser observada por participantes remotos o que puede anexarse a la memoria de la reunión.

Si dos o más estaciones generan información de audio al mismo tiempo, como cuando los participantes aplauden o se ríen, el software de control instruye a una de las cámaras para tomar una vista global de la sala y deja la dirección de la fuente de audio fija, es decir, se registra la información de audio de una de las computadoras (Rodríguez, 1998).

5.3.2 Formato de los datos de audio.

Al registrar los datos de audio que son enviados por las computadoras cliente, el servidor necesita primero conocer el formato en el que la información será recibida para su decodificación o su almacenamiento y consulta posterior. Conforme a las herramientas

utilizadas por el programa cliente de manejo de sonido, éste es registrado a 8 kHz (8000 muestras por segundo), en datos de 8 bits, (con valores de 0 a 255, en donde 128 representa el silencio absoluto) utilizando un esquema de codificación μ -law. De esta manera, los datos de audio recibidos por el servidor deben ser precedidos por el encabezado que se muestra en la Figura 9, cada uno de los campos tiene una longitud de 4 bytes. El identificador de archivo es un número especial utilizado para identificar al bloque de datos como información de audio. El campo de longitud de datos es opcional ya que por ejemplo, si se desea decodificar el sonido mientras es recibido desde los programas cliente, no se puede saber de antemano que tan larga será la transmisión, en ese caso, los datos son leídos hasta el final del bloque de datos. El valor de 1 en el campo de canales de lectura indica que el audio se registra en modo *Mono* y no en *Stereo*. Debido a que la longitud del encabezado debe ser de 32 bytes, el espacio que queda entre el fin de los datos del formato y el fin del encabezado es rellenado con 0's.

0x2e736e64	identificador de archivo
32	tamaño de encabezado (bytes)
0 ó longitud	longitud de datos (opcional)
1 (μ -law)	codificación
8000	lecturas por segundo
1	canales de lectura

Figura 9. Encabezado de los datos de audio manejados por el servidor de audio.

El formato μ -law (pronunciado “mu-law”) puede ser utilizado tanto en plataformas PC-Windows, Macintosh así como en varios tipos de sistemas UNIX. Este formato es un estándar internacional para la compresión de audio, generalmente utilizado para grabar voz en datos de 16 y 8 bits, con una relación de compresión de 2:1. Es comúnmente utilizado en aplicaciones de telefonía y en Internet para codificar archivos de audio, casi siempre con extensiones “.au” o “.snd”. Cada lectura del tipo μ -law es registrada en 8 bits utilizando un método logarítmico, a diferencia de otros formatos que utilizan un método lineal, por ejemplo el formato AIFF. Esto quiere decir que el espacio entre los diferentes niveles de sonido crece progresivamente conforme el nivel de audio es incrementado, obteniéndose como resultado un rango dinámico mayor a los censados de 8 bits normales (ESCAtech, 1999). Con este formato se puede llegar a representar un rango dinámico equivalente a la utilización de 12 bits con un esquema de codificación lineal (Van Rossum, 1994). El formato μ -law fue popularizado por Sun Microsystems y Next Computers, por lo que el nombre de este formato es frecuentemente intercambiado por archivos Sun-AU o archivos de sonido Next. Otros nombres para designar este formato son mu-law y u-law (Silicon Graphics, 1999). En el apéndice A se muestra una comparación entre el formato μ -law y otros formatos para la compresión de audio.

5.4 Aplicación cliente.

La aplicación cliente consiste principalmente de un Thread o hilo de ejecución que se encuentra censando permanentemente el nivel de sonido a través de un micrófono

instalado en cada una de las computadoras cliente. La clase AudioRecorder (ver Figura 7), después de activar a SoundBiteSenderUDP y de comunicarle la información acerca de la aplicación servidor, entra en un estado de espera para después terminar su ejecución, dejando a SoundBiteSenderUDP hacer el resto del trabajo. SoundBiteSenderUDP cuenta con un canal de comunicación abierto hacia el servidor del software de control de la Sala de Reuniones Electrónicas, a través del cual le notifica si existe información de audio que debiera ser registrada, en caso de que así sea, el software de control notifica a AudioServerUDP que debe cambiar la dirección de su fuente de sonido a la de la computadora cliente que lo acaba de solicitar.

5.4.1 El Medical Collaboration Project de la Universidad de Michigan.

Dada la carencia de una herramienta para registrar sonido dentro de las clases contenidas en las bibliotecas del lenguaje Java, fue necesario evaluar otras posibilidades, entre ellas, algunas utilerías escritas en lenguaje C, ensamblador y software comercial para el manejo de contenido multimedia. De todas las alternativas exploradas, un conjunto de clases escritas en Java como parte del Medical Collaboration Project de la Universidad de Michigan resultó ser la mejor opción. Este proyecto está enfocado al desarrollo de un conjunto de bibliotecas y aplicaciones para facilitar la colaboración asíncrona entre radiólogos y médicos generales. Los integrantes de este proyecto han desarrollado un sistema para grabar la utilización de aplicaciones y applets escritas en Java (U. Michigan, 1999). Además, desarrollaron una clase con la cual puede ser capturado sonido en

computadoras con el sistema operativo Solaris y Windows95, la clase SoundBite. Esta clase utiliza una biblioteca escrita en C++ para acceder los datos de la tarjeta de audio y almacenarlos en memoria para su posterior utilización. Además de poder acceder la información de audio desde un programa escrito en Java, esta biblioteca cuenta una licencia que permite su uso para el desarrollo de otras aplicaciones y se distribuye con todo y código fuente. Es por esto que la clase SoundBite resultó ser la herramienta ideal.

5.4.2 La clase SoundBite.

Aunque la clase SoundBite proporcionaba mucha de la funcionalidad requerida para el desarrollo de este proyecto, era necesario hacer algunas modificaciones. La clase SoundBite original grababa toda la información de audio en memoria y esta información sólo podía ser almacenada o transmitida al final de una sesión de grabación. Esto provocaba que los primeros programas de prueba agotaran este recurso al ser utilizados por períodos largos de tiempo. Fue entonces que se creó la clase SoundBiteSenderUDP modificando la funcionalidad de SoundBite para que la información de audio fuera transmitida por la red a otra computadora o escrita directamente a un archivo tan pronto como ésta fuera adquirida por el micrófono.

5.4.3 SoundBiteSenderUDP y WriterThreadUDP.

La clase `SoundBiteSenderUDP` fue construida como una subclase de `SoundBite`, heredando de ésta su funcionalidad. Algunos de los métodos de `SoundBite` fueron redefinidos en `SoundBiteSenderUDP` y fue necesaria la creación de un nuevo componente, la clase `WriterThreadUDP` (ver Figura 7). Al igual que su clase padre, `SoundBiteSenderUDP` es primordialmente un hilo de ejecución (`Thread`) que se encuentra censando de manera ininterrumpida el audio registrado por el micrófono de la computadora donde se ejecuta, pero en vez de almacenar los datos en memoria, como lo hace `SoundBite`, la información es pasada a `WriterThreadUDP`. La clase `WriterThreadUDP` crea también un hilo de ejecución independiente que se encuentra activo siempre que `SoundBiteSenderUDP` está obteniendo información de audio. `WriterThreadUDP` se encarga de tomar la información de audio y mandarla a la computadora servidor a través de la red, liberando a `SoundBiteSenderUDP` de esta tarea.

La re-utilización del software del Medical Collaboration Project se realizó con permiso del autor original.

5.4.4 Detección de voz.

Aún cuando el programa cliente se encuentra censando permanentemente el micrófono a través de la clase `SoundBiteSenderUDP`, no se manda información alguna al servidor cuando no se detectan datos en el micrófono. Para detectar si la persona que utiliza la computadora en donde se ejecuta la aplicación cliente está hablando debe definirse un

umbral de sonido. Cuando el nivel de sonido obtenido por el micrófono sobrepasa, por un período de tiempo razonable, el valor de un umbral pre-definido, se manda una señal al servidor solicitando el control de piso y la información de audio empieza a ser transmitida al servidor de audio. Debido a que el valor 128 representa el silencio absoluto, si uno de los datos se encuentra entre $128 + \text{umbral}$ y $128 - \text{umbral}$, el dato no debe ser tomado en cuenta, el micrófono ha registrado sólo ruido ambiental. Si el dato se encuentra fuera de este rango, entonces se debe considerar como información de audio. El valor de umbral es necesario ya que los datos de entrada pueden experimentar interferencias que provoquen que el sonido nunca tenga el valor de 128 que representa el silencio absoluto (Munnik et. al., 1994), ver Figura 10.

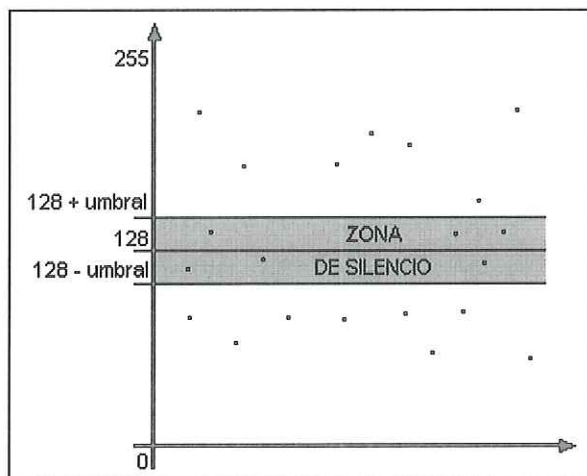


Figura 10. Los datos de audio dentro de la zona gris son considerados como silencio.

También es necesario que el valor de sonido sobrepase el valor de umbral por un cierto período de tiempo para informar al servidor que una participación se está realizando.

SoundBiteSenderUDP hace la solicitud de control de piso cuando registra que más del 10% de la información censada se encuentra fuera de la zona de silencio. Esto es, si al hacer 8000 lecturas se detectan más de 800 con información fuera del área de silencio, entonces el usuario debe estar hablando. Lo anterior con la finalidad de evitar que sonidos de muy corta duración como el golpeteo de una pluma, un estornudo o el cerrar de una puerta sean considerados como participaciones y se realice una asignación de control de piso errónea.

5.5 Comunicación Cliente-Servidor.

La comunicación entre los programas cliente y el servidor se realizó utilizando el soporte de Java para la familia de protocolos de red TCP/IP. Este conjunto de especificaciones fue originalmente diseñado para permitir la comunicación entre sistemas de cómputo diferentes, como un proyecto de la Agencia de Proyectos de Investigación Avanzada de E.U. (DARPA). El Protocolo de Control de Transporte / Protocolo de Internet, (Transport Control Protocol / Internet Protocol) TCP/IP, permite el acceso público a sus protocolos de red, facilita la integración entre todos los ambientes de computación que deseen operar bajo un esquema de red (Naugle, 1994) y su popularidad dió origen a Internet.

La designación de nombres para programas servidor dentro de un ambiente de red es un problema difícil, ya que múltiples servidores pueden estarse ejecutando en la misma computadora. La convención más común para resolver este problema es el de asignar a los

programas servidor un número entero llamado puerto. El servidor entonces escucha las peticiones de las aplicaciones cliente en un puerto predefinido para el servicio que presta en particular (Robbins et. al., 1996). De esta forma, la dirección de red de la computadora donde se ejecuta y el número de puerto a través del cual proporciona su servicio identifican de manera única a las aplicaciones servidor. Esta combinación de dirección y puerto es conocida como Socket. (Naugle, 1994). Sin embargo, el número de puerto para una aplicación servidor no puede establecerse de manera arbitraria. Los puertos menores a 1024 se encuentran reservados para servicios privilegiados, importantes para la operación de algunos sistemas operativos y aún cuando es posible utilizar un número de puerto mayor a 1024, es recomendable asignar a nuevas aplicaciones números superiores a 50000 (Leffler et. al., 1989).

El servidor AudioServerUDP tiene asignado como puerto por omisión el 1900, mientras la aplicación servidor del software de control (MultiServer), escucha peticiones de servicio en el puerto 8189.

5.5.1 Conexión.

Al iniciar una sesión de trabajo, el servidor del software de control de la sala comienza a escuchar peticiones de servicio por parte de los clientes en su puerto correspondiente. Cuando cada una de las aplicaciones cliente comienza su ejecución, establece de manera inmediata una conexión con el servidor para la transmisión de los datos

de control. Mientras que el software de control de la sala mantiene esta conexión activa durante toda la sesión, el servidor de audio no crea una conexión explícita con sus respectivos clientes para realizar la transmisión de información.

Cuando el software de control termina una sesión de trabajo, las conexiones entre éste y las aplicaciones cliente son finalizadas.

5.5.2 Transmisión de datos.

Dentro de la familia de protocolos TCP/IP existen dos protocolos para transportar datos de una computadora hacia otra, un protocolo orientado a conexión y un protocolo sin conexión.

El protocolo orientado a conexión es el llamado TCP (Transport Control Protocol). En este modelo de transporte de datos la aplicación servidor espera una solicitud de conexión de uno de los clientes, cuando esta conexión se establece, el intercambio de datos se realiza utilizando descriptores de archivo (como si la conexión fuera un archivo local) y no es necesario incluir en el mensaje la dirección del servidor destino. Una vez establecida, una conexión TCP es manejada como una secuencia continua de bytes y no como porciones discretas de información. El TCP fue diseñado específicamente para ofrecer un flujo de información confiable entre dos puntos utilizando una red no confiable (Tanenbaum, 1996). Este protocolo cuenta con algoritmos robustos para la detección de

errores que utilizan números de secuencia y mensajes de confirmación de datos recibidos al mandar datos de una estación a otra. Al transmitir información de un punto a otro de la red, los datos son divididos en paquetes que son enviados de manera individual. La utilización de números de secuencia permite ordenar los paquetes de manera correcta en la computadora receptora y los mensajes de confirmación son utilizados para asegurar que la información llegó a su destino de manera íntegra. Aunque todo esto se realiza con el fin de que la transmisión de la información sea lo más confiable posible, debe considerarse también que el proceso de verificación agrega cierto retardo al envío y la recepción de los datos (Tanenbaum, 1996).

El protocolo UDP, Protocolo de Datagramas de Usuario (User Datagram Protocol), es el que no requiere conexión y es básicamente el Protocolo de Internet (IP, Internet Protocol) con un pequeño encabezado extra (Tanenbaum, 1996). Cuando la comunicación no es orientada a conexión el cliente manda al servidor un sólo mensaje, el servidor realiza el servicio solicitado y generalmente regresa el resultado obtenido a la aplicación cliente (Robbins et. al., 1996). El UDP provee un servicio de transporte de datos no confiable, sin necesidad de conexión, no tiene soporte para enviar mensajes de confirmación desde el receptor al emisor del mensaje al recibir datos, no tiene un mecanismo para ordenar los paquetes de datos que recibe y se pueden perder o duplicar paquetes sin enviar un mensaje de error a quien los envía. Lo único que el UDP ofrece es la asignación y el manejo de números de puerto para identificar de manera única las aplicaciones que se ejecutan en una computadora. Este protocolo es generalmente más rápido en la transmisión de información

que TCP debido al bajo sobre-procesamiento (overhead) involucrado en su funcionamiento y es comúnmente utilizado en aplicaciones que no necesitan un transporte de información muy confiable o en aplicaciones en que el tiempo es importante, por ejemplo, aplicaciones de manejo de redes, servidores de nombres, sistemas distribuidos de archivos (NFS), transmisión de audio o video en una red local de datos, etc. (Santifaller, 1991). A diferencia de TCP, los datos transmitidos con UDP son enviados y recibidos como paquetes individuales.

La transmisión de datos de control entre el software de control de la Sala de Reuniones Electrónicas y el cliente de audio se realizó utilizando TCP. Esto debido a que se debía asegurar que la información de control fuera recibida de manera correcta y en orden. Además, la velocidad de transmisión no era una prioridad para estos datos. Para establecer esta conexión se utilizó la clase Socket del paquete `java.net` de Java. La clase Socket es la representación de una conexión de red TCP en Java (Hughes et. al., 1997).

La transmisión de datos entre el cliente y el servidor de audio utiliza el protocolo UDP. Entre estas dos aplicaciones los datos fluyen sólo desde el cliente hacia el servidor y dada la aplicación, el hecho de perder un paquete de información o recibir alguno duplicado o fuera de orden se considera que no afecta mucho la calidad de audio final que el servidor recibe. En este caso, además, existen situaciones en las que se requiere que la transmisión sea lo más rápido posible, como cuando se desea reproducir el audio “en tiempo real” o enviarlo a personas localizadas fuera de la Sala de Reuniones Electrónicas. El cambio de

control de piso también se dificulta con la utilización de TCP. El tener que mantener varias sesiones de comunicación abiertas simultáneamente o estar creando y destruyendo conexiones provoca problemas de retardo y sincronización en la señal de audio.

El enviar un paquete UDP involucra la creación de un objeto del tipo `DatagramPacket`, el cual consiste en el cuerpo del mensaje a enviar y la dirección destino, este `DatagramPacket` puede entonces enviarse al receptor a través de la red. Para recibir un paquete UDP primero es necesario crear un `DatagramPacket` para ahí depositar el paquete que se recibe de la red. Una vez recibido el paquete, la dirección origen y el contenido del mensaje pueden ser extraídos del `DatagramPacket`. Con UDP no se pueden restringir las direcciones de donde se desea recibir información, el protocolo recibe el primer paquete que llega con la dirección de la computadora en donde se encuentra activo (Hughes et. al., 1997). La clase `WriterThreadUDP` es la encargada, en la aplicación cliente, del manejo del audio, de construir los paquetes UDP con la información de audio generada por `SoundBiteSenderUDP` y enviarlos al servidor de audio.

La clase `DatagramSocket` es utilizada tanto para enviar como para recibir `DatagramPackets`. De la misma forma que en TCP, a un `DatagramSocket` se le debe asignar un puerto en el cual pueda recibir información (Hughes et. al., 1997). Las clases `DatagramPacket` y `DatagramSocket` forma parte del paquete `java.net`. `AudioServerUDP` cuenta con un `DatagramSocket` para recibir los paquetes de audio que los clientes envían (ver Figura 6).

5.6 Seguridad.

El esquema de seguridad utilizado por la aplicación para el manejo de audio es un tanto primitivo, restringiendo el rango de direcciones de red de donde puede recibir información tomando como base solamente la información que el software de control de la Sala de Reuniones Electrónicas provee. Utilizar un método de seguridad más elaborado podría traer repercusiones en el desempeño de la aplicación, produciendo una menor calidad de audio. Un esquema más estricto podría también entorpecer la participación de los usuarios en las sesiones de trabajo disminuyendo el intercambio de información.

5.7 Configuración.

No existe la necesidad de configurar el servidor de audio ya que la información que necesita para operar la toma del resto del software de control de la sala, una vez configurado éste último, el servidor de audio puede ser utilizado.

Por otro lado, las aplicaciones cliente requieren un poco más de atención. La configuración de los clientes es fácil y no tiene que realizarse sino cuando se hacen cambios en la computadora servidor. Los datos que se tienen que configurar son: la dirección de red de la computadora servidor, el número de puerto del software de control de la sala y el

número de puerto del servidor de audio. Una vez configurados estos parámetros con valores válidos, las aplicaciones cliente están listas para operar.

5.8 Utilización.

Esta aplicación está totalmente integrada al software de control de la sala y su utilización debiera ser solamente como un subconjunto de éste. Ni el operador de la sala ni los usuarios de la misma debieran notar diferencias entre el software de control y el servidor de audio. Incluso, los datos de audio generados con esta aplicación, deben verse como algo generado por el software de control de la Sala de Reuniones Electrónicas.

Capítulo 6. Pruebas y resultados.

Las pruebas que a continuación se describen se realizaron dentro de la Sala de Reuniones Electrónicas del CICESE. La computadora que funciona como servidor dentro de la sala cuenta con un procesador Pentium de 200 Mhz, 64 MB de memoria RAM y un espacio de almacenamiento de 10 GB. Las computadoras cliente tienen procesadores Pentium de 166 Mhz, 16 MB de memoria RAM y dispositivos de almacenamiento con capacidades mayores a los 2 GB. Todas las computadoras utilizadas cuentan con una tarjeta de audio y un micrófono.

6.1 Pruebas realizadas al servidor y al cliente de audio.

El sistema para manejar el audio dentro de la Sala de Reuniones Electrónicas del CICESE, tanto en su porción cliente como su parte servidor, fue construido como un sistema independiente. Una vez terminado, también fue probado de manera aislada antes de ser integrado al resto del software de control de la sala.

Para probar el servidor y los clientes de audio de manera independiente se creó una interfaz gráfica, en forma de una ventana, desde donde es posible seleccionar de manera manual la fuente de audio deseada. En esta ventana, se representan las computadoras cliente como un conjunto de botones, los cuales tienen como etiqueta la dirección de red de la computadora que representan. La selección de la fuente de audio se realiza presionando

alguno de los botones de la interfaz. El nombre de la computadora seleccionada es también mostrado en forma de un pequeño letrero en la parte inferior de la ventana (ver Figura 11).

Al utilizar la interfaz presentada en la Figura 11 se manipula directamente el comportamiento del servidor de audio, esto permitió probar la funcionalidad del servidor y la comunicación entre éste y las aplicaciones cliente. Las direcciones de red que aparecen en los botones de la interfaz muestran los nombres de las computadoras en donde pueden ejecutarse programas cliente y desde donde es posible enviar audio al servidor. Esto es, si una computadora trata de enviar información de audio al servidor, pero la dirección de dicha computadora no es visible en la interfaz, la información es desechada.

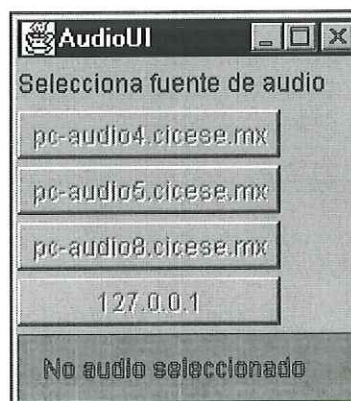


Figura 11. Interfaz de la aplicación para probar de manera aislada el sistema de audio (servidor).

En la Figura 11 se puede observar que ninguno de los botones puede ser seleccionado, esto debido a que el servidor no ha recibido ninguna conexión de parte de alguna de las computadoras desde donde es posible recibir información de audio. El letrero de la parte inferior indica que tampoco se encuentra seleccionada alguna computadora

cliente como fuente de audio. Cuando el servidor recibe una petición de conexión desde uno de los programas cliente, el botón asociado a su dirección de red se habilita y el cliente puede ser elegido por el usuario de la interfaz como fuente de audio. La información de las computadoras cliente válidas es almacenada en un archivo de texto con un formato similar a algunos archivos de configuración utilizados en sistemas UNIX. Para la interfaz que se muestra en la Figura 11 se utilizó el siguiente archivo:

```
# este es un comentario del archivo audiohosts
# es el archivo donde se declara que maquinas pueden
# conectarse como clientes al servidor de audio

pc-audio4.cicese.mx
pc-audio5.cicese.mx
pc-audio8.cicese.mx
127.0.0.1
```

En este archivo los espacios y líneas en blanco así como los caracteres de tabulado son desechados, las líneas de texto que inician con el caracter “#” son considerados como comentarios y tampoco son tomados en cuenta al construir la tabla de direcciones de las computadoras cliente. La utilización de este archivo es innecesaria al utilizar el servidor de audio junto al resto del software de control de la sala, ya que la información de las máquinas cliente es proporcionada por el módulo de configuración del software de control.

Se creó también un programa cliente de prueba a través del cual se puede establecer comunicación con la aplicación servidor. Este programa de prueba manda la información de audio registrada en la estación de la sala donde se ejecuta la aplicación servidor. Este programa tiene una interfaz en forma de línea de comando y requiere como parámetros la

dirección de red y el número de puerto en donde se encuentra ejecutándose el servidor de audio.

Con el fin de probar la aplicación para el manejo de audio como un componente independiente, tanto el servidor de audio como la interfaz para su manipulación discutida anteriormente, fueron instalados en la computadora servidor de la Sala de Reuniones Electrónicas. Copias de los programas cliente de prueba fueron también instaladas en varias estaciones de la sala para establecer conexiones con el servidor de audio. Después de configurar al servidor con las direcciones de las computadoras cliente y a los programas cliente con la dirección de red y puerto del programa servidor se realizaron las pruebas al software. Una persona activaba el servidor de audio y lo manipulaba desde la computadora servidor de la sala mientras dos personas inicializaban y operaban programas cliente en dos de las computadoras de la Sala de Reuniones Electrónicas. El audio recibido en el servidor de audio era reproducido inmediatamente. La calidad del audio reproducido es bastante buena, sin embargo, es perceptible un retraso de entre 1 y 3 segundos entre la generación de la señal de sonido en la computadora cliente y su reproducción en la aplicación servidor. El lapso de tiempo necesario para hacer el cambio de fuente de audio resultó ser mínimo. No fue posible realizar pruebas de este tipo con un número mayor de usuarios o por períodos de tiempo mayores a 45 minutos.

Habiendo comprobado la operación del servidor de audio se procedió a incorporarlo al software de control de la Sala de Reuniones Electrónicas. Las pruebas al

software de control de la sala fueron realizadas cuando el servidor de audio había ya sido incluido y son descritas a continuación.

6.1.1 Pruebas al software de control de la Sala de Reuniones Electrónicas.

Una vez integrado al software de control de la sala, se realizaron pruebas con el fin de verificar su funcionalidad. Tanto el servidor de audio como los programas cliente para la transmisión del audio funcionaron de manera correcta en conjunción con los demás módulos del software de control de la sala. El manejo del control de piso con el uso de la información de audio de las computadoras cliente resultó ser una de las funciones más útiles en la administración de recursos de la sala, permitiendo manipular las cámaras de video sin necesidad de un operador.

Aún cuando las pruebas realizadas al software de control fueron también realizadas con grupos reducidos de personas, éstas fueron hechas en sesiones de más de 30 minutos.

6.2 Calidad del audio registrado.

Al igual que en el momento de las pruebas realizadas en el servidor y los clientes de audio como un componente independiente, la calidad de audio obtenida al utilizarse ya integrado con el software de control de la sala es adecuada. Sin embargo, el retraso en la

transmisión de la información de audio también se mantiene al utilizar el servidor dentro del software de control.

6.2.1 Valores de silencio absoluto.

Un resultado inesperado fue descubierto al probar los programas cliente. El valor teórico de “silencio absoluto” (supuestamente siempre igual a 128) no siempre fue el mismo en las computadoras donde el software fue probado. Esto provocó que se tuviera que calibrar este valor de silencio calculando la media de los datos de audio registrados por un lapso de tiempo en el que no era usado el micrófono de la computadora, encontrando por ejemplo, algunas computadoras con valores de silencio absoluto de 92 y 110. Esto es causado por las diferencias en los dispositivos con los que cuentan las computadoras de la sala (micrófonos de diferente calidad, diferentes tarjetas de audio).

6.2.2 Valores umbral de detección de voz.

Las diferencias en los dispositivos de las computadoras provocaron también valores de umbral de voz diferentes. Micrófonos más sensibles requerían valores de umbral menores a los que eran necesarios en micrófonos con una menor sensibilidad. Mientras en algunas computadoras un valor umbral de 5 era suficiente, en otras se tenían que utilizar valores de hasta 15. Estos valores umbral tuvieron que ser calculados a manera de prueba y error en las computadoras cliente en donde se presentaban problemas como el de siempre

registrar audio, aún cuando nadie utilizara el micrófono, o no registrar sonido alguno cuando alguien realizaba una participación.

6.3 Longitud de los archivos de audio.

Los archivos de audio generados durante las sesiones de prueba de 30 minutos del software de control, ocuparon un espacio de alrededor de 15 megabytes.

Capítulo 7. Discusión.

En este capítulo se presenta una discusión acerca de los resultados de las pruebas realizadas a la aplicación para el manejo de audio y al software de control de la Sala de Reuniones Electrónicas del CICESE.

7.1 Funcionamiento.

Las pruebas al módulo de audio de manera independiente se hicieron por espacio de hasta 45 minutos y la duración de cada una de las pruebas hecha al software de control, con el módulo de audio integrado, fue de cuando menos 30 minutos. Ni las pruebas realizadas al módulo de audio, ni las realizadas a todo el software de control de la sala de reuniones, consumieron de manera importante los recursos de las computadoras donde se realizaron y la cantidad de audio que podían registrar era sólo limitada por el espacio de almacenamiento disponible.

La calidad del audio obtenida al realizar las pruebas al módulo de manejo de audio como una aplicación independiente fue suficiente. Esta calidad no se vio afectada al integrar el módulo de audio al resto del software de control de la sala y el funcionamiento de la biblioteca SoundBite, desarrollada en la Universidad de Michigan, fue el esperado.

Los archivos de audio generados en las sesiones de 30 minutos ocuparon un espacio de alrededor de 15 megabytes. Esto concuerda con las especificaciones del formato de audio μ -law. Este formato registra 8000 muestras de 8 bits por segundo (Van Rossum, 1994), por lo que dos minutos de datos de audio ocupan un espacio de almacenamiento de aproximadamente 1 megabyte. Si multiplicamos estos dos minutos por 15 obtenemos entonces 30 minutos de audio, ocupando un espacio aproximado de 15 megabytes.

7.2 Protocolo de transporte de datos.

La calidad del audio no se vio tampoco afectada por la utilización del protocolo UDP como medio de transporte a través de la red local de datos, protocolo que no cuenta con mecanismos de control de conexión ni de corrección de errores. Inicialmente se utilizó el protocolo TCP debido a su facilidad de uso y sus mecanismos para el control de errores. Sin embargo, buscando aumentar la velocidad de transmisión de datos, se decidió cambiar al protocolo UDP para evitar con esto el procesamiento que el TCP realiza para el establecimiento y mantenimiento de conexiones y para corregir los errores de transmisión. La ganancia en la velocidad de transmisión fue pequeña, pero perceptible. La red local no era muy utilizada al realizar las pruebas, por lo que la probabilidad de colisiones de paquetes UDP debe haber sido muy baja. Desgraciadamente no fue posible conseguir una herramienta con la que se pudiera medir de manera precisa la diferencia en la utilización de ambos protocolos para la plataforma en la que la aplicación fue desarrollada. El retardo que se obtiene al utilizar TCP en vez de UDP es de entre 1 y 2 segundos, aproximadamente.

7.3 Manejo de control de piso.

Usar el audio como un mecanismo de control de piso resultó ser útil para los participantes en las sesiones de prueba. Debido a que el habla es una forma natural de comunicación, resultó para los usuarios un mecanismo muy familiar para iniciar una participación. El que los usuarios utilicen la voz como herramienta para pedir en forma explícita el control de piso les permitió además mantener las manos libres para realizar otras actividades como accionar el teclado o mover el ratón de sus computadoras, esto en contraste con otros métodos como el conocido “Push to Talk” (Presione para hablar), en donde el usuario debe interrumpir la actividad que realiza, o abandonar la sección de la pantalla en la que trabaja, para poder comunicarse. Este protocolo de tipo social es muy similar a la forma en que se conduce una plática cotidiana y es por supuesto más intuitivo y fácil de usar que otros protocolos de tipo tecnológico (como el “Push to Talk”) (Watson, 1996). Utilizar este mecanismo, junto con el módulo de control para manipular las cámaras de la Sala de Reuniones Electrónicas, permite que el usuario pueda ser enfocado automáticamente, sin la necesidad de un operador de cámaras.

7.4 Reutilización del módulo de manejo de audio de la Sala de Reuniones Electrónicas.

El módulo de audio se diseñó para que pudiera ser fácilmente modificado o integrado a otras aplicaciones. De hecho, durante el desarrollo de este trabajo fue modificado en varias ocasiones para cumplir con diferentes funciones. Primeramente se

construyó como una herramienta independiente para después ser integrado al software de control de la Sala de Reuniones Electrónicas y al final se hicieron modificaciones para facilitar el almacenamiento y recuperación de los datos de audio. Los cambios hechos a la aplicación para su modificación no fueron muchos, así que, de la misma forma en que el servidor de audio fue incluido como un componente del software de control de la sala, puede también ser incluido en alguna otra aplicación que requiera del manejo de audio.

7.5 Otras consideraciones.

Uno de los requerimientos del software de control de la sala era ofrecer la posibilidad de redireccionar los datos de audio a otras aplicaciones para su transferencia a participantes remotos. Sin embargo, esto no pudo ser implementado ya que involucra que dos aplicaciones (el software de control y la aplicación para la distribución del audio, NetMeeting, por ejemplo) tengan acceso, al mismo tiempo, a la tarjeta de audio de la computadora servidor. Aún cuando esto es posible con tarjetas de audio con capacidad de comunicación full duplex (se puede leer y escribir en la tarjeta al mismo tiempo), esto requiere acceder el dispositivo a muy bajo nivel, lo cual no es posible con el tipo de bibliotecas que se utilizaron en el desarrollo del servidor de audio. Una solución a este problema es agregar una tarjeta de audio extra. Esta situación también se hubiera podido resolver construyendo una aplicación para emular un comportamiento tipo full duplex en el dispositivo de audio, pero la implementación de esta aplicación está fuera del alcance del presente trabajo.

Al final se realizaron modificaciones al servidor de audio para que cada una de las participaciones fuera registrada en un archivo de audio por separado. Así, en vez de tener un sólo archivo con los datos de sonido de toda la reunión, las sesiones pueden segmentarse por participación, para de esta forma poder acceder de manera más fácil y rápida alguna porción particular de la sesión. Estas modificaciones no fueron incorporadas al software de control de la Sala de Reuniones Electrónicas.

7.6 Trabajo futuro.

Al servidor de audio podría agregarse una configuración más flexible que permitiera al operador de la sala cambiar dinámicamente entre grabar las participaciones en un archivo, reproducirlas mientras son generadas o enviarlas a un lugar remoto sin tener que detener la sesión de trabajo. Incrementar el nivel de seguridad resulta ser un tema un tanto complicado, pero el esquema expuesto en este trabajo pudiera ser mejorado utilizando nombres de usuarios y contraseñas.

Se puede también cambiar la utilización de la clase SoundBite por otra herramienta más independiente de la plataforma. Actualmente se encuentra en evaluación la nueva versión del Java Media Framework (JMF versión 2.0 Beta), una serie de herramientas para el manejo de contenido multimedia, la cual parece ser una muy buena opción para resolver el problema de la portabilidad.

Capítulo 8. Conclusiones.

El objetivo de este trabajo de tesis, diseñar y desarrollar un servidor de audio para la Sala de Reuniones Electrónicas del CICESE, fue cumplido satisfactoriamente. Durante el desarrollo de esta aplicación y su integración con el software de control de la sala se llegó a las conclusiones que a continuación se describen.

Para la realización de este trabajo fue preferible utilizar el protocolo de transporte de información UDP en vez del protocolo TCP para la transmisión de datos de audio. Aún cuando el protocolo UDP no tiene mecanismos para la corrección de errores de transmisión, la calidad del audio que se transmite desde los programas cliente al servidor de audio no se deterioró de manera significativa al realizar las pruebas dentro de la Sala de Reuniones Electrónicas y se evitó el retardo que el TCP agrega a la transmisión, al realizar manejo de conexiones y recepción y envío de mensajes de control. La forma en que se genera la información de audio, un determinado número de muestras por segundo, se presta también para la utilización de UDP. Una vez que la información correspondiente a un conjunto de lecturas se encuentra disponible para su transmisión al servidor, ésta es encapsulada en un paquete UDP y enviada a través de la red de datos. Mientras se realiza la transmisión de un paquete de datos UDP, un nuevo ciclo de lectura de audio se completa para una transmisión de datos posterior. Tomando en cuenta además que la comunicación se realiza dentro de una red local de datos, la posibilidad de que los paquetes de

información se pierdan, se dupliquen o arriben fuera de orden a su destino es pequeña (Comer, 1996).

La calidad de audio obtenida utilizando el formato μ -law es suficiente para establecer un canal de comunicación entre dos personas y para registrar las participaciones dentro de la memoria de una sesión de trabajo. Aún cuando este formato es considerado de baja calidad con respecto a otros (ver apéndice A), resulta ser suficiente cuando el sonido consiste solamente en datos de voz. Además, la cantidad de datos necesarios para su codificación no es muy grande, por lo que esta información puede ser almacenada en archivos o transmitida a través de una red de datos. Una ventaja extra del μ -law es que en él se puede representar un rango de datos mayor a otros esquemas de codificación (Munnik, 1994).

La facilidad para acceder los recursos disponibles en una red de datos a través de las clases contenidas en las bibliotecas `java.net` del lenguaje Java facilitaron el desarrollo de esta aplicación bajo el esquema Cliente-Servidor. Sin embargo, el que Java sea un lenguaje interpretado provoca un retardo en la transmisión y recepción de los datos. Este retardo, aunque sensible cuando se debe transmitir audio en tiempo real, no tiene tanta importancia cuando lo único que es necesario es registrar las participaciones de los usuarios de la sala en la memoria de la sesión de trabajo.

Es necesario que Java incluya soporte para registrar audio. El que se haya recurrido a bibliotecas externas, escritas en C++, para poder censar la información de audio resta

portabilidad a esta aplicación. Siendo la portabilidad uno de los atributos más importantes de Java, es necesario que este tipo de limitaciones o dependencias sean eliminadas.

Actualmente se trabaja en esta dirección con el desarrollo de la 2a. versión del Java Media Framework (JMF), un conjunto de bibliotecas para la captura y transmisión de contenido multimedia utilizando Java. El JMF 2.0 se encuentra en la actualidad en estado Beta y proporciona, entre otras cosas, una funcionalidad similar a la lograda en este trabajo con la mezcla del protocolo UDP y la clase SoundBite.

Las diferencias en la calidad de los dispositivos utilizados al registrar audio pueden ocasionar problemas de lectura. Los diferentes niveles de sensibilidad presentes en los micrófonos de la Sala de Reuniones Electrónicas provocaron variaciones tanto en los valores de silencio absoluto como en los valores umbral utilizados. Este problema puede ser resuelto utilizando dispositivos homogéneos y de mayor calidad.

Aunque el servidor de audio de la Sala de Reuniones Electrónicas del CICESE tiene un nivel de funcionalidad aceptable, existen secciones de él que pueden ser mejoradas o extendidas y lograr con esto aplicaciones que integren otro tipo de interacciones tales como envío de mensajes o transmisión de video.

Literatura Citada.

- Bannon, L. y K. Schmidt. 1993. CSCW: Four characters in search of a context. Studies in Computer Supported Cooperative Work. Elsevier Science Publishers.
- Bass, B. 1980. Team productivity and individual member competence. Small Group Behavior. 431-504 pp.
- Brooks, F. 1975. The Mythical Man-Month: Essays on Software Engineering. Addison-Wesley. Reading Mass.
- Coleman, David. 1997. Electronic Meetings as Today's Presentations, en Groupware Collaborative Strategies for Corporate LANs and Intranets. Prentice Hall. New Jersey. 183-191 p.
- Coleman, David. 1997a. Groupware Technology and Applications: An Overview of Groupware, en Groupware Collaborative Strategies for Corporate LANs and Intranets. Prentice Hall. New Jersey. 3-41 p.
- Comer, Douglas E. y David L. Stevens. 1996. Internetworking with TCP/IP Volume III, Client-Server Programming and Applications, BSD Socket Version. 2a. Edición. Prentice Hall. New Jersey
- Ellis, C.A., S.J. Gibbs, G.L. Rein. Enero 1991. Groupware: Some Issues and Experiences. Communications of the ACM. Vol. 34, No. 1.
- ESCAtech Media Inc. 1999. The Sound page.
<http://www.escatech.com/soundpage/au.html>
- Flanagan, David. Mayo 1997. JAVA In A Nutshell. 2a. Edición. O' Reilly & Associates. Sebastopol, California.
- Fowler, Martin y Kendall Scott. Septiembre 1997. UML Distilled, Applying the Standard Object Modeling Language. Addison-Wesley.
- Grudin, J. 1989. Why Groupware applications fail: Problems in design and evaluation. Office: Tech. People 4, 3. 245-264 pp.
- Grudin, J. 1998. Why CSCW applications fail: Problems in the design and evaluation of organizational interfaces. Proceedings of the Second Conference on Computer-Supported Cooperative Work (Portland Oregon, Sept. 26-28). ACM. New York. 85-93 pp.

- Haywood, Martha. 1998. *Managing Virtual Teams: Practical Techniques for High-Technology Project Managers*. Ed. Artech House. Boston. 199 p.
- Hindus, Debby y Chris Schmandt. Noviembre 1992. *Ubiquitous Audio: Capturing Spontaneous Collaboration*. CSCW 92 Proceedings. ACM. 210-217 pp.
- Hughes, Merlin, Conrad Hughes, Michael Shoffner, Maria Winslow. 1997. *Java Network Programming*. Manning Publications. Greenwich, Connecticut.
- Leffler, Samuel J., Robert S. Fabry, William N. Joy, Phil Lapsley. 1989. *An Advanced 4.3 BSD Interprocess Communication Tutorial*. BSD UNIX Programmer's Manual. Hewlett-Packard. Palo Alto, California.
- Lowe, Fran. 1995. *Technology Tools for Increasing Group Participation, Concensus, and Productivity*.
- Mathur, Amit G. y Atul Prakash. Noviembre 1993. *On Transport Protocols for Audio Conferencing in CSCW Environments*. Proc. of the Workshop on the Role of Real-Time in Multimedia. Durham, N.C.
- Munnik, Joshua, Oostendorp. 1994. *The Sound Blaster Book*. SYBEX. Alameda, California. p. 325.
- Naugle, Matthew. 1994. *Network Protocol Handbook*. McGraw-Hill.
- Nunamaker, Jay F., R. Alan Dennis, Joseph L. Valacich, Douglas R. Vogel, Joey F. George. Enero 1991. *Electronic Meeting Systems to Support Group Work*. Communications of the ACM. Vol. 34, No. 7.
- Nunamaker, Jay F., Robert O. Briggs, Daniel D. Mittleman. 1995. *Electronic Meeting Systems: Ten Years of Lessons Larned*. *Groupware: Technology and Applications*. Prentice Hall. New Jersey. 146-193 pp.
- Robbins, Kay A. y Steven Robbins. 1996. *Practical UNIX Programming, A Guide to Concurrency, Communication, and Multithreading*. Prentice Hall. New Jersey.
- Rodríguez, Marcela D. Octubre, 1998. *Memoria Organizacional para Revisiones Técnicas en Salas Electrónicas de Reuniones*. Tesis, Maestría en Ciencias. CICESE.
- Santifaller, Michael. 1991. *TCP/IP and NFS: Internetworking in a UNIX Environment*. Addison Wesley.
- Silicon Graphics Inc. 1999. *Creating μ -law/Sun-AU/Next and AIFF/AIFC Audio Files*. [Http://www.oz.nthu.edu.tw/~wohong/sgi/webmagic/CreatingAIFC.html](http://www.oz.nthu.edu.tw/~wohong/sgi/webmagic/CreatingAIFC.html)

Soriano, Ma. Teresa y Jesús Favela. Marzo 1997. Reporte Técnico: Trabajo Cooperativo Asistido por Computadora. CICESE. México.

Sun Microsystems. 1998. Java Platform 1.1.5 Core API Specification.
<http://java.sun.com/products/jdk/1.1/>

Tanenbaum, Andrew S. 1996. Computer Networks. 3a. Edición. Prentice Hall. New Jersey.

U. Michigan. 1999. Medical Collaboration Project. <http://spartan.si.umich.edu>

Van Rossum, Guido. Febrero 1994. FAQ: Audio File Formats.
[ftp://mitpress.mit.edu/pub/
Computer-Music-Journal/Documents/SoundFiles/SoundFile.FAQ.3.08](ftp://mitpress.mit.edu/pub/Computer-Music-Journal/Documents/SoundFiles/SoundFile.FAQ.3.08)

Watson, Anna y Martina Angela Sasse. 1996. Evaluating Audio and Video Quality in Low-Cost Multimedia Conferencing Systems. Artículo aceptado para su publicación en *Interacting with Computers*.

Apéndice A. Tabla comparativa de algunos formatos utilizados en la compresión de audio.

Formato de sonido	Plataforma	Ventajas	Desventajas	Nivel de compresión
RealAudio	Windows Macintosh UNIX	Puede tocarse en tiempo real	Calidad de sonido inferior	*****
AIFF 16 bits	Macintosh	Buena calidad de sonido	Tamaño grande de archivos	*
AIFF 8 bits	Macintosh	Tamaño medio de archivos	Sonido "granulado"	**
QuickTime 16 bits	Windows Macintosh UNIX	Muy buena calidad de sonido	Tamaño grande de archivos	*
QuickTime 8 bits	Windows Macintosh UNIX	Tamaño medio de archivos	Sonido "granulado"	**
WAV 16 bits	Windows Macintosh	Buena calidad de sonido	Tamaño grande de archivos	*
WAV 8 bits	Windows Macintosh	Tamaño medio de archivos	Sonido "granulado"	**
u-Law	Windows Macintosh UNIX	Tamaño medio de archivos	Calidad pobre de sonido	***
MPEG Audio	Windows Macintosh	Tamaño pequeño de archivos, buena calidad de sonido, puede tocarse en tiempo real	Alto compromiso entre compresión y calidad de sonido	****
MIDI	Windows Macintosh UNIX	Tamaño muy pequeño de archivos	Puede solo utilizar sonidos MIDI	*****
QuickTime Music	Macintosh con QuickTime e instrumentos musicales QT	Tamaño muy pequeño de archivos	Puede solo utilizar sonidos MIDI	*****
ESCAMixer	Windows Macintosh	Tamaño pequeño de archivos, muy buena calidad de sonido	Uso limitado con voz	***** (infinito)

En esta tabla se muestran solamente las ventajas y desventajas más importantes de cada uno de los formatos. El nivel de compresión se refiere a la relación entre tiempo de reproducción y el tamaño del archivo reproducido, entre menor es el número de caracteres * en la casilla de nivel de compresión, menor es la relación de compresión que se obtiene con dicho formato.

Información extraída de The Sound HomePage
<http://www.escatech.com/soundpage/au.html>