



COLEGIO DE POSTGRADUADOS
INSTITUCIÓN DE ENSEÑANZA E INVESTIGACIÓN EN CIENCIAS AGRÍCOLAS

CAMPUS MONTECILLO

POSTGRADO DE SOCIOECONOMÍA, ESTADÍSTICA E INFORMÁTICA
CÓMPUTO APLICADO

**MATERIAL DIDÁCTICO PREPARADO
CON U-BOOKS Y REALIDAD VIRTUAL**

JORGE ALBERTO LÓPEZ CUEVAS

T E S I S

PRESENTADA COMO REQUISITO PARCIAL
PARA OBTENER EL GRADO DE:

MAESTRO EN CIENCIAS

MONTECILLO, TEXCOCO, EDO. DE MÉXICO

2012

La presente tesis titulada: **Material Didáctico preparado con U-Books y Realidad Virtual**, realizada por el alumno: **Jorge Alberto López Cuevas** bajo la dirección del Consejo Particular indicado, ha sido aprobada por el mismo y aceptada como requisito parcial para obtener el grado de:

MAESTRO EN CIENCIAS

SOCIOECONOMÍA ESTADÍSTICA E INFORMÁTICA
CÓMPUTO APLICADO

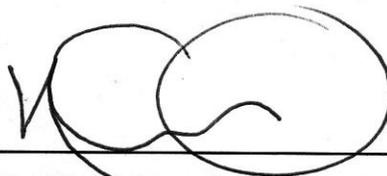
CONSEJO PARTICULAR

CONSEJERO



Dr. Juan Ricardo Bauer Mengelberg

ASESOR



Dr. Martiniano Castro Popoca

ASESOR



M.C. Margarita Cruz Millán

Montecillo, Texcoco, Estado de México, 1 de agosto de 2012

MATERIAL DIDÁCTICO PREPARADO CON U-BOOKS Y REALIDAD VIRTUAL

Jorge Alberto López Cuevas, M.C.

Colegio de Postgraduados, 2012

El análisis de diversos estilos de aprendizaje parece indicar que el uso de componentes de realidad virtual podría incrementar la comprensión, asimilación y retención del material de una clase presencial. Se agregaron elementos al paquete de software SPRP que implementa los libros sin estructura (U-Books) para desplegar simultáneamente objetos en más de un monitor conectado a la computadora. Lo mismo sucede con varios pares de bocinas. Se diseñó e implementó un dispositivo que permite conectar 3 pares de bocinas a una computadora. Para conectar varios monitores a una computadora se usa una tarjeta de video que ofrece 4 puertos, o se conecta otra computadora a la del maestro y se usan los monitores de esta última, en la que se ejecuta un programa del SPRP que recibirá comandos y archivos de la computadora principal (que hace de servidor en la red) y mostrará los objetos que le indique el programa que utiliza el maestro. El concepto básico es que se puede asociar a un elemento de una sucesión otro objeto, que se exhibirá en forma simultánea pero en otro monitor. Se describen los conceptos fundamentales de los U-Books, y se muestra cómo se despliegan los contenidos de los archivos que se deben exhibir como parte de una sucesión. Se agregó a los U-Books la posibilidad de incluir pizarrones virtuales como ítems del libro, y se describe cómo se crean y usan estos pizarrones. Un ejemplo ilustra el uso de varios monitores y pares de bocinas.

Palabras clave: Realidad virtual, múltiples monitores, múltiples bocinas, libros sin estructura, pizarrón interactivo, tecnología educativa.

DIDACTIC MATERIAL DESIGNED WITH U-BOOKS AND VIRTUAL REALITY

Jorge Alberto López Cuevas, M.C.

Colegio de Postgraduados, 2012

The analysis of various learning styles seems to indicate that the use of virtual reality components could increase the understanding, assimilation and retention of material in face-to-face classes. Elements that allow several objects to be displayed simultaneously in more than one monitor connected to the computer were added to the software package SPRP, which implements the unstructured books (U-Books). The same applies to several pairs of speakers. A device that connects 3 pairs of speakers to a computer was designed and implemented. A video card with 4 ports is used to connect 3 monitors to the computers. In the absence of such a card, another computer is connected to the presenter's computer using a local network. A program developed for this purpose will receive commands and files from the main computer (which thus acts as the network server) and displays the objects on its own monitors. The basic concepts of U-Books that several sequences of files can be created and displayed in order. To show a file on a different monitor, another file is associated to the one being shown with the indication that it should be displayed simultaneously but on another monitor. The basic concepts of U-Books are described and the way its elements are displayed is shown. The possibility of including virtual whiteboards as items of the book, as well as how to create and use them, is also described in detail. A simple example illustrates the use of multiple pairs of speakers and monitors for simultaneous displays.

Keywords: Virtual reality, multiple monitors, multi-speakers, unstructured book, interactive whiteboard, educational technology.

AGRADECIMIENTOS

Al Colegio de Postgraduados, por brindarme el apoyo y las herramientas para continuar mi formación académica.

Al Consejo Nacional de Ciencia y Tecnología (CONACYT) por el financiamiento brindado durante la realización de mis estudios de Postgrado.

A los integrantes de mi consejo Particular:

Dr. Juan Ricardo Bauer Mengelberg, por haber hecho posible este trabajo, por sus inigualables clases, pláticas y consejos, que contribuyeron ampliamente a mi formación profesional.

M.C. Margarita Cruz Millán, por sus valiosos comentarios y revisión de este trabajo.

Dr. Martiniano Castro Popoca, por su valiosa asesoría y revisión de este trabajo.

Dr. David Hebert Del Valle Paniagua, por el apoyo brindado y sus consejos.

M.C. Edgar Ramírez, por su valioso apoyo y sus aportaciones.

DEDICATORIA

A mis padres, porque creyeron en mí, me sacaron adelante dándome ejemplos dignos de trabajo y entrega... Porque gracias a ustedes, hoy puedo ver alcanzada una meta más en mi vida.

A mi Abuelita, que con la sabiduría de Dios, me has enseñado a ser quien soy.

A mis hermanas, que con su amor me han enseñado a salir adelante, gracias por su paciencia, gracias por preocuparse por su hermano menor.

Gracias a ti Liz por ayudarme a ser una mejor persona, por tu amor y sobre todo gracias por enseñarme a creer en mí y motivarme hacer las cosas de la mejor manera.

A mis Sobrinas y Cuñados, gracias por estar conmigo en todo momento, gracias por el cariño y apoyo moral que siempre he recibido de ustedes y con el cual he logrado culminar esta etapa de mi vida, que es para mí la mejor prueba de cariño y agradecimiento.

A mis amigos gracias por permitirme formar parte de su vida, gracias por presionarme para terminar este trabajo y sobre todo gracias por estar en las buenas y en las malas.

ÍNDICE

CAPÍTULO I. INTRODUCCIÓN.....	1
1.1 PLANTEAMIENTO DEL PROBLEMA.....	1
1.2 OBJETIVO GENERAL.....	2
1.2.1 OBJETIVOS ESPECÍFICOS.....	2
1.4 LA ORGANIZACIÓN DEL TRABAJO.....	3
1.5 METODOLOGÍA.....	4
CAPÍTULO II. MARCO TEÓRICO.....	5
2.1 LAS TEORÍAS RELACIONADAS CON EL APRENDIZAJE.....	5
2.1.2 ESTILOS DE APRENDIZAJE.....	5
2.1.3 MODELOS DE ESTILOS DE APRENDIZAJE.....	7
2.1.3 ESTRATEGIAS DE ENSEÑANZA.....	19
2.2. REALIDAD VIRTUAL.....	22
2.2.1 ANTECEDENTES DE LA REALIDAD VIRTUAL.....	24
2.2.2 MODELO, NIVELES Y TIPOS DE REALIDAD VIRTUAL.....	26
2.2.3 NIVELES DE TECNOLOGÍA DE LA REALIDAD VIRTUAL.....	26
2.3 REALIDAD VIRTUAL Y LA EDUCACIÓN.....	27
2.3.1 BENEFICIOS DE LA REALIDAD VIRTUAL EN LA EDUCACIÓN.....	28
CAPITULO III. U-BOOKS.....	30
DESCRIPCIÓN PANORÁMICA DE U-BOOKS.....	30
3.1 ÍTEMS DE U-BOOKS.....	31
3.1.1 PIZARRÓN COMO ÍTEM.....	32
3.2 SUCESIONES.....	33
3.2.1 ¿QUÉ SON LAS SUCESIONES EN U-BOOKS?.....	33
3.2.2 SUBSUCESIONES.....	33
3.2.3 ELEMENTO ASOCIADO A UN ELEMENTO DE UNA SUCESIÓN.....	34
3.2.4 SUCESIONES DE OBJETOS PREDISEÑADOS (PARA PIZARRONES).....	36
3.3 CONSTRUCCIÓN DE SUCESIONES.....	36
3.4 LECTURA DE UNA SUCESIÓN.....	39
3.5 LO QUE NO SE INCLUYÓ DE U-BOOKS.....	40
3.6 REALIDAD VIRTUAL EN U-BOOKS.....	41
3.7 HARDWARE Y SOFTWARE PARA LA REALIDAD VIRTUAL Y EL U-BOOK.....	43
CAPITULO IV. MÚLTIPLES PANTALLAS.....	44
4.1 TARJETAS DE VIDEO.....	44
4.1.1 CONECTORES DE VIDEO.....	47
4.2 MONITORES TOUCH-SCREEN Y MULTI-TOUCH.....	54
4.3 MÚLTIPLES PANTALLAS Y EL U-BOOK.....	55
4.3.1 PREPARACIÓN DEL ARCHIVO DE CONFIGURACIÓN.....	58
4.3.1.2 DIAGRAMA DEL PROCESO DE “CONFIG.MON.EXE”.....	67

4.3.2 PREPARACIÓN DE FORMAS Y MONITORES.....	68
4.4 IMPACTO SOBRE EL ASPECTO DE APRENDIZAJE.....	75
CAPITULO V. EL USO DE PIZARRONES VIRTUALES EN EL U-BOOK.....	76
5.1 PIZARRA DIGITAL INTERACTIVA.....	76
5.1.1 PIZARRÓN DIGITAL INTERACTIVO PORTÁTIL.....	77
5.1.2 COMPONENTES QUE INTEGRAN UNA PIZARRA DIGITAL INTERACTIVA.....	78
5.1.3 VENTAJAS Y DESVENTAJAS DE LA PIZARRA DIGITAL INTERACTIVA.....	79
5.2 PIZARRÓN SIMULADO POR EL U-BOOK.....	80
5.2.1 PIZARRONES INTERACTIVOS Y DUMMY EN SPRP.....	81
5.2.2 CATÁLOGO DE PIZARRONES.....	82
5.2.4 ACCIONES QUE PERMITE UN PIZARRÓN INTERACTIVO.....	84
5.3 PINTAR (O DIBUJAR) SOBRE EL PIZARRÓN.....	85
5.3.1 ACCIONES QUE REALIZA EL SPRP PARA PINTAR.....	86
5.3.2 NAVEGACIÓN DE UN PIZARRÓN INTERACTIVO.....	87
5.4.1 ACCIONES QUE REALIZA EL SPRP PARA ESCRIBIR.....	91
5.5 COMO SE IMPLEMENTAN LOS PIZARRONES EN UN U-BOOK.....	93
5.6 INSERTAR OBJETOS PREDISEÑADOS.....	95
5.6.1 CUADRICULA PARA IMÁGENES PREDISEÑADAS.....	96
5.6.2 ACCIONES QUE REALIZA EL SPRP PARA INSERTAR UNA IMAGEN PREDISEÑADA.....	98
5.7 MANDAR IMAGEN A UN PIZARRÓN DUMMY.....	101
5.8 IMPACTO DE LOS PIZARRONES EN LOS ASPECTOS RELACIONADOS CON EL APRENDIZAJE.....	103
CAPITULO VI. AUDIO.....	105
6.1 TARJETAS DE AUDIO.....	105
6.1.1 TARJETAS DE AUDIO MULTICANAL.....	108
6.2 BOCINAS (MÚLTIPLES SALIDAS).....	109
6.2.1 INTERFACE DE MÚLTIPLES BOCINAS.....	110
6.2.1.1 CONTROLADOR DE MÚLTIPLES BOCINAS.....	111
6.2.1.2 INTERFAZ EN ARDUINO.....	117
6.2.1.3 INTERFAZ INCLUIDA EN LOS PROGRAMAS DE SPRP.....	119
6.3 IMPACTO SOBRE EL ASPECTO DE APRENDIZAJE.....	125
CAPITULO VII. COMUNICACIÓN ENTRE DOS COMPUTADORAS.....	126
7.1 TIPOS DE REDES.....	127
7.1.1 RED DE ÁREA METROPOLITANA (MAN).....	128
7.1.1 RED DE ÁREA EXTENSA(WAN).....	128
7.1.3 RED DE ÁREA LOCAL (LAN):.....	128
7.2 MÉTODOS DE CONEXIÓN ENTRE DOS COMPUTADORAS.....	129
7.3 MÚLTIPLES COMPUTADORAS.....	131
7.3.1 ARCHIVOS COMPARTIDOS.....	132

7.3.2 COMUNICACIÓN ENTRE PROGRAMAS.....	134
7.4 RESULTADO DE LA CONEXIÓN EXITOSA.....	140
7.5 IMPACTO SOBRE EL ASPECTO DE APRENDIZAJE.....	140
CAPITULO VIII. EJEMPLO DE UNA CLASE O PRESENTACIÓN.....	141
CAPÍTULO IX. CONCLUSIONES	146
CAPÍTULO X. BIBLIOGRAFÍA.....	147
ANEXOS.....	150

ÍNDICE DE CUADROS

Tabla 1. Modelos de estilos de aprendizaje.....	20
Tabla 2. La tabla THEBLACKBOARDS de la base de datos SITEBASE.....	82
Tabla 3. Los campos de la tabla ItemBlackSpecs del libro.	94
Tabla 4. Factores que intervienen en una Mezcladora.	108
Tabla 5. Características de ATMEGA168.	112
Tabla 6. Lista de componentes para circuito electrónico.	115
Tabla 7. Acciones para el controlador ARDUINO.....	119
Tabla 8. Acciones y tipos de mensajes.	136
Tabla 9. Descripción de mensajes de comunicación.	137

ÍNDICE DE FIGURAS

Figura 1. Modelos de Estilos de aprendizaje.....	9
Figura 2. Cuadrantes Cerebrales de Herrmann.....	10
Figura 3. El Modelo de los Cuadrantes Cerebrales de Herrmann.....	10
Figura 4. El Modelo de Felder y Silverman.....	11
Figura 5. Modelo de Kolb.	12
Figura 6. Modelo de aprendizaje.....	12
Figura 7. Modelo de Programación Neurolingüística.	14
Figura 8. Modelo de los hemisferios cerebrales.	16
Figura 9. Modelo de de las Inteligencias Múltiples de Gardner.	17
Figura 10. Estrategias De Enseñanza.....	19
Figura 11. Un ítem de un U-Book.....	31
Figura 12. Una sucesión que contiene una subsucesión.....	34
Figura 13. Conector S-Video.....	47
Figura 14. Conector RCA-Video por Componentes.....	48
Figura 15. Conectores VGA.	48
Figura 16. Conector RCA-Video Compuesto.....	49
Figura 17. Conector SCART.	50
Figura 18. Conector Coaxial RF.....	50
Figura 19. Conector DVI.	51
Figura 20. Conector FireWire.....	52
Figura 21. Conector SDI.	52
Figura 22. Conector HDMI.	53
Figura 23. Conector DISPLAYPORT.....	54
Figura 24. Configuración de dos pantallas.	55
Figura 25. Tarjeta de Video ATI –RADEON HD5770.	56
Figura 26. Configuración con dos monitores y un proyector.....	56
Figura 27. Configuración típica para una computadora personal y un proyector.	57
Figura 28. Puertos de video de una computadora personal.	57
Figura 29. Mensaje de error “No hay configuración de pantallas”.....	59
Figura 30. Mensaje de error “Existen variaciones en la resolución de los monitores”.....	59
Figura 31. Forma MDI.	60
Figura 32. Manera de ajustar la forma MDI.	60
Figura 33. Forma MDI ajustada a dos monitores.....	61
Figura 34. Forma uno para la pantalla uno completa.	61
Figura 35. Forma dos para la pantalla dos completa.....	62
Figura 36. Propiedades de una forma.	62
Figura 37. Propiedades de configuración de dos pantallas.	64
Figura 38. Forma MDI, para tres monitores.....	65
Figura 39. Forma MDI ajustada a tres monitores.	65
Figura 40. Forma tres para la pantalla tres completa.	65
Figura 41. Propiedades de configuración de tres pantallas.	66

Figura 42. Diagrama del proceso de CONFIG.MON.exe.....	67
Figura 43. Totalidad de formas disponibles para dos monitores.....	68
Figura 44. Las formas ya ajustas a los dos monitores.....	69
Figura 45. Mensaje de error “No existe el archivo de configuración”.....	70
Figura 46. La forma para ítems que son archivos PDF.....	71
Figura 47. La forma para ítems que NO son archivos PDF.....	71
Figura 48. SPRP monitor uno configurado.....	72
Figura 49. Las formas del segundo monitor quedaron bien configuradas.....	73
Figura 50. SPRP Monitor uno y dos configurado.....	73
Figura 51. Diagrama del proceso de configuración de formas y monitores.....	74
Figura 52. Pizarrón Multimedia.....	77
Figura 53. Tipos de Pizarrones.....	82
Figura 54. Acción de “Pintar” en un pizarrón interactivo.....	85
Figura 55. Funcionamiento de la pila (Stack) de pizarrones.....	89
Figura 56. Navegación del Stack.....	89
Figura 57. Acción de escribir en el Pizarrón.....	90
Figura 58. Propiedades para escribir.....	90
Figura 59. Frame para la acción “Escribir” del pizarrón interactivo.....	92
Figura 60. Insertar un objeto prediseñado.....	95
Figura 61. Una imagen colocada en el cuadro (2,3) de una cuadrícula (3,6).....	96
Figura 62. Inserción de una imagen prediseñada en forma consecutiva.....	96
Figura 63. Inserción de 3 imágenes prediseñada en una columna única 1.....	97
Figura 64. Mandar imagen a un pizarrón DUMMY.....	103
Figura 65. Puerto de audio de una computadora personal.....	106
Figura 66. Tarjeta de audio 2.1 canales.....	106
Figura 67. Mezclador de Volumen.....	107
Figura 68. Funciones de un Mezclador.....	107
Figura 69. Distribución de audio de 5.1 canales.....	109
Figura 70. Proceso del controlador para múltiples bocinas.....	111
Figura 71. Controlador ARDUINO.....	112
Figura 72. Un relevador y su circuito.....	113
Figura 73. Diagrama del circuito electrónico.....	113
Figura 74. Circuito cara superior.....	114
Figura 75. Circuito cara inferior.....	114
Figura 76. Circuito impreso.....	115
Figura 77. Montaje del controlador ARDUINO.....	116
Figura 78. Controlador final.....	117
Figura 79. Interfaz de ARDUINO.....	117
Figura 80. Asignación de bocinas a formas y pantallas.....	120
Figura 81. Configuración para tres pares de bocinas.....	122
Figura 82. Configuración de un par de bocinas, para el monitor uno.....	123
Figura 83. Configuración de un par de bocinas, para el monitor dos.....	123
Figura 84. Configuración de tres bocinas, para el monitor dos.....	124

Figura 85. Excepción de asignación de bocinas.....	124
Figura 86. Cable de Ethernet (figura izquierda) y Tarjeta de Red (figura derecha).....	130
Figura 87. Cable paralelo.....	130
Figura 88. Cable USB para red.....	130
Figura 89. Comunicación entre dos computadoras.....	132
Figura 90. Acceso a la carpeta compartida.....	133
Figura 91. Conexión a otra computadora.....	133
Figura 92. Carpetas compartidas.....	134
Figura 93. Restricción de acceso.....	134
Figura 94. Estructura de mensajes de comunicación entre dos computadoras.....	136
Figura 95. Forma de SPRPShow-Other que espera el mensaje de conexión.....	138
Figura 96. Forma de SPRPBuildShow para configurar la conexión.....	138
Figura 97. Equipos conectados a la red LAN.....	139
Figura 98. Apariencia y Personalización de Windows.....	150
Figura 99. Configuración de pantallas de Windows.....	150
Figura 100. Conector PCI.....	159
Figura 101. Sonido Surround.....	160
Figura 102. Selección del modelo de ARDUINO.....	161
Figura 103. Propiedades del ARDUINO.....	161
Figura 104. Opción Verify del ARDUINO.....	163
Figura 105. Opción Upload del ARDUINO.....	163
Figura 106. Mensaje de fin de proceso del ARDUINO.....	163
Figura 107. Sección del Control de Windows.....	164
Figura 108. Sistema y seguridad de Windows.....	164
Figura 109. Lista de puertos conectados de Windows.....	164
Figura 110. Componentes de Visual Basic.....	165
Figura 111. Propiedades de una carpeta en Windows.....	167
Figura 112. Opciones para compartir carpeta en Windows.....	168
Figura 113. Opciones de permisos para una carpeta de Windows.....	168
Figura 114. Opciones de uso compartido.....	169
Figura 115. Opciones de carpetas públicas.....	169
Figura 116. Símbolo del sistema.....	170
Figura 117. Ejemplo de descripción IP.....	170
Figura 118. Ejemplo de nombre de equipo.....	170
Figura 119. Componentes de Visual Basic.....	171

CAPÍTULO I. INTRODUCCIÓN

1.1 Planteamiento del problema

Adelantamos aquí una definición inicial – y totalmente imprecisa e incluso no del todo acertada - de los U-Books de Bauer, sin la cual resulta incomprendible el resto de esta introducción. Un U-Book es una colección de archivos; se pueden armar - muchos y diferentes - subconjuntos ordenados de los mismos que se ofrecen a lectores del libro. Un ejemplo es una colección de fotografías tomadas en una excursión de un grupo de alumnos. A un interesado en el partido de fútbol que jugaron los muchachos, se le muestran ciertas fotografías, incluyendo una del río porque uno de los alumnos se tuvo que meter para sacar la pelota que allí cayó; los que les interesa el lugar que visitaron, armarán una sucesión (así se llaman los subconjuntos ordenados) con las fotos que contienen imágenes de sitio, el río y las hermosas praderas. La foto del río está en ambas sucesiones, lo que ejemplifica precisamente el uso de los ítems en cualquier sucesión.

Observe que una presentación en PowerPoint no es más que un conjunto ordenado de elementos (diapositivas). En un U-Book, se almacenarían por separado estas, y se armaría una sucesión de ellas que constituiría la presentación.

La definición de los objetivos de la investigación consistió en determinar si se pudieran agregar a los U-Books elementos de realidad virtual para hacer que, cuando se utilice el software correspondiente, es decir el paquete SPRP, para impartir clases, se pudiera aumentar el impacto en los aspectos del aprendizaje.

Esto resultó en que se seleccionara para el enfoque de este trabajo elaborar herramientas que permitieran a un maestro preparar el material de una clase y presentarlo de modo de conseguir ciertos objetivos que se fija para cada sesión. Se incluyó el uso de las mismas tecnologías en presentaciones de cualquier naturaleza, donde quizá no se trata de que la audiencia aprenda, pero sí que preste atención, comprenda el material y lo relacione con los conceptos y conocimientos que poseen para aumentar el impacto de la comunicación.

Esto resultó en que se agregaran ciertos elementos a los U-Books, en especial los pizarrones como elementos de sucesiones, al tiempo que se ampliaron las posibilidades en cuanto a cómo se muestran estas sucesiones: usar más de un monitor o proyector y más de un par de bocinas, con el propósito de simular situaciones de tipo realidad virtual.

1.2 Objetivo General

Contar con un paquete de software y los elementos de hardware que permitan mostrar una sucesión de modo que la audiencia pueda ver varios objetos en forma simultánea, desplegados en varios monitores, cada uno de los cuales tendrá asociado un par de bocinas, y permitirá al presentador usar un pizarrón virtual para aumentar sus posibilidades de explicar y detallar aspectos de lo que presenta.

1.2.1 Objetivos específicos

Se presentan a continuación los objetivos concretos que se establecieron como criterio de finalización de la investigación.

- Incluir como parte de los U-Books los pizarrones como ítems o como elementos de sucesiones;
- Ampliar el concepto y posibilidades de los elementos asociados a otros en sucesiones para agregar a lo que ya tenía (poder mostrar dos objetos en las mitades de un monitor o proyector) de modo que se pudieran desplegar hasta 3 objetos en monitores diferentes;
- Actualizar el paquete SPRP – que implementa los U-Books – para incorporar los nuevos elementos. En especial, agregar los algoritmos que determinarán qué monitor y par (o pares) de bocinas se usarán para cada uno de los objetos que se muestran en forma simultánea;

- Construir un dispositivo que permita conectar varios pares de bocinas a una computadora, y que pueda encender o apagar las bocinas con comandos de software;
- Estudiar el uso de una tarjeta de video de 4 puertos que permita el uso de hasta 3 monitores conectados a la computadora;
- Descubrir un método que permita el uso de más de un monitor visible a la audiencia en ausencia de una tarjeta de video;
- Elaborar todos los programas necesarios para preparar una sesión de lectura de sucesiones para aprovechar los nuevos dispositivos;
- Estudiar el impacto que pudieran tener presentaciones o clases impartidas con esta tecnología, especialmente para determinar los modos en que deben usarse para lograr los objetivos fijados para cada sesión;
- Probar todo el sistema y lograr por lo menos una versión funcional, que pueda sufrir cambios a posteriori pero que permita usar el producto.

1.4 La organización del trabajo

El marco teórico en el que se inscribe esta investigación tiene 3 aspectos diferentes: la educación y en especial, los estilos de aprendizaje y el uso de TIC en la enseñanza, mismos que se discutirán en el Capítulo 2; los U-Books y el SPRP que es la herramienta seleccionada para impartir clases, que ocupan el Capítulo 3; y los elementos de HW (Hardware) que se encontraron - y en algunos casos completaron - para poder usar los dispositivos que se ofrecerían, que se comentarán a medida que aparezcan en el resto de la tesis.

A continuación se separaron en capítulos los temas principales de la investigación: el Capítulo 4 es sobre el uso de varios monitores conectados a una computadora; el Capítulo 5 está dedicado a los pizarrones virtuales; el Capítulo 6 muestra como se usan varios pares de bocinas conectadas a una computadora con el dispositivo

creado para tal efecto; el Capítulo 7 describe el uso de una computadora adicional conectada a la del presentador. Tras un breve ejemplo en el Capítulo 8 que se incluyó para mostrar cómo se pueden usar estas nuevas funcionalidades, las conclusiones ocupan el último capítulo.

Se delegaron muchas explicaciones detalladas y rutinas de programas a los anexos para permitir la lectura fluida del material sin dejar de presentar detalles para los lectores interesados.

1.5 Metodología

Se partió de las decisiones tomadas, que resultaron en la formulación de los objetivos específicos (concretos). Por lo tanto se siguieron los siguientes pasos:

1. Estudiar el U-Book y el paquete SPRP, especialmente todos los componentes que se verán afectados por los trabajos a realizar.
2. Estudiar la teoría de modelos de aprendizaje, con el objeto de concentrar el esfuerzo en elementos que se dirigen al uso de la Realidad Virtual con U-Books.
3. Plantear y discutir alternativas en cuanto a dispositivos que se pudieran ocupar para ofrecer elementos de Realidad Virtual en clases, especialmente contemplando restricciones en cuanto a las erogaciones que significarían.
4. Encontrar y completar los elementos de HW (HardWare) necesarios.
5. Diseñar y elaborar los programas que se necesitan para usar los dispositivos como parte del SPRP.
6. Implementar los pizarrones virtuales.
7. Implementar la conexión de otra computadora a la del maestro.
8. Incorporar los elementos nuevos al SPRP.
9. Probar a fondo el sistema.

CAPÍTULO II. MARCO TEÓRICO

Se presentan elementos de tres temas: estilos de aprendizaje, realidad virtual y la realidad virtual en la educación. Como se notará, se estudió con cierto detalle el tema de los estilos de aprendizaje. El motivo de este estudio extenso es que hay cierta disparidad de opiniones de cómo se definen estos estilos, y se pretendió conocer todos ellos para compararlos en el sentido de qué podría aportar la teoría a los elementos de una clase usando los U-Books.

Quizá se note la ausencia de otro tema: la educación basada en TIC. A pesar de que casi todo lo que se comenta e hizo se inscribe en ese tema general, se decidió no documentar la literatura que se consultó sobre este tema, por dos motivos. Por un lado, la literatura es muy amplia, pero en general está enfocada a dos temas: las clases presenciales y las clases a distancia. En cuanto a las primeras, sí hay trabajos sobre el uso de dispositivos y el impacto de su uso sobre el aprendizaje.

Por ejemplo, muchos versan sobre el uso de “Clickers” puesto que permiten una interacción de los alumnos con el material y le proporciona al maestro una retroalimentación. Puesto que inicialmente estos iban a ser incluidos en nuestra investigación, pero no en el modo en el que se usan en general, se decidió eliminar el tema por completo. En términos coloquiales se puede decir que íbamos a descubrir el hilo negro: inventar un dispositivo que permitiera al maestro saber en forma instantánea cuáles de los alumnos habían entendido algo que acababa de decir.

2.1 Las teorías relacionadas con el aprendizaje

2.1.2 Estilos de aprendizaje

El Programa Nacional de Educación 2001-2006 (SEP), menciona que el nuevo entorno de la sociedad del conocimiento brinda oportunidades extraordinarias para innovaciones orientadas al desarrollo de modalidades educativas más adecuadas, dentro de una concepción de una educación integral que abarque la formación de la

afectividad, la expresión artística, la interacción social y el ejercicio de los diferentes tipos de inteligencia.

Las últimas investigaciones en la neurofisiología y en la psicología han dado como resultado un nuevo enfoque sobre cómo los seres humanos aprendemos: no hay un modo único de aprender: cada persona tiene una forma o estilo particular de establecer relación con el mundo y por lo tanto para aprender. Con respecto a este enfoque, se han desarrollado distintos modelos que aproximan una clasificación de distintas formas de aprender.

La enseñanza centrada en el aprendizaje obliga a “diseñar, incorporar y difundir acciones que lleven a los alumnos a asumir y entender los contenidos de aprendizaje planteados, a través del auto aprendizaje y la responsabilidad compartida”; así como una nueva visión del proceso de enseñanza-aprendizaje, en la cual se considera que cada persona aprende de manera diferente y posee un potencial, conocimientos y experiencias distintas, es decir, existen diversos estilos de aprendizaje, a partir de los cuales procesamos la información recibida del medio y la transformamos en conocimiento.

El término “estilo de aprendizaje” se refiere al hecho de que cada persona utiliza su propio método o estrategias para aprender. Aunque las estrategias varían según lo que se quiera aprender, cada uno tiende a desarrollar ciertas preferencias o tendencias globales, tendencias que definen un estilo de aprendizaje. Son los rasgos cognitivos, afectivos y fisiológicos que sirven como indicadores relativamente estables de cómo los alumnos perciben interacciones y responden a sus ambientes de aprendizaje, es decir, tienen que ver con la forma en que los estudiantes estructuran los contenidos, forman y utilizan conceptos, interpretan la información, resuelven los problemas y seleccionan medios de representación (visual, auditivo, kinestésico). Los rasgos afectivos se vinculan con las motivaciones y expectativas que influyen en el aprendizaje, mientras que los rasgos fisiológicos están relacionados con el género y ritmos biológicos, como puede ser el de sueño-vigilia, del estudiante. (Woolfolk, 1995).

La noción de que cada persona aprende de manera distinta a las demás permite buscar las vías más adecuadas para facilitar el aprendizaje. Sin embargo hay que tener cuidado de no “etiquetar”, ya que los estilos de aprendizaje, aunque son relativamente estables, pueden cambiar: pueden ser diferentes en situaciones diferentes y son susceptibles de mejoras. Y cuando a los estudiantes se les enseña según su propio estilo de aprendizaje, aprenden con más efectividad.

2.1.3 Modelos de estilos de aprendizaje

Antes de presentar diversas teorías sobre el tema, estableceremos el criterio general formulado para poder extraer de cada una de las teorías los elementos que se dirigen específicamente a este criterio. Se señala que esto se inscribe dentro de las teorías del proceso de informar que usa Bauer, y que no corresponden a una teoría en especial ni a investigaciones publicadas que lo expresen precisamente de ese modo. Y, al final, los U-Books son de Bauer, y por lo tanto, es su responsabilidad proveerlos de los elementos que él mismo considera importantes para que los que decidan aplicarlos en sus clases o material didáctico logren sus objetivos. Y esta investigación, puesto que fue conducida por el Dr. Bauer, forzosamente se inscribió en este marco: contribuir a lograr los objetivos del consejero, en especial, el uso de la realidad virtual en los U-Books.

Independientemente del modelo de aprendizaje, consideramos que una presentación o clase presencial debe contemplar cinco aspectos de la relación presentador-audiencia:

Recepción del Material: el alumno debe ver, oír y en los casos que aplique, sentir el mensaje que envía el comunicador.

Atención: el alumno debe estar atento a lo que le llega en ese momento.

Comprensión Del Material: el alumno debe entender el material en cuanto a los términos y lenguaje en los que se le comunica.

Interpretación y Asimilación: el alumno interpreta el material y lo relaciona con conocimientos previos.

Retención: el alumno recordará el material en diversos grados, tanto en el sentido cuantitativo (qué recuerda) y en los detalles o la profundidad con lo que lo recuerda.

Gran parte de lo que se plantea en esta investigación está enfocado precisamente a esta secuencia de actividades. Tras asegurarse de que los alumnos tendrán acceso al material, en el sentido físico de que no hay obstáculos como que no se oye lo que dice el maestro, o no se ve o puede leer lo que muestra, se comienza por conseguir la atención de un oyente. A continuación se ha adaptado el contenido del mensaje que se envía en este instante para que lo reciba del modo planificado, y finalmente en la forma que permita su asimilación o comprensión de acuerdo al estado cognitivo del oyente.

En otras palabras, los componentes del U-Book que se ofrecerán a los maestros para preparar sus clases pretenden proporcionarle las herramientas para lograr (definitivamente) los primeros dos puntos: recepción y atención. La posibilidad de usar versiones e idiomas diferentes de los ítems podrá contribuir a la comprensión, y el resto es función del maestro: debe planear el material de modo que si logra los tres aspectos mencionados, se produzcan los siguientes.

Se han desarrollado distintos modelos y teorías sobre estilos de aprendizaje los cuales ofrecen un marco conceptual que permite entender los comportamientos diarios en el aula, cómo se relacionan con la forma en que están aprendiendo los alumnos y el tipo de acción que puede resultar más eficaz en un momento dado.

Los modelos más conocidos y utilizados en cuanto a estilos de aprendizaje se muestran en la Figura 1.



Figura 1. Modelos de Estilos de aprendizaje.

Aun cuando estos modelos contienen una clasificación distinta y surgen de diferentes marcos conceptuales, todos ellos tienen puntos en común que permiten establecer estrategias para la enseñanza a partir de los estilos de aprendizaje.

2.1.3.1 El Modelo de los Cuadrantes Cerebrales de Herrmann

Ned Herrmann elaboró un modelo que se inspira en los conocimientos del funcionamiento cerebral. Figura 2.



Figura 2. Cuadrantes Cerebrales de Herrmann.

Hace una analogía de nuestro cerebro con el globo terrestre con sus cuatro puntos cardinales. A partir de esta idea representa una esfera dividida en cuatro cuadrantes, que resultan del entrecruzamiento del hemisferio izquierdo y el derecho del modelo Sperry, y de los cerebros cortical y límbico del modelo McLean. Los cuatro cuadrantes representan cuatro formas distintas de operar, de pensar, de crear, de aprender y, en suma, de convivir con el mundo. (Herrmann, 1999).

Las características de estos cuatro cuadrantes se ejemplifican en la Figura 3.

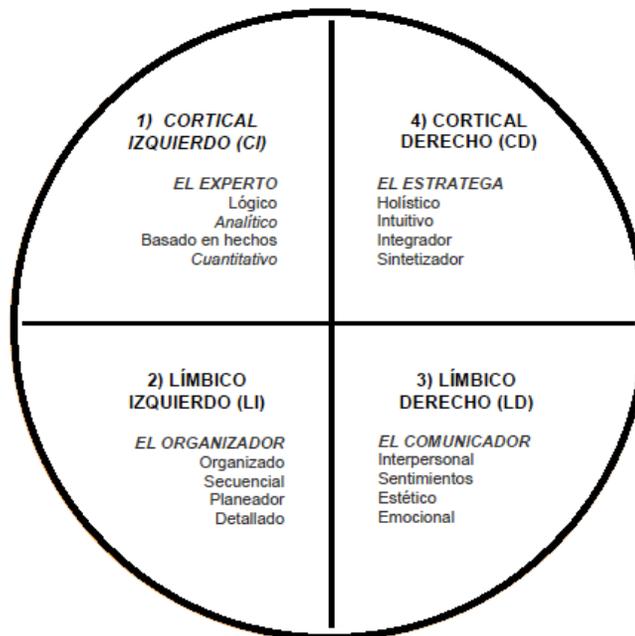


Figura 3. El Modelo de los Cuadrantes Cerebrales de Herrmann.

2.1.3.2 Modelo de estilos de aprendizaje de Felder y Silverman

El modelo de Felder y Silverman (Figura 4), clasifica los estilos de aprendizaje a partir de cinco dimensiones (Felder, 2005), las cuales están relacionadas con las respuestas que se puedan obtener a las siguientes preguntas ¿Qué tipo de información perciben preferentemente los estudiantes? ¿A través de qué modalidad sensorial es más efectivamente percibida la información cognitiva? ¿Con qué tipo de organización de la información está más cómodo el estudiante a la hora de trabajar? ¿Cómo progresa el estudiante en su aprendizaje? ¿Cómo prefiere el estudiante procesar la información?

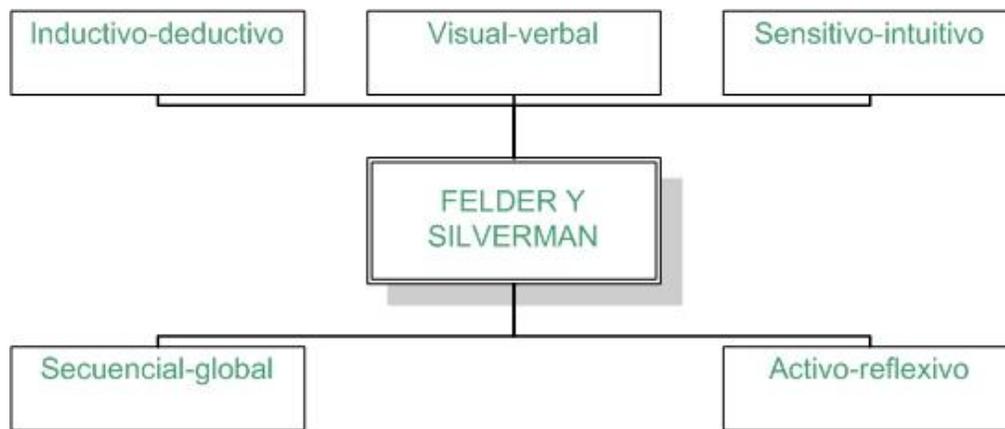


Figura 4. El Modelo de Felder y Silverman.

De acuerdo a esta información, los estudiantes se clasifican en cinco dimensiones: Sensitivos, Intuitivos, Visuales, Verbales, Activos, Reflexivos, Secuenciales, Globales, Inductivo, Deductivo.

2.1.3.3 Modelo de Kolb

El modelo de estilos de aprendizaje elaborado por Kolb (Figura 5), supone que para aprender algo debemos trabajar o procesar la información que recibimos. (Kolb, 1981).

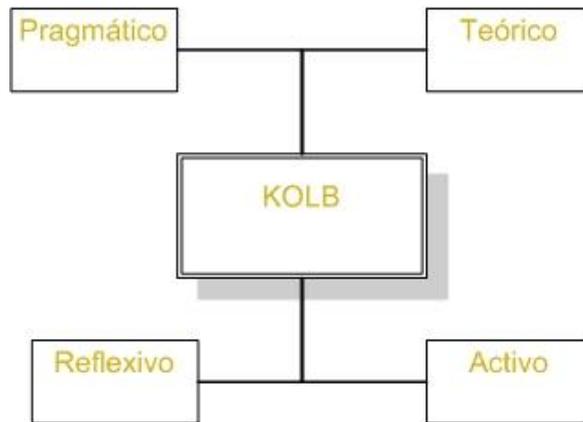


Figura 5. Modelo de Kolb.

Las experiencias que tengamos, concretas o abstractas, se transforman en conocimiento cuando las elaboramos de alguna de estas dos formas:

- a) Reflexionando y pensando sobre ellas: alumno reflexivo.
- b) Experimentando de forma activa con la información recibida: alumno pragmático.

Según el modelo de Kolb (Figura 6), un aprendizaje óptimo es el resultado de trabajar la información en cuatro fases.

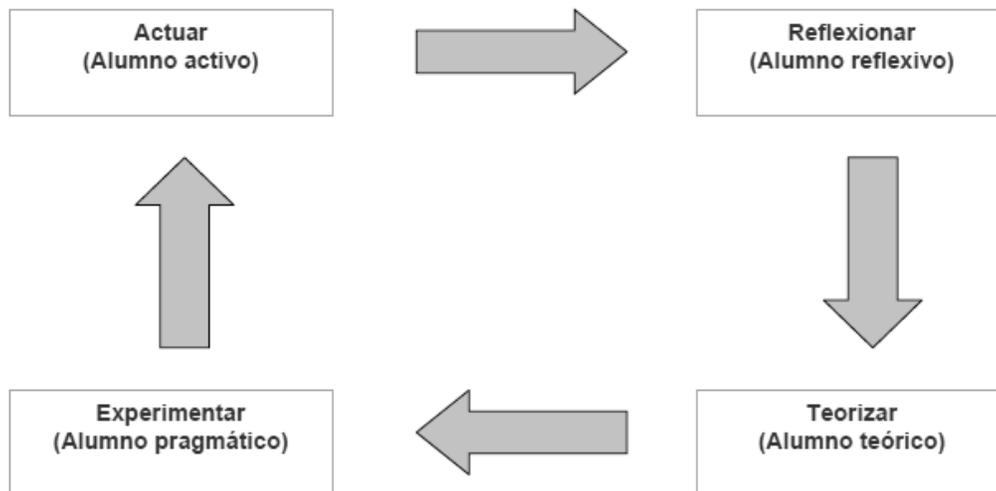


Figura 6. Modelo de aprendizaje.

En la práctica, la mayoría de nosotros tendemos a especializarnos en una, o como mucho en dos, de esas cuatro fases, por lo que se pueden diferenciar cuatro tipos de alumnos, dependiendo de la fase en la que prefieran trabajar:

1. Alumno activo
2. Alumno reflexivo.
3. Alumno teórico.
4. Alumno pragmático.

Un aprendizaje óptimo requiere de las cuatro fases, por lo que será conveniente presentar nuestra materia de tal forma que garanticemos actividades que cobren todas las fases de la rueda de Kolb.

Alumnos Activos: Los alumnos activos se involucran totalmente y sin prejuicios en las experiencias nuevas. Disfrutan el momento presente y se dejan llevar por los acontecimientos.

Suelen ser entusiastas ante lo nuevo y tienden a actuar primero y pensar después en las consecuencias. Llenan sus días de actividades y tan pronto disminuye el encanto de una de ellas se lanza a la siguiente. Les aburre ocuparse de planes a largo plazo y consolidar proyectos, les gusta trabajar rodeados de gente, pero siendo el centro de la actividades.

Alumnos Reflexivos: Los alumnos reflexivos tienden a adoptar la postura de un observador que analiza sus experiencias desde muchas perspectivas distintas.

Recogen datos y los analizan detalladamente antes de llegar a una conclusión. Para ellos lo más importante es esa recogida de datos y su análisis concienzudo, así que procuran posponer las conclusiones todo lo que pueden.

Son precavidos y analizan todas las implicaciones de cualquier acción antes de ponerse en movimiento.

Alumnos Teóricos: Los alumnos teóricos adaptan e integran las observaciones que realizan en teorías complejas y bien fundamentadas lógicamente.

Piensan de forma secuencial y paso a paso, integrando hechos dispares en teorías coherentes.

Les gusta analizar y sintetizar la información y su sistema de valores premia la lógica y la racionalidad. Se sienten incómodos con los juicios subjetivos, las técnicas de pensamiento lateral y las actividades faltas de lógica clara.

Alumnos Pragmáticos: A los alumnos pragmáticos les gusta probar ideas, teorías y técnicas nuevas, y comprobar si funcionan en la práctica. Les gusta buscar ideas y ponerlas en práctica inmediatamente, les aburren e impacientan las largas discusiones discutiendo la misma idea de forma interminable. Son básicamente gente práctica, apegada a la realidad, a la que le gusta tomar decisiones y resolver problemas. Los problemas son un desafío y siempre están buscando una manera mejor de hacer las cosas.

2.1.3.4 Modelo de la Programación Neurolingüística de Bandler y Grinder



Figura 7. Modelo de Programación Neurolingüística.

Este modelo, también llamado visual-auditivo-Kinestésico (Figura 7), toma en cuenta que tenemos tres grandes sistemas para representar mentalmente la información, el visual, el auditivo y el Kinestésico. Utilizamos el sistema de representación visual siempre que recordamos imágenes abstractas (como letras y números) y concretas. El sistema de representación auditivo es el que nos permite oír en nuestra mente

voces, sonidos, música. Cuando recordamos una melodía o una conversación, o cuando reconocemos la voz de la persona que nos habla por teléfono estamos utilizando el sistema de representación auditivo. Por último, cuando recordamos el sabor de nuestra comida favorita, o lo que sentimos al escuchar una canción estamos utilizando el sistema de representación Kinestésico. (Bandler, Grinder, 1979).

Sistema de representación visual.- Los alumnos visuales aprenden mejor cuando leen o ven la información de alguna manera. En una conferencia, por ejemplo, preferirán leer las fotocopias o transparencias a seguir la explicación oral, o, en su defecto, tomarán notas para poder tener algo que leer.

La capacidad de abstracción y la capacidad de planificar están directamente relacionadas con la capacidad de visualizar.

Sistema de representación auditivo: Cuando recordamos utilizando el sistema de representación auditivo lo hacemos de manera secuencial y ordenada. Los alumnos auditivos aprenden mejor cuando reciben las explicaciones oralmente y cuando pueden hablar y explicar esa información a otra persona.

El sistema auditivo no permite relacionar conceptos o elaborar conceptos abstractos con la misma facilidad que el sistema visual y no es tan rápido.

Sistema de representación Kinestésico: Cuando procesamos la información asociándola a nuestras sensaciones y movimientos, a nuestro cuerpo, estamos utilizando el sistema de representación Kinestésico.

Aprender utilizando el sistema Kinestésico es lento, mucho más lento que con cualquiera de los otros dos sistemas, el visual y el auditivo.

2.1.3.5 Modelo de los hemisferios cerebrales

Cada hemisferio (Figura 8), es el responsable de la mitad del cuerpo situada en el lado opuesto: es decir, el hemisferio derecho dirige la parte izquierda del cuerpo, mientras que el hemisferio izquierdo dirige la parte derecha. (VerLee, 1986).



Figura 8. Modelo de los hemisferios cerebrales.

Cada hemisferio presenta especializaciones que le permite hacerse cargo de tareas determinadas:

- El hemisferio izquierdo está más especializado en el manejo de los símbolos de cualquier tipo: lenguaje, álgebra, símbolos químicos, partituras musicales;
- Es más analítico y lineal, procede de forma lógica;
- El hemisferio derecho es más efectivo en la percepción del espacio, es más global, sintético e intuitivo. Es imaginativo y emocional.

Aunque cada persona utiliza permanentemente todo su cerebro, existen interacciones continuas entre los dos hemisferios, y generalmente uno es más activo que el otro.

El hemisferio izquierdo es descrito a veces como analítico debido a que se especializa en reconocer las partes que constituyen un conjunto. El proceso del hemisferio izquierdo es también lineal y secuencial; pasa de un punto al siguiente de modo gradual, paso a paso. Es especialmente eficiente para procesar información verbal y para codificar y decodificar el habla.

En tanto que el hemisferio izquierdo se ocupa de separar las partes que constituyen un todo, el derecho se especializa en combinar esas partes para crear un todo: se dedica a la síntesis. Busca y construye relaciones entre partes separadas.

2.1.3.6 Modelo de las Inteligencias Múltiples de Gardner

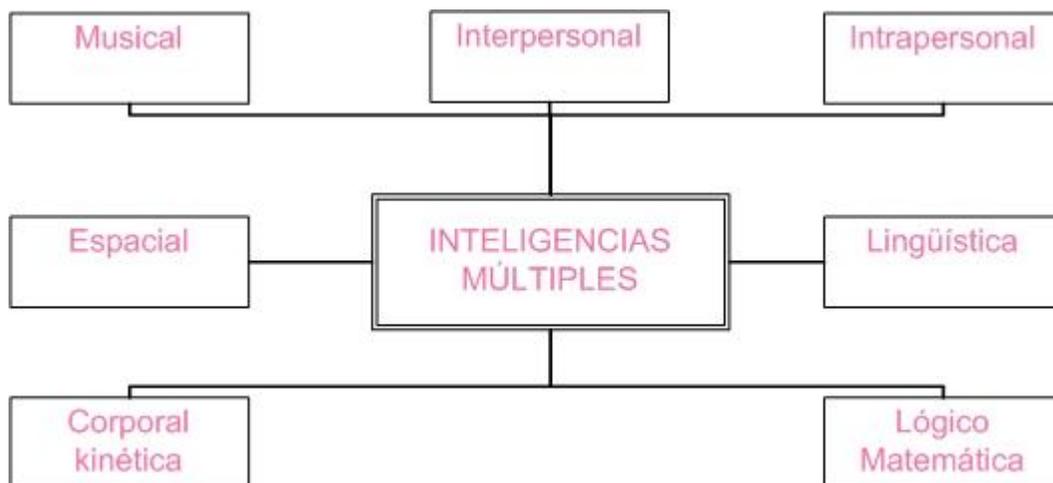


Figura 9. Modelo de las Inteligencias Múltiples de Gardner.

Según el análisis de las siete inteligencias (Figura 9), todos somos capaces de conocer el mundo a través del lenguaje, del análisis lógico-matemático, de la representación espacial, del pensamiento musical, del uso del cuerpo para resolver problemas o hacer cosas, de una comprensión de los demás individuos y de una comprensión de nosotros mismos. Los individuos se diferencian en cuanto a la intensidad de estas inteligencias y en las formas en que recurren a esas mismas inteligencias y las combina para llevar a cabo diferentes labores, para resolver diversos problemas y progresar en distintos ámbitos. (Gardner, 1987).

Gardner proveyó un medio para determinar la amplia variedad de habilidades que poseen los seres humanos, agrupándolas en siete categorías o “inteligencias”:

1. Inteligencia lingüística: la capacidad para usar palabras de manera efectiva, sea en forma oral o de manera escrita. Esta inteligencia incluye la habilidad para manipular la sintaxis o significados del lenguaje o usos prácticos del lenguaje. Algunos usos incluyen la retórica (usar el lenguaje para convencer a otros de tomar un determinado curso de acción), la mnemónica (usar el lenguaje para recordar información), la explicación (usar el lenguaje para informar) y el metalenguaje (usar el lenguaje para hablar del lenguaje).

2. La inteligencia lógico matemática: la capacidad para usar los números de manera efectiva y razonar adecuadamente. Esta inteligencia incluye la sensibilidad a los esquemas y relaciones lógicas, las afirmaciones y las proposiciones (si-entonces, causa-efecto), las funciones y las abstracciones.
3. Los tipos de procesos que se usan al servicio de esta inteligencia incluyen: la categorización, la clasificación, la inferencia, la generalización, el cálculo y la demostración de la hipótesis.
4. La inteligencia corporal-kinética: la capacidad para usar todo el cuerpo para expresar ideas y sentimientos (por ejemplo un actor, un mimo, un atleta, un bailarín) y la facilidad en el uso de las propias manos para producir o transformar cosas (por ejemplo un artesano, escultor, mecánico, cirujano).
5. Incluye habilidades físicas como la coordinación, el equilibrio, la destreza, la fuerza, la flexibilidad y la velocidad así como las capacidades auto-perceptivas, las táctiles y la percepción de medidas y volúmenes.
6. La inteligencia espacial: la habilidad para percibir de manera exacta el mundo visual-espacial (por ejemplo un cazador, explorador, guía) y de ejecutar transformaciones sobre esas percepciones (por ejemplo un decorador de interiores, arquitecto, artista, inventor). Esta inteligencia incluye la sensibilidad al color, la línea, la forma, el espacio y las relaciones que existen entre estos elementos. Incluye la capacidad de visualizar, de representar de manera gráfica ideas visuales o espaciales.
7. La inteligencia musical: la capacidad de percibir (por ejemplo un aficionado a la música), discriminar (por ejemplo, como un crítico musical), transformar (por ejemplo un compositor) y expresar (por ejemplo una persona que toca un instrumento) las formas musicales. Esta inteligencia incluye la sensibilidad al ritmo, el tono, la melodía, el timbre o el color tonal de una pieza musical.
8. La inteligencia interpersonal: la capacidad de percibir y establecer distinciones en los estados de ánimo, las intenciones, las motivaciones, y los sentimientos

de otras personas. Esto puede incluir la sensibilidad a las expresiones faciales, la voz y los gestos, la capacidad para discriminar entre diferentes clases de señales interpersonales y la habilidad para responder de manera efectiva a estas señales en la práctica (por ejemplo influenciar a un grupo de personas a seguir una cierta línea de acción).

9. La inteligencia intrapersonal: el conocimiento de sí mismo y la habilidad para adaptar las propias maneras de actuar a partir de ese conocimiento. Esta inteligencia incluye tener una imagen precisa de uno mismo (los propios poderes y limitaciones), tener conciencia de los estados de ánimo interiores, las intenciones, las motivaciones, los temperamentos y los deseos, y la capacidad para la autodisciplina, la auto-comprensión y la autoestima.

2.1.3 Estrategias De Enseñanza



Figura 10. Estrategias De Enseñanza.

Las estrategias de enseñanza (Figura 10), son los procedimientos o recursos utilizados por el agente de enseñanza para promover aprendizajes (Verlee, 1986), algunas de sus características se muestran en la tabla 1:

Tabla 1. Modelos de estilos de aprendizaje.

ESTRATEGIA	MODELO(S) DE ESTILOS DE APRENDIZAJE	APLICACIÓN
Metáfora	<ul style="list-style-type: none"> • Modelo de los Hemisferios Cerebrales. • Modelo de Felder y Silverman. • Modelo de Inteligencias Múltiples. • Modelo de Kolb. • Modelo de los Cerebrales. 	<p>Presentación de un tema nuevo. Conexión de conceptos. Estimulación de la imaginación. Medio para organizar y recordar la información. Favorece los procesos de clasificación e integración.</p>
Pensamiento Visual	<ul style="list-style-type: none"> • Modelo de Programación Neurolingüística. • Modelo de los Hemisferios Cerebrales. 	<p>Representar relaciones en aquellas asignaturas que se basan en la observación. Recordar información. Efectuar funciones matemáticas. Representación gráfica de conceptos, ampliando la comprensión del mismo. Resolución de problemas que impliquen relaciones espaciales. Favorece el desarrollo de capacidades visuales, la clarificación del pensamiento y la comunicación de ideas a otros.</p>

Fantasía	<ul style="list-style-type: none"> • Modelo de los Hemisferios Cerebrales. • Modelo de Felder y Silverman. 	<p>Resolución de problemas. Tratamiento de temas que requieren de concientización y empatía, como los valores, derechos humanos, preservación del medio ambiente, etc. Promueve la experiencia cinestésica y la respuesta emocional. Estimula el involucramiento y con ello aumenta la motivación por aprender. Favorece el desarrollo de la creatividad y el uso de la imaginación.</p>
Aprendizaje Multisensorial	<ul style="list-style-type: none"> • Modelo de Programación Neurolingüística. • Modelo de Inteligencias Múltiples. • Modelo de los Cuadrantes 	<p>Ampliación de conocimiento sensorial. Estimulación de la habilidad verbal para describir sensaciones, útil en clases de literatura y redacción. Favorece la memoria. Desarrollo del pensamiento abstracto. Provee un método adicional para comprender un tema. Permite liberar energía física. Permite mejorar la memoria.</p>
Experiencia Directa	<ul style="list-style-type: none"> • Modelo de Kolb • Modelo de Inteligencias Múltiples. • Modelo de los Cuadrantes Cerebrales. 	<p>Estimula un involucramiento más personal. Fomentan un conocimiento propio. Fomentan las habilidades interpersonales. Favorecen el desarrollo de la empatía. Favorece un sentido del todo más que de las partes.</p>

2.2. Realidad Virtual

Desde hace varios años, las computadoras se han utilizado para apoyar el proceso enseñanza-aprendizaje en diversos niveles de estudio. La realidad virtual se ha desarrollado de manera importante desde los años ochenta a escala mundial; sin embargo no fue usada en ambientes educativos hasta los años noventa.

Desde hace más de tres décadas, se han utilizado computadoras para apoyar el proceso enseñanza aprendizaje, tanto de manera grupal como individual. Esta área de estudio se llama Aprendizaje asistido por computadora, o CAL por sus siglas en inglés, y ha evolucionado a la par de los desarrollos tecnológicos.

En los años ochenta se dio el boom de las computadoras personales, permitiendo a escuelas y a estudiantes tener computadoras a relativamente bajo costo. Esto ha permitido apoyar el aprendizaje de diversas áreas del conocimiento en alumnos de todos los niveles, inclusive desde edad preescolar.

El incremento de la velocidad de procesamiento, la mayor capacidad de gráficos y la reproducción de sonido en las computadoras han permitido que la tecnología de realidad virtual pueda utilizarse en el proceso enseñanza-aprendizaje.

La realidad virtual fue creada en los años 70. Se comenzó a utilizar en simulaciones de escenarios de guerra y para entrenamiento de pilotos militares y comerciales. En los años ochenta la tecnología de realidad virtual fue adaptada para su uso en la industria privada, farmacéutica, petróleo y automotriz.

La realidad virtual se define como un medio tecnológico compuesto por una simulación tridimensional por computadora, donde el participante utiliza varios sentidos sensoriales para interactuar con dicha simulación, dando en esta la sensación de estar mentalmente inmerso (presente). A la simulación tridimensional se le ha llamado ambiente o “mundo” virtual (Sherman y Craig, 2003).

La realidad virtual: Es un fenómeno que percibimos a través de los sentidos que se desarrolla en un espacio ficticio. La Realidad Virtual es un sistema interactivo que

permite sintetizar un mundo tridimensional ficticio, creando una ilusión de realidad. La realidad Virtual es una técnica fotográfica de 360 grados, que permite moverse hacia arriba o hacia abajo, realizar acercamientos o alejamientos; la diferencia es que en la realidad virtual tú tienes el control absoluto de los movimientos; este te ofrece un realismo de estar en el sitio.

Virtual, en Informática, significa 'algo simulado', creado por la computadora para llevar a cabo determinado fin. La Realidad Virtual es considerada en muchos aspectos como el interface definitivo entre los seres humanos y la computadora.

Básicamente consiste en simular todas las posibles percepciones de una persona, como los gráficos para la vista, sonido, tacto e incluso sensaciones de aceleración o movimiento. Todas estas sensaciones diferentes deben ser presentadas al usuario de forma que se sienta inmerso en el universo generado por las computadoras, hasta el punto de dejar de percibir la realidad y ser engañado, sentirse transportado (al otro lado de la pantalla) como si se tratase de un universo nuevo.

La tecnología actual está permitiendo unos sistemas interactivos computarizados tan rápidos e intuitivos que la computadora desaparece de la mente del usuario, dejando como real el entorno generado por la computadora, por lo que puede ser un mundo de animación en el que nos podemos adentrar.

La realidad virtual es una tecnología que lleva más de 40 años en desarrollo, pero es ahora cuando está apareciendo en todas partes como simuladores de vuelo, cirugía, estrategias militares, arqueología, química, geología, modelación entre otros.

Los avances de la tecnología nos han dado medios de comunicación más eficientes como el internet: basta con tener una computadora para poder comunicarnos a cualquier lugar del mundo y acceder a miles de fuentes y recursos en cuestión de minutos, ahorrándonos tiempos, costos y esfuerzo. Pero actualmente ya no es suficiente: necesitamos llegar a otro nivel de interactividad de usuario-maquina, si no explotar los sentidos y poder percibir esto como si fuera el mundo real.

En el ámbito de la investigación el acceso a las fuentes de información de internet es un medio muy útil que permite obtener información. Este tipo de consultas pretenden ofrecer información grafica ligada entre sí para dar el efecto de una visita. En la actualidad la mayoría de estas visitas ocurren en forma bidimensional; pero en ciertos casos puede ser necesario o conveniente visualizar algún sitio como lo haría mente la mente del usuario, es decir en 3D. Un tipo de ejemplos de esto sería visitas guiadas a museos, ciudades, edificaciones, zonas arqueológicas. Este tipo de visitas serian más interesantes y educativamente más efectivas.

2.2.1 Antecedentes de la realidad virtual

La realidad virtual comenzó en los 70's como material para una clase de aviación en el departamento de defensa de los Estados Unidos, para hacer simulaciones de vuelo, practicando y no arriesgando vidas.

- 1965: Surge el concepto de Realidad Virtual, cuando Ivan Sutherland publico un artículo titulado "The ultimate display", el cual describía el concepto básico de Realidad Virtual;
- 1966: Ivan Sutherland creo el primero casco visor de Realidad Virtual, al montar tubos de rayos catódicos en un armazón de alambre; este visor fue llamado "Espada de Damocles";
- 1968: Ivan Sutherland y Davis Evans, crean el primer generador de escenarios con imágenes;
- 1971: Redifon Ltd en el Reino Unido comienza a fabricar simuladores de vuelo con displays gráficos;
- 1972: General Electric, bajo la comisión de la Armada norteamericana, desarrolla el primer simulador computarizado de vuelo. Los simuladores de vuelo marcarán la pauta para la Realidad Virtual;

- 1973: Bui-Tuong Phong presenta sus tesis de doctorado: “Iluminación de imágenes generadas por Computadora”;
- 1976: P.J Kilpatrick publica su tesis de doctorado: “El uso de la Cinemática en un Sistema Interactivo Grafico”;
- 1977: Dan Sandin y Richard Sayre inventan un guante sensitivo a la reflexión;
- 1979: Eric Howlett (LEEP System, Inc.). Diseñan la perspectiva óptica mejorada de extensión larga (Large Expanse Enhanced Perspective Optics, LEEP);
- Jaron Lanier es uno de los primeros generadores de aparatos de interfaz sensorial, acuño la expresión “Realidad Artificial”, también colabora en el desarrollo de aparatos de interface realidad virtual, como guantes y visores;
- 1980: La Compañía StereoGraphics hace las gafas de visión estéreo;
- 1982: Sucede uno de los acontecimientos históricos en el desarrollo de los simuladores de vuelo, cuando Thomas Furness presentó el simulador más avanzado que existe, contenido en su totalidad en un casco;
- 1982: Thomas Zimmerman patenta un Electro-guante que inventó mientras investigaba sobre cómo controlar con la mano un instrumento musical virtual;
- 1985: Scott Fisher considerado uno de los "Padres Fundadores" de la realidad virtual. El creó el VISIO- CASO más avanzado en la Nasa Ames Center, por todas partes empiezan a surgir equipos de desarrollo trabajando en lo que era la tecnología de la realidad virtual, y se empiezan a ver los primeros resultados comerciales;
- 1987: La compañía Inglesa Dimensión Internacional desarrolla un Software de construcción de mundos tridimensionales sobre computadoras personales;

- 1988: Scott Foster inventa un dispositivo para la generación de sonido tridimensional;
- 1989: ATARI saca al mercado la primera máquina de galería de vídeo juegos con tecnología 3D. En ese mismo año Autodesk presenta su primer sistema de realidad virtual para computadoras personales.

2.2.2 Modelo, Niveles y Tipos de Realidad Virtual

Existen dos tipos de realidad virtual, dependiendo del hardware utilizado: realidad virtual de escritorio y realidad virtual de inmersión completa. La primera es utilizada por una persona o un grupo reducido de personas, donde el ambiente virtual es visualizado en un monitor de computadora (o varios monitores), y los participantes pueden interactuar con el mismo utilizando aparatos especiales, tales como un ratón (mouse), una pantalla táctil (Touch-Screen), tarjetas de audio o tarjetas de video.

En la realidad virtual de inmersión completa el usuario utiliza un casco de realidad virtual o un dispositivo que cubre el ángulo de visión, donde se despliega el ambiente virtual en el mismo, utiliza sensores de movimiento y guantes de datos. Este equipo permite generar en el usuario un alto grado de inmersión, siempre y cuando el equipo y el ambiente estén correctamente aplicados y calibrados.

2.2.3 Niveles de tecnología de la realidad virtual

En la práctica de Realidad Virtual se usan diversos dispositivos de hardware. La tecnología utilizada en Realidad Virtual puede ser clasificada en cuatro categorías o niveles:

- **Nivel de Entrada:** Es el que utiliza un computadora personal o estación de trabajo, e implementar un sistema "Window on a World", es decir, la representación del mundo a través de la pantalla, como si se tuviera una "ventana al mundo";

- **Nivel Básico:** Cuando se agrega unas ampliaciones básicas de interacción (Guante, Gafas estereoscópicas, graficas 2D);
- **Nivel Avanzado:** El siguiente paso en la tecnología de Realidad Virtual es un mejor despliegue de las gráficas y mejor manejo de información de entrada. Esto generalmente se logra con aceleradores de gráficas, e incorporación de procesadores en paralelo;
- **Nivel Inmersivo:** En este nivel, se han agregado al sistema de Realidad Virtual dispositivos más inmersivos, que le añaden realismo: HMD (Head-mounted display), Boom, elementos táctiles.

2.3 Realidad Virtual y la Educación

Desde hace aproximadamente quince años, universidades e instituciones de investigación de todo el mundo han estudiado la aplicación de la tecnología de realidad virtual en la educación.

Los estudiantes utilizan ambientes virtuales de manera presencial, en primera persona y manipulando directa y activamente los objetos virtuales, esto se sitúa dentro de la filosofía de aprendizaje del constructivismo, donde se pregona el “aprender haciendo”.

Sweller indica que se aprende mejor utilizando una combinación de información visual (gráfica) y auditiva. (Sweller 1988).

La realidad virtual en la educación se ha utilizado mundialmente desde hace aproximadamente 15 años. Su aplicación en México es mucho más reciente, realizándose de manera importante quizás desde hace apenas unos 6 años.

Los medios de aprendizaje son los canales que se utilizan para comunicar el conocimiento (visuales, auditivos, táctiles, etc.), mientras los materiales son los vehículos para transmitir los conocimientos (carteles, grabaciones, plastilina, u otro).

Por ello se pueden estructurar por parte del maestro, múltiples y diversas formas de comunicar a los alumnos el conocimiento.

Algunas personas aprenden mejor si “ven” la información que están aprendiendo; otras personas aprovechan más su canal auditivo, mientras que otro grupo aprende mejor haciendo las cosas a través del tacto, que es la Kinestesia. Estas características son las que se quieren destacar debido a que la realidad virtual por medio de los sistemas de computadora, pueden aprovechar las formas de aprendizaje de los alumnos y potenciar el proceso comunicativo en las aulas.

Existen un sin número de teorías respecto a la mejor forma de transmitir el aprendizaje a los alumnos. La forma de aprendizaje tipo VAC (Visual, Auditivo y Kinestésico) es una de las teorías más actuales y adecuadas para concebir aprendizaje utilizando medios de comunicación audiovisual multisensorial. Este modelo de Reid (1996) se adapta al medio de información que se propone utilizar.

En este modelo de aprendizaje se estudian las modalidades sensoriales de cada individuo y el factor de dependencia e independencia individual y social, tomando en cuenta además el tipo de medios que se proponen utilizar en la transferencia de información.

Esta clasificación y sus características no valoran la inteligencia ni la aptitud personal de los alumnos. Se trata de preferencias que pueden variar y además suele predominar más de una. Lo importante es que el creador de material pedagógico sepa que los diferentes estilos de aprendizaje pueden ser enseñados y desarrollados con la finalidad de mejorarlos.

2.3.1 Beneficios de la realidad virtual en la educación

Winn (1993) ha propuesto que la realidad virtual aplicada en ambientes de educación puede contribuir con estos factores:

- Los estudiantes exploran y analizan información utilizando sus sentidos de la vista, oído y tacto. Esto puede contribuir a acortar el tiempo de aprendizaje;
- La realidad virtual facilita la manipulación y análisis de modelos complejos y grandes, que se pueden analizar desde cualquier ángulo y punto de vista;
- La información multisensorial puede apoyar diversos estilos de aprendizaje;
- La realidad virtual puede mejorar la comprensión de información compleja y abstracta, ya que realiza reificación (mostrar de manera concreta conceptos abstractos);
- Los ambientes virtuales en línea (colaborativos) conforman un espacio de trabajo adecuado para que interactúen social y pedagógicamente alumnos, maestros e investigadores;
- Con la realidad virtual, los estudiantes pueden realizar actividades y aprender ciertos conceptos científicos más fácil y rápidamente.

CAPITULO III. U-BOOKS

Descripción panorámica de U-Books

En (Bauer Mengelberg, 2007) se definió este término en un artículo como un libro sin estructura (Unstructured Book). Se trata de una colección de elementos, llamados *ítems*, que representan archivos multimedia. Estos ítems no tienen significado propio (excepto el representado por sus contenidos, es decir, lo que representa una imagen o un video, o el significado de un texto.) Se utilizan vía *sucesiones*, que son subconjuntos ordenados de ítems.

Para explicar este concepto, supongamos que separamos los capítulos y secciones de un libro, de modo que cada sección sea un ítem. El libro completo sería una sucesión que presentaría, una tras otra, las secciones del libro. Si se lee una sección por separado, puede carecer de sentido o no tener una interpretación evidente. Al formar parte de un libro, en cambio, adquiere la interpretación que le confiere su posición en el libro y la relación con las restantes sucesiones.

Bauer dice que una sucesión transforma elementos (información) sin estructura en información semi-estructurada. Por lo tanto, al producto de software que implementa los U-Books lo denomina SP (structured publishing).

Los U-Books ofrecen una funcionalidad llamada “restricted privileges”, de modo que le agregó RP al nombre del paquete, resultando en **SPRP**. A pesar de que esta funcionalidad no se usa en esta investigación, se describirá brevemente en una sección posterior.

Se describen solamente los elementos de los U-Books que se utilizaron en la investigación que condujo a esta tesis, y al nivel necesario para entender y usar el material resultante. Las descripciones se agruparon en: ítems, sucesiones, construcción de sucesiones y lecturas de sucesiones.

Quizá convenga hacer una aclaración. El nombre U-Book puede ser confuso, puesto que no necesariamente se trata de un libro electrónico. Se puede usar el concepto y

el paquete de software para otros fines, entre otros: usar un conjunto de canciones o fotografías, elaborar un libro de texto, o como se verá aquí, preparar clases presenciales o presentaciones de cualquier tipo. Un ejemplo de un uso no evidente de U-Books se puede consultar en (Bauer Mengelberg 2011), donde se los usa para ofrecer material de una conferencia a los asistentes para facilitarles la selección de los trabajos a los que asistirán, contemplando que hay varias sesiones simultáneas.

Se han incluido en este mismo capítulo dos secciones adicionales: el uso de la realidad virtual en U-Books y los elementos de Hardware y Software que se necesitan para poder aprovechar la realidad virtual en ellos.

3.1 Ítems de U-Books

Un ítem se puede considerar como la conjunción de dos elementos: un archivo y una descripción del mismo, como se ilustra en la Figura 11. La descripción sirve para saber de qué archivo se trata, y el contenido del archivo es lo que se exhibirá a un lector.

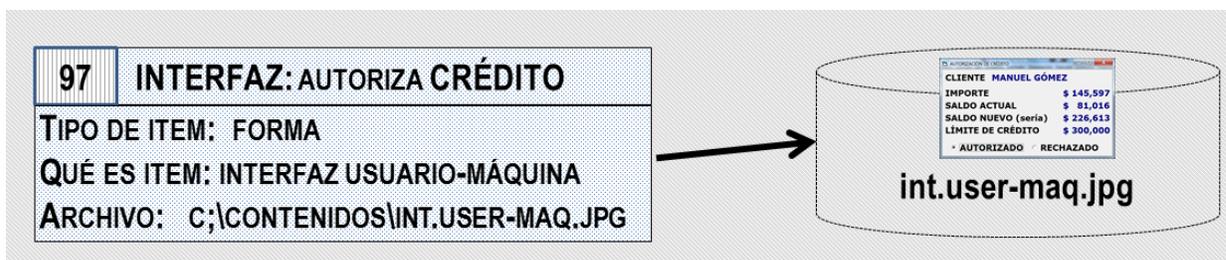


Figura 11. Un ítem de un U-Book.

De hecho, un ítem puede corresponder a – describir- varios archivos, y no solo uno. El U-Book usa el término versión para indicar archivos diferentes pero que corresponden a un mismo ítem.

Por ejemplo el ítem 38 (los ítems tienen un número único) es: Biografía de Galileo Galilei. Sin embargo, hay 3 “versiones” de esta biografía. Una es un texto breve con algunos datos. Otra es una presentación en PowerPoint que contiene varias imágenes y textos donde se describe a este científico, mientras que la tercera es una película (video clip) sobre la vida del mismo. Cuando a alguien se le muestre la biografía, por algún mecanismo el sistema decide cuál de las versiones (archivos) usará.

Lo mismo sucede con el idioma: se pueden asociar archivos en diferentes idiomas a un mismo ítem. No consideramos necesario proporcionar un ejemplo de esta posibilidad.

El poder asociar más de un archivo al mismo ítem originó que se usaran dos tablas: la de los ítems (que además del número y una breve descripción, tiene varios descriptores y datos técnicos relacionados con los archivos y sus contenidos); y la tabla en la que se especifican los archivos asociados al ítem. En esta tabla se almacenan las direcciones de los archivos, entre otros datos. Se hace notar que estos detalles no son necesarios para entender el resto de esta tesis.

De este modo, en el resto del trabajo, cuando se hable de mostrar un ítem, se trata de hacer que el lector vea el contenido del archivo asociado al ítem.

Un ítem puede corresponder a un archivo que contenga un texto, un archivo en formato pdf, una imagen, un video o audio clip, pero también puede ser una presentación en PowerPoint o Flash, o un archivo de un paquete de MS Office (Access, Excel, Word, Visio). Además, hay otro tipo de ítem que nos interesa en particular en este trabajo.

3.1.1 Pizarrón como ítem

Un ítem puede ser un pizarrón, concepto que se describe a detalle en el Capítulo 5. De ese modo, cuando se invoque el ítem el sistema ofrecerá un pizarrón virtual. Este concepto es central a esta investigación, y si no resulta claro en este punto lo será en cuanto se necesite comprenderlo.

De los datos que describen al ítem conviene destacar dos. Se usa un campo denominado *What_is_item* en el que se indica si corresponde a un archivo o a un pizarrón. Cabe agregar que en U-Books un ítem también puede corresponder a otros objetos, como lo son páginas de internet y consultas en tiempo real a alguna base de datos o programas de computadora (que se ejecutan como consecuencia de su invocación).

También hay un campo denominado *ItemType* que se puede utilizar para una primera clasificación de los ítems. Por ejemplo, en el U-Book que usaremos como ejemplo y que consiste de elementos descriptivos (documentación) de un sistema de información, hay diversos tipos de contenidos: programas de computadora, bases de datos, instructivos de operación, documentos elaborados durante el análisis, etc. Cuando se crea un libro se definen los tipos de ítems que se desean para ese libro, y cuando se introduce un ítem se especifica a cuál de los tipos pertenece. Los interesados en mayor detalle de cómo se describen los ítems en los libros pueden obtenerlos en (Bauer Mengelberg, 2012).

3.2 Sucesiones

3.2.1 ¿Qué son las Sucesiones en U-Books?

El otro concepto fundamental de los U-Books es el de una sucesión, que constituye el modo de usar los ítems. Una sucesión es un subconjunto ordenado de los ítems de un libro. Se habla de los *elementos* de una sucesión, puesto que además de ítems pueden representar otro tipo de objetos, como se verá más abajo.

Se pueden armar tantas sucesiones como sea necesario o deseable, y se pueden crear en cualquier momento. El SPRP almacena las sucesiones en dos tablas: en una describe la sucesión misma, y en la otra guarda los elementos de la misma: el número de ítem o el dato necesario para informar al programa que muestra la sucesión del objeto que debe ofrecer al lector.

3.2.2 Subsucesiones

Se puede incluir una sucesión como elemento de otra. En este caso, hablaremos de una subsucesión. Este concepto es recursivo, ilustrado en la figura 12, es decir la subsucesión puede a su vez contener otras Subsucesiones.

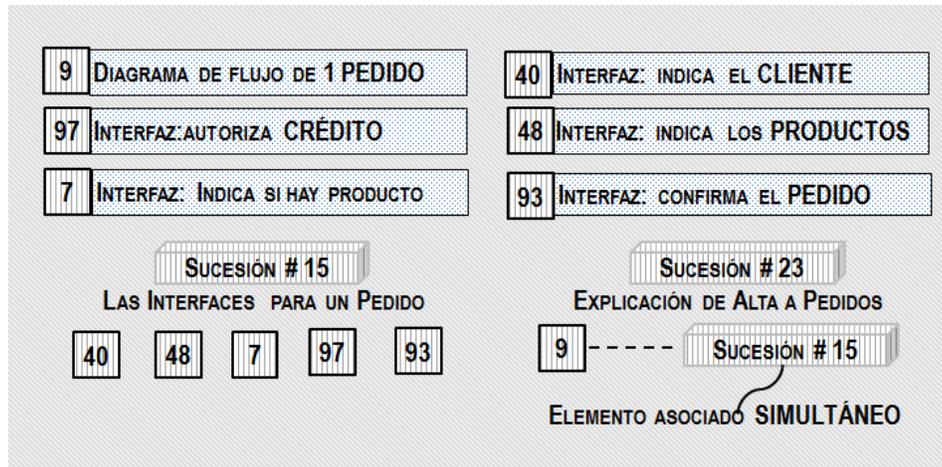


Figura 12. Una sucesión que contiene una subsucesión.

3.2.3 Elemento asociado a un elemento de una sucesión

A un elemento de una sucesión se le puede asociar otro elemento (que puede ser un ítem o una subsucesión). Cuando se usa esta facilidad, se indica si este elemento asociado se mostrará en forma simultánea o alterna. Simultánea significa que se mostrarán al mismo tiempo el elemento “original” y su asociado, para lo cual se usan dos monitores o proyectores o se divide un monitor en dos partes, y se muestra un objeto en cada uno de ellos. Si es alterno, el asociado se ofrece al lector (con un botón cuya etiqueta le informa de qué se trata el alterno): si este decide verlo, se le muestra el asociado en lugar del original, y cuando termina de verlo, regresa al elemento original.

Ejemplos de estos elementos asociados

Supongamos que estamos describiendo el movimiento de la Tierra alrededor del Sol. Ofrecemos al oyente – con un botón de comando - la posibilidad de ver la biografía de Galileo. Lo hacemos asociando al elemento “tierra gira alrededor del sol” el ítem “biografía de Galileo”. Observe que para lograr este efecto incluiríamos en la sucesión:

Elemento #24 (posición 24 dentro del conjunto ordenado de elementos de la sucesión): ítem 915 (cuya descripción es “tierra gira alrededor del sol”).

Elemento asociado: ítem 234 (biografía de Galileo) con la indicación de que el elemento se muestra en forma “alterna”.

En cambio, supongamos que estamos hablando de un programa de computadora. Tenemos un diagrama que muestra diversos componentes del programa, y al mismo tiempo, uno tras otro, las formas con las que interactúan los usuarios del programa con la computadora. Si dispusiéramos de un solo proyector, mostraríamos en la parte izquierda del monitor el diagrama, y en la parte derecha, una tras otra, una imagen de la interfaz. Es importante señalar que también se podrían mostrar videos en esta parte derecha (¡también son ítems!)

Para implementar esta sucesión, se incluye como elemento #7 (posición en la sucesión) el ítem “Programa para introducir pedidos”. Le asociamos (en forma “simultánea”) la sucesión 18 – definida previamente – que tiene los elementos:

1. Ítem 1345: Forma para captura del cliente que formula el pedido
2. Ítem 2587: Forma en la que se introducen los artículos solicitados por el cliente y la cantidad de cada uno.
3. Ítem 2594: La misma forma pero que indica si hay o no existencia suficiente de ese artículo para surtir el pedido.
4. Ítem 3115: La forma para confirmar el pedido, que muestra los precios y el importe total del pedido. Se introduce el lugar y fecha de entrega solicitada de los productos.
5. Ítem 4854: La forma en la cual el encargado de Crédito autoriza el pedido.

Observación: el control del presentador (avanzar al siguiente elemento, retroceder al anterior, etc.) está en la subsucesión. Al terminar esta subsucesión, el control regresa a la sucesión principal (se muestra el siguiente elemento, es decir, el que se estaba mostrando “termina”. Naturalmente, el presentador puede regresar al elemento anterior: en ese caso, también aparecerá nuevamente la subsucesión asociada.

Estos conceptos son fundamentales para comprender el resto de este trabajo, puesto que precisamente son los que se agregaron para aprovechar los elementos de realidad virtual que introduciremos más abajo.

3.2.4 Sucesiones de objetos prediseñados (para pizarrones)

Como se verá más adelante, se pueden definir sucesiones de imágenes que se agregarán a un pizarrón virtual de los que usa el U-Book. Resumiendo o adelantando el concepto, se trata de usar una imagen elaborada previamente y “pegarla” al pizarrón.

Una analogía que puede servir para ilustrar este concepto – aunque no es del todo plausible – es un maestro que está dando una clase en un pizarrón (real, es decir usando un gis o un plumón). Tiene que hacer un diagrama del sistema digestivo, y no confía en poder hacerlo en vivo. Por lo tanto, lleva una cartulina con el diagrama y lo adhiere al pizarrón. Esto puede causar risa a sus alumnos (o confusión) pero le resuelve un problema. Veremos que si lo hace con un pizarrón virtual (de los que usamos aquí) los alumnos no notarán la diferencia excepto por el tiempo que tardaría el profesor en hacer el diagrama, y quizá por la calidad o claridad del diagrama mismo.

Estas sucesiones no se distinguen mayormente de otras, aunque son más limitadas: sus elementos sólo pueden ser imágenes, y a sus elementos no se les pueden asociar otros elementos. Sin embargo pueden contener Subsucesiones, pero estas serán del mismo tipo que las de objetos prediseñados.

3.3 Construcción de sucesiones

No se muestran aquí las interfaces que se usan para construir una sucesión, puesto que son muchas y sería difícil presentarlas todas además de que no agregaría nada a la descripción que se incluyó. Una sucesión se crea en dos pasos (aunque el sistema contiene uno adicional).

Paso 1. Se crea una nueva sucesión. Se indica un número (o se le delega esa tarea al programa) y se incluye una descripción que permite distinguirla de las restantes. Como muchos otros objetos de un U-Book, esta descripción se puede introducir en varios idiomas, de modo que un usuario vea de qué se trata en el idioma en el cual está utilizando el software.

También se indican algunos datos de la sucesión, especialmente el tipo: definida por el autor del libro o de un lector, de objetos prediseñados, etc. Se pueden indicar fechas de activación y desactivación, y algunos datos que permiten al programa ofrecer o no algunas funciones para los otros pasos.

Paso 2. Se editan los elementos de la sucesión (agregan, quitan o modifican). También se puede modificar el orden y realizar otras operaciones que no se detallan puesto que no intervienen en los usos de sucesiones que se describen aquí.

El sistema ofrece la lista de los elementos que ya se habían introducido. Se reordenan o eliminan los elementos que lo ameriten. Se agregan elementos adicionales. Solo describiremos esta actividad, puesto que las otras son fáciles de imaginar. Se señala que las interfaces son fáciles de entender y usar, con algunas excepciones resultantes de la necesidad de comprender algún concepto para poder usar las respectivas funciones.

Esencialmente, se pueden agregar elementos en cualquiera de estos modos.

- Un elemento “nuevo”. Se indica si es un ítem o una sucesión (que se convierte en una subsucesión de la que se está creando). Se proporciona el número, Para ello se puede indicar el número (si se lo conoce) o invocar una lista (filtrada) de los ítems o sucesiones, según el caso. La necesidad de filtrar la lista que se presenta para elegir el ítem o sucesión deseado nace de la posibilidad de que haya un gran número de ítems o sucesiones. ¡Un U-book puede tener millones de ítems!;
- A continuación el sistema valida si el ítem o sucesión ya están en el U-Book.
- Se le puede asociar a este elemento –digamos que el original en este contexto– otro elemento, y una vez más, se indica de qué se trata en forma análoga a la del elemento original. El sistema pregunta si el asociado se mostrará como *alternativo* o *simultáneo*;
- Una lista (ordenada) de ítems al mismo tiempo. Esto a su vez se puede hacer de varias maneras.

- Elegir uno tras otro de una lista de ítems “disponibles” (existentes en el libro.) Esta lista se puede filtrar de varios modos, por ejemplo por los campos de la tabla: ItemType y what_is_ítem y de varias otras basadas en otros descriptores de los ítems. Se arma una lista y se agregan los ítems a la sucesión (cada uno se convertirá en el siguiente elemento de la sucesión);
- Agregar todos los elementos de otra sucesión (supongamos que es la #48). Observe que esto no es equivalente a agregar la sucesión misma. Supongamos que la sucesión que se está construyendo es la #511. Aquí se agregan uno por uno los elementos de la sucesión #48 como elementos de la nueva sucesión #511. La diferencia es que si se cambia la sucesión #48, este cambio no se reflejará en la nueva sucesión, mientras que si se la hubiera agregado como subsucesión, un cambio en la #48 se reflejaría cuando se muestre la #411. Solicitamos al lector nos disculpe el uso de los números, pero pensamos que facilitaría la explicación;
- Finalmente, se puede usar un componente de SPRP denominado RISP para elaborar una lista de ítems a agregar a la sucesión. El RISP (Relate Items in Structured Publishing) permite agregar a los ítems descriptores adicionales precisamente para este fin: construir una lista de ítems para agregarlos a una sucesión. Al lector interesado en este tema se lo refiere una vez más a (Bauer Mengelberg, 2012) que trata precisamente de este tema.

Hay una facilidad adicional: se puede copiar una sucesión para crear otra, misma que posteriormente se modificará del modo indicado por el nuevo uso. Por ejemplo, si se quisiera agregar algo a una presentación existente sin afectar la anterior, se usaría esta facilidad. Observe que se puede lograr el mismo resultado incluyendo como único elemento (inicial) de la nueva sucesión a la “fuente” como subsucesión; la ventaja que ofrece el copiado es que le asigna a la nueva sucesión los atributos de la original y no solo sus elementos.

Como se mencionó, el sistema ocupa otra función: detecta si la sucesión contiene “ciclos”. La posibilidad de que esto suceda es consecuencia de la inclusión de Subsucesiones. Supongamos que la sucesión #56 tiene como uno de sus elementos la sucesión #15, que a su vez contiene a la #56 como elemento (esto puede suceder de modo un poco menos evidente, es decir, en otra capa.) Esto resultaría en una ejecución infinita de la sucesión #56. El programa descubre estos ciclos y le informa al usuario dónde suceden y no grabará la sucesión en la base de datos hasta que se corrija la deficiencia.

3.4 Lectura De Una Sucesión

Naturalmente el objetivo de los U-Book es ofrecer material a los lectores: esto se hace mediante las sucesiones. El lector invoca una sucesión mediante el número de la misma, que proporciona o selecciona de una lista de las sucesiones definidas.

El programa prepara el “desplegado” – se llama *show* en el SPRP – de la sucesión elegida. Esto consiste en mostrar uno tras otro los elementos de la sucesión. Lo que el lector recibe es el contenido de los ítems indicados por la sucesión. Si el elemento es una subsucesión, mostrará uno tras otro sus elementos, hasta llegar al ítem. Se ubica el archivo correspondiente y se muestra en la forma que ve el usuario, para lo cual se usan los objetos gráficos apropiados según el tipo de archivo: audio, texto, imagen, etc.

No se describe el programa que efectúa estas tareas, excepto en forma parcial en el resto de las secciones de esta tesis, especialmente en lo que se refiere a los elementos asociados simultáneos, en los cuales el programa debe indicar en qué “forma” los muestra (en una mitad de un monitor o en otro dispositivo, como se explica en la sección siguiente).

3.5 Lo que no se incluyó de U-Books

Debido a la naturaleza de esta tesis, pero especialmente dada la longitud de una descripción de los componentes y conceptos de U-Books que no se incluyeron, procederemos a presentar algunos de estos componentes ausentes.

Se mencionó la posibilidad de ofrecer un ítem en varias versiones y/o idiomas. Esto permite adecuar el material que ve un lector a sus preferencias o posibilidades. Ejemplos podrían ser el idioma en el que se le presenta, o una versión para expertos y otra para los que se inician en el tema. La versión e idioma que se usa para mostrar cada ítem se determina de diversos modos: digamos que el fundamental es que el lector indique sus preferencias, pero el creador de la sucesión puede seleccionar la versión o idioma en la que se mostrará algún elemento de la sucesión, independientemente del lector.

Otro aspecto fundamental de los U-Books es lo que Bauer llamó “privilegios restringidos”: se trata de que ciertos usuarios no puedan usar (ni ver) ciertos ítems. Esto puede ser por imposición del autor: (el lector ya compró un capítulo pero no le ofrecemos los siguientes porque todavía no los pagó es el ejemplo más desagradable del uso de estas facilidades. Pero también puede ser que un lector de un U-Book declare que no le interesan ciertos temas o materiales. Una persona “lee” un catálogo de un museo – preparado con una sucesión de un U-Book – pero indica que no quiere ver los muebles ni los libros, solo los cuadros y esculturas. El SPRP contiene todos los elementos que permiten usar estas restricciones. Una vez más referimos al lector interesado al multicitado (Bauer Mengelberg, 2011) en el que se explican estos conceptos y que contiene abundantes ejemplos.

Naturalmente el SPRP tiene un módulo de Control de Acceso a las funciones, es decir, los usuarios de un libro podrán usar las funciones para las cuales están autorizados. Puede haber lectores que podrán crear sus propias sucesiones, mientras otros no tendrán ese privilegio. Lo mismo sucede con todas las funciones del paquete. El sistema de control utilizado es un RBAC (Role Based Access Control)

donde a cada usuario se le asigna un rol (de hecho son varios). El rol a su vez especifica las funciones que puede utilizar el usuario.

El SPRP usa un concepto de SITE, el que se pueden crear tantos U-Books como se desee. Un usuario lo es del SITE, pero también lo será de los libros en los que el administrador los incluya. El SITE se puede contemplar como un “programa” y una base de datos general, que contiene todo lo que se aplica a todos los libros del SITE.

3.6 Realidad virtual en U-Books

La posibilidad de mostrar más de un ítem en forma simultánea ofrece posibilidades de usar varios dispositivos. El motivo fundamental que motivó esta posibilidad fue la de simular una realidad para los oyentes de una clase o presentación. Se trata de aprovechar varios monitores (o proyectores, televisores o simplemente varios pares de bocinas) para reforzar los cinco impactos deseados de este tipo de comunicación. Usaremos la nomenclatura de una clase y sus alumnos, aunque se aplican los conceptos a cualquier tipo de comunicación presentador-audiencia.

Recepción: al usar varios monitores o bocinas se pueden obviar obstáculos físicos. Es importante señalar que se puede mostrar el mismo objeto en dos monitores, y no solo objetos diferentes.

Atención: el mensaje solo llega si el alumno presta atención en ese momento. Los nuevos dispositivos proporcionan al maestro opciones para incluir “trucos” o artificios para captar la atención de los alumnos, típicamente cuando desea comunicar algo que considera importante o fundamental.

Comprensión: el mensaje debe combinar contenido y forma de modo que los alumnos (cada uno de ellos) lo entiendan. Un ejemplo sencillo es el idioma del mensaje, o el modo en el que se le comunica; si un alumno no entiende una palabra, abreviatura, término técnico o similar, no lo entenderá. Y si contiene términos o conceptos que no entiende, no servirá de nada el mensaje. Naturalmente el pizarrón simultáneo puede ser de gran utilidad para este fin.

Asimilación: el alumno procesará el mensaje: esencialmente lo combinará con conocimientos que ya tiene. Aquí es donde se espera el mayor impacto del desplegado simultáneo de objetos. Por ejemplo, se puede “dejar visible” un diagrama general mientras se presentan sus elementos por separado. Esto será de ayuda al alumno que, al recibir una explicación de un componente, tiene que relacionarlo con el “todo”.

Retención: se trata de que el alumno recuerde el contenido, o la mayor parte del mismo posible. Al ofrecer más posibilidades gráficas y sonoras, incluyendo la de dibujar en vivo alguna cosa, el maestro puede preparar su clase incluyendo elementos diseñados a conseguir este aspecto de la educación.

Un alumno que mira la imagen proyectada por un proyector conectado a la computadora fácilmente se distraerá, especialmente si el material le resulta aburrido, difícil de comprender o sencillamente no le interesa. El presentador lo complementará con comentarios, pero aun así perderá la atención de varios de sus alumnos, especialmente en función de la duración de sus mensajes. Hay mucha literatura que refleja investigaciones del uso de TIC para impartir clases, y se encuentra un denominador común: el uso de una presentación tipo PowerPoint por sí sola no necesariamente es un buen modo de comunicarse con los alumnos. Para subsanar esta deficiencia, se han usado videos como material didáctico, pero se descubrió que si no están muy bien hechos (producidos) su impacto puede ser nulo.

Esto resultó en la búsqueda de otros elementos de la tecnología que complementen los que se usan habitualmente. Sin embargo, en muchos casos, su uso requiere de conocimientos de informática mayores a los que tienen los maestros. La investigación que se describe aquí incluyó como objetivo facilitar el uso de elementos adicionales de TIC sin tener que estudiar computación o profundizar en el estudio de algún paquete de software. Naturalmente este objetivo es adicional al de mejorar la comunicación en el sentido de incrementar su efecto en los 4 aspectos señalados.

Para lograr estos objetivos, se le agregaron a los U-Books (de hecho, al SPRP) los siguientes componentes para el desplegado de los elementos de una sucesión:

- **Varios Monitores:** en lugar de usar un monitor visible a la audiencia (típicamente proyectado) se pueden usar varios de ellos, por ejemplo, con 2 proyectores o un proyector y un televisor.
- **Varias Bocinas:** se podrán usar diferentes bocinas para objetos simultáneos. Claro que será un solo audio, pero se oirá en un par diferente de bocinas, lo que se puede aprovechar para simular una conversación, o simplemente para conseguir la atención de los oyentes.
- **Pizarrón “Simulado” o Virtual:** se usa una forma para escribir o pintar como si fuera un pizarrón electrónico.

3.7 Hardware y Software para la realidad virtual y el U-Book

En el resto de esta tesis se describirán los elementos de realidad virtual anunciados en la sección anterior. Se mostrará cómo se usan los monitores cuándo hay más de uno, cómo se asocian las bocinas a los objetos que se muestran (incluyendo el modo de dirigir los sonidos a las bocinas indicadas por la sucesión) y cómo se definen y usan los pizarrones.

CAPITULO IV. MÚLTIPLES PANTALLAS

El uso de dos o más pantallas se está extendiendo cada vez más. Según algunos estudios, el uso de múltiples monitores en forma extendida puede aumentar la productividad en más de un 40%, un dato que no sorprende cuando consideramos que en sistemas operativos multitarea el Escritorio se llena rápidamente de iconos y ventanas.

Poner dos monitores en forma extendida puede ser una manera económica de mejorar el uso de la computadora ya que:

- Incrementa el área de visualización;
- Incrementa la productividad;
- Permite realizar tareas diferentes en forma simultánea;
- En algunos casos puede ser más económico que un solo monitor con la misma área.

Algunas desventajas:

- El uso de dos monitores puede abarcar mucho espacio en el escritorio, lo que lo convierte en un lugar menos apto para trabajar;
- Al usar dos monitores conectados entre sí, aumenta el consumo eléctrico;
- Puede en realidad ser costoso y se contradice con una de las ventajas mencionadas, puesto que para conectar dos monitores a una misma PC se necesita una tarjeta grafica adicional que permita el “dual-monitor”;
- Implica movimientos continuos en las cervicales, lo que a la larga puede ocasionar dolencias en el cuello.

4.1 Tarjetas de Video

En la actualidad es imposible pensar en una computadora sin **tarjeta de video**. Sin embargo, a principios de los 60' las computadoras se comunicaban visualmente con sus usuarios mediante la emisión de mensajes que se mostraban en primitivas impresoras de línea, en general tipo “télex”. A fines de esa década las impresoras

comenzaron a ser sustituidas por monitores, proporcionando una imagen electrónica al espacio de trabajo del usuario; a su vez este componente es el que más ha evolucionado a lo largo del tiempo.

La primera tarjeta gráfica fue desarrollada por **IBM** en 1981, conocida como **MDA (Monochrome Display Adapter)**: trabajaba en modo texto y era capaz de representar 25 líneas de 80 caracteres en pantalla. Se usaba con monitores monocromáticos, de tonalidad normalmente verde y de ahí parte su denominación. Durante años esta tarjeta fue tomada como el estándar en tarjetas de video monocromo.

En 1981 llegaron también los primeros colores y gráficos con la **CGA (Color Graphics Adapter)** que trabajaba tanto en modo texto como gráfico. En modo texto representaba lo mismo que la MDA, pero el texto era menos legible debido a que los caracteres se basaban en una matriz de puntos más pequeña. En modo gráfico podía representar cuatro colores con una resolución de 320x200 puntos. La CGA contaba con cuatro veces más memoria (16 Kb) que la MDA y podía conectarse a monitores RGB que eran capaces de emitir color.

En 1982 Van Suwannukul, fundador de **Hercules Computer Technology**, desarrolló la **HGC (Hércules Graphics Card)**. Sus posibilidades eran abrumadoras comparadas con las anteriores, puesto que además del modo texto, podía gestionar dos páginas gráficas con una resolución de 720x348 puntos en pantalla. Con ello combina la estupenda legibilidad en modo texto de la MDA con las capacidades gráficas de la CGA, ampliando incluso la resolución. Sin embargo la tarjeta HGC no era capaz de mostrar color por pantalla por lo que no llegó a estandarizarse como la CGA. La tarjeta Hércules tenía una memoria de 643 Kb y no era totalmente compatible con las tarjetas de IBM.

En 1985 IBM presentó la **EGA (Enhanced Graphics Adapter)** que era compatible con **MDA** y **CGA**; con una resolución de 640x350 puntos, se podían representar 16 colores diferentes de una paleta de 64. También se aumentó la RAM de video hasta

256 Kb para tener espacio suficiente para representar varias páginas gráficas, tal como lo hacía la **HGC**.

En 1990 IBM desarrolló la **VGA (Video Graphics Array)** que tuvo una aceptación masiva, lo que llevó a compañías como ATI, Cirrus Logic y S3 Graphics a trabajar sobre dicha tarjeta para mejorar la resolución y el número de colores. Así nació el estándar **SVGA (Super Video Graphics Array)** con el que se alcanzaron los 2 MB de memoria de vídeo, así como resoluciones de 1024 x 768 puntos a 256 colores de los 262 de una paleta y de ese modo constituyeron un nuevo paso en la consecución de gráficos de alta calidad. Al ser compatibles con las tarjetas anteriormente mencionadas se pudo utilizar el software desarrollado sin problemas.

La evolución de las tarjetas gráficas dio un giro importante en 1995 con la aparición de las primeras tarjetas 2D/3D, fabricadas por **Matrox, Creative, S3 y ATI**, entre otros fabricantes. Dichas tarjetas cumplían el estándar SVGA, pero incorporaban funciones 3D. En 1997, **3DFX** lanzó el chip gráfico **Voodoo**, con una gran potencia de cálculo, así como nuevos efectos 3D (Mip Mapping, Z-Buffering, Antialiasing). A partir de ese punto, se suceden una serie de lanzamientos de tarjetas gráficas como **Voodoo2 de 3dfx, TNT y TNT2 de NVIDIA**. La potencia alcanzada por dichas tarjetas fue tal que el puerto **PCI** al cual se conectaban se quedó corto. Intel desarrolló el puerto **AGP (Accelerated Graphics Port)** que solucionaría los cuellos de botella que empezaban a aparecer entre el procesador y la tarjeta. Desde 1999 hasta 2002, **NVIDIA** dominó el mercado de las tarjetas gráficas (absorbiendo incluso a 3dfx) con su gama **GeForce**. En ese período, las mejoras se orientaron hacia el campo de los algoritmos 3D y la velocidad de los procesadores gráficos. Sin embargo, las memorias también necesitaban mejorar su velocidad, por lo que se incorporaron las **memorias DDR** a las tarjetas gráficas. Las capacidades de memoria de vídeo en la época pasan de los 32 Mb de GeForce, hasta los 64 y 128 Mb de GeForce 4.

Actualmente, **NVIDIA y ATI** se reparten el liderazgo del mercado con sus series de chips gráficos GeForce y Radeon, respectivamente.

El soporte nativo de Windows para varios monitores, sin embargo, deja mucho que desear, pues se limita a ofrecer un área supletoria para colocar ventanas e iconos. El resto de la interfaz no parece tener en cuenta las necesidades de quienes usan dos o más monitores. Los principales fabricantes de tarjetas gráficas (ATI y nVidia) y algunos programadores han intentado arreglar esta carencia.

4.1.1 Conectores de video

Actualmente se ofrece una gran cantidad de conectores de vídeo en el mercado para uso doméstico y profesional; la diferencia entre ellos es el número de pines y de los conectores que se usan, y cómo se usan para transmitir la crominancia, luminancia, audio y video. La crominancia es el componente de la señal de vídeo que contiene las informaciones del color. Por otra parte, la luminancia da la luz o brillo.

Video analógico

- **S-Video:** también conocido como Y/C (o erróneamente conocido como Super-Video), es un tipo de señal analógica de vídeo. S-Video (Figura 13), tiene más calidad que el video compuesto, ya que el televisor dispone por separado de la información de brillo y la de color, mientras que en el vídeo compuesto se encuentran juntas. Esta separación hace que el cable S-Video tenga más ancho de banda para la luminancia y consiga más trabajo efectivo del decodificador de crominancia. Existen otras variantes del conector MiniDin con diferente número de pines (por ejemplo los teclados y ratones, que es MiniDIN6).



Figura 13. Conector S-Video.

- **Video por Componentes:** Utiliza tres conectores de tipo RCA (Figura 14), verde, azul, y rojo. Cada uno lleva un tipo de información: el verde lleva el brillo, y el rojo y el azul llevan la crominancia. Transmite video en alta definición hasta 1080p sin señal de audio;



Figura 14. Conector RCA-Video por Componentes.

- **VGA:** Acrónimo de Video Graphics Array (Matriz de video y gráficos) (Figura 15). Es el tipo de conexión más utilizado en los monitores de PC de cualquier tipo, ya sean CRT o LCD; también la utilizan las televisiones de plasma o LCD. Es un conector de 15 pines que se diseñó en 1987 y durante años ha sido el estándar en lo que se refiere a hardware gráfico de cualquier tipo, hasta la llegada de la señal de video digital. Existe también una versión Mini-VGA, que se utiliza en algunos ordenadores portátiles: su función es la misma; la diferencia estriba en el tamaño y el hecho que no lleva los tornillos para anclar el conector;



Figura 15. Conectores VGA.

Audio y video analógico

- **Video Compuesto:** Usa un cable con un conector RCA (Figura 16), que habitualmente es de color amarillo (para diferenciarlo de otros cables RCA). El mismo cable lleva la señal de video completa. De hecho es uno de los que “peor” calidad de imagen tiene si se compara con otras soluciones mejores. Frecuentemente suele venderse un kit de tres cables RCA:
 - Amarillo para Vídeo, el mismo cable transmite luminancia (brillo) y crominancia (color) sobre un cable coaxial de 75 Ohmios (75Ω).
 - Negro o blanco (Left, canal Izquierdo, Mono) para el audio.
 - Rojo (Right, canal Derecho, Mono) para el audio.



Figura 16. Conector RCA-Video Compuesto.

- **SCART o Euro-Conector** (Figura 17): Conecta dos dispositivos, por ejemplo una Televisión y un DVD, mediante un solo cable, que transmite tanto video, como audio estéreo. Tiene 21 pines, aunque no tiene por qué utilizar todos los pines para la transmisión de audio/video. El Euro-Conector es bidireccional - puede enviar y/o recibir información. Este tipo de conector se incluye en televisores de todas las gamas, videos VHS, sintonizadores de TDT, videoconsolas, etc. Es muy utilizado en toda Europa;



Figura 17. Conector SCART.

- **Coaxial RF** (Figura 18): Es el clásico cable de antena, pero también es un cable de señal de video: lleva la señal de video y audio estéreo. Transforma las dos señales de audio moduladas en una señal de radiofrecuencia (RF);



Figura 18. Conector Coaxial RF.

Video digital

- **DVI**: Acrónimo de Digital Video Input (Entrada de video digital) (Figura 19), transmite señal de video digital en alta definición. Se utiliza sobre todo para conectar monitores de pantalla plana LCD o plasma a la tarjeta grafica de un ordenador. Es compatible con la señal VGA, pudiendo tener el mismo cable

un conector DVI en un extremo y un VGA del otro, o utilizando un adaptador en caso de ser necesario. Hay varios tipos de DVI, que se diferencian en el número de pines que tiene el conector:

- DVI-D: Transmite únicamente la señal digital.
- DVI-A: Transmite únicamente señal analógica.
- DVI-I: Transmite señal analógica y digital, es el que suelen utilizar las tarjetas graficas de ordenador.



Figura 19. Conector DVI.

- **FireWire o IEEE 1394 o iLink (Sony)** (Figura 20): Se trata de una tecnología desarrollada por Apple para la entrada y salida de datos en serie a alta velocidad (alcanza los 400 megabits por segundo de una manera bastante estable), e interconexión de dispositivos digitales. Se utiliza para transferir todo tipo de datos pero es muy utilizada para dispositivos multimedia como videocámaras y cámaras de fotos. Hay dos tamaños: el FireWire normal, con 6 pines; y el mini FireWire con 4 pines, que normalmente es el que tienen las cámaras de fotos.



Figura 20. Conector FireWire.

- **SDI y HD-SDI** (Figura 21): Es poco utilizado para uso doméstico, pero se trata de un estándar reconocido a nivel profesional. Existen dos versiones, single-link y dual-link. Su versión estándar soporta resoluciones de hasta 565p. Su versión HD-SDI soporta hasta 720p, y las versiones dual-link soportan hasta 1080p. Su principal característica es transmitir señales de video digital sin comprimir en una transmisión en serie, a través de un cable coaxial normal.



Figura 21. Conector SDI.

Audio y video digital

- **HDMI**: Acrónimo de High Definition Multimedia Interface (Figura 22), interfaz multimedia de alta definición. Es el más utilizado por televisiones de tipo LCD y Plasma que admitan imagen en alta definición, y algunos monitores para ordenador de última generación. Es el equivalente a una conexión DVI pero con el audio estéreo en alta definición incluido. Existen cables de DVI a HDMI, muy útiles para conectar un ordenador a una televisión de pantalla

plana, teniendo en cuenta que para transmitir audio necesitaremos un cable aparte, cualquier conexión de HDMI a otro tipo de conector perderá el audio en la transformación. Hay dos tipos de HDMI: de enlace simple y de doble enlace. Aunque este último soporta resoluciones superiores a 1080p, el más común es la de enlace simple. Hay varias versiones de HDMI:

- **HDMI v1.0:** Transmite video en alta definición hasta 1080p y audio de 8 canales a 192 kHz y 24 bits.
- **HDMI v1.1:** Igual que el anterior pero soporta DVD Audio.
- **HDMI v1.2:** Igual que las anteriores pero soporta transmisión de DSD para Super Audio CD.
- **HDMI v1.3:** Además de lo anterior soporta resoluciones superiores a 1080p, mayor cantidad de bits de color, y audio de alta definición como Dolby TrueHD y DTS-HD, formato utilizado por los discos Blue-Ray de Sony.



Figura 22. Conector HDMI.

- **DisplayPort** (Figura 23): Es un conector muy similar al HDMI en sus características técnicas, pero libre de licencias y cánones. Se suele incluir en algunas tarjetas gráficas, pero es raro encontrarlo en televisores. Su principal inconveniente es su incompatibilidad con DVI y HDMI.



Figura 23. Conector DISPLAYPORT.

4.2 Monitores Touch-Screen y Multi-touch

Una pantalla táctil o Touch-Screen es una pantalla que mediante un toque directo sobre su superficie permite la entrada de datos y órdenes a una computadora, y a su vez muestra los resultados introducidos previamente, actuando como periférico de entrada y periférico de salida de datos. Actualmente hay pantallas táctiles que pueden instalarse sobre una pantalla normal. Las pantallas táctiles son de una tecnología llamada “**Resistivas**”.

Las pantallas **Resistivas** están formadas por varias capas. Cuando presionamos sobre la pantalla, hay dos capas que entran en contacto: esto produce un cambio en la corriente eléctrica y se detecta la pulsación. Aparte de su precio, la mayor resistencia al polvo o al agua ha contribuido a su popularidad. Sin embargo, el uso de múltiples capas hace que su brillo se reduzca en aproximadamente un 25 por ciento. Pueden usarse con los dedos o el stylus: los objetos afilados pueden dañarlas.

Uno de los principales inconvenientes que se les atribuye a las pantallas resistivas es la imposibilidad de detectar varias pulsaciones, lo que se denomina el Multi-touch.

Otra tecnología de pantallas son las “**Capacitivas**” o **Multi-touch**. Estas pantallas se tienen que manejar mediante el dedo o un objeto que disponga de capacitancia, no siendo aptas para los típicos stylus. Por otro lado, pueden detectar varias pulsaciones simultáneas o gestos, lo que permite diversas formas de actuar con ellas, aumentando su capacidad para ser controladas. Las pulsaciones o gestos no

requieren presión, basta con deslizar el dedo para controlar la pantalla del dispositivo.

Al ser utilizadas directamente con el dedo, sin objetos intermedios, y sin requerir presión alguna, la experiencia para el usuario al manejar este tipo de pantallas es bastante buena. La impresión es de rapidez, de inmediatez, siempre que el sistema operativo y el programa que estemos manejando ayuden a obtener estos resultados.

También tienen sus limitaciones: el uso de los dedos puede reducir la precisión y por lo tanto se limitan las posibilidades del software que pueden ejecutar.

4.3 Múltiples Pantallas y el U-Book

Una presentación en el U-Book puede usar varias “formas”, es decir, se pueden mostrar dos o más objetos en forma simultánea en más de una forma: para ello se usan varios monitores. En cada monitor, se puede mostrar un objeto ocupando todo el monitor, como se muestra en la figura 24.

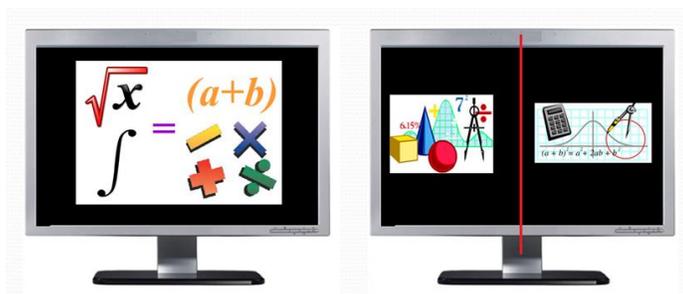


Figura 24. Configuración de dos pantallas.

Para poder realizar las pruebas y el desarrollo del software de múltiples pantallas en el U-Book se requirió del siguiente equipo:

- Una tarjeta de video ATI - Radeon HD5770 (Figura 25), con tecnología ATI-eyefinity (es la combinación de hardware y software, que permite hasta 3 pantallas en forma extendida);



Figura 25. Tarjeta de Video ATI –RADEON HD5770.

- Conectores HDMI, DVI y DisplayPort;
- Un monitor Touch-Screen, un proyector y un monitor (para poder realizar la configuración de 3 monitores en 1 sola computadora).

Todo el hardware se instaló en una computadora de escritorio y la configuración fue la que ilustra la (Figura 26).

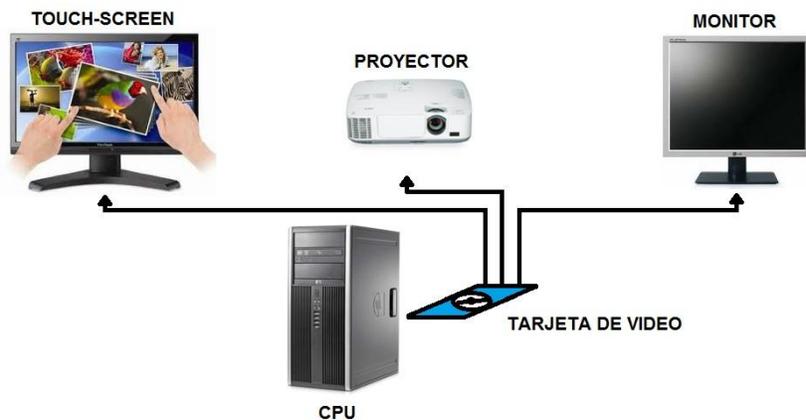


Figura 26. Configuración con dos monitores y un proyector.

Regularmente las tarjetas de video tienen varios puertos, para los cuales se utilizaron dos conectores DVI y uno de tipo DisplayPort.

También se puede usar una configuración distinta, que no requiere una tarjeta de video de varios puertos. Se usa una computadora portátil, a la que se le puede agregar un monitor o proyector puesto que tienen incorporada una tarjeta de video que lo permite. De este modo, la configuración que usaría un U-Book en este caso sería la que se muestra en la Figura 27.



Figura 27. Configuración típica para una computadora personal y un proyector.

Para esta configuración se usaron los puertos de la laptop: todas las computadoras modernas de ese tipo contienen al menos un puerto de video 1 VGA y otras un puerto extra HDMI, ilustrados en el Figura 28.



Figura 28. Puertos de video de una computadora personal.

Todas las configuraciones de múltiples pantallas tienen que ser en escritorio extendido, para poder usar diferentes formas en cada monitor. Para configurar las múltiples pantallas se utilizan las aplicaciones de Windows (*Explicación técnica, ver Anexo 1*).

Observación: como se verá más adelante (Capítulo VII) hay otro modo de utilizar varios monitores para una presentación usando el SPRP: se trata de usar dos computadoras conectadas en red. En la segunda computadora se ejecuta un programa, y se utiliza una técnica P-to-P (program to program) para desplegar en los monitores de esta los objetos que determina el programa que se ejecuta en la computadora “principal”.

4.3.1 Preparación del archivo de configuración

Para poder configurar las múltiples pantallas hay que ejecutar el sistema **CONFIG.MON.exe** antes de utilizar el **SPRPBuildShow**. Como se verá lo mismo sucede cuando se ocupe otra computadora conectada.

El programa **CONFIG.MON.exe** permite configurar las formas que utilizará una presentación o clase para trabajar en múltiples monitores conectados a la computadora (máximo 3 monitores). El programa configura las formas de cada monitor y guarda estas configuraciones en un archivo de texto. Cuando se ejecuta el **SPRP** (que permite usar un U-Book) se invoca este archivo de configuración que contiene toda la información necesaria para habilitar los monitores y dimensionar las formas que se mostrarán en cada uno de ellos.

Se presenta en “pasos” la manera en que el sistema **CONFIG.MON.exe** genera el archivo de configuración cuando hay conectados 2 monitores:

1. Se invoca el programa **CONFIG.MON.exe**.
2. Automáticamente el sistema detecta cuántos monitores hay conectados a la computadora en ese momento. (*Explicación técnica, ver Anexo 2*).
3. El programa busca (en el directorio raíz, C:) el archivo correspondiente al número de monitores. El archivo se llama: Setup_monitores_N.txt

Donde “N” es el número de monitores (paso 1).

Ejemplo: Setup_monitores_2.txt tiene la configuración para 2 monitores.

3.1 Si “NO” encuentra el archivo, el programa desplegará un mensaje diciendo que no se encuentra el archivo de configuración (Figura 29).

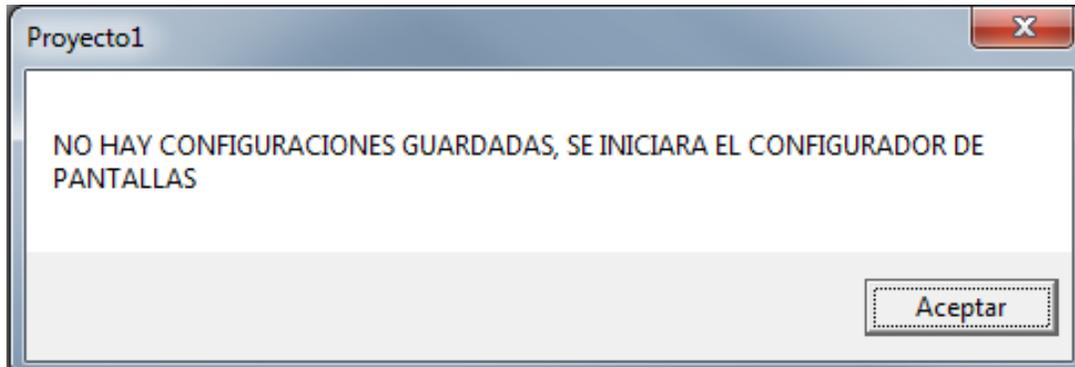


Figura 29. Mensaje de error “No hay configuración de pantallas”.

En caso que la configuración guardada no coincida con la configuración actual, desplegará el mensaje mostrado en la Figura 30.

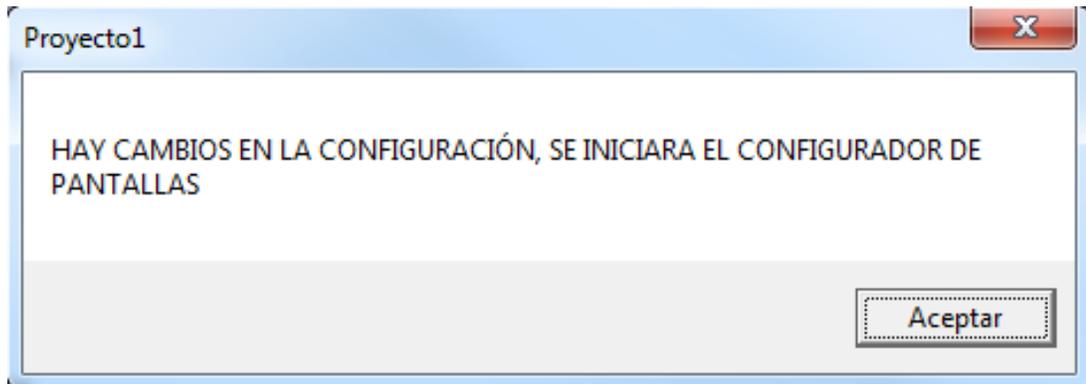


Figura 30. Mensaje de error “Existen variaciones en la resolución de los monitores”

Crearé el archivo de configuración una vez que haya mostrado todas las formas posibles y se hayan configurado una por una.

Ejemplo (en caso de que dos monitores estén conectados):

- Se desplegará una forma **MDI** (Figura 31)(*Explicación técnica, ver Anexo 3*);



Figura 31. Forma MDI.

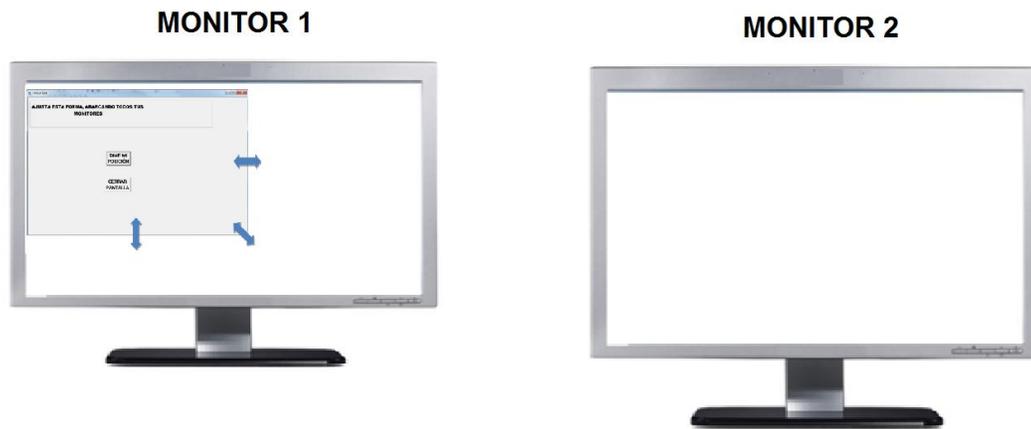


Figura 32. Manera de ajustar la forma MDI.

- Para ajustar la forma MDI (Figura) hay que arrastrar de la orilla, hasta que abarque la totalidad de los monitores;



Figura 33. Forma MDI ajustada a dos monitores.

El MDI se tiene que ajustar (manualmente) para abarcar el espacio total de los monitores, es decir inicia en la pantalla “principal” y hay que arrastrar los bordes para que abarquen todos los monitores extendidos. La altura del MDI se ajustará al menor de los monitores.

Tras ajustar el tamaño, se oprime el botón “CERRAR PANTALLA”.

- Se desplegara la FORMA 1(Figura 34);

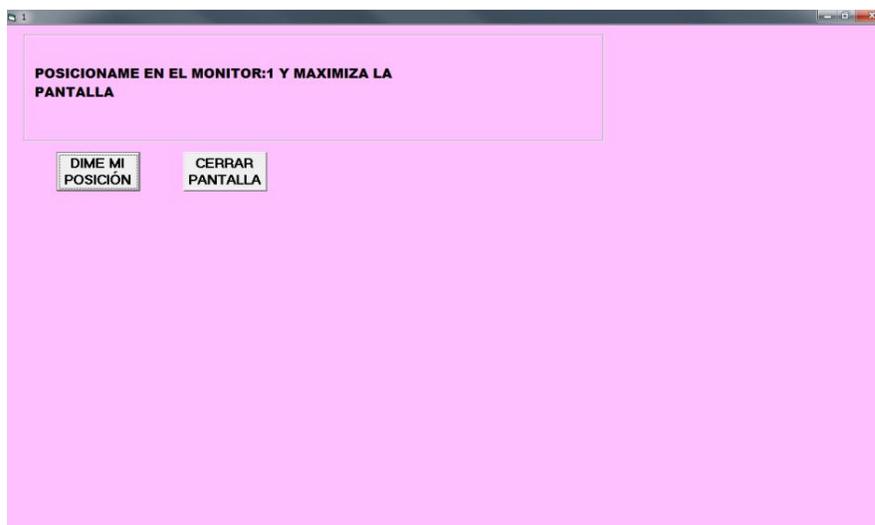


Figura 34. Forma uno para la pantalla uno completa.

La forma 1, se ajusta automáticamente en la pantalla principal (*Explicación técnica, ver Anexo 4*). Si se necesitara modificar las dimensiones y la posición, se arrastran los bordes a la posición correcta.

Tras ajustar el tamaño o verificar el tamaño, se oprime el botón “CERRAR PANTALLA”.

- Se desplegara la FORMA 2 (Figura 35);

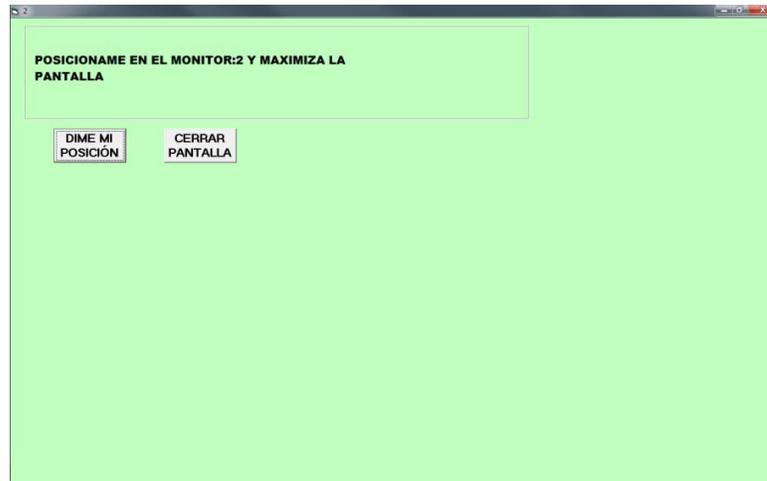


Figura 35. Forma dos para la pantalla dos completa.

La forma 2, se ajusta automáticamente en la pantalla 2 (*Explicación técnica, ver Anexo 4*). Una vez más se arrastran los bordes a la posición correcta.

Tras ajustar el tamaño o verificar el tamaño, se oprime el botón “CERRAR PANTALLA”. Todas las formas tienen un botón que dice “DIME MI POSICIÓN”, si se le oprime despliega una leyenda indicando su tamaño y posición de esa forma en particular (Figura 36).



Figura 36. Propiedades de una forma.

Una vez configuradas todas las formas disponibles para estos monitores se creará el archivo de configuración, que siempre reside en el directorio raíz de la computadora. (*Explicación técnica, ver Anexo 5*).

3.2 Si “SI” existe el archivo, lo abre, obtiene las “medidas” (ancho y alto de la pantalla) y posición relativa de cada forma previamente configurada. Estos datos se almacenan en memoria. (*Explicación técnica, ver Anexo 5*).

4. A partir de los datos guardados, prepara todas las formas disponibles para cada monitor (ya con las medidas configuradas, ancho, alto y posición).

Ejemplo 2 monitores:

- Monitor 1 y 2:
 - Forma MDI.
- Monitor 1:
 - Forma (0): el monitor total.
 - Forma (1): la mitad izquierda.
 - Forma (2): la mitad derecha.
- Monitor 2:
 - Forma (0): el monitor total.
 - Forma (1): la mitad izquierda.
 - Forma (2): la mitad derecha.

Toda la información se presenta en la Figura 37.



Figura 37. Propiedades de configuración de dos pantallas.

En esta pantalla se ofrecen las siguientes acciones:

- Ver la cantidad y la información de cada una de las pantallas conectadas;
- Ver la cantidad y las propiedades de cada forma configurada;
- Las propiedades de la forma MDI;
- Volver a configurar cada forma;
- Visualizar una o dos formas en cada monitor.

- **Ejemplo 2 monitores**

Para el caso de 3 monitores (que como vimos sólo es posible si la computadora tiene una tarjeta de video de 3 puertos) no se presenta todo el detalle, solamente “las imágenes” para configurar la forma **MDI** y la tercer Forma, ya que la para las Formas uno y dos el proceso es similar al anterior.

- Ajuste de MDI (Forma 38);



Figura 38. Forma MDI, para tres monitores.

- Para ajustar la forma MDI (Figura 39) hay que arrastrar de la orilla, hasta que abarque la totalidad de los monitores;



Figura 39. Forma MDI ajustada a tres monitores.

- Ajuste de la Forma 3 (Figura 40);

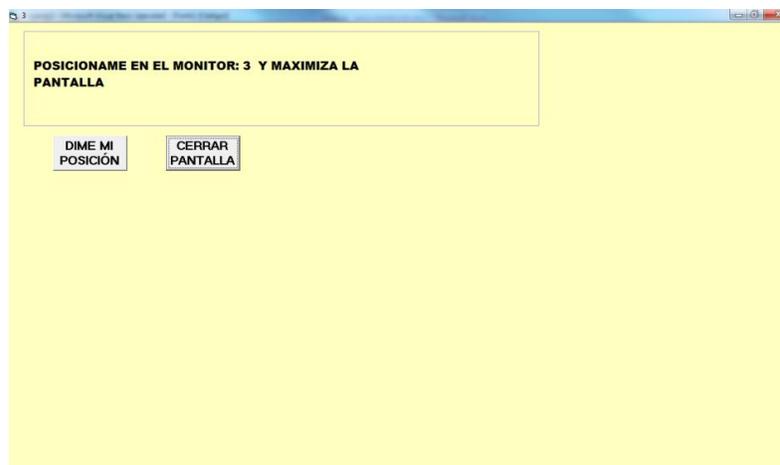


Figura 40. Forma tres para la pantalla tres completa.

La forma 3, se ajusta automáticamente en la pantalla 3 (*Explicación técnica, ver Anexo 4*). Se pueden arrastrar los bordes a la posición correcta en caso que se quiera modificar las dimensiones y/o la posición.

Tras ajustar el tamaño o verificar el tamaño, se oprime el botón “CERRAR PANTALLA”.

Una vez configuradas todas las formas disponibles para estos monitores se creará el archivo de configuración. (*Explicación técnica, ver Anexo 5*).

Toda la información del ajuste de 3 monitores es presentada en 1 sola pantalla (Figura 41).



Figura 41. Propiedades de configuración de tres pantallas.

4.3.1.2 Diagrama del proceso de "CONFIG.MON.exe"

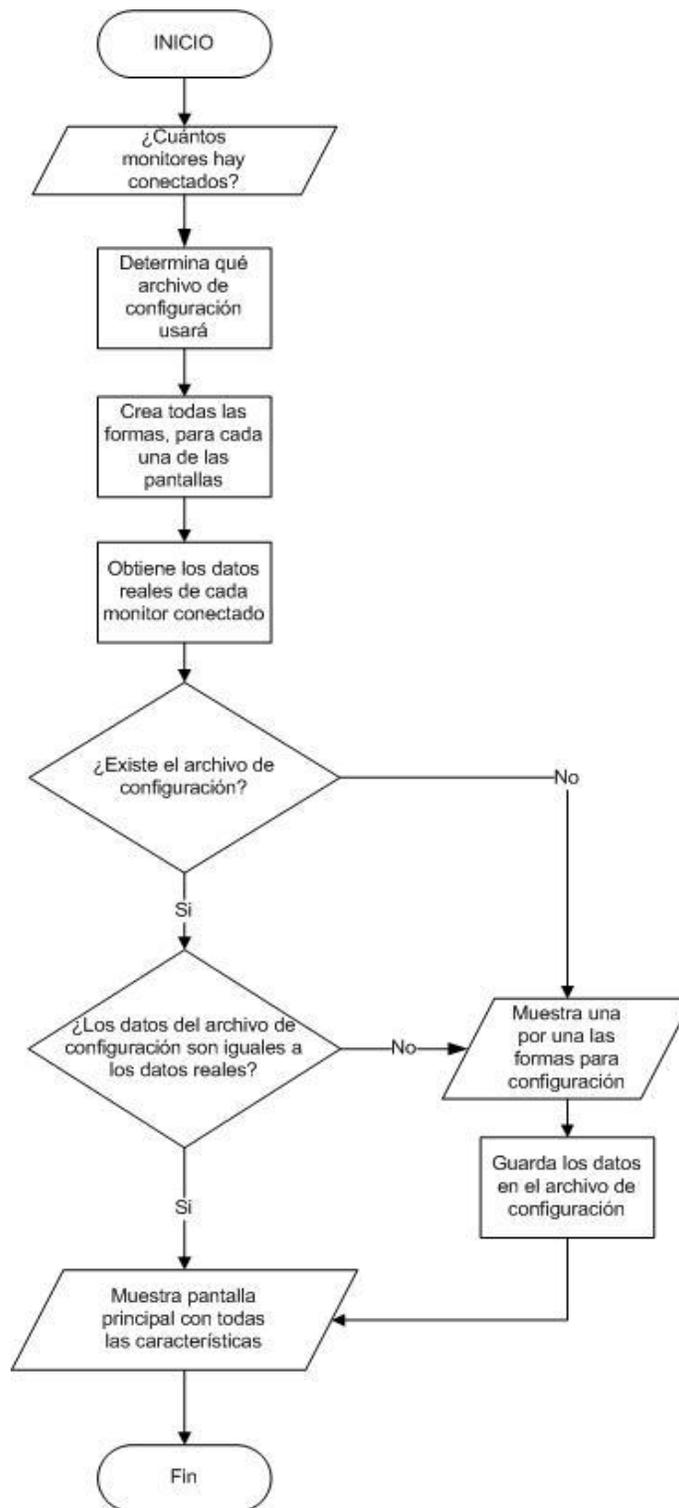


Figura 42. Diagrama del proceso de CONFIG.MON.exe.

4.3.2 Preparación de formas y monitores

Una vez creado el archivo de configuración, el **SPRPBuildShow** determina la cantidad de monitores conectados en modo extendido (*Explicación técnica, Anexo 2*) y de ese modo sabe cuál archivo de configuración utilizará. Posteriormente hace la configuración de todas las formas y las muestra para comprobar que estén correctamente dimensionadas. Si no es el caso, hay que volver a ejecutar el archivo de configuración.

Ejemplo de 2 monitores conectados:

Dos monitores tienen 7 formas, **A**, **B**, **C**, **D**, **E** y **F** y la forma extra **G** (la MDI; que abarca la totalidad de los monitores). Antes de leer el archivo de configuración las formas se verían como en la figura 43.

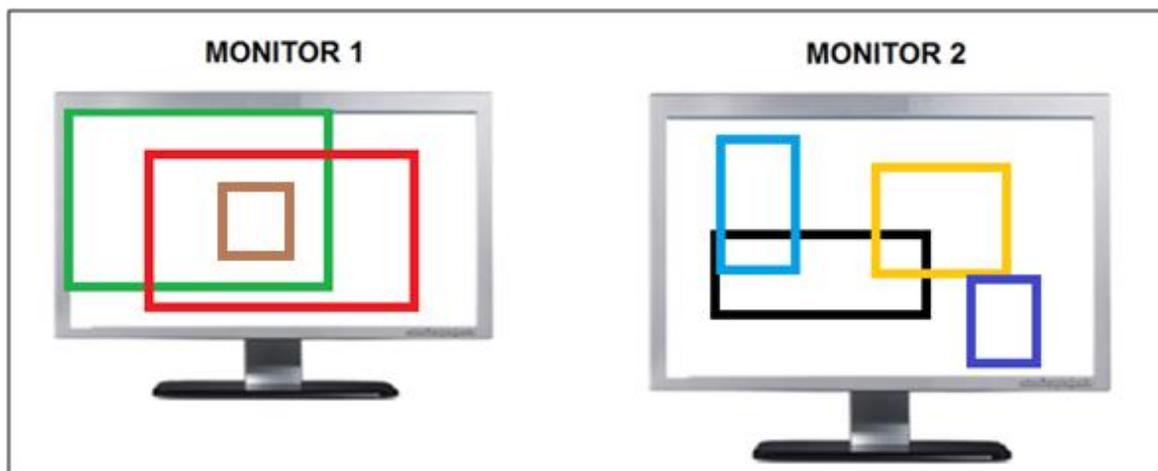


Figura 43. Totalidad de formas disponibles para dos monitores.

Una vez configuradas las formas con los siguientes parámetros:

- Forma **G**-(monitor 0, posición: **total (0)**) → se indica como monitor cero, que significa que abarca la totalidad de ambos monitores; su altura siempre será la del monitor de menor altura. Como se vio, esta forma la usa el programa como forma MDI, es decir, la forma “padre” de las restantes;
- Formas **A**-(monitor 1, posición: **total (0)**);
- Formas **B**-(monitor 1, posición: **izquierda(1)**);

- Formas **C**-(monitor 1, posición: **derecha(2)**);
- Formas **D**-(monitor 2, posición: **total (0)**);
- Formas **E**-(monitor 2, posición: **izquierda (1)**);
- Formas **F**-(monitor 2, posición: **derecha (2)**).

Después de estos ajustes, las formas tendrán la posición que se muestra en la Figura 44.

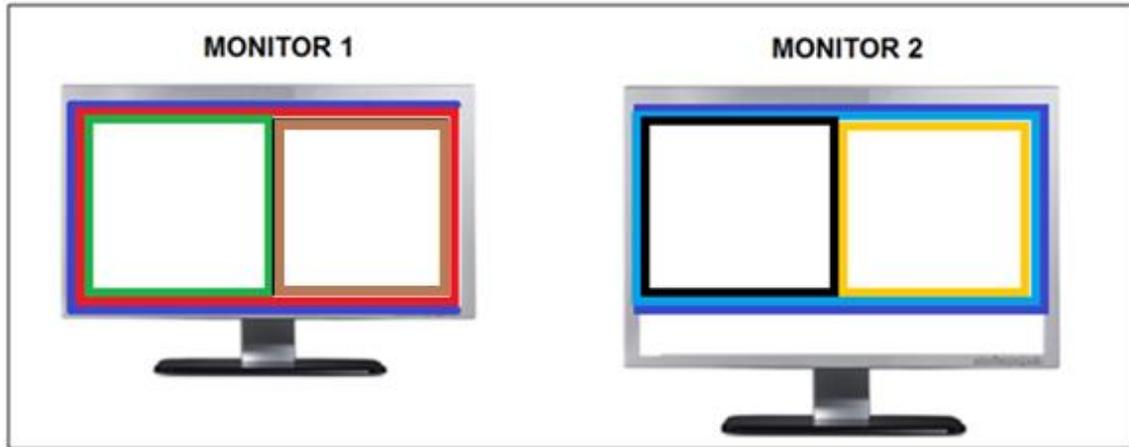


Figura 44. Las formas ya ajustas a los dos monitores.

Se presenta en pasos la manera en que el programa **SPRPBuildShow** dimensiona y posiciona las formas que usará en cada monitor conectado, con la aclaración de que el programa que se ejecuta en la otra computadora – si hay una conectada – lo hace en modo similar:

1. Se invoca el programa que se usará.
2. Automáticamente el sistema detecta cuántos monitores hay conectados a la computadora en ese momento. *Explicación técnica: (Explicación técnica, ver Anexo 2).*
3. El programa busca (en C:) el archivo correspondiente al número de monitores. Como vimos, el archivo se llama: Setup_monitores_2.txt y contiene la configuración para 2 monitores. *(Explicación técnica, ver Anexo 5).*

3.1 Si no se encuentra el archivo, el sistema desplegará un mensaje (Figura 45) diciendo que no se encuentra el archivo de configuración, y que se ejecute el **CONFIG.MON.exe** para volver a configurar todas las formas.

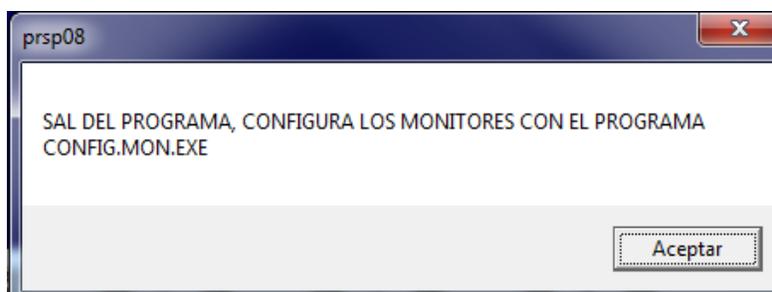


Figura 45. Mensaje de error “No existe el archivo de configuración”.

- 3.2 Si está el archivo, lo abre, obtiene las “medidas” (ancho y alto de las pantallas) y posición relativa de cada forma previamente configurada. (*Explicación técnica, ver Anexo 6*).
4. El programa **SPRPBuildShow** (y el similar en la otra computadora) contiene un módulo (.bas) llamado Configuración_y_formas. Este módulo contiene las rutinas necesarias para poder configurar 1 forma con un cierto tamaño ya establecido. Existen 2 formas base que se usarán: estas formas contienen todos los objetos que usa una sucesión: una de ellas se usa exclusivamente para ítems que representan archivos PDF (Figura 46), mientras que la otra ITEMS (Figura 47) se usa para todos los restantes tipos de objetos, incluyendo textos, imágenes, audio, video, etc. Ninguna de estas formas tiene una posición establecida: su posición se determina en tiempo de ejecución de acuerdo a las dimensiones de las formas en las que se colocan. Esto se hace al inicio del programa y solo una vez. Estas dos formas base, se clonarán dependiendo del número de monitores.

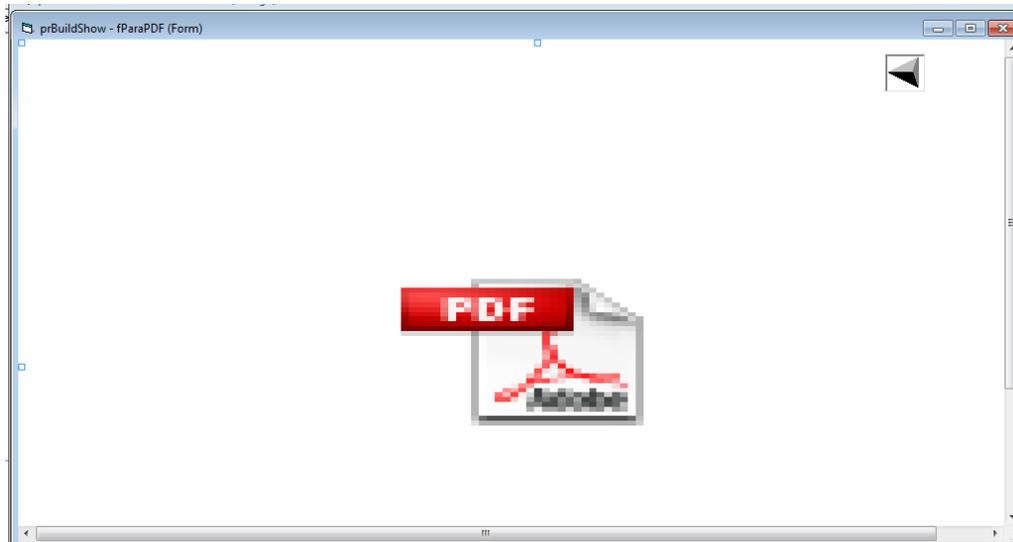


Figura 46. La forma para ítems que son archivos PDF.

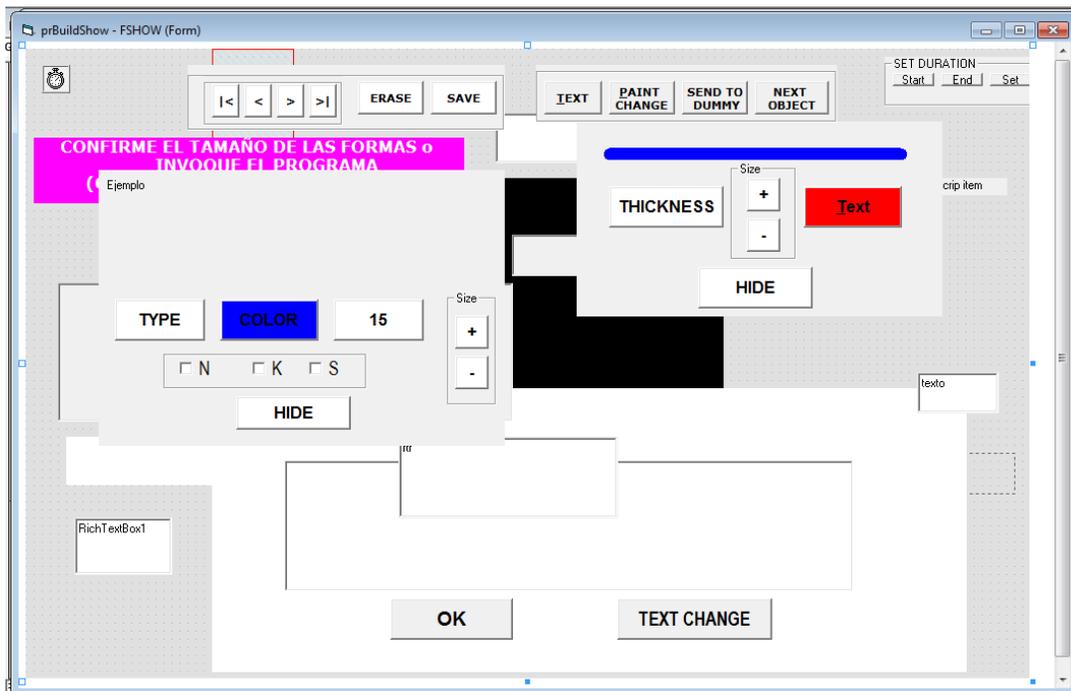


Figura 47. La forma para ítems que NO son archivos PDF.

Una vez que se obtienen los datos del archivo de configuración, el programa procede del siguiente modo:

- Redimensiona y posiciona la forma MDI, posteriormente se muestra. *(Explicación técnica, ver Anexo 7).*

- Asignar las posiciones y dimensiones de todas las formas hijas posibles. Lo hace una por una; para el caso de dos monitores son 6 formas. (*Explicación técnica, ver Anexo 8*).
- Posiciona todos los objetos que contienen las dos formas base. (*Explicación técnica, ver Anexo 9*).
- Oculta todos los objetos y muestra las formas para verificar que tengan las dimensiones correctas y estén donde deben estar (*Explicación técnica, ver Anexo 10*).

El monitor 1 (Figura 48) quedaría de la siguiente forma (cada forma tiene un borde rojo para identificarlas fácilmente).



Figura 48. SPRP monitor uno configurado.

Como se ve, las tres formas son “correctas”, ya que sobresale una línea roja que esta exactamente a la mitad y los bordes se alcanzan a distinguir con la línea roja.

El monitor 2 quedaría de la siguiente forma (Figura 49), cada forma tiene un borde rojo para identificarlas fácilmente. En este caso las tres formas son correctas.

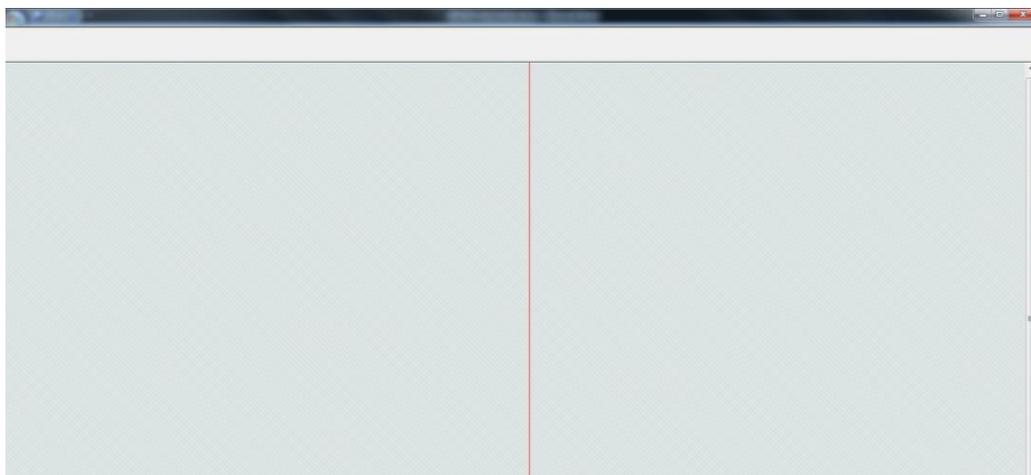


Figura 49. Las formas del segundo monitor quedaron bien configuradas.

La forma en que se verían las formas en 2 monitores se presenta en la Figura 50, con sus dimensiones correctas.

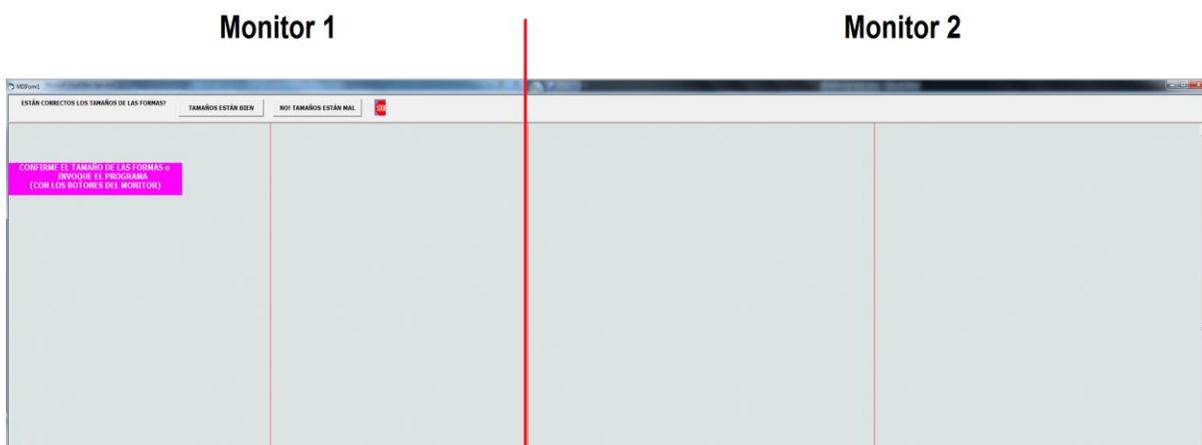


Figura 50. SPRP Monitor uno y dos configurado.

Si son correctas hay que oprimir el botón etiquetado como **TAMAÑOS ESTÁN BIEN**, y se procede al resto del programa, es decir, la selección de un libro, la selección y lectura de una sucesión, etc. Si no son correctas hay que volver a generar el archivo de configuración con el programa **CONFIG.MON.exe**.

4.3.2.1 Diagrama del proceso de preparación de formas y monitores

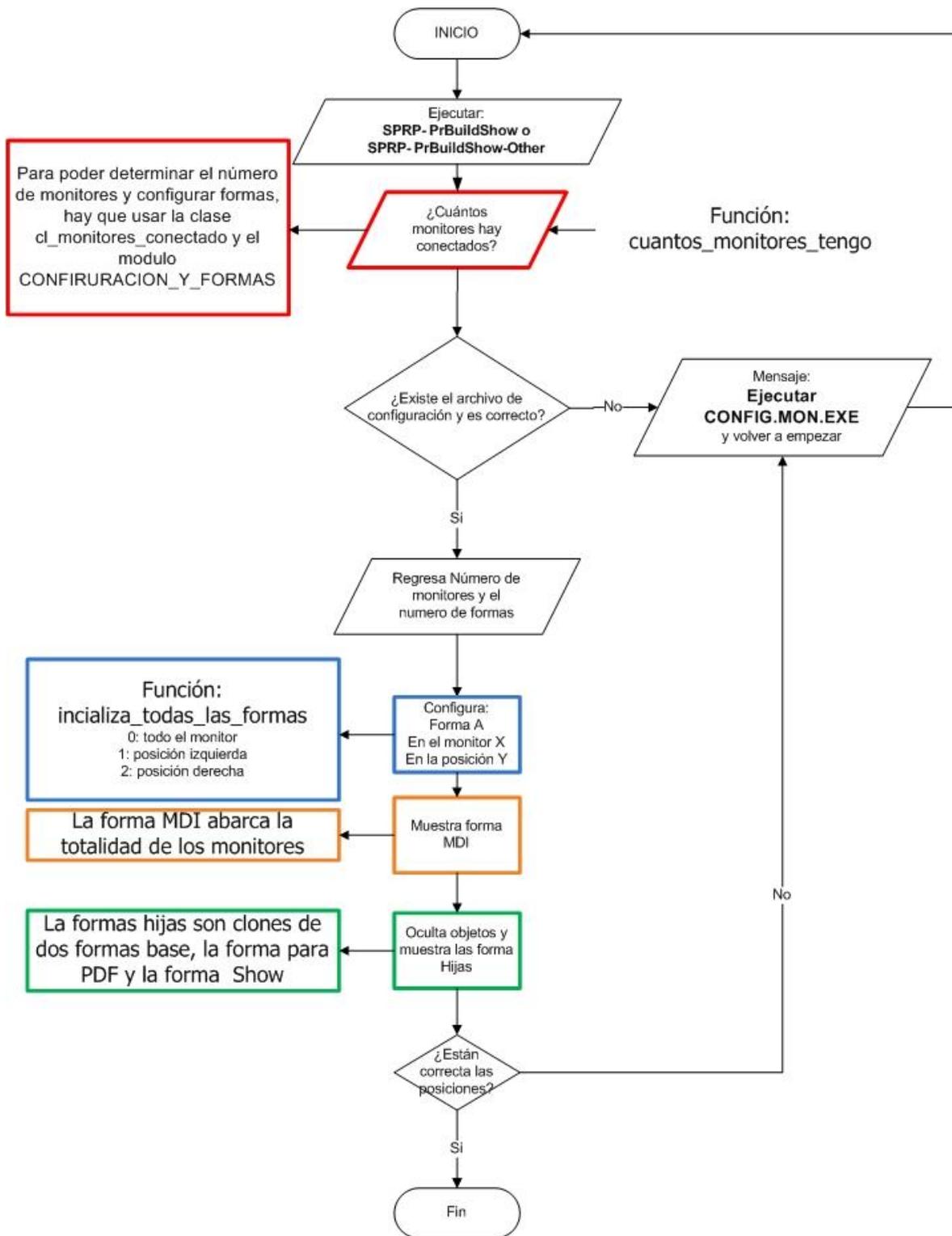


Figura 51. Diagrama del proceso de configuración de formas y monitores.

4.4 Impacto sobre el aspecto de Aprendizaje

El impacto de utilizar múltiples pantallas reside en la posibilidad de visualizar varios ítems de las Subsucesiones en forma simultánea. En muchas de las presentaciones una sola diapositiva no alcanza a ejemplificar algún tema en particular, ya que el área de visión es a un solo monitor. Se tendrían que hacer varias diapositivas de ese tema, y como se muestran una tras otra podría darse el caso de estar avanzando y retrocediendo continuamente para comprender el material (en especial, relacionar lo que se exhibe en una imagen con lo de las demás). El poder mostrar varias imágenes al mismo tiempo puede contribuir a la comprensión y asimilación del material presentado. Adicionalmente, puede reducir la irritación (casi inevitable) que produce volver a ver una imagen ya vista y de ese modo conseguir mantener la atención del espectador.

Con el uso de múltiples pantallas, podemos visualizar diferentes contenidos: el límite es el uso de 3 monitores por computadora, con la posibilidad de dividir las pantallas y mostrar diferentes formas con diferentes contenidos en sus mitades.

Como se explicó en el Capítulo III (U-Books) para preparar la sucesión, se incluyen elementos asociados simultáneos para aprovechar la disponibilidad de las formas adicionales. También se mencionó en esa explicación que esta posibilidad es recursiva: a un elemento asociado a otro, a su vez se le puede asociar un elemento, aunque esto no se puede realizar como aquí se enuncia. Será necesario especificar como elemento asociado una sucesión de un elemento, y a este sí se le puede asociar otro.

CAPITULO V. EL USO DE PIZARRONES VIRTUALES EN EL U-BOOK

Una técnica de exposición prevalente es el uso de un pizarrón o dispositivo semejante: el pizarrón real, rotafolio, pizarrón electrónico o un pizarrón que se podría denominar “simulado” que interpretamos como la interacción del presentador con su monitor que resulta en que la audiencia vea lo que está escribiendo o dibujando.

En cualquiera de estos, el expositor puede escribir o pintar, ya sea en forma exclusiva o en conjunción con otro tipo de proyecciones de objetos, que pueden estar en la mitad del mismo monitor o en un dispositivo adicional.

La Pizarra Digital Interactiva es un periférico que permite al presentador compartir con una audiencia datos, imágenes y recursos multimedia, entre otros.

Como se describió anteriormente, el SPRP ofrece la inclusión de pizarrones como ítems, y de mostrar simultáneamente más de un objeto a la audiencia. De ese modo, el presentador podrá usar un pizarrón tanto en forma aislada o aprovecharlo para ejemplificar o ampliar un objeto que se despliega en otro monitor.

Se describirán estos pizarrones en cuanto a lo que ofrecen al instructor y cómo se implementan en sucesiones de un U-Book. En particular, se podrán agregar a un pizarrón imágenes “prediseñadas”, es decir, preparadas *ad hoc*. Como se mencionó anteriormente, esto puede ser de utilidad cuando se trata de hacer un diagrama o mostrar una fórmula complicada, de modo que en lugar de tener que hacerla en tiempo real, la prepara de antemano y la incluye en la imagen del pizarrón.

5.1 Pizarra Digital Interactiva

El material de esta sección se extrajo de (Martínez y Santillán, 2006).

El pizarrón electrónico o interactivo DPI (Pizarra Digital Interactiva) usado en aulas es una superficie destinada a desplegar la información de la computadora mediante un proyector. Funciona como una pantalla común que permite visualizar y manipular

la información. Se le nombra interactivo porque brinda a los usuarios la posibilidad de elaborar, almacenar y modificar la información.

Existen en el mercado gran variedad de pizarrones interactivos (Pizarrón multimedia) (Figura 52), pero estos necesariamente tienen que ser usados con un proyector y otros dispositivos de entrada. El pizarrón multimedia puede definirse como un dispositivo o periféricos de entrada (input) que permite al usuario crear, alimentar, ordenar información e introducirla en la memoria, al igual que lo haría con otro dispositivo de entrada, como el teclado o el mouse (ratón). Según el mecanismo que tengan para manipular la información, es posible distinguir dos tipos de pizarrones:

- **Hard boards:** tienen una superficie magnética a lo largo de la pantalla y necesitan plumones especiales para actuar sobre ellos.
- **Soft boards:** poseen una membrana “táctil” en toda su superficie que es sensible tanto a las plumas electrónicas (plumones de borrado en seco) como al tacto de los dedos.

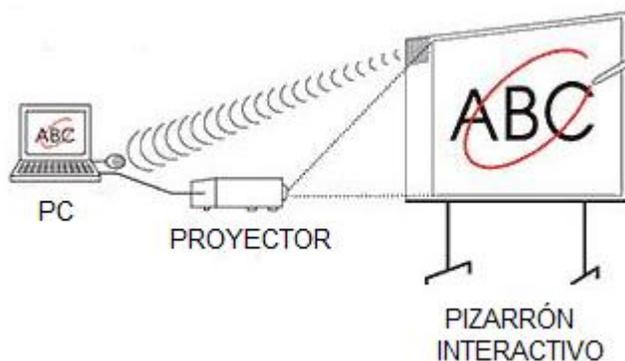


Figura 52. Pizarrón Multimedia.

5.1.1 Pizarrón Digital Interactivo Portátil

Se dice que un pizarrón interactivo es portátil cuando cumple una de las dos funciones siguientes:

- Se puede trasladar fácilmente de una clase a otra y de un lugar a otro;
- Además se puede impartir la clase desde cualquier lugar del aula y se puede utilizar cualquier superficie de proyección aunque sea una pantalla enrollable o una pantalla gigante en un auditorio.

En el primer caso estamos ante un accesorio que se suele adherir a una superficie rígida para convertirla en una pizarra interactiva. En el segundo estamos ante una PDiP (Pizarra Digital Interactiva Portátil) tipo tableta que se conecta al ordenador sin cables (por Radio Frecuencia o Bluetooth), en algunos casos, hasta permite varios alumnos actúen simultáneamente en trabajos en equipo o en competición) y permite controlar la computadora y hacer anotaciones desde cualquier lugar del aula. Incluso en este último caso, se puede utilizar un monitor o una TV y se elimina el video proyector.

5.1.2 Componentes que integran una Pizarra Digital Interactiva

Un pizarrón interactivo debe incluir como mínimo los siguientes elementos:

- Computadora para procesar el material multimedia. Esta computadora debe ser capaz de reproducir toda la información multimedia almacenada en disco o material que se encuentra en línea. El sistema operativo del equipo tiene que ser compatible con el software de pizarrón proporcionado;
- Proyector, con objeto de ver la imagen del ordenador sobre el pizarrón. Hay que prever una luminosidad y resolución suficiente (Mínimo 2000 Lumen ANSI y 1024x768). El proyector conviene colocarlo en el techo y a una distancia del pizarrón que permita obtener una imagen luminosa de gran tamaño;
- Medio de conexión, a través del cual se comunican la computadora y el pizarrón. Existen conexiones a través de bluetooth, cable (USB, paralelo) o conexiones basadas en tecnologías de identificación por radiofrecuencia;
- Pantalla interactiva, sobre la que se proyecta la imagen y que se controla mediante un puntero especial o incluso con el dedo. Tanto los profesores como los alumnos tienen a su disposición un sistema capaz de visualizar e incluso interactuar sobre cualquier tipo de documentos, Internet o cualquier información de la que se disponga en diferentes formatos, como pueden ser las presentaciones multimedios, documentos de disco o vídeos;
- Software proporcionado por el fabricante o distribuidor y que generalmente permite: gestionar el pizarrón; capturar imágenes y pantallas; disponer de

plantillas de diversos recursos educativos, de herramientas tipo zoom; convertidor de texto manual a texto impreso y reconocimiento de escritura, entre otras;

- Apuntadores especiales, en caso que no pueda ser controlado el pizarrón interactivo con los dedos.

5.1.3 Ventajas y Desventajas de la Pizarra Digital Interactiva

Las Pizarra Digital Interactivas de multimedia que se instalan en aulas para clases tienen muchas funcionalidades. Hay una variedad de ellas en el mercado con características diferentes. Las ventajas que ofrecen incluyen:

- Interactividad entre presentador y su audiencia (profesor y alumno);
- Contenido multimedia almacenada localmente y en algunos casos soporte en línea;
- Mejora la impartición de la lección proporcionando las herramientas para realzar temas clave y escribir anotaciones sobre imágenes de computadora;
- Los presentadores cuentan con herramientas para poder ejemplificar algún tema en particular;
- Facilita la comprensión de la audiencia (alumnos), especialmente en el caso de conceptos complejos dada la potencia para reforzar las explicaciones utilizando vídeos, simulaciones e imágenes con las que es posible interaccionar;

Sin embargo, tienen algunas desventajas, que incluyen:

- Costos elevados de adquisición, mantenimiento y SW;
- En muchos de los casos se requiere una configuración especial, instalación, montaje y calibración;
- Es difícil trasladar de un lado a otro;
- Requiere mantenimiento periódico;
- Hay que capacitar del personal que la opera;

- Está limitado al material que se encuentra almacenado en la computadora, en caso de requerir material extra se necesita de soporte en línea (acceso a internet);
- No ofrece flexibilidad en cuanto a que el presentador incluya su propio material y poder interactuar con lo que agregó.

5.2 Pizarrón Simulado por el U-Book

Un pizarrón es una “forma” con la que el presentador puede interactuar ya sea escribiendo textos, dibujando “líneas” o pegar imágenes diseñadas previamente al efecto, las que como se dijo se denominan “imágenes prediseñadas”.

Un pizarrón simulado por el U-Book, posee muchas de las características de un pizarrón interactivo, como escribir, pintar, colocar objetos, borrar, etc. También tiene herramientas que le ayudan al presentador poder dar una clase o una presentación y poder interactuar con una audiencia. La diferencia es que no se requiere un apuntador especial: se puede usar un mouse, un mouse-pen o con el dedo o un objeto conveniente si se cuenta con una pantalla táctil. Además se puede visualizar ya sea con un proyector, en una pantalla táctil, en un televisor, o en una pantalla normal.

El pizarrón es una forma que contiene un objeto de tipo PictureBox. Cuando se escribe un texto (en una caja de texto que se invoca al efecto) este se incluye en la imagen en la posición indicada por el presentador. Cuando dibuja algo, se incluye automáticamente en la imagen. Cuenta con una barra de herramientas: los componentes que integran la barra de herramientas se configuran dependiendo del tipo de pizarrón que se definió, siendo precisamente las opciones que se ofrecen parte de dicha definición.

En el caso del pegado de imágenes prediseñadas, se indica previamente (como parte de la preparación de la sucesión) la posición de las imágenes.

El U-Book usa dos tipos de pizarrones: el que denominamos normal: el presentador dibuja en un monitor visible a la audiencia (o la imagen del que usa se clona a uno que ven); y el llamado interactivo, donde el presentador escribe y dibuja en un monitor que no ve la audiencia, y cuando así le conviene, hace que aparezca la misma imagen en un dispositivo visible, al que se ha llamado **pizarrón DUMMY**. Se seleccionó este término en lugar de denotarlo como “esclavo”.

Trabajar en un monitor invisible a la audiencia facilita al presentador la interacción con las herramientas, medios y recursos del U-Book.

Podemos distinguir tres funciones básicas de los pizarrones usado:

- Sirve como un pizarrón común: puede pintar, escribir y borrar;
- Posibilita la interactividad, es decir, pone al alcance los materiales almacenados en una sucesión para manipularlos de acuerdo con las necesidades, por ejemplo: las imágenes prediseñadas.
- Permite mandar el contenido de un pizarrón a otro pizarrón.

5.2.1 Pizarrones Interactivos y DUMMY en SPRP

El SPRP permite usar un pizarrón que solo ve el presentador, en el cual dibuja o escribe. En el momento deseado, envía la imagen a otra forma, que es la que ve la audiencia. Nos referiremos a pizarrones con los que puede interactuar el usuario (Figura 53) como un pizarrón “interactivo”, mientras que la imagen que verá la audiencia se llamará un pizarrón “DUMMY” (se seleccionó este término en lugar de denotarlo como “esclavo”). El pizarrón DUMMY, no tiene herramientas (NO permite ninguna El pizarrón DUMMY, no tiene herramientas (NO permite ninguna interacción).

Todos los pizarrones aprovechan el recurso de múltiples pantallas y las acciones para posicionarse en una en que se posicionan.



Figura 53. Tipos de Pizarrones.

5.2.2 Catálogo de pizarrones

Se decidió ofrecer, como parte del SPRP; un conjunto de pizarrones con atributos diferentes, para obtener cierto grado de uniformidad pero también para evitar a los usuarios la necesidad de especificar los atributos de los pizarrones que usan en sus sucesiones. Para ello, se creó un catálogo de pizarrones que ofrece el SPRP, para lo cual se agregó la tabla a la base de datos del SITE. Solo los administradores del SITE podrán agregar o modificar los pizarrones que se ofrecen.

La Tabla 2 muestra *THEBLACKBOARDS* de la base de datos SITEBASE con una breve explicación de cada uno de sus campos. Como en muchos otros objetos del SPRP, las descripciones se pueden especificar en los idiomas que se incluyen en una instalación particular. Por omisión se incluyen en inglés y en español, pero se pueden agregar en otros idiomas, cuyo número estará especificado en otra tabla de la misma base de datos llamada THELANGUAGES.

Tabla 2. La tabla THEBLACKBOARDS de la base de datos SITEBASE.

CAMPO	TIPO	DESCRIPCIÓN
blackboard_num	Número	Número de Pizarrón
description_idi1	Texto	Descripción en idioma inglés
description_idi2	Texto	Descripción en idioma español
description_idi3	Texto	Descripción en el idioma número 3
description_idi4	Texto	Descripción en el idioma número 4

description_idi5	Texto	Descripción en el idioma número 5
description_idi6	Texto	Descripción en el idioma número 6
description_idi7	Texto	Descripción en el idioma número 7
description_idi8	Texto	Descripción en el idioma número 8
puede_ser_elemento_de_sucesion	Sí/No	Si/No Puede ser elemento de una sucesión.
lo_pueden_usar_ITEMS	Sí/No	Si/No Puede usar items.
es_forma_MDICHILD	Sí/No	Si/No Se muestra con show o show modal.
comandos_en_MDI	Sí/No	Si/No Los comandos están en la forma Pizarrón.
manda_imágenes_otro_blackb	Sí/No	Si/No Manda imágenes a otro Pizarrón.
erase_available	Número	Permite borrar 0=No 1=Si.
offer_navigation	Número	Valor es la suma de 0 = nada, 1 = primero, 2 = siguiente, 4=anterior, 8 = ultimo, 15 permite navegación total.
offer_ink_paint_color	Número	Ofrece color para pintar, 0=No, 1=alterna (switchear) y 2=ofrece Paleta de colores.
offer_ink_write_color	Número	Ofrece color para escribir, 0=No, 1=alterna (switchear) y 2=ofrece Paleta de colores.
offer_back_color	Número	Ofrece color para fondo del pizarrón, 0=No, 1=alterna (switchear) y 2=ofrece Paleta de colores.
offer_font_type_change	Número	Ofrece cambio de tipografía para pintar, 0=No, 1=alterna (switchear) y 2=ofrece Paleta de tipografía.
offer_font_size_change	Número	Ofrece tamaño de tipografía, 0=no, 1=alterna (switchear), 2= flechas (+, -).
offer_line_width	Número	Ofrece tamaño de línea, 0=no, 1=alterna (switchear), 2= flechas (+, -).
offer_TEXTING	Sí/No	Permite Escribir.
offer_painting	Sí/No	Permite Pintar.
use_write_sw	Número	Código (en programas) del producto usado.
use_paint_sw	Número	Código (en programas) del producto usado.
PERMITE_PREDESIGNED	Número	Permite usar objetos prediseñados .0=No, 1=Si.
divisiones_horiz_default	Número	Número de divisiones horizontales.
divisiones_vertic_default	Número	Número de divisiones verticales.
permite_cambiar_divisiones	Sí/No	Permite cambiar divisiones.
ofrece_save_imágenes	Número	Permite guardad el Pizarrón 0=No, 1= Si.
color0_fondo	Número	Color de fondo, opción 0.
color0_tinta_pintura	Número	Color de tinta para pintar, opción 0.
color0_tinta_texto	Número	Color de tinta para escribir opción 0.
font0_tipo	Texto	Nombre de tipografía, opción 0.
font0_tamano	Número	Tamaño de tipografía, opción 0.
FONT0_NEGRITA	Sí/No	Ofrece negritas, opción 0.
FONT0_ITALICS	Sí/No	Ofrece itálicas, opción 0.
color1_fondo	Número	Color de fondo, opción 1.
color1_tinta_pintura	Número	Color de tinta para pintar, opción 1.
color1_tinta_texto	Número	Color de tinta para escribir opción 1.

font1_tipo	Texto	Nombre de tipografía, opción 1.
font1_tamano	Número	Tamaño de tipografía, opción 1.
FONT1_NEGRITA	Sí/No	Ofrece negritas, opción 1.
FONT1_ITALICS	Sí/No	Ofrece itálicas, opción 1.
line_points0	Número	Tamaños de grosor de líneas, opción 0.
line_points1	Número	Tamaños de grosor de líneas, opción 1.

5.2.3 Implementación de pizarrones en SPRP

Para implementar los pizarrones en SPRP, puesto que el pizarrón es un objeto PictureBox se agregó a la forma fshow un objeto de este tipo llamado "Pic_PIZARRON". De hecho, la forma fshow es la que usa el SPRP para desplegar cualquier tipo de objeto, y para el programa, el pizarrón no es más que otro de ellos.

Se desarrolló una clase llamada `cl_UN_PIZARRON_INTERACTIVO` y cada vez que se utiliza uno de ellos se crea una instancia de esta clase para asignar las características. Esto resultará en que se habiliten (o desactiven) las acciones que tendrá el pizarrón. Esta clase contiene las funciones y métodos que usará un pizarrón interactivo. Para el pizarrón DUMMY se creó un clase llamada `CL_UN_PIZARRON_DUMMY` que incluye los métodos para poder mostrar una imagen, además de las del constructor, que se encarga de indicar el monitor en el cual aparecerá y de desactivar todas las funciones que tiene la forma show relacionadas con los pizarrones (puesto que no hay interacción con un pizarrón DUMMY).

Todos los objetos (pizarrón y herramientas), se posicionan correctamente cada vez que un se usa un pizarrón. La ubicación de los objetos depende de la forma y del monitor en el que se mostrarán. En especial el tamaño del PictureBox será diferente si se muestras en una mitad del monitor o si ocupará la totalidad del mismo.

5.2.4 Acciones que permite un pizarrón interactivo

En un pizarrón interactivo, el usuario puede invocar cualquiera de las siguientes funciones, con la salvedad de que el tipo de pizarrón lo permita, puesto que en las especificaciones del tipo de pizarrón contienen las opciones que ofrecen:

- Pintar (o dibujar);
- Escribir texto;
- Insertar imágenes prediseñadas;
- Mandar una copia exacta del pizarrón interactivo si tiene asociado un pizarrón DUMMY.

Las acciones de un pizarrón se describen a detalle a continuación en sendas secciones.

5.3 Pintar (o dibujar) sobre el pizarrón

El pizarrón interactivo cuenta con herramientas que le permiten hacer trazos o representar algo con líneas. Para esta característica se usa el ratón o un dispositivo tipo “pluma” electrónica. Si cuenta con un monitor táctil (Touch-Screen o Multi-touch) podrá usar un dedo o un lápiz Stylus.

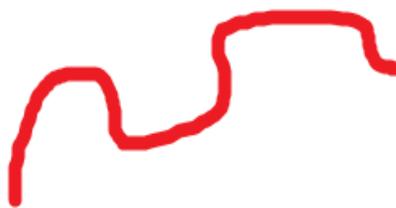


Figura 54. Acción de “Pintar” en un pizarrón interactivo.

Se pueden cambiar tanto el color como el grosor de las líneas de pintura: se invocan los cambios con el Botón “PAINT CHANGE” (Figura 54). Cuando solo se ofrece un color y grosor de tinta alterno (y no la especificación dinámica de estos datos) en lugar del botón “PAINT CHANGE” aparece uno con la leyenda “ALTERNATE COLOR”, este botón solo cambiar a los colores alternos que se tienen definidos.

5.3.1 Acciones que realiza el SPRP para pintar

Las acciones que realizan los programas **SPRPBuildShow** y **SPRPShow-Other** son las mismas y estas acciones están divididas en dos partes, cada parte utiliza los métodos que posee el PictureBox instanciado como Pic_PIZARRON y se invocan cada vez que se da un click con el mouse (o dispositivo que se esté utilizando) o se mueve el ratón. Las acciones son:

- **MOUSEMOVE**: se utilizó este método del PictureBox, en conjunción con el **MOUSECLICK** del mismo objeto. Se definió la siguiente subrutina:

```
Private Sub Pic_PIZARRON_MouseMove(Button As Integer, Shift As Integer, X As Single, Y As Single)....
```

‘Observación: se omitió aquí reproducir otras instrucciones que se usan cuando es pizarrón DUMMY o interactivo, así como algunas validaciones.

```
Pic_PIZARRON.PSet (X, Y)
```

```
End Sub
```

Esta subrutina recibe los parámetros “X” y “Y”, que indican la posición del mouse y usando el método Pset colocamos un punto; como el mouse está en movimiento, el resultado es una línea.

Para colocar el color y el grosor, se usan otros métodos del PictureBox; previamente se asignaron los valores apropiados a *color* y *grosor* con los métodos DrawWidth (grosor) y ForeColor (color).

```
With Pic_PIZARRON
```

```
.AutoRedraw = True
```

```
.ScaleMode = vbPixels
```

```
.DrawWidth = Grosor
```

```
.ForeColor = Color
```

```
End With
```

Se tiene que activar el método AutoRedraw para poder dibujar; el método ScaleMode se especifica en pixeles, para poder pintar, insertar imágenes y poder grabarlas. Si no tiene estas validaciones, lo que pintáramos no se vería reflejado: cuando se guardara la imagen del pizarrón no se guardaría nada.

5.3.2 Navegación de un pizarrón interactivo.

El método MOUSEUP del pizarrón se invoca cada vez que se termina una línea continua, y sirve para hacer una copia exacta del pizarrón para almacenarla en una pila (stack), puesto que se ofrece la navegación (atrás, adelante, al final, al principio), las que resultan en que se regrese o avance a una imagen elaborada previamente. Para ello un pizarrón interactivo tiene una barra herramientas en donde se ofrecen las opciones de navegación (Adelante, Atrás, Final, Principio).

Cada vez que se realiza una acción (pintar, escribir o insertar un objeto prediseñado), el contenido del pizarrón es almacenado en un PILA (Stack de pizarrones), se desarrollaron rutinas especiales que permiten almacenar el contenido de un pizarrón en una pila, así como las opciones de invocar cualquier parte de esta pila.

```
Private Sub Pic_PIZARRON_MouseUp(B As Integer, S As Integer, X As Single, Y As Single)
    Flag = False
    If usare_write Then Exit Sub ' usare_write establece como parte del pizarrón
    EL_PIZARRON_INTERACTIVO.agrega_este_pic_al_stack
End Sub
```

Para poder usar una pila se desarrollaron los siguientes métodos y funciones que se detallan a continuación.

Cuando se usa un pizarrón interactivo (PictureBox), antes de mostrarlo en una pantalla, se construyen todas las propiedades que tendrá. Se verifica que tenga el atributo que ofrece esta navegación; si no lo tiene se desactivan todas las opciones de navegaciones. Para la navegación se construirá una pila de pizarrones con una subrutina que se invoca por el constructo de la cl_UN_PIZARRON_INTERACTIVO llamada "construye_el_stack_de_pictures".

```
Private Sub construye_el_stack_de_pictures(FORMITA As Form) ' le llega la formita
Dim numPics As Integer, I As Integer, picName As String
numPics = 50 ' es el tamaño del stack
For I = 1 To numPics
    picName = "Pic_" & I
    Set stack_de_pics(I) = FORMITA.Controls.Add("VB.PictureBox", picName, FORMITA)
```

```

    With stack_de_pics(l)
        .HEIGHT = FORMITA.Pic_PIZARRON.HEIGHT
        .WIDTH = FORMITA.Pic_PIZARRON.WIDTH
        .Left = 0 // .Top = 0 // .BorderStyle = 0 // .Visible = false
    End With
Next l
stack_de_pics(1).Picture = Nothing
End If
End Sub

```

Esta subrutina crea un arreglo de 50 objetos PictureBox (se pueden definir más, pero no es muy probable que alguien necesite más de unos pocos) y los añade a la forma fshow, para los cual se usa el método Controls.Add. Este arreglo de PictureBox contendrá una copia exacta del pizarrón.

Después de escribir, pintar o insertar una imagen prediseñada, el contenido del Pizarrón interactivo es almacenado en la pila. Solo se recorre la posición donde se insertará., y se usa la subrutina “agrega_este_pic_al_stack” para agregar el nuevo elemento a la pila.

```

    der = der + 1
    Call crea_un_archivo_con_el_picture
    stack_de_pics(der).Picture = Nothing
    stack_de_pics(der).Picture = LoadPicture(nombre_archivo_picture_del_pizarron)

```

Para poder copiar el contenido de un PictureBox a la pila (Figura 55) es necesario guardarlo como imagen en un archivo temporal, acción que efectúa el método “crea_un_archivo_con_el_picture” cuya explicación técnica se incluyó como Anexo 23. Posteriormente, con el método LoadPicture y especificando la ubicación del archivo, se carga la imagen a la pila. Este proceso de generar un archivo para cargarlo posteriormente es necesario.

Es importante señalar que si se igualaran el contenido de dos PictureBox de la pila, puesto que estos se crearon dinámicamente, cambian muchas propiedades que al “igualar” los dos PictureBox, estarían en blanco.

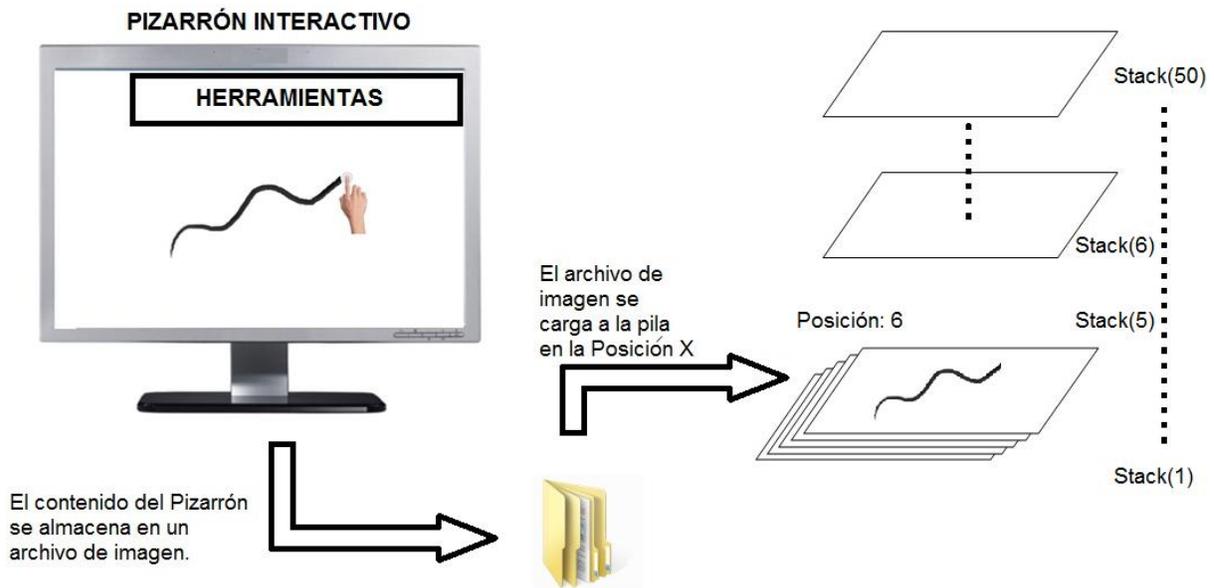


Figura 55. Funcionamiento de la pila (Stack) de pizarrones.

La navegación usa un apuntador (Figura 56) que indica la última posición del objeto almacenado en la pila y otro que indica la posición actual. Cada botón de navegación invoca la acción correspondiente. Las rutinas que efectúan estas operaciones se han incluido en el Anexo 24.



Figura 56. Navegación del Stack.

Naturalmente para todos los casos se verifica que no se desborda la pila.

5.4 Escribir texto sobre el pizarrón

El pizarrón interactivo cuenta con herramientas que le permiten escribir textos. Para invocar esta función se ofrece el botón “TEXT” del pizarrón. La Figura 57 muestra los botones que se ofrecen, con la observación de que no estarán los botones que no apliquen al pizarrón utilizado. Por ejemplo, si no hay una sucesión de objetos prediseñados para el pizarrón, no se verá el botón “NEXT OBJECT” que es precisamente el que causará la inclusión de la siguiente imagen de esa sucesión en la imagen del pizarrón. Una situación análoga sucede si no hay pizarrón DUMMY o si el pizarrón no ofrece escribir o dibujar.



Figura 57. Acción de escribir en el Pizarrón.

Para que aparezca la herramienta ilustrada en la Figura 58 en la forma, debemos seleccionar la posición donde insertaremos el texto, cosa que se logra con un doble click en el lugar donde se desea que se inserte el texto.



Figura 58. Propiedades para escribir.

Hay dos casos:

- Si no se cuenta con un *mouse-pen* o con el monitor táctil, se ofrece una caja de texto en al cual el presentador escribe el texto deseado (con un teclado);
- Si cuenta con un software externo (para el *mouse-pen* o un monitor táctil) simplemente tocando la pantalla táctil saldrá un recuadro (Software que se

incluye en cualquiera de los dos dispositivos de entrada), donde simplemente dibujando con el dedo las letras o escribiendo con él lápiz “Stylus”, automáticamente se convertirán a caracteres (letras o números). Al cerrar esta aplicación, el texto se inserta en la caja de texto del pizarrón.

Finalmente, con la tecla ENTER o el botón OK, el texto se refleja en el pizarrón.

Para cambiar las herramientas de tipografía se tiene que invocar con el Botón “TEXT CHANGE”. Cuando solo se ofrece una tipografía alterna (color, tamaño y tipo de letra) y no la especificación dinámica de estos datos, en lugar del botón “PAINT CHANGE” aparece uno con la leyenda “TEXT CHANGE”, este botón solo cambiara las propiedades alternas a la tipografía.

5.4.1 Acciones que realiza el SPRP para escribir

Las acciones que realiza el programa SPRPBuildShow (y por ende, el SPRPShow-Other) están divididas en dos partes: posiciona un Frame e inserta el texto en la imagen, cada una de las cuales utiliza métodos de objetos PictureBox y que se invocan a partir de que se invoca el “TEXT”.

Para posicionar el recuadro donde colocaremos el texto, que posteriormente se insertara en el pizarrón interactivo, se utiliza el método del PictureBox DBLCLICK, para lo cual se implementó la siguiente función

```
Private Sub Pic_PIZARRON_DblClick()  
If usare_write Then  
    Me.KeyPreview = False  
    frm_write_piz.Top = YY  
    frm_write_piz.Left = XX  
    frm_write_piz.Visible = True  
    .....  
    .....’Se omitió código que valida otras acciones  
End If  
End Sub
```

Esta subrutina posiciona un Frame (Figura 59) que contiene la caja de texto y los botones apropiados en la posición que se le indica al darle el doble click, usando las propiedades TOP y LEFT la posición inicial. Posteriormente hacemos visible este objeto.

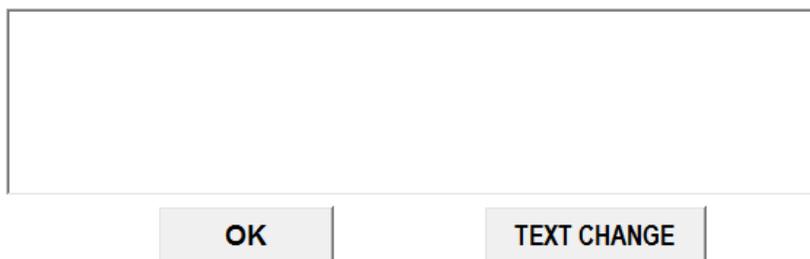


Figura 59. Frame para la acción "Escribir" del pizarrón interactivo.

Para colocar el texto escrito en la posición correcta cuando se confirme el mismo (con ENTER o el botón OK) se programó el siguiente método, que usa variables cuyos valores se establecieron previamente:

```
Private Sub cmd_ok_write_Click()  
    Pic_PIZARRON.Font = Tipo de letra  
    Pic_PIZARRON.FontBold = Negritas  
    Pic_PIZARRON.FontItalic = Itálicas  
    Pic_PIZARRON.FontUnderline = Subrayado  
    Pic_PIZARRON.ForeColor = Color de texto  
    Pic_PIZARRON.FontSize = Tamaño  
    Pic_PIZARRON.CurrentX = Posición X  
    Pic_PIZARRON.CurrentY = Posición Y  
    Pic_PIZARRON.Print Me.txt_write_pic.Text  
    frm_write_piz.Visible = False  
    usare_write = False  
    Me.txt_write_pic.Text = ""  
    EL_PIZARRON_INTERACTIVO.agrega_esto_pic_al_stack  
End Sub
```

Con el método Print del PictureBox se coloca el texto: en este caso es el contenido del cuadro de texto que está en el Frame, y se inserta en las mismas coordenadas en las que está el Frame. Posteriormente se limpia la caja de texto y se oculta el Frame. Una vez insertado el texto, el contenido del pizarrón se almacena en una pila que se usará para la navegación (atrás, adelante, al final, al principio).

5.5 Como se implementan los pizarrones en un U-Book

El U-Book permite incluir pizarrones en una sucesión de dos maneras:

- Como tal (pizarrón): se indica que el elemento de la sucesión es un pizarrón, que deberá estar especificado en el catálogo de pizarrones.
- Como un ítem tipo pizarrón: se crea un ítem del libro y se especifica en el campo “what_is_item” que se trata de un pizarrón (a diferencia de los otros valores “ítem”, “query”, “link”, que indica que se invocará una página de internet). Ahí se indica cuál pizarrón del catalogo se desea (por su número, puesto que en el catálogo están numerados) y se podrán modificar ciertos atributos para esta instancia en particular.

Un ítem pizarrón en U-Books

Se crea un ítem del libro y se indica que se trata de un pizarrón. El dato más importante es el número de pizarrón, que se refiere al catálogo de pizarrones.

Se pueden – optativamente - agregar atributos o especificaciones. Para ello, el **SPRP** usa una tabla adicional llamada “ItemBlackbSpecs”. En la tabla 3, se detallan los campos de esta tabla, con breves comentarios. Para los campos en los que no alcanzan dichos comentarios se explican más abajo. Es importante señalar que esta tabla complementa lo que se indicó ya en la tabla de los ítems (THEITEMS), en especial, el número de pizarrón que ocupará. También se señala que se pueden cambiar las dimensiones de la cuadrícula especificada para ese número de pizarrón en el catálogo, así como los colores y fuentes. Asimismo, los campos que terminan en 0 (cero) o 1 indican las alternativas proporcionadas para los pizarrones que no

ofrecen cambios dinámicos a los colores y fuentes, y solo permiten alternar los que se especifican con 0 y 1 respectivamente.

Tabla 3. Los campos de la tabla ItemBlackSpecs del libro.

CAMPO	TIPO	DESCRIPCIÓN
item_num	Número	
numero_sucesion_objetos	Número	Numero de la sucesión de imágenes prediseñadas que se mostrarán. Vale 0 (cero) si no la usa
como_muestra_obj_predis	Número	Especifica una regla (explicación sigue).
hilera_unica	Número	Hilera única, se indica ¿Cuál?
columna_unica	Número	Columna única, se indica ¿Cuál?
divisiones_horizontales	Número	El número de divisiones horizontales de la cuadrícula.
divisiones_verticales	Número	Número de divisiones verticales.
back_color0	Número	Color de fondo, opción 0.
text_color0	Número	Color de tinta para escribir, opción 0.
ink_color0	Número	Color de tinta para pintar opción 0.
text_font_type0	Texto	Nombre de tipografía, opción 0.
text_font_size0	Número	Tamaño de tipografía, opción 0.
puntos_linea0	Número	Tamaños de grosor de líneas, opción 0.
back_color1	Número	Color del pizarrón mismo fondo, opción 1.
text_color1	Número	Color de tinta para escribir, opción 1
ink_color1	Número	Color de tinta para pintar opción 1
text_font_type1	Texto	Nombre de tipografía, opción 1
text_font_size1	Número	Tamaño de tipografía, opción 1.
puntos_linea1	Número	Tamaños de grosor de líneas, opción 1.
usa_DUMMY_en_la_otra_compu	Sí/No	Usa DUMMY Si/No.

Detalles adicionales de algunos de los campos de la tabla

- **como_muestra_obj_predis:** Cómo se colocan los objetos prediseñados:
 - **“0” consecutivo puro:** inicia del cuadro (1,1) y ocupa sucesivamente los cuadros siguientes de la hilera, y cuando se llena esta pasa a la siguiente.
 - **“1” reemplazo:** Siempre se usa el mismo cuadro (la imagen reemplaza lo que tenía el cuadro).
 - **“2” semi-consecutivo:** cuando se indica una hilera o columna única.
- **hilera_unica:** si no vale “0”, las imágenes se colocarán sucesivamente en los cuadros de la hilera especificada (no se usan las restantes hileras de la cuadrícula).
- **columna_unica:** *ut supra*, pero se usa una sola columna.

Observaciones:

El U-Book no permite asociar un pizarrón (como tal) a un elemento de una sucesión; si se desea esta funcionalidad, se deberá usar un ítem-pizarrón como asociado.

No se pueden incluir objetos prediseñados en un pizarrón que se usa como elemento. En cambio, como se desprende de la tabla correspondiente, a un ítem-pizarrón se le puede asociar una sucesión de ítems.

5.6 Insertar Objetos prediseñados

El pizarrón interactivo cuenta con herramientas que permiten insertarle objetos prediseñados (Figura 60). Cuando en una sucesión se incluye un elemento tipo “ítem-pizarrón” que tiene una sucesión de objetos prediseñados, las imágenes indicadas por dicha sucesión se incluirán en la imagen del pizarrón por invocación (“NEXT OBJECT”).

Para la inserción de una imagen se calcula la posición en la cual se insertará en la imagen del pizarrón y el tamaño de la nueva imagen. Esta operación tiene dos pasos. Las explicaciones de la cuadrícula se incluyeron en la siguiente sección.

Primero se determina el cuadro de la cuadrícula que se ocupará, de acuerdo a las reglas especificadas como parte de la definición de las especificaciones del ítem como pizarrón descritas anteriormente. A continuación se usan las dimensiones de la cuadrícula para calcular el tamaño de la nueva imagen y su posición relativa al PictureBox del pizarrón. Finalmente, se inserta la imagen en el lugar definido, y como siempre, se agrega la nueva imagen del pizarrón a la multicitada pila imágenes.

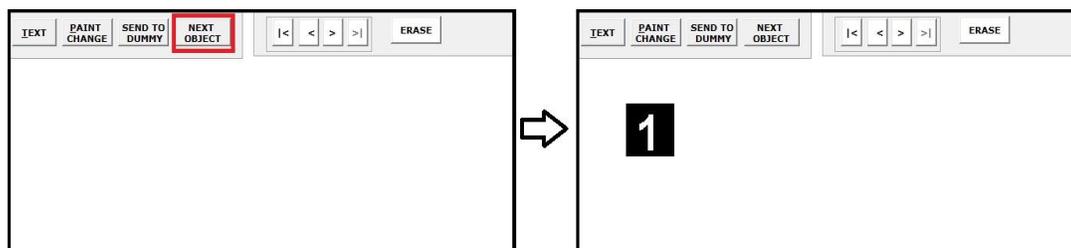


Figura 60. Insertar un objeto prediseñado.

5.6.1 Cuadrícula para imágenes prediseñadas

Para determinar la posición en la que se muestra una imagen prediseñada, se define una cuadrícula (grid) para el pizarrón. Como parte del pizarrón que se usa, se incluyen reglas de cómo se usarán los cuadros de esta cuadrícula, que se define especificando el número de subdivisiones horizontales y verticales de la imagen del pizarrón. Estos datos se usan como pares. Ilustramos este concepto con el ejemplo ilustrado por la Figura 61.

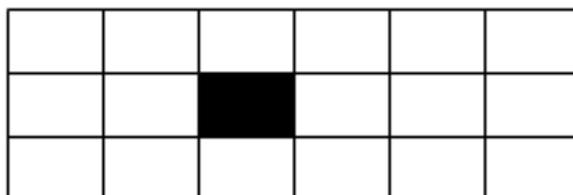


Figura 61. Una imagen colocada en el cuadro (2,3) de una cuadrícula (3,6).

Como ya se adelantó cuando se describieron los campos de la tabla ItemBlackbSpecs, la regla para determinar cuál de los cuadros ocupará la imagen que se inserta:

- **Consecutivo:** Inserta las imágenes prediseñadas (Figura 62), desde las posición (1,1) a las posición (3, 6), de forma consecutiva. Cuando llega al último cuadro, ocupara nuevamente la posición (1,1), tras borrar todo el resto del pizarrón.



Figura 62. Inserción de una imagen prediseñada en forma consecutiva.

- **Reemplazo (posición única):** Se insertan todas las imágenes en la misma posición. Para esta regla, hay que usar los campos hilera y columna única. Se

observa que la alternativa de indicar una cuadrícula de 1 solo cuadro no es equivalente, puesto que de hacerlo, la imagen ocupará todo el pizarrón.

- **Columna única:** Se indica la Columna en donde se insertarán las imágenes prediseñadas. Por ejemplo si se tiene un pizarrón con una cuadrícula (3,6) y se tienen tres imágenes prediseñadas y se especifica como columna única la primera, insertara los objetos sucesivamente en las posiciones (1,1), (2,1) y (3,1) (Figura 63). En caso que hubiera más imágenes, se reutiliza la (1,1) y se borran las otras dos.

(1,1)					
(2,1)					
(3,1)					

Figura 63. Inserción de 3 imágenes prediseñada en una columna única 1.

- **Hilera única:** se procede del mismo modo que en el caso de la columna única, excepto que ahora se usan los cuadros de 1 hilera.

El tamaño de las imágenes prediseñadas, se ajustan a las dimensiones de un cuadro de la cuadrícula (todos los cuadros son del mismo tamaño). El constructor de la clase “cl_UN_PIZARRON_INTERACTIVO” invoca un método llamado “PON_DIV” que se detalla a continuación.

```

Private Sub PON_DIV(Pic As PictureBox, divisionesX, divisionesY)
Dim WIDTH As Double
Dim HEIGHT As Double
Dim X As Integer
Dim Y As Integer
Dim INCREMENTO_X As Integer
Dim INCREMENTO_Y As Integer
If divisionesY = 0 Or divisionesX = 0 Then Exit Sub
If divisionesY > 0 Then WIDTH = Pic.ScaleWidth / divisionesY
If divisionesX > 0 Then HEIGHT = Pic.ScaleHeight / divisionesX
INCREMENTO_X = 0
INCREMENTO_Y = 0

```

```

ReDim CUADRICULA(1 To divisionesX, 1 To divisionesY)
For X = 1 To divisionesX
    For Y = 1 To divisionesY
        CUADRICULA(X, Y).POS_X = INCREMENTO_X
        CUADRICULA(X, Y).POS_Y = INCREMENTO_Y
        CUADRICULA(X, Y).TAM_W = WIDTH
        CUADRICULA(X, Y).TAM_H = HEIGHT
        INCREMENTO_X = INCREMENTO_X + WIDTH + 0.01
    Next Y
    INCREMENTO_X = 0
    INCREMENTO_Y = INCREMENTO_Y + HEIGHT + 0.01
Next X
End Sub

```

Recibe como parámetro el PictureBox así como el número de divisiones horizontales y verticales. Al principio de la subrutina se define:

- Variables que utilizará posteriormente;
- Verifica que los valores de divisiones horizontales sean positivos;
- Calcula la altura y el ancho de cada una de las divisiones;
- Determina las dimensiones del arreglo que contendrá todos los valores;
- Llena para cada cuadro el alto, ancho, posición X y posición Y. Almacena los valores en un arreglo de tipo TYPE llamado CUADRICULA.

5.6.2 Acciones que realiza el SPRP para insertar una imagen prediseñada

Las acciones que realiza el programa **SPRPShow-Other** difieren de las correspondientes en el programa **SPRPBuildShow**, puesto que no solicitan las imágenes a insertar del mismo modo. Esto se debe a que en la otra computadora no hay “datos” ni archivos. El programa le pide al otro (el SPRPBuildShow) mediante un comando *ad hoc* que le proporcione la imagen que debe insertar.

Cuando el presentador solicita que se agregue la siguiente imagen (prediseñada) al pizarrón, el **SPRPBuildShow** invoca una rutina llamada

“necesito_dibujar_siguiete_objeto_predisenado” contenida en la clase de pizarrones interactivos.

```
Private Sub necesito_dibujar_siguiete_objeto_predisenado()  
Dim arch_obtenido As String  
ultimo_obj_predis_mostrado = ultimo_obj_predis_mostrado + 1  
If ultimo_obj_predis_mostrado > sus_atributos.CUANTOS_OBJETOS_PREDEFINIDOS Then  
    MsgBox "YA NO HAY OBJETOS"  
    Exit Sub  
End If  
arch_obtenido = SUC_OBJETOS_PREDIS.dame_sigite_objeto_predisenado  
If Trim(arch_obtenido) = "YA NO HAY OBJETOS PREDIS" Then Exit Sub  
Call inserta_el_objeto_en_el_pizarron(arch_obtenido)  
End Sub
```

Se realizan los siguientes pasos:

- Determina si hay más imágenes prediseñadas; en caso que no existan más objetos prediseñadas, notifica que “Ya no hay objetos”. Si hay imágenes pendientes (es decir, no se ha acabado la sucesión) invoca una función llamada “SUC_OBJETOS_PREDIS.dame_sigite_objeto_predisenado”, que regresa el nombre del archivo que contiene la imagen prediseñada (*Explicación técnica, ver Anexo 25*);
- Una vez que se obtiene la ruta del archivo que contiene la imagen solicitada, se invoca la rutina “inserta_el_objeto_en_el_pizarron(arch_obtenido)” a la cual se le proporciona la ruta obtenida. Esta rutina a su vez invoca el método “determina_donde_pone_objeto_predisenado”, que calcula el tamaño y la posición de la nueva imagen en base a las reglas mencionadas anteriormente (*Explicación técnica, ver Anexo 26*);
- Tras obtener la posición y el tamaño, se carga a un objeto temporal Image la imagen que tiene como parámetro el path de la imagen prediseñada y con el método PaintPicture de PictureBox, se carga el contenido del Image al picture. Naturalmente se le proporcionan el origen de la imagen, la posición “X” y “Y”, el ancho y la altura del cuadro, datos que como vimos se calcularon

previamente. Finalmente el contenido del pizarrón se almacena en la pila de imágenes.

```
Private Sub inserta_el_objeto_en_el_pizarron(file As String)
    If borrar_lo_que_tenia Then limpia_el_pizarron
    determina_donde_pone_objeto_predisenado
    With FF
        .Image1.Picture = Nothing
        .Image1.Picture = LoadPicture(file)
        .Image1.Top = 0
        .Image1.Left = 0
        .Image1.BorderStyle = 0
        .Image1.Visible = True
    End With
    With CUADRICULA(hil_calculada, col_calculada)
        FF.Pic_PIZARRON.PaintPicture FF.Image1.Picture, .POS_X, .POS_Y, .TAM_W,
        .TAM_H
    End With
    Call agrega_este_pic_al_stack
End Sub
```

En SPRP la lectura de una sucesión puede ocupar una computadora adicional a la que usa el presentador, conectada en red con esta. Este concepto se describe a detalle en la sección de múltiples computadoras del Capítulo VII, Adelantamos que la comunicación será de tipo “program-to-program”, de modo que hay un programa llamado SPRPShow-Other en ejecución en la otra computadora, y la comunicación entre los programas se realiza vía mensajes enviados a sendos sockets y los archivos se transmiten colocándolos en una carpeta compartida o pública.

Si una imagen es solicitada por un pizarrón que se muestra en la “otra computadora”, el programa SPRPShow-other solicita la imagen a su contraparte enviando un mensaje en el cual le notifica precisamente eso: que requiere la siguiente imagen prediseñada. La consecuencia es un mensaje de regreso (de la computadora principal a la “otra”) en el que se notifica que revise la carpeta de archivos compartidos (o que ya no hay imágenes para mostrar). Se le proporciona la imagen a

la otra computadora mediante un archivo que siempre tendrá la misma ruta y nombre; es decir, la computadora principal copia el archivo con la imagen solicitada al archivo de ese nombre en la carpeta compartida.

De ahí en adelante las acciones son las mismas que las descritas para el pizarrón desplegado en la computadora del presentador.

5.7 Mandar Imagen a un pizarrón DUMMY

El pizarrón interactivo cuenta con herramientas que le permiten mandar una copia exacta a su pizarrón DUMMY, asociado cuando el tipo de pizarrón interactivo usado indica que lo necesita (de hecho, se puede desactivar esta opción como parte de las especificaciones del ítem-pizarrón).

El pizarrón interactivo y el DUMMY, puede ser mostrado en cualquier monitor de una computadora o de otra computadora: una vez más, esto se detallará en la sección correspondiente del Capítulo VII.

Se invoca el envío de una imagen a un pizarrón DUMMY con el botón “SEND TO DUMMY”. Esto ocasiona que se guarde el contenido del pizarrón interactivo como archivo (temporal) de imagen (*Explicación técnica, ver Anexo 23*) que luego se cargará al pizarrón DUMMY.

Para saber dónde se grabará este archivo temporal, se determina dónde se encuentra el DUMMY: si está en la misma computadora o en la otra, donde esto funciona en ambos sentidos. Según estos casos, se ocupa un archivo

- De la computadora del pizarrón interactivo (en el caso que el pizarrón DUMMY también se muestra ahí);
- en la carpeta compartida cuando el DUMMY no está en la misma computadora que el pizarrón interactivo.

Observación técnica: la necesidad de pasar por un archivo en lugar de copiar el contenido del PictureBox al otro surge de que esta operación no funciona. A pesar de que se puede copiar un picture de uno a otro PictureBox, al escribir y modificar de otros modos el PictureBox se cambia algunos atributos que hacen

que esta operación falle. Esto se comprobó con numerosos intentos que reflejaron alternativas para lograr el resultado.

Cuando el DUMMY se encuentra en la otra computadora, se notifica al programa que tiene instanciado el DUMMY que tiene que revisar la capeta compartida, ya que ahí está un nuevo archivo. Estas acciones las ejecuta la rutina “manda_picture_al_DUMMY”, como se muestra con el código siguiente.

```
Public Sub manda_picture_al_DUMMY()  
    If Not usa_DUMMY Then Exit Sub  
        crea_un_archivo_con_el_picture  
    If DUMMY_esta_en_la_otra_compu Then  
        Call dile_a_la_otra_que_muestre_esto_en_DUMMY  
    Else  
        Call  
un_pizarron_DUMMY_mio.carga_un_picture_desde_archivo(nombre_archivo_picture_del_pizarron)  
    End If  
End Sub
```

Después de haber creado el archivo, como ya se tiene la ruta del archivo, independientemente donde se encuentre, se carga el archivo al DUMMY, con la siguiente subrutina:

```
Public Sub carga_un_picture_desde_archivo(nombre_del_archivo_del_pic)  
    FF.Pic_PIZARRON.Picture = LoadPicture(nombre_del_archivo_del_pic)  
End Sub
```

Se usa la propiedad LoadPicture del PictureBox e indicando la ruta del archivo, todo el proceso se indica en la Figura 64.

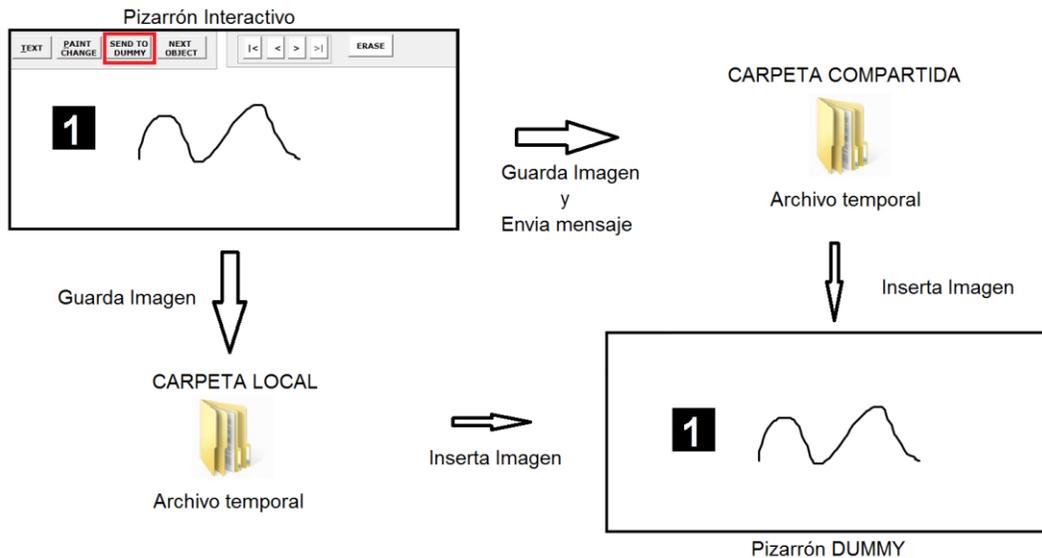


Figura 64. Mandar imagen a un pizarrón DUMMY.

5.8 Impacto de los pizarrones en los aspectos relacionados con el aprendizaje

El uso de la interactividad ha probado ser de utilidad en muchos productos de software. Por ejemplo, en las versiones nuevas de PowerPoint se puede pintar encima de una diapositiva y mostrar la imagen alterada de este modo a la audiencia. Para ello, se usa el monitor del presentador para ejecutar el programa PowerPoint en modo fuente, y se envía la presentación a un monitor que se declara como “extendido” (y no “clonado”).

El SPRP permite hacer algo parecido, pero lo efectúa en forma distinta, como se describió anteriormente. Aquí se muestra una imagen independiente (el pizarrón) que puede reemplazar o completar otro objeto. Esto permite “comentar” en forma visual un video o un audio, o cualquier otro tipo de objeto que se muestre. Indudablemente constituye un auxilio importante a un instructor que pretende aclarar o ampliar el contenido de lo que está presentando. La contribución de la posibilidad de usar estos dispositivos con el SPRP es la facilidad de preparación del material, que no implica conocimientos técnicos avanzados por parte del usuario.

El impacto sobre el efecto de la comunicación - el mensaje - sobre el recipiente (que en la literatura se llama cliente del proceso que lo informa) del uso de este tipo de

artificios puede ser muy significativo. Naturalmente intervienen componentes adicionales, no sólo el medio por el cual se envía el mensaje. El contenido, la forma y la adecuación de ambos al medio usado son fundamentales: cualquier inconsistencia no solo puede anular algún efecto positivo, sino convertirlo en un factor negativo. Exhibir simultáneamente objetos relacionados ayuda al cliente a integrar los conocimientos presentados, no solo entre ellos, sino con los que ya tiene. El uso de un pizarrón no tiene que ser comentado, pero la simultaneidad con otro tipo de objetos que ve u oye el cliente ofrece muchos nuevos elementos para ayudarle a relacionar conceptos.

Finalmente, como ya se había comentado, los dispositivos adicionales pueden ser un arma valiosa para atraer y mantener la atención de la audiencia. El presentador o el que prepare el material (o ambos) deberán saber aprovechar los nuevos recursos en cada ocasión.

CAPITULO VI. AUDIO

En un U-Book se busca la siguiente funcionalidad: asociar bocinas a las formas en las que se muestran los objetos, de modo que un audio (o video) que se muestra en una forma en particular solo se oirá en las bocinas que se asocien a dicha forma.

Los materiales auditivos se han considerado desde hace tiempo como un importante recurso educativo ya sea para la enseñanza o simplemente como recurso de presentaciones, ya que la mayor parte de la información que reciben las personas se realiza a través del sentido de la vista y del oído. La principal razón para la utilización de mezclar imagen y audio en los procesos educativos es que resultan motivadoras, sensibilizan y estimulan el interés de los estudiantes o espectadores de una presentación hacia un tema determinado, de modo que facilitan la instrucción completando las explicaciones auditivas. Además permiten al presentador manejar la atención de su audiencia, que como se indicó anteriormente, consideramos un aspecto didáctico preponderante.

En el amplio conjunto de los recursos educativos conviene distinguir entre recursos clásicos (explicaciones orales con ayuda de la pizarra, libros de texto, apuntes...) y recursos tecnológicos, que requieren el manejo de un instrumento para poder transmitir los mensajes de la enseñanza: el U-BOOK integra todos estos elementos. Dentro de estos recursos tecnológicos se engloban los medios audiovisuales y las nuevas Tecnologías de la Información y la Comunicación (TIC).

6.1 Tarjetas de audio

Una tarjeta de sonido o placa de sonido es una tarjeta de expansión para computadoras que permite la salida de audio bajo el control de un programa informático llamado controlador (en inglés driver). El típico uso de las tarjetas de sonido consiste en usar un programa que actúa de mezclador para que las aplicaciones multimedia del componente de audio suenen y puedan ser gestionadas. Estas aplicaciones multimedia engloban composición y edición de video o audio,

presentaciones multimedia y entretenimiento (videojuegos). Algunos equipos (como los personales) tienen la tarjeta ya integrada (Figura 65): tienen 1 o 2 salidas de audio y una entrada de audio (un micrófono u otro tipo de dispositivo que provea el sonido a procesar).



Figura 65. Puerto de audio de una computadora personal.

Mientras que otros requieren tarjetas de expansión (Figura 66), que requieren una instalación, la mayoría tienen un conector PCI (Peripheral Component Interconnect, Interconexión de Componentes Periféricos). (*Explicación técnica, ver Anexo 11*).



Figura 66. Tarjeta de audio 2.1 canales.

El mezclador de audio es una parte de la tarjeta de sonido que se encarga de mezclar los sonidos que llegan a la tarjeta procedente de diferentes fuentes. La forma de mezclar los sonidos (Figura 67) se puede controlar mediante programas, que pueden ser de la propia tarjeta de sonido o del sistema operativo. En los sistemas operativos de Microsoft Windows hay un programa llamado control de volumen o control de grabación (según el modo en el que esté) que realiza esa función. En cada modo hay un conjunto de controles que permiten controlar el volumen y balance izquierda-derecha (para los sonidos estéreo) de cada fuente de sonido. También pueden existir otros controles para el flujo de sonido procedente de otros dispositivos (como tarjetas sintonizadoras de TV, etc.)

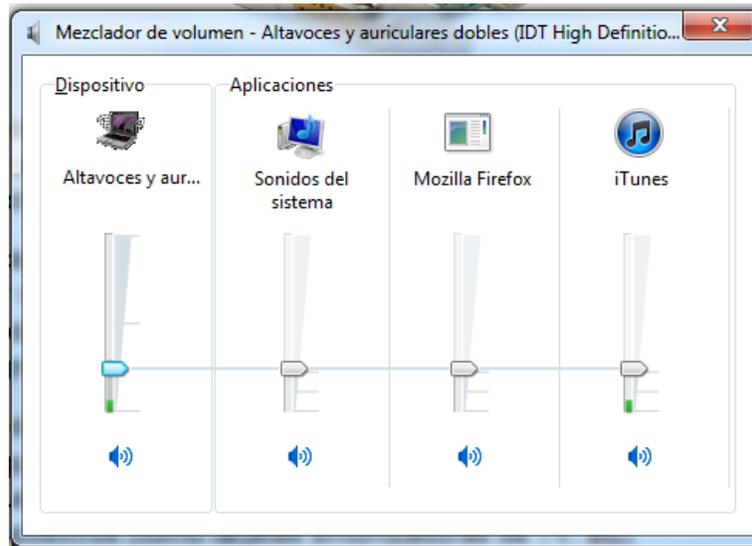


Figura 67. Mezclador de Volumen.

En la figura 68, se muestran los principales controles que intervienen en la mezcla del sonido para obtener el sonido que sale por los altavoces y el que se graba (con la grabadora de sonidos, por ejemplo).

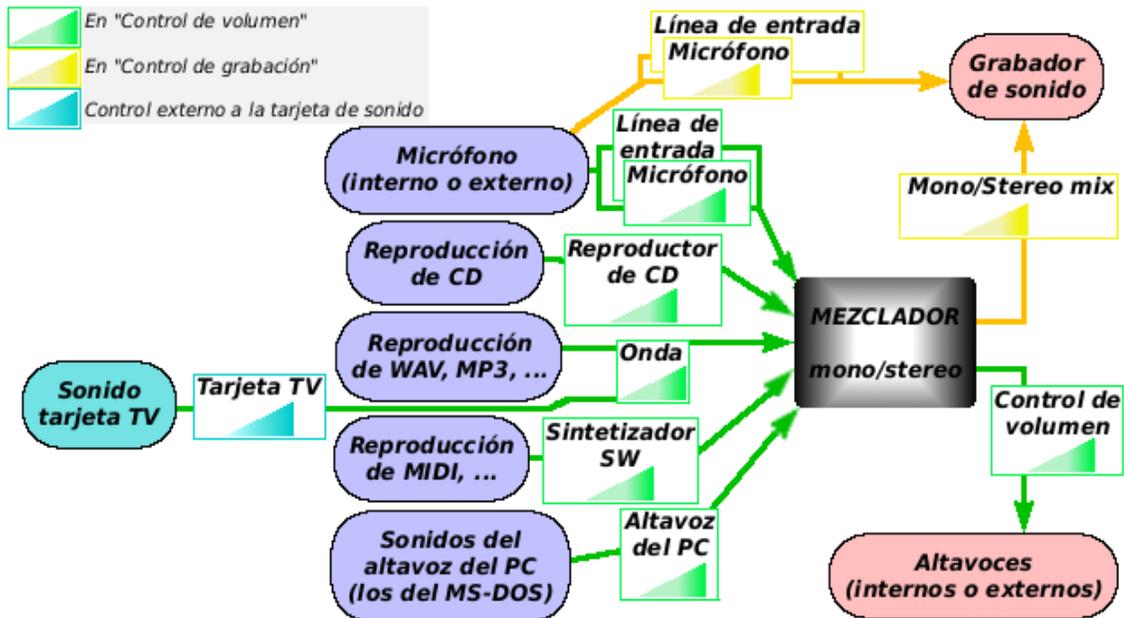


Figura 68. Funciones de un Mezclador.

En la Tabla 4, se muestran los elementos que intervienen en la mezcladora.

Tabla 4. Factores que intervienen en una Mezcladora.

CONTROL	CANALES	SONIDO CONTROLADO
Onda	tereo	Controla el sonido generado por la computadora al reproducir WAV, MP3, etc. También el sonido generado al reproducir CD-Audio con algunos programas (como Windows Media Player o Media Player Classic) o al reproducir archivos MIDI con otros programa.
Sintetizador SW	tereo	Sonido producido al reproducir archivos MIDI con la mayor parte de los programas.
Reproductor de CD	tereo	Sonido producido al reproducir CD-Audio con la mayor parte de los programas.
Micrófono	mono	Sonido que entra desde el micrófono (externo o interno).
Línea de entrada	tereo	También controla el sonido que procede del micrófono.
Aux	tereo	Controla el sonido desde una toma de entrada externa (ó auxiliar) a la PC.
SPDIF	mono	Interfaz digital de algunos dispositivos. Poco habitual.
Altavoz del PC	mono	Sonido procedente del altavoz de la PC. Este sonido es el que se oye en los programas de MS-DOS, (normalmente pitidos).
Control de volumen	tereo	Sonido mezclado con destino a los altavoces.

6.1.1 Tarjetas de audio multicanal

Se denomina sistema de sonido multicanal tanto al diseño de la pista de audio que tiene tres o más canales (típicamente canal derecho e izquierdo y otro para los sub-graves, más otros canales especializados), como también al equipo en sí (con tres o más altavoces) capaz de reproducir dicha pista. (Igoe, 1998)

Sistema 2.1: Consta de 3 altavoces: canal izquierdo y derecho, y otro para sub-graves (subwoofer).

Sistema 4.1: Consta de 5 altavoces: canales frontales izquierdo y derecho, dos traseros, izquierdo y derecho, y otro para sub-graves (subwoofer).

Sistema 5.1 (Figura 69): Consta de 6 altavoces, como los que usan los sistemas de sonido surround (*Explicación técnica, ver Anexo 12*), La notación 5.1 hace referencia a la forma en que es distribuido el sonido. En este caso, 5 altavoces que tratan de forma independiente un rango determinado de frecuencias. Cuando se trata de 5 altavoces se distribuyen del siguiente modo: central (emite sonidos medios o de voz), delantero izquierdo y derecho (emite sonidos de todo tipo, a excepción de los bajos), trasero izquierdo y derecho (emiten sonidos de ambientación). Por último ".1" hace referencia al canal de subwoofer (emite todos los sonidos con frecuencias hasta aproximadamente los 100 Hz).

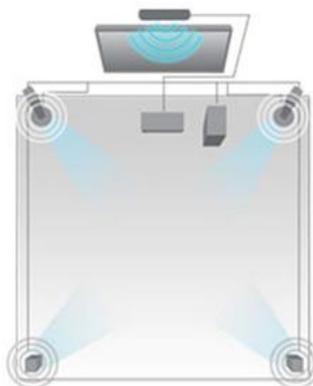


Figura 69. Distribución de audio de 5.1 canales.

Sistema 6.1: Se añade un altavoz central en la parte posterior con respecto a 5.1.

Sistema 7.1: Coloca dos altavoces más en la parte lateral con respecto a 5.1.

Sistema 7.2: Se añade un subwoofer en la parte posterior con respecto a 7.1

6.2 Bocinas (múltiples salidas)

Hay diversos modos para implementar múltiples bocinas en una sola computadora y cada bocina con audios diferentes:

- Usar una tarjeta de múltiples canales, en específico una de 5.1. El inconveniente es que todo el material multimedia - audio y video - tiene que ser procesado de tal manera que el audio esté separado por canales. Este proceso se tiene que hacer cada vez que se utilice material de audio, lo cual

es oneroso en tiempo y recursos. En caso de no contar con una tarjeta de 5.1 canales, no tendría sentido procesar de este modo el audio.

Si tenemos dos audios diferentes que queremos escuchar en bocinas diferentes, se tienen que mezclar y procesar en canales diferentes. El U-Book no permite visualizar dos videos o audios al mismo tiempo, puesto que se pierde el objetivo del U-Book en cuanto a obtener la atención de la audiencia. Simplemente causaría confusión en los oyentes.

- Usar una mezcladora de audio profesional, lo que aumentaría la complejidad ya que no solo manipularíamos el U-Book, sino también se tendría que operar la consola y sincronizar todo el audio.
- La alternativa seleccionada para el SPRP es que se creó un dispositivo electrónico capaz de alternar (switch) bocinas cada vez que el U-Book lo requiera. Esto produce un efecto similar al de una consola pero operada directamente por el programa (SPRP), el cual indicaría en qué bocinas se oíría un audio en particular,

6.2.1 Interface de múltiples bocinas

Se partió de la necesidad de poder manipular el audio de varias bocinas. Puesto que el U-Book no permite múltiples audios simultáneos, se desarrolló hardware y software simplemente para poder apagar y prender bocinas. Con esto se puede dirigir un audio en particular a la (o las) bocina(s) deseadas.

Todo el desarrollo se hizo a partir del número de bocina y solo se permiten tres juegos de bocinas, por la cantidad máxima de monitores que se pueden conectar a la computadora. Cada par de bocinas se asocia a un monitor o una forma de un monitor (recordemos que se pueden “simular” dos monitores mostrando 2 formas en un mismo dispositivo), éste proceso se muestra en la Figura 70. El controlador está compuesto por 2 partes:

- Controlador (interruptor);
- Aplicación en el SPRP que determinará que acción debe realizar el controlador.

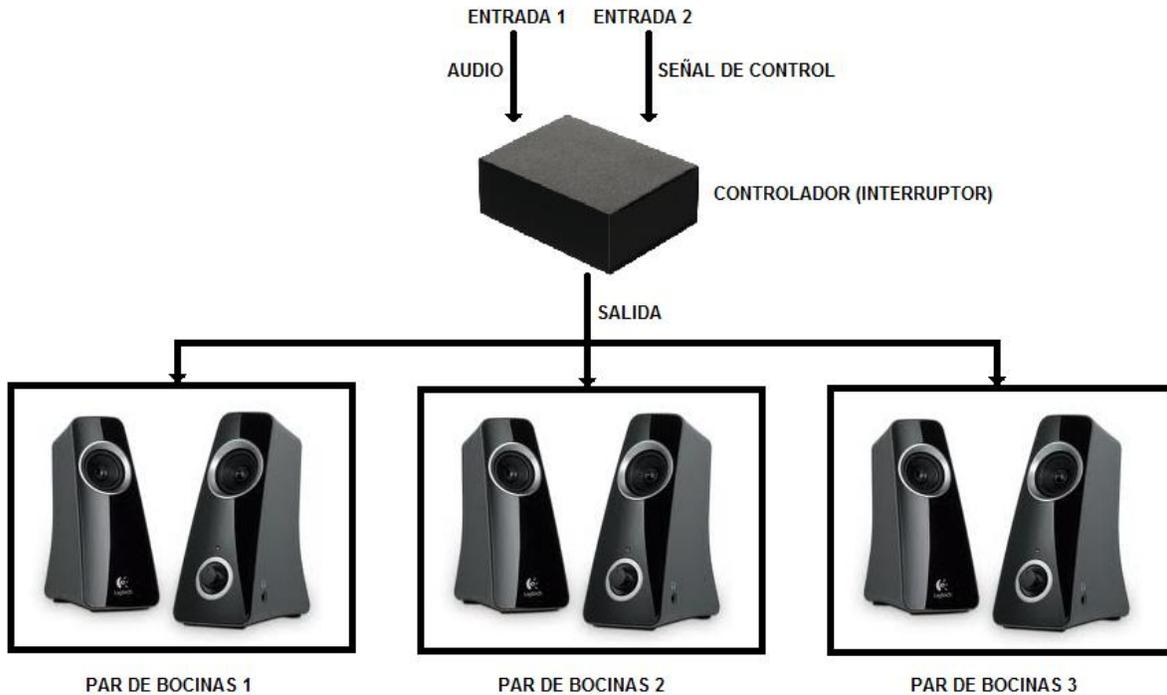


Figura 70. Proceso del controlador para múltiples bocinas.

Una de las entradas es un mini-plug macho de audio (2.1 canales, canal izquierdo y derecho; no se contempla el uso de un subwoofer). La segunda entrada es la de control: determina cuál par de bocinas se prenderá o apagará; esta entrada se conecta a la computadora por medio de USB.

6.2.1.1 Controlador de múltiples bocinas

Se desarrolló un controlador usando como apoyo una plataforma ARDUINO (Figura 71) – porque es hardware libre - basada en una placa con un microcontrolador y un entorno de desarrollo. ARDUINO está diseñada para facilitar el uso de la electrónica.

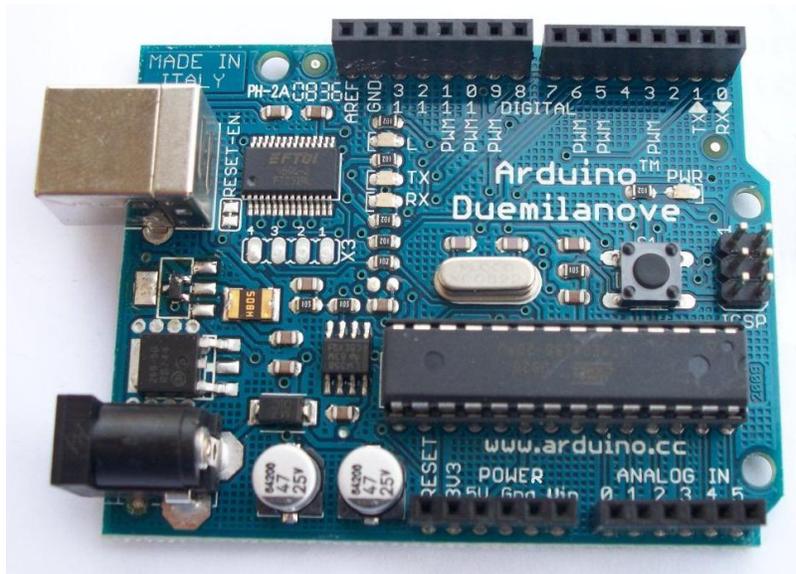


Figura 71. Controlador ARDUINO.

El hardware consiste en una placa con un microcontrolador Atmel-168 y puertos de entrada/salida. Por otro lado el software consiste en un entorno de desarrollo que implementa el lenguaje de programación Processing/Wiring y el cargador de arranque (boot loader) que corre en la placa ARDUINO. Otras características se muestran en la Tabla 5.

Tabla 5. Características de ATMEGA168.

Voltaje operativo	5 V
Voltaje de entrada recomendado	7-12 V
Voltaje de entrada límite	6-20 V
Pines de entrada y salida digital	14 (6 proporcionan PWM - Pulse Width Modulation)
Pines de entrada analógica	6
Intensidad de corriente	40 mA.
Memoria Flash	16KB (2KB reservados para el bootloader)
SRAM	1 KB
EEPROM	512 bytes
Frecuencia de reloj	16 MHz

ARDUINO se puede utilizar para desarrollar objetos interactivos autónomos o se puede conectar directamente a aplicaciones en la computadora como son Macromedia Flash, Processing, Max/MSP, Pure Data, Visual Basic, .NET, etc.

Para complementar el ARDUINO se desarrolló un circuito, cuya función es prender o apagar relevadores. Un relevador (Figura 72) es un dispositivo electromecánico que funciona como un interruptor controlado por un circuito eléctrico en el que, por medio de una bobina y un electroimán, se acciona un juego de uno o varios contactos que permiten abrir o cerrar otros circuitos eléctricos independientes. Un relevador también es capaz de controlar un circuito de salida de mayor potencia que el de entrada: en un sentido amplio se puede considerar como un amplificador eléctrico.

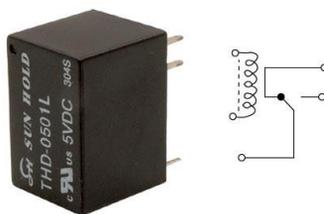


Figura 72. Un relevador y su circuito.

Se diseñó un circuito con tres relevadores (Figura 73); cada uno se usó como interruptor y no como amplificador. Se usó el modelo THD-0501L que es un relevador de 5 Volts porque el ARDUINO tiene un voltaje de alimentación de 5 volts. Además, tiene 1 polo y 2 tiros, lo que significa que tiene una línea con dos interruptores, abierto o cerrado (apagado o prendido). A cada relevador se le asignó un indicador visible – un diodo-LED, de color rojo - para indicar si está apagado o prendido el relevador.

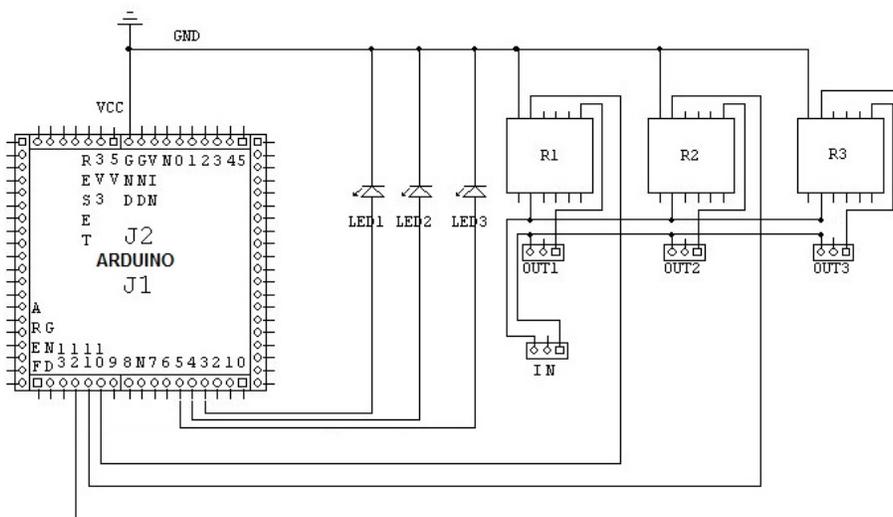


Figura 73. Diagrama del circuito electrónico.

Una vez simulado el circuito se diseñó la placa fenólica (PCB, *printed circuit board*, *tarjeta de circuito impreso*). Se usó un software llamado **KIDCAD** para diseñar circuitos impresos: se elaboro el circuito a dos capas y posteriormente se generan archivos GERBER-FILES. Esos archivos se mandaron a una empresa que se dedica a imprimir circuitos, el circuito impreso consta de dos caras, la primera es la parte superior (Figura 74) y la segunda es la parte posterior (Figura 75).

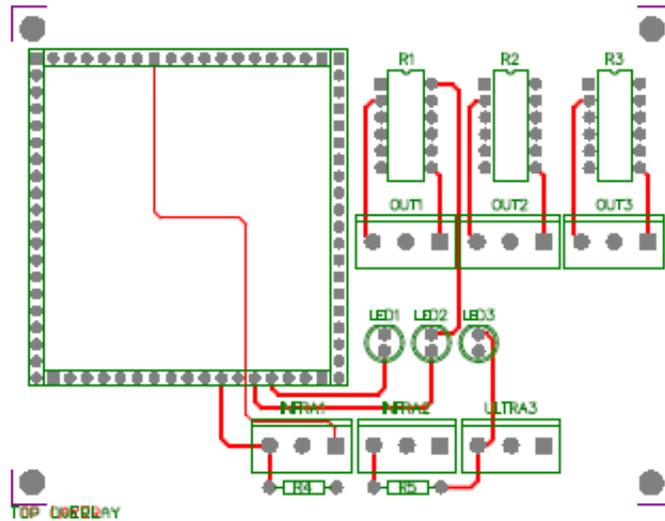


Figura 74. Circuito cara superior.

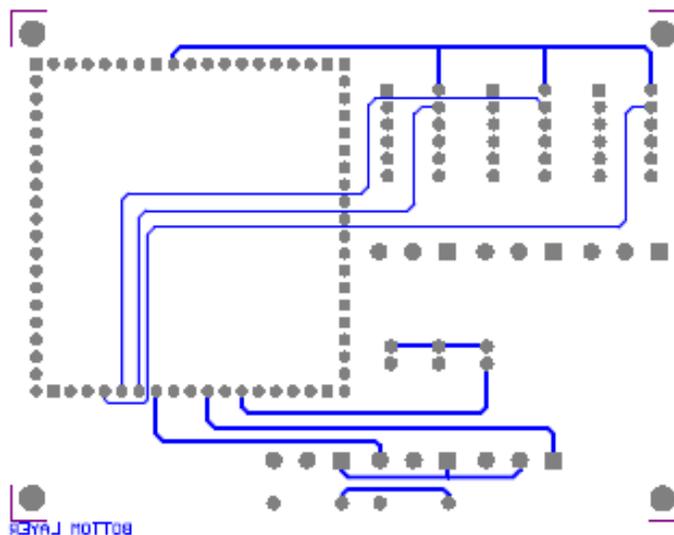


Figura 75. Circuito cara inferior.

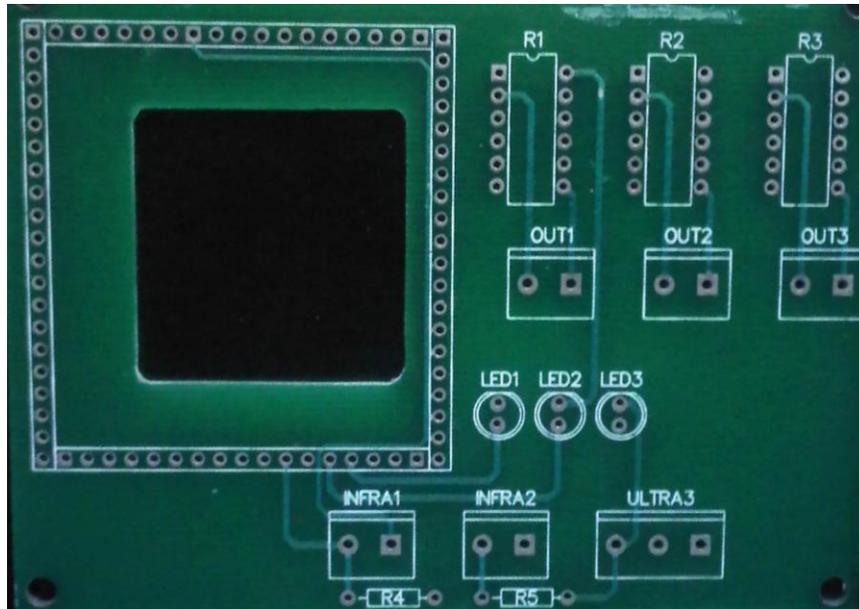


Figura 76. Circuito impreso.

Una vez impreso el circuito (Figura 76) se coloca todos los componentes que se muestran en la Tabla 6 y se coloca el ARDUINO. El ARDUINO es montado boca abajo sobre el circuito, ilustrado en la Figura 77.

Tabla 6. Lista de componentes para circuito electrónico.

COMPONENTES	CANTIDAD
ARDUINO	1
Relevadores	3
Headers 	3
Led's 	3
Out- Bases	3

	
<p>Plug Macho</p> 	<p>1</p>
<p>Plugs Hembras</p> 	<p>3</p>

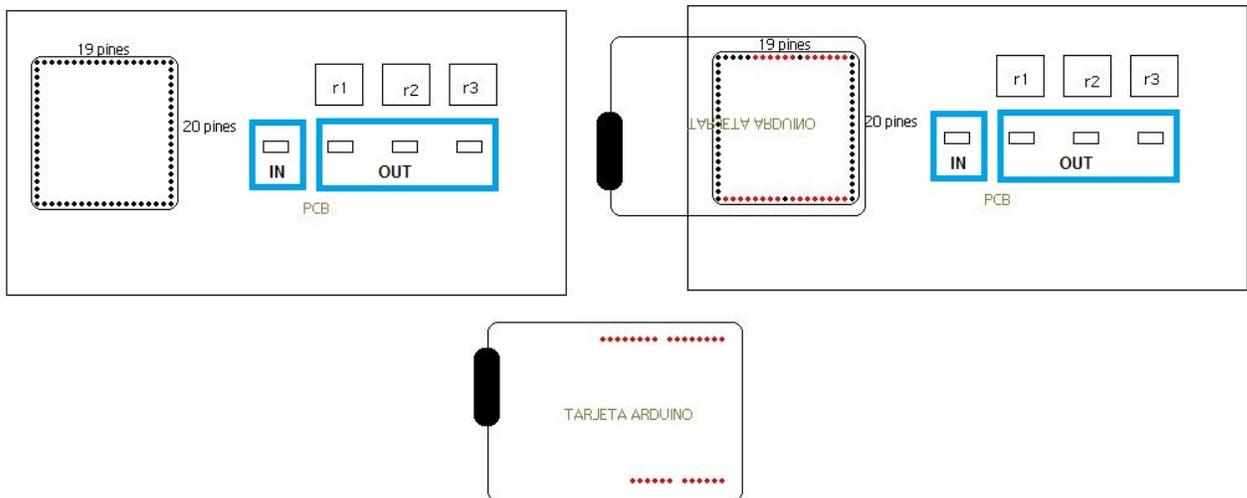


Figura 77. Montaje del controlador ARDUINO.

La Figura 78 muestra el circuito final.

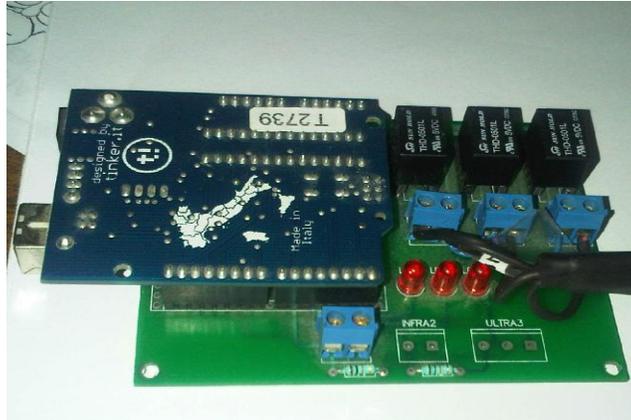


Figura 78. Controlador final.

6.2.1.2 Interfaz en ARDUINO

El entorno de desarrollo es libre y se puede descargar gratuitamente de la página de ARDUINO (Figura 79).

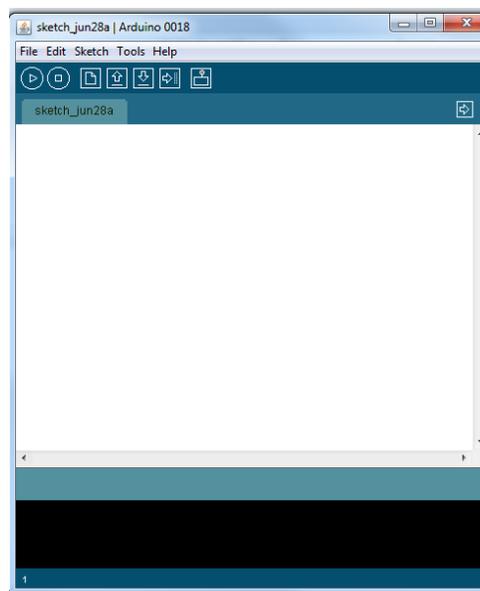


Figura 79. Interfaz de ARDUINO.

Se elaboró un programa que se guardó en el circuito integrado que tiene la tarjeta ARDUINO. Los pasos para la elaboración de este programa son los siguientes:

- Se declaró una variable de tipo entero llamada userInput inicializada con 0 (valor que apaga todas las bocinas), y es la variable que recibe mensajes desde el programa de SPRP.

```
int userInput=0;
```

- Se declararon 6 variables de tipo entero que funcionan como salidas: la salida 10 (Par de bocina 1), 11 (Par de bocina 2), y 12 (Par de bocina 3) que controlan los RELEVADORES, y las salidas 3,4, 5 que son los indicadores de apagado y encendido, respectivamente:

```
int RELE1=10;
```

```
int LED1=3;
```

```
int RELE2=11;
```

```
int LED2=4;
```

```
int RELE3=12;
```

```
int LED3=5;
```

Se declara la variable y el número de pin que usará.

- Todas las variables declaradas se especifican como de salida y como hay una comunicación con el **SPRP**, se abre el puerto serial con un valor de 9600 baudios. El programa se deja en un ciclo, y se mantiene en escucha para recibir acciones.

```
Serial.begin(9600);
```

```
pinMode(RELE1,OUTPUT);
```

```
pinMode(LED1,OUTPUT);
```

```
pinMode(RELE2,OUTPUT);
```

```
pinMode(LED2,OUTPUT);
```

```
pinMode(RELE3,OUTPUT);
```

```
pinMode(LED3,OUTPUT);
```

- Se programaron varios “if anidados”; como el programa está en un ciclo, se analizan los valores que llegan a la entrada serial y se ejecuta la acción apropiada, mostrada en la Tabla 7.

Tabla 7. Acciones para el controlador ARDUINO.

VALOR DE MENSAJE	ACCIÓN
0	Apaga todas las bocinas
1	Prende par de bocinas 1
2	Prende par de bocinas 2
3	Prende par de bocinas 3
4	Apaga par de bocinas 1
5	Apaga par de bocinas 2
6	Apaga par de bocinas 3

Para poder apagar y prender los relevadores, se utiliza la palabra reservada `digitalWrite` seguida de la variable y de su valor (HIGH, LOW), que indica apagado o encendido.

`digitalWrite(RELE3,HIGH);` → Encendido

`digitalWrite(RELE3,LOW);` → Apagado

- Hay que compilar el proyecto y exportarlo a la memoria del ARDUINO. (*Explicación técnica, ver Anexo 13*).

6.2.1.3 Interfaz incluida en los programas de SPRP

Visual Basic es compatible con la Interfaz de ARDUINO: en este caso la comunicación se hace a través del puerto serial. Para eso se agregó al proyecto una forma llamada **fbocinas** en la que se agregó el componente MSCOMM. (*Explicación técnica, ver Anexo 15*). Contiene todas las subrutinas que se usarán para las bocinas y el ARDUINO (Figura 80).

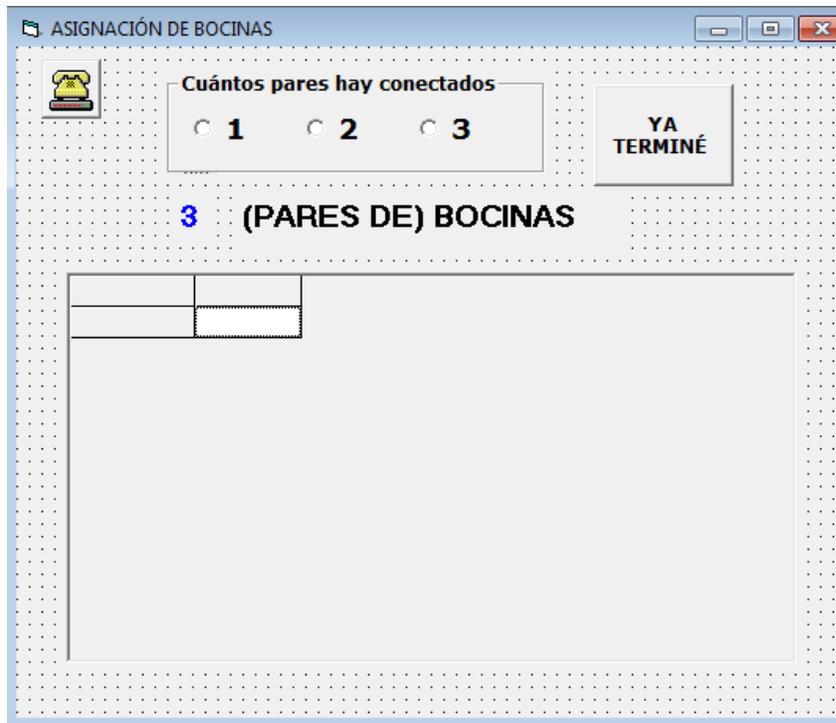


Figura 80. Asignación de bocinas a formas y pantallas.

Se creó un módulo llamado **mod_LAS_Bocinas**, que contiene todas las subrutinas que se usarán para las bocinas y el ARDUINO.

Se desarrollaron los siguientes métodos:

Una función que inicializa el puerto de comunicaciones, llamada inicializa_arduino y recibe como parámetro el puerto:

```

With fbocinas.MSComm1
    .RThreshold = 1
    .InputLen = 1
    .settings = "9600,n,8,1" "baud rate, parity, bit number and stop bit
    .CommPort = por " cambiar de acuerdo al número de puerto
    .DTREnable = False
    .PortOpen = True
    que_lei = .Input
    inicializa_arduino = True
End With

```

Para detectar el número de puerto hay que conectar el ARDUINO a la computadora por medio de USB. (*Explicación técnica, ver Anexo 14*). Los baudios tienen que corresponder a los mismos que se indicaron en el programa de ARDUINO.

Esta función regresa el valor “verdadero” o “falso” de acuerdo a que existe o no una conexión con el ARDUINO. Posteriormente esta variable se usará para asignar bocinas a las formas.

Se hicieron dos Subrutinas, una para apagar bocinas **apaga_bocinas** y otra para encender bocinas **enciende_bocinas**: ambas reciben como parámetro el par de bocinas que se desea apagar o prender.

```
Sub apaga_bocinas(cual_bocina As Integer)
    If Not tiene_arduino_instalado Then Exit Sub
    With fbocinas.MSComm1
        .Output = CStr(cual_bocina)
    End With
End Sub

Sub enciende_bocina(cual_bocina As Integer)
    ' 1 2 3 enciende bocinas 4 5 6 apaga la correspondiente
    ' 7 enciende todas 0 apaga todas
    Dim arg As String
    If cual_bocina = 0 Then
        arg = "7"
    Else
        arg = CStr(cual_bocina)
    End If
    fbocinas.MSComm1.Output = arg
End Sub
```

El componente MSComm tiene una propiedad Output, donde se indica qué acción realizará el ARDUINO, según la tabla de mensajes que se describió anteriormente.

Se desarrolló una subrutina que asigna los pares de bocinas a formas o pantallas; esto solo se hace si está conectado el controlador ARDUINO. Si no está conectado, el SPRP usa el juego de bocinas que Windows tiene conectadas por default.

Por ejemplo si tenemos conectados 2 monitores y el controlador ARDUINO, saldrá una ventana de configuración (Figura 81), en la cual se tiene que indicar cuántas bocinas conectaremos a la placa: en este caso indicaremos los 3 pares de bocinas, y aparecerá una lista de todas las formas que el U-Book puede usar. Para cada forma se indican los pares de bocinas que usará.

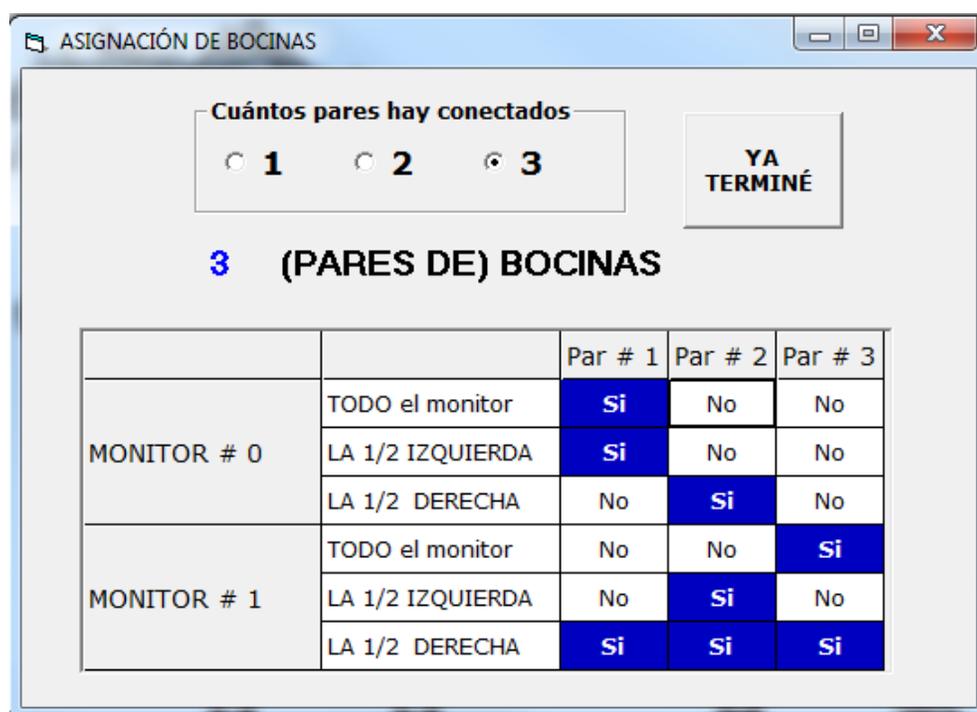


Figura 81. Configuración para tres pares de bocinas.

Con esta configuración, por ejemplo, si se muestra algo en la parte izquierda del monitor #0, el sonido se oirá solamente en la bocina 1, en cambio si se ocupa la parte derecha del mismo monitor #0, sólo se oirá en la bocina 2.

- La forma completa del monitor 1 usará el par de bocinas 1;
- La forma izquierda del monitor 1 usara el par de bocinas 1;
- La forma derecha del monitor 1 usara el par de bocinas 2;
- La forma completa del monitor 2 usara el par de bocinas 3;
- La forma izquierda del monitor 2 usara el par de bocinas 2;
- La forma derecha del monitor 2 usara el par de bocinas 1, 2 Y 3.

Como consecuencia de estas asignaciones, un objeto que sea material multimedia (audio o video) se escuchará en sus respectivas bocinas. Tomemos los siguientes casos:

- La forma completa del monitor 1 usará el par de bocinas 1, como se muestra en la Figura 82:

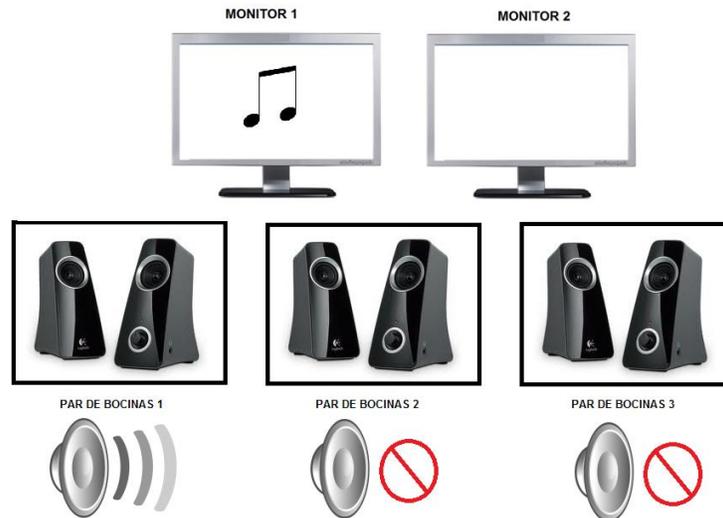


Figura 82. Configuración de un par de bocinas, para el monitor uno.

Solo se escuchara el audio en la bocina 1, las demás estarán apagadas.

- La forma derecha del monitor 2 usara el par de bocinas 3, como se muestran en la Figura 83.

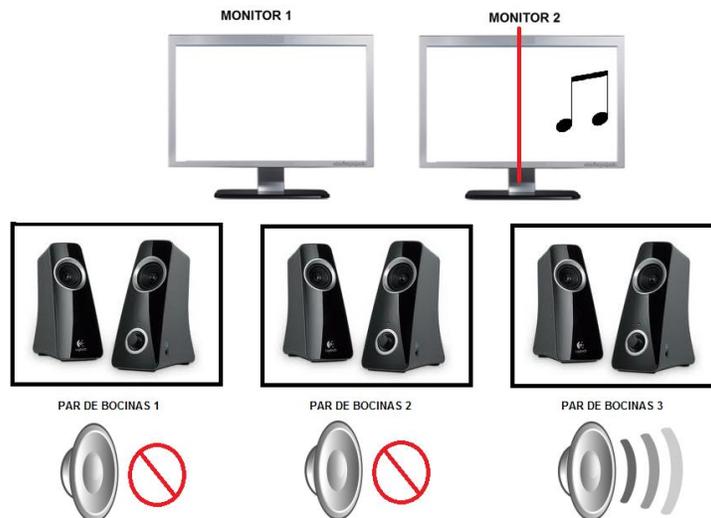


Figura 83. Configuración de un par de bocinas, para el monitor dos.

Solo se escuchará el audio en la bocina 3, las demás estarán apagadas.

- Se puede asignar 1 o hasta las 3 bocinas a la misma forma (Figura 84), por ejemplo:

La forma derecha del monitor 2 usará el par de bocinas 1, 2 y 3.

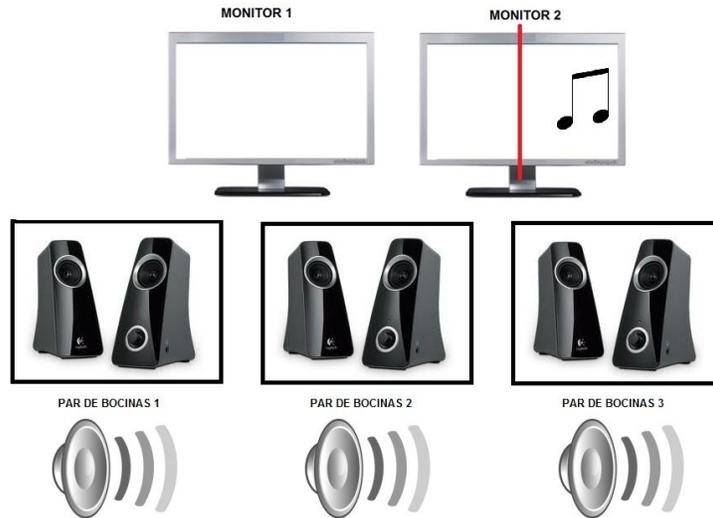


Figura 84. Configuración de tres bocinas, para el monitor dos.

Se escuchará el audio en los 3 pares.

- Nunca se reproducirán dos audios o dos videos simultáneos (el U-Book nunca solicitará este tipo de acción) (Figura 85).

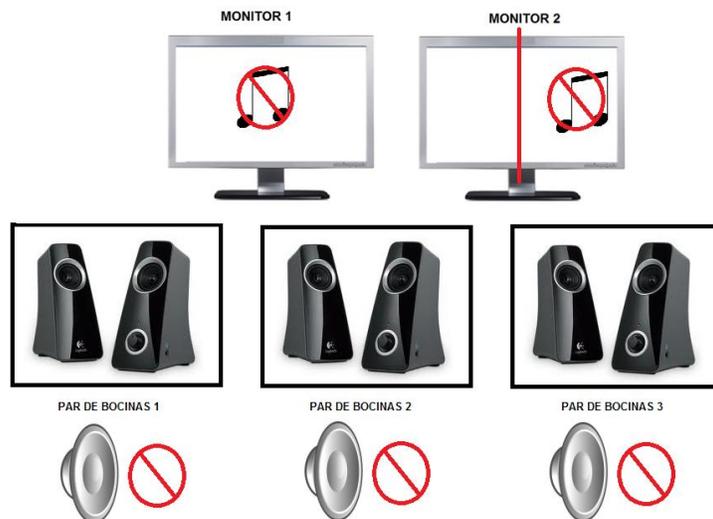


Figura 85. Excepción de asignación de bocinas.

Si la sesión está utilizando el ARDUINO, cada vez que se envía algo a una forma se encienden las bocinas asignadas a esa forma y se apagan las restantes.

La forma de asignar las bocinas es mandando valores del 0 al 7 según la forma y su asignación al par o pares bocinas que tiene asociada una forma. Utiliza un arreglo llamado **bocinas_por_Forma** el cual se declara de la siguiente forma:

Public bocinas_por_Forma(0 To 3, 0 To 2, 1 To 3) As Integer

Donde el 0 a 3 indica el monitor, 0 a 2, indica la forma, y 1 a 3 indica el par de bocinas. El valor que almacena es un entero donde 0 indica que ese par de bocinas no usará y 1 que sí usará ese par de bocinas.

Por ejemplo la forma derecha del monitor 2 usará el par de bocinas 3:

bocinas_por_Forma(2, 2, 3)= 1

Los demás valores del arreglo estarán en cero. Cuando llega un elemento de la sucesión que es multimedia (audio o video), apaga las bocinas y solo prende los pares que usará la forma que se usará para ese elemento (*explicación técnica, ver Anexo 16*).

6.3 Impacto sobre el aspecto de Aprendizaje

Al contar con la facilidad de usar bocinas diferentes para distintos objetos, el maestro puede dirigir un sonido a diversas bocinas, y eso le permitirá atraer la atención de los alumnos, que instintivamente buscarán la fuente del sonido.

Por ejemplo, se puede simular una conversación entre dos “monitores”. Sin la posibilidad de dividir el sonido, aunque la imagen salga en otro monitor, el alumno no percibirá que habla uno o el otro. Se ha comprobado que en general, los alumnos prestan atención a la fuente del sonido en mayor grado que a la de las imágenes, aunque esto deja de ser el caso si el sonido es imperceptible o incomprensible. No hace falta agregar que el uso de música en muchos casos puede constituir una herramienta valiosa para aumentar la comunicación de algún material visual.

CAPITULO VII. COMUNICACIÓN ENTRE DOS COMPUTADORAS

La conveniencia de usar una computadora adicional a la del presentador, pero en la que hay un programa que obedece las instrucciones que le llegan de la computadora “principal” surgió inicialmente como alternativa que permitiera el uso de más de un proyector para una presentación, pero en ausencia de los puertos que permitieran conectar el segundo proyector. Se conecta el segundo proyector a la computadora adicional, y esta a su vez se conecta a la del presentador, que usa el primer proyector. La comunicación entre las computadoras conectadas se hace tipo P-to-P (program to program): el programa del presentador (SPRPBuildShow) envía y recibe mensajes a un programa llamado **SPRPShow-Other** que se ejecuta en la otra computadora.

Cabe señalar que tras lograr que funcionara esta configuración se descubrió que se podría aprovechar para otros aspectos. Por ejemplo, el presentador podría usar la otra computadora como su pizarrón interactivo, o viceversa, mostrar el pizarrón DUMMY en la otra computadora.

El contenido de este capítulo se ha dividido en secciones correspondientes a los siguientes temas: Tipos de Redes; Métodos de conexión entre dos computadoras; Múltiples computadoras: Resultado de la conexión exitosa; Impacto sobre el aspecto de aprendizaje. Algunas descripciones se incluyeron para proveer un marco teórico además de explicar cómo se usan dos computadoras para una presentación o clase con los U-Books.

El concepto de dos computadoras conectadas

La comunicación de dos computadoras, nace de la necesidad de transmitir datos o compartir recursos como impresoras, scanner y otro tipo de dispositivos. Existen muchos tipos de conexión y muchos tipos de configuraciones de redes.

Una red de computadoras es un conjunto de conexiones físicas y programas de protocolo, empleados para conectar dos o más computadoras.

Es posible usar **contraseñas o identificaciones** de usuarios para restringir el acceso a algunos sectores. O para hacer que los documentos sean sólo de lectura y **no puedan borrarse o grabarse** por accidente.

Las conexiones físicas se establecen a través de un conjunto de materiales tales como cables par trenzados, cables coaxiales, fibra óptica, adaptadores de red, que permiten la comunicación entre dos o más computadoras, por medio de transmisión de información en códigos binarios.

El conjunto de software utilizado para la transmisión de datos a través de las conexiones físicas se denomina protocolos de comunicación, estos permiten la sincronización en la transferencia de datos por las redes hasta llegar a su destino. El protocolo más utilizado es TCP/IP.

Una de las necesidades del U-Book es armar sucesiones, sub-sucesiones y poder visualizarlas en múltiples pantallas, pero tenemos la limitante de cuantos monitores podemos conectar a una computadora o si contamos con una tarjeta de video, conectores y cables, en caso de no contar con tales recursos, simplemente con un cable y algunas configuraciones podemos conectar dos computadoras. Se desarrollaron aplicación para poder hacer la conexión, un protocolo de comunicación y la manera de poder compartir recursos, en este caso solo se trata de enviar y recibir archivos y comandos.

7.1 Tipos de Redes

Se distinguen diferentes tipos de redes según su tamaño (en cuanto a la cantidad de equipos), su velocidad de transferencia de datos y su alcance. Las redes privadas pertenecen a una misma organización. Generalmente se dice que existen tres categorías de redes:

- MAN (Red de área metropolitana)
- WAN (Red de área extensa)
- LAN (Red de área local)

7.1.1 Red de área metropolitana (MAN)

Una MAN (Red de área metropolitana) conecta diversas LAN cercanas geográficamente (en un área de alrededor de cincuenta kilómetros) entre sí a alta velocidad. Por lo tanto, una MAN permite que dos nodos remotos se comuniquen como si fueran parte de la misma red de área local. Una MAN está compuesta por conmutadores o routers conectados entre sí con conexiones de alta velocidad (generalmente cables de fibra óptica).

Las redes MAN también se aplican en las organizaciones, en grupos de oficinas corporativas cercanas a una ciudad, estas no contiene elementos de conmutación, los cuales desvían los paquetes por una de varias líneas de salida potenciales. Estas redes pueden ser públicas o privadas.

7.1.1 Red de área extensa(WAN)

Una red de área extensa, con frecuencia denominada WAN, acrónimo de la expresión en idioma inglés **wide area network**, es un tipo de red de computadoras capaz de cubrir distancias desde unos 100 hasta unos 1000 km, provee de servicio a un país o un continente. Muchas WAN son construidas por y para una organización o empresa particular y son de uso privado; otras son construidas por los proveedores de internet (ISP- Internet Service Provider) para proveer de conexión a sus clientes.

Hoy en día, Internet proporciona WAN de alta velocidad, y la necesidad de redes privadas WAN se ha reducido drásticamente, mientras que las redes privadas virtuales que utilizan cifrado y otras técnicas para hacer esa red dedicada, aumentan continuamente.

7.1.3 Red de Área Local (LAN):

La red de área local (Local Area Network: LAN) es la más simple de todas las conexiones que existen, ya que solo enlaza computadoras de un área pequeña como un edificio u oficina, Así mismo, una LAN puede estar conectada con otras LANs a cualquier distancia por medio de línea telefónica y ondas de radio.

Pueden ser desde 2 computadoras, hasta cientos de ellas. Todas se conectan entre sí usando precisamente la topología LAN. La computadora que se encarga de llevar el control de la red es llamada "servidor" y a las computadoras que dependen del servidor, se les llama "nodos" o "estaciones de trabajo".

Los nodos de una red pueden ser una computadora que cuenta con su propio CPU, disco duro y software y tienen la capacidad de conectarse a la red en un momento dado.

Las LANs son capaces de transmitir datos a velocidades muy rápidas, algunas inclusive más rápido que por línea telefónica; pero las distancias son limitadas.

Asimismo, en este tipo de red hay dos sistemas para colocar las computadoras que se van a conectar en red, a saber: El sistema Cliente / Servidor y el sistema Par a Par. En el sistema de cliente/servidor hay una computadora central a la que se conectan las demás. Su tarea es administrar los datos y almacenar los programas. En el sistema par a par todas las máquinas son iguales, lo que es ideal solo cuando no hay más que cuatro o cinco computadoras.

7.2 Métodos de conexión entre dos computadoras

Como se verá, la necesidad del **U-Book** es poder enviar a y recibir mensajes de UNA computadora adicional, y por ello se seleccionó una configuración sencilla LAN.

El modo tradicional para conectar dos computadoras es hacer una conexión por cable, puesto que proporciona un enlace fiable y efectivo. Existen varias alternativas para conectar dos computadoras de esta manera:

- Ethernet (Figura 86): Es el método preferido para realizar este tipo de enlaces ya que posibilita una conexión de alta velocidad y una mínima configuración. Para esto se requiere que las computadoras cuenten con una tarjeta de red;



Figura 86. Cable de Ethernet (figura izquierda) y Tarjeta de Red (figura derecha).

- Cables serie o paralelo (Figura 87): También llamado DCC (Direct Cable Connection) cuando se usa el sistema operativo Windows, ofrece un rendimiento bajo pero funciona, cosa que no siempre sucede con otras alternativas.



Figura 87. Cable paralelo.

- Cables USB (Figura 88): No se deben usar para conectar dos ordenadores directamente. Esto podría causar daños eléctricos graves. Sin embargo, existen cables especiales USB diseñados para conectar directamente dos computadoras. De hecho, estos cables simulan una conexión Ethernet, con la ventaja de que ofrecen mayor velocidad de transmisión. Cabe señalar que este aspecto en general no es de gran peso excepto si se transmiten archivos grandes de una computadora a la otra.



Figura 88. Cable USB para red.

Aparte de conectar dos computadoras de forma directa, también pueden conectarse indirectamente por medio de elementos de red. Este método requiere dos cables de red (Ethernet), cada una de las computadoras conectado al dispositivo. Existen varios equipos para interconectar computadoras.

- Hubs ethernet, switches y routers;
- Hubs USB;
- Cable telefónico.

La implementación de la red usando cualquiera de estos métodos suele ser más costoso pero tiene la ventaja de poder conectar un buen número de computadoras y crear una mini red LAN.

En los últimos años, la tecnología sin hilos (Wireless) ha ganado mucha popularidad en redes caseras. Al igual que las soluciones de cable, existen diferentes modelos de tecnología Wireless.

- Wifi;
- Bluetooth;
- Infrared.

7.3 Múltiples computadoras

Para el uso de dos computadoras conectadas el SPRP tiene dos programas que se ejecutan simultáneamente en las respectivas computadoras. De ese modo se establecerá la comunicación P-to-P, como se ilustra en la Figura 89.

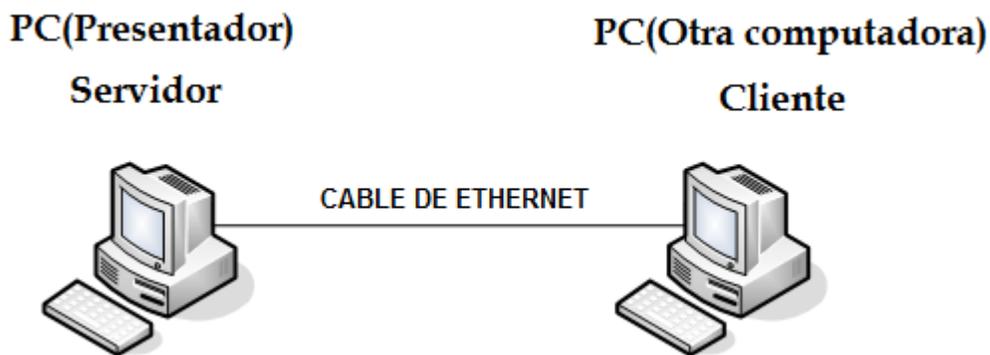


Figura 89. Comunicación entre dos computadoras.

Se tiene que conectar el cable de red entre ambas computadora (Servidor con Cliente).

7.3.1 Archivos compartidos

Antes de hacer la conexión entre computadoras (servidor y cliente), en la computadora servidor hay que compartir una carpeta en la que se depositaran todos los archivos de la sucesión que usará la computadora cliente. Es importante señalar que la computadora cliente no puede usar archivos que no residen en una carpeta compartida. Es importante saber que en el caso del sistema operativo WINDOWS 7, se crea una carpeta "PUBLIC".

Hay varias formas de compartir archivos y depende mucho del sistema operativo que se use. Se pueden compartir carpetas o se pueden hacer carpetas públicas. (*Explicación técnica, ver Anexo17*). El nombre de la carpeta compartida o pública que usa el SPRP siempre se llama "COMUNICACION_SPRP".

Para que la carpeta "COMUNICACION_SPRP" sea compartida se debe crear en la unidad C, por ejemplo C:\ COMUNICACION_SPRP. Si se desea que la carpeta sea pública hay que crearla en el directorio de documentos compartidos, C:\Users\Public\Documents\ COMUNICACION_SPRP (Figura 90).

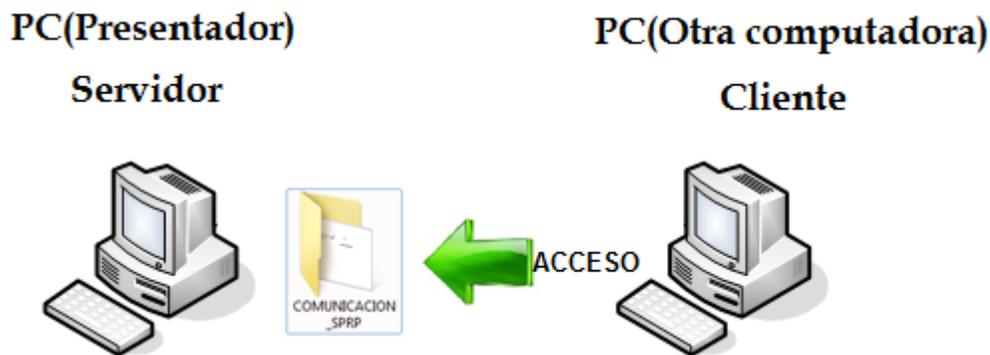


Figura 90. Acceso a la carpeta compartida.

Hay mucha incompatibilidad de conexión entre diferentes sistemas operativos. Es conveniente que el inicio de una sesión de Windows de la computadora cliente no tenga contraseña: hay que verificar que cuando se haga la conexión y al compartir archivos no haya restricciones y no se necesite la contraseña del servidor, para evitar tener que proporcionar la contraseña cuando el cliente se conecte. Se recomienda cambiar la Configuración de uso compartido avanzado, (*Explicación técnica, ver Anexo 18*).

Después de haber compartido la carpeta, quitado la contraseña y haber agregado los permisos de la carpeta, y modificado la configuración del uso compartido, se recomienda hacer una prueba de conexión:

- Hay que averiguar la IP (*Explicación técnica, ver Anexo 19*) o el nombre de equipo de red (*Explicación técnica, ver Anexo 20*) del servidor.
- En la computadora cliente seleccionar inicio y la opción ejecutar (Figura 91), teclear **NOMBRE DE EQUIPO** o la **IP de la otra computadora (servidor)**, y darle enter.



Figura 91. Conexión a otra computadora.

- Si toda la configuración es correcta, aparecerá una ventana donde estará la carpeta compartida (Figura 92).

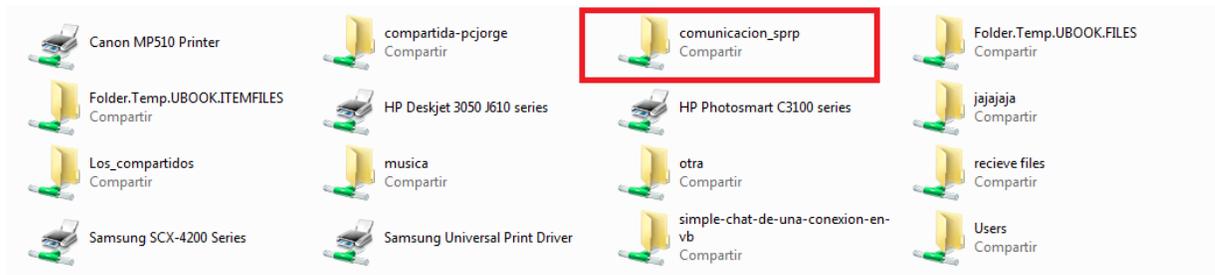


Figura 92. Carpetas compartidas.

- En el caso contrario aparecerá una ventana donde habrá que indicar el nombre y la contraseña de la computadora del servidor (Presentador) y posteriormente aparecerá la ventana con la carpeta compartida (Figura 93).



Figura 93. Restricción de acceso.

7.3.2 Comunicación entre programas

Para poder hacer que dos aplicaciones se comuniquen por medio de un conexión LAN, en este caso el **SPRPBuildShow** (servidor/presentador) y **SPRPShow-Other** (cliente/otra), es necesario añadir el componente Winsock a una forma del proyecto (*Explicación técnica, ver Anexo 21*). Este componente tiene varios métodos, solo hay que configurarlo con algunos parámetros:

- La configuración es muy similar en ambos programas. En el caso de **SPRPBuildShow** (servidor/presentador), se agrega una subrutina que tiene dos parámetros: la IP y/o el nombre de la computadora que vamos a conectar y el puerto, que siempre es el 123 y conectar el socket.

```

With Winsock1
    .Close
    .RemoteHost = IP o nombre de la cliente
    .RemotePort = 123
    .Connect
End With

```

- En el caso **SPRPShow-Other**(cliente/otra) solo hay que indicar el número de puerto (que también siempre será 123) y ponerlo a escuchar:

```

With Winsock1
    .Close
    .LocalPort = 123
    .Listen
End With

```

- En ambos programas para aceptar la conexión se agrega una función llamada Winsock1_ConnectionRequest(requestID), donde requestID es el número de identificador de la conexión. Para establecer la conexión se tiene que indicar el ID:

```

Winsock1.Accept requestID

```

- Hay que tener una subrutina que manda una cadena, y por lo tanto usa un parámetro que contiene el mensaje que se manda. Se usa una método del socket SendData, para mandar el mensaje:

```

fSOCKETDELAOTRA.Winsock1.SendData mensaje

```

- Para poder recibir el mensaje se creó una subrutina que invoca el método GetData del socket. Le llega la cadena tipo Cadena y se almacena en la variable caam, que posteriormente se procesa de acuerdo al contenido:

Todos los mensajes tienen la misma estructura, que consta de 3 partes, como se muestra en la Figura 94:

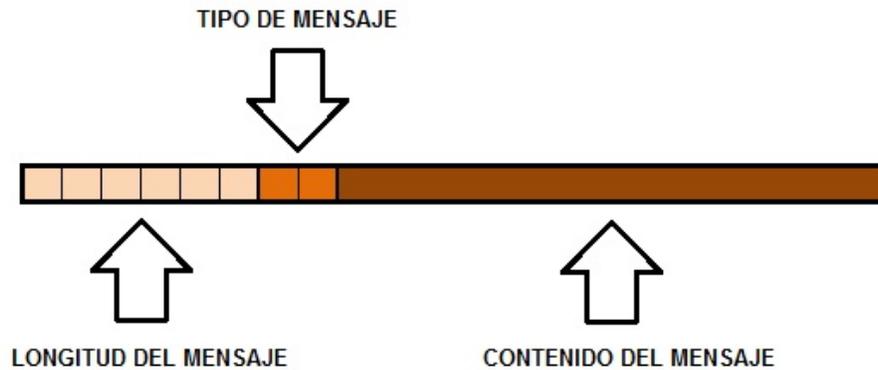


Figura 94. Estructura de mensajes de comunicación entre dos computadoras

- La primera parte es la longitud del mensaje son 6 dígitos.
- El tipo de mensaje: es un código del SPRP de 2 dígitos, la tabla de mensajes se muestra en la Tabla 8:

Tabla 8. Acciones y tipos de mensajes.

CÓDIGO	ACCIÓN	TIPO DE MENSAJE	
		COMPUTADORA PRINCIPAL	OTRA COMPUTADORA
1	Bienvenida	Salida	Entrada
2	Muestra un Objeto	Salida	Entrada
5	Muestra PICTURE en su DUMMY	Entrada/Salida	Entrada/Salida
7	Finaliza DUMMY	Entrada	Salida
8	Instancia y Muestra DUMMY	Salida	Entrada
9	Muestra Pizarrón	Salida	Entrada
10	Pide objeto prediseñado	Entrada	Salida
11	Coloca imagen prediseñada	Salida	Entrada
12	Ya no hay mas imágenes prediseñadas	Salida	Entrada
22	Saludo de regreso	Entrada	Salida
99	Finalizo pizarrón interactivo	Entrada	Salida
87	FIN de la Sucesión	Salida	Entrada

- El contenido del mensaje (Tabla 9), es una cadena que puede tener mucha información. De ese modo está conformado de muchos pedazos que contienen lo necesario para la comunicación.

Tabla 9. Descripción de mensajes de comunicación.

CÓDIGO	DESCRIPCIÓN
1	Inicio de conexión entre la computadora principal y la otra computadora.
2	Muestra un Objeto cualquiera: jpg, pdf, Word, video, audio, RTF, etc.
5	Manda mensaje para que muestre el Contenido (Imagen) de un pizarrón INTERACTIVO a su DUMMY.
7	Manda acción cuando se finaliza un DUMMY
8	Instancia y Muestra DUMMY en su forma adecuada.
9	Muestra un Pizarrón interactivo.
10	Pide objeto prediseñado a la computadora principal.
11	Alerta que ya se colocó una imagen prediseñada en la carpeta compartida y que la muestre.
12	Ya no hay más imágenes prediseñadas, oculta el botón de imágenes prediseñadas.
22	Saludo de regreso y se indica cuantos monitores están conectados en la otra máquina.
99	Finalizo pizarrón interactivo y se oculta el pizarrón interactivo.

Se crearon estructuras especiales (en VB se llaman TYPE), para cada uno de los tipos de mensaje. En el caso de un mensaje de entrada, para los diversos tipos hay sendas subrutinas que interpretan el mensaje: decodifican las cadenas según el tipo de mensaje, es decir, las descomponen y almacenan en variables del TYPE correspondiente. (*Explicación técnica, ver Anexo 22*).

Para enviar un mensaje se arma una cadena, concatenando datos y especificando su longitud.

Para realizar la conexión hay que ejecutar las aplicaciones en ambos programas.

Ejecutar el programa **SPRPSHOW-Other** en la computadora cliente (la “otra” computadora) y una vez que se verificaron los tamaños de las formas aparecerá la pantalla mostrada en la Figura 95: seleccionar el botón Poner a la escucha y simplemente hay que esperar la conexión, que le llega con un mensaje que le envía la otra computadora.

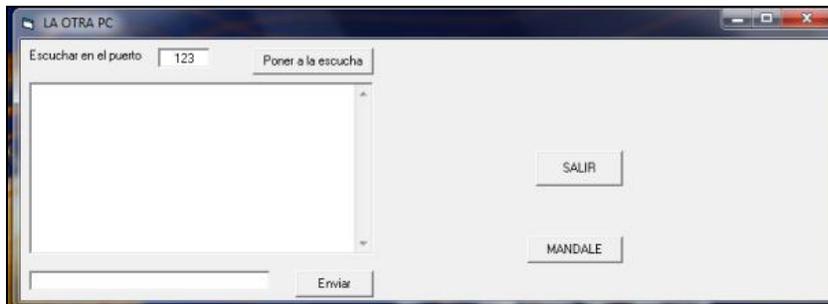


Figura 95. Forma de SPRPSHOW-Other que espera el mensaje de conexión.

Ejecutar el programa el **SPRPBUILDShow** del SPRP en la computadora servidor (Presentador). Después de la configuración de pantallas aparecerá una pantalla de configuración, como se muestra en la Figura 96: se “checa” que SE CONECTARÁ OTRA COMPUTADORA, para que aparezcan las opciones de conexión.

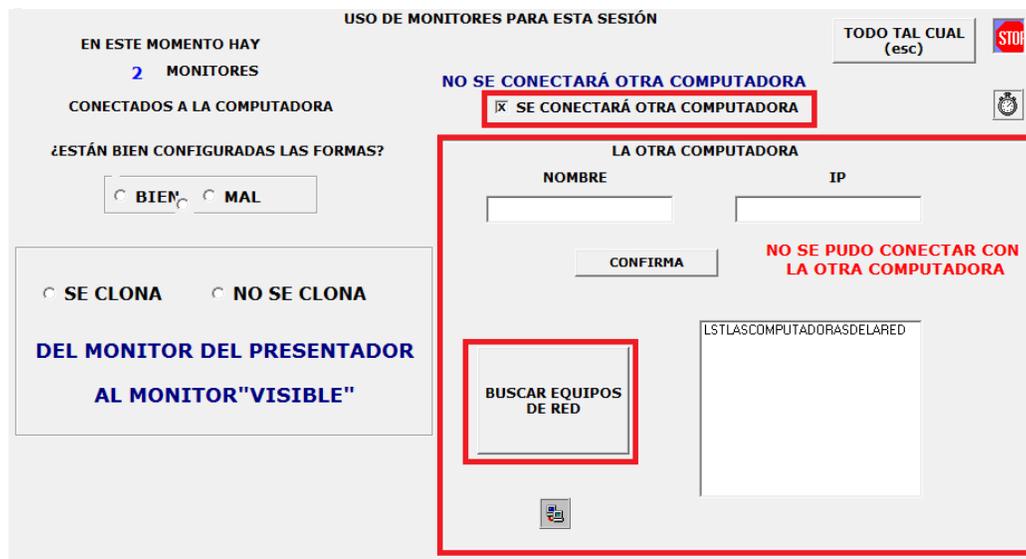


Figura 96. Forma de SPRPBUILDShow para configurar la conexión.

Si se selecciona la opción de BUSCAR EQUIPOS DE LA RED que se está ocupando, aparecerá una lista con todos los equipos conectados a la red LAN (Figura 97): en ella se elige el nombre del cliente deseado y dar clic en aceptar.

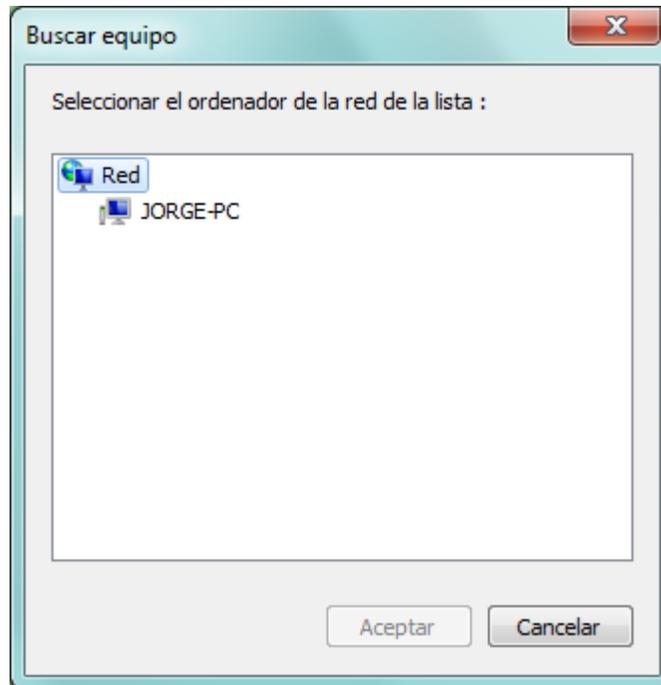


Figura 97. Equipos conectados a la red LAN.

En caso de que no aparezca el nombre del cliente en la lista, hay que cerrar esa pantalla y teclear ya sea el NOMBRE DE EQUIPO o la IP de la computadora cliente que se desea conectar.

Al seleccionar CONFIRMAR saldrá una leyenda validando la conexión. Si se establece la conexión, ambos programas continúan. En caso contrario solo sale la leyenda “error de conexión” en la forma, y en ese caso hay que comprobar lo siguiente:

- Checar que el cable esté bien conectado;
- Verificar el Nombre de Equipo o la IP del cliente sea correcto;
- Asegurarse que se está ejecutando el programa SPRPShow-other en la otra computadora, el cliente.

7.4 Resultado de la conexión exitosa

Si se pudo establecer la conexión, ambos programas se están ejecutando y “escuchando” mutuamente. En este caso, la computadora cliente (es decir, el programa que se ejecuta en ella) le informa al servidor el número de monitores que están conectados a ella y por lo tanto, le ofrece al SPRP para mostrar sucesiones.

El programa agregará esos monitores a los que ya tiene, y todos ellos estarán disponibles para mostrar objetos de una sucesión.

Como se vio anteriormente, al cargar el programa SPRPShow-other se configuran las formas de los monitores de esa computadora. Del mismo modo, cuando se carga el programa en la otra computadora, se realizan tareas similares a las descritas para la asignación de bocina a dichas formas (siempre que esté conectado un ARDUINO a dicha computadora).

7.5 Impacto sobre el aspecto de Aprendizaje

De hecho, contar con otra computadora conectada es equivalente a conectar otros monitores (y quizá bocinas) a la computadora. Esto es de utilidad cuando no se dispone de una tarjeta de video con la capacidad de soportar el número deseado de monitores. Como se mencionó en el Capítulo V (Pizarrones), también ofrece ventajas para desplegar dichos elementos.

Ya se comentaron las ventajas que ofrece esta segunda computadora en comparación con el uso de una sola (además de contar con los monitores adicionales). Uno de ellos es que el presentador puede usar dos teclados y mouse.

CAPITULO VIII. EJEMPLO DE UNA CLASE O PRESENTACIÓN

Para ejemplificar el uso de los dispositivos y elementos que se agregaron para su aprovechamiento con un U-Book, se presenta como se mostraría una misma sucesión en la presencia de diferentes configuraciones. Se han seleccionado solo algunas configuraciones para explicar cómo se aprovechan los monitores. Señalamos que se decidió no utilizar diagramas para ilustrar las situaciones para no alargar aún más la tesis, puesto que las descripciones vía texto deben ser suficientes para ilustrar lo que se explica.

En la primera, en la computadora del presentador hay un proyector conectado, pero se usa como pantalla extendida (y no clonado). Es importante señalar lo obvio: puede ser (en general será el caso) que la audiencia NO VEA el monitor del presentador, de modo que lo que se muestre en ese monitor, para que se vea en el proyector, debe ser enviado al proyector. El SPRP realiza esta tarea, que allí se conoce como “clonar la imagen a otro monitor”). Se indica a cuál de las formas se debe enviar, y la clonación resulta en que se exhiba el mismo objeto que en la forma fuente, es decir, a partir de la cual se clona. Observe que el SPRP permite ejecutar un mismo archivo multimedia en ambas formas, pero más vale que se trate del mismo si contiene elementos de audio. Esto puede resultar en un desfase de los audios (en el video no importa). Si se cuenta con varias bocinas (con el ARDUINO) se resuelve el problema: al monitor principal no se le asignan bocinas, de modo que solo se oirán las que se asocien al proyector.

En la otra, se agrega a esta configuración de los monitores de la computadora principal, una segunda computadora que tiene una configuración idéntica, es decir, el proyector es un monitor extendido y la pantalla de la computadora puede no ser visible. Observe que si se trata de una computadora de escritorio con tarjeta de video que lo permite (es decir, permite conectar un proyector y otro monitor) se podrían usar dos proyectores o un proyector y un televisor.

Nos limitaremos a presentar el uso de un pizarrón interactivo en estas configuraciones, y supondremos que el uso del pizarrón se ofrece mientras que en otro monitor se muestra “algo” (que puede ser un texto, una imagen, un audio o un video). De hecho en U-Books hay otros tipos de ítems: se puede declarar como tal una consulta (en vivo) a una base de datos o una página de internet, o ejecutar algún programa. Usaremos precisamente esta última posibilidad.

Supongamos que, como parte de una presentación o sesión de capacitación a usuarios, deseamos explicar una función de un programa. Produciremos un video (con un producto como CAMTASIA) y le programamos “pausas”, en las cuales explicaremos algunos conceptos usando un pizarrón interactivo. Podemos usar un pizarrón DUMMY si alcanzan los dispositivos; caso contrario permitiremos que los oyentes vean nuestra interacción con el pizarrón (en particular, cuando escribo algo en mi monitor táctil, verán “mi letra”).

Construimos la sucesión del siguiente modo (supongamos que no tiene otros elementos, aunque naturalmente puede contenerlos y no afectará el que nos ocupa).

Elemento 1: un pizarrón interactivo. Le asocio el ítem “video de la ejecución” e indico que se mostrarán en forma simultánea. Veremos cómo se aprovechan los monitores disponibles para esta situación. Para los lectores que se dieron cuenta (no se puede asignar un elemento asociado a un pizarrón) les informamos que se hace incluyendo como elemento 1 un “ítem-pizarrón”, al que sí se le puede asociar el video. De hecho, podría definir la sucesión al revés. Elemento 1: video, asociado, el pizarrón (otra vez como ítem).

Configuración más frecuente: tengo un proyector “clonado”. Envío el pizarrón a la mitad izquierda de mi monitor, y el asociado (video) a la mitad derecha. Aprovecho las pausas para explicar el video. Si lo hiciera al revés, aparecería el video en la mitad izquierda.

Configuración 1 proyector “extendido”.

Mostramos el video en la mitad izquierda del proyector. Luego enviamos el pizarrón interactivo a “mi monitor” (invisible a la audiencia) y usamos la mitad derecha del proyector como pizarrón DUMMY. Esto hace que podamos escribir en todo el monitor, aunque la imagen se verá reducida en el proyector). Los que hemos tenido que escribir con letra pequeña usando un mouse comulgarán con nosotros cuando decimos que nos gusta esta posibilidad del SPRP.

Configuración: monitor del presentador visible y proyector extendido.

Mostramos el VIDEO en el proyector, y usamos el monitor como pizarrón interactivo (sin DUMMY).

Ahora veamos las posibilidades que nos da contar con otra computadora, en especial, su uso más frecuente: le conectamos otro proyector.

Enviamos el video a la otra computadora, y manejamos “la mía” como pizarrón. Si nos resulta más fácil interactuar con el monitor de la otra (usar el teclado y mouse de aquella) lo hacemos al revés.

Estamos trabajando en varios aspectos del SW que permitan “dirigir” los objetos sin hacerle caso a las reglas del SPRP. Esencialmente, estas reglas son como siguen:

- Numera los monitores del 0 al mayor disponible: 0, 1, etc. Si hay otra computadora, los números se asignan a continuación de los de la computadora del presentador.
- En cada monitor, como se vio, se definen 3 formas (el monitor total y cada una de sus mitades).
- El programa usará el “siguiente” monitor para los elementos asociados simultáneos que mostrará.

El SPRP ya contiene diversos modos de cambiar estas asignaciones automáticas. Se especifica en la sucesión, como parte del elemento, dónde se muestra. Como la sucesión se podrá mostrar en distintas configuraciones, esto no es tan evidente como parece, y se usan varios artificios para obtener los resultados (asignaciones) deseados en todas las configuraciones.

Observación sobre el uso de una computadora adicional

Pudiera parecer que es lo mismo que manejar dos sucesiones o presentaciones (una en cada computadora). Esto no es así: el mismo programa controla ambas presentaciones, y no hace falta “manejar” la otra computadora, cosa que podría ser difícil en una situación. De hecho, hemos usado el monitor de una computadora como dispositivo “visible” cuando no teníamos más que un proyector. Se coloca la computadora de modo que la vea la audiencia, y se la usa como si fuera otro monitor conectado a la “mía”. Mandamos el video a la otra computadora, y usamos la propia como pizarrón interactivo: si el proyector está clonado, no usará DUMMY.

Pensamos que con estas explicaciones, será posible que un lector interesado en el tema formule otras configuraciones y se imagine cómo las usaría. Repetimos que lo importante es que se puede manejar todo desde una sucesión. Y recordamos al lector que se puede enviar el DUMMY de un pizarrón interactivo a la “otra” computadora, en ambos sentidos. En otras palabras, si no tenemos suficientes monitores para usar un pizarrón que no vea la audiencia (o está clonado el nuestro al proyector) usamos el pizarrón interactivo en la otra computadora, y enviamos el pizarrón DUMMY a la nuestra. En varias ocasiones hemos usado este artificio: manejamos la sucesión en una laptop conectada a un proyector (en modo clonado): le conectamos la computadora de escritorio con el monitor táctil. Ahora escribimos y pintamos en el monitor touch, y mostramos todo en la computadora principal, de modo que se vea en el proyector.

Opinamos que el único modo de apreciar y terminar de entender el potencial de estas herramientas es ver una presentación en vivo, o mejor todavía, elaborar una presentación e implementarla.

Dado el caso, se puede interactuar con la otra computadora con sus propios dispositivos, en especial el teclado y el ratón. Es importante señalar en este sentido que aunque se muestren varios objetos seguidos en la otra computadora (como una serie de imágenes) el control que hace que se muestre la siguiente lo tendrá la computadora principal, es decir, la del presentador.

CAPÍTULO IX. CONCLUSIONES

La primera conclusión es la formulada por el consejero, con la aprobación de los asesores del Comité Particular: se lograron todos los objetivos que se fijaron para la investigación.

En particular, los U-Books ahora incluyen la posibilidad de no solo asociar un elemento a otro de una sucesión, sino hacer esto en forma recursiva. También se pueden incluir ítems que representen un pizarrón virtual, en lugar de solamente archivos de material multimedia.

El SPRP calcula donde mostrará los diferentes elementos de una sucesión, para lo cual al inicio de una sesión determina los monitores que están conectados, permite configurar las formas que se muestren en cada uno de ellos, y si es preciso, trabajar con otra computadora en modo Program-to-Program para exhibir elementos usando este otro ordenador.

El SPRP también puede encender las bocinas conectadas a cualquiera de las dos computadoras enviando una señal al dispositivo (ARDUINO). Para aprovechar esta funcionalidad, puede asignar los pares de bocinas que se asociarán a cada una de las formas que usará en esa sesión de trabajo.

Se hicieron y depuraron todos los programas auxiliares que requiere la preparación de los monitores, formas y bocinas de una sesión.

El resultado es una versión del paquete SPRP que contiene todos los elementos necesarios para ofrecer material del modo descrito en esta tesis.

La investigación o trabajo que quedó pendiente se puede resumir en dos temas fundamentales: la reprogramación de todo el paquete en VB.Net, misma que estaba a cargo de otro alumno que finalmente decidió no realizarla, y la programación de ciertas páginas de Internet que permitan definir y leer sucesiones sin tener que instalar el SPRP en la computadora del lector. Es importante señalar que ninguno de estos temas estaba incluido en el proyecto de investigación que resultó en esta tesis.

CAPÍTULO X. Bibliografía

- Anderson, L.W., and Krathwohl, D. (2001). A Taxonomy for Learning, Teaching and Assessing: a Revision of Bloom's Taxonomy of Educational Objectives. Longman, New York.
- Bauer Mengelberg, J.R. (2007). The concept of an unstructured book and the software to publish and read it. Information and Beyond: Part II: Journal of Issues in Informing Science and Information Technology. Vol 4. Santa Rosa. CA: Informing Science Press. P. 801-810.
- Bauer Mengelberg, J.R. (2011). The Use of U-books to Inform Delegates to a Conference. Proceedings of Informing Science & IT Education Conference (InSITE). Journal of Issues in Informing Science and Information Technology.
- Bauer Mengelberg, J. R. (2012). The use of Keywords by Context to Relate Items of a U-book. Proceedings of Informing Science & IT Education Conference (InSITE). Journal of Issues in Informing Science and Information Technology.
- Bandler, R., Grinder, J. (1979) Frogs into Princes: Neuro Linguistic Programming. Real People Press. 149 p.
- Cabero, J., Duarte, A. y Barroso, J. (1997). La piedra angular para la incorporación de los medios audiovisuales, informáticos y nuevas tecnologías en los contextos educativos: la formación y el perfeccionamiento del profesorado. Edutec: Revista de Tecnología Educativa. Ed. 8.
- Felder, R. M. y Brent, R. (2005). Understanding Student Differences. Journal of Engineering Education. Ed. 16.
- Gardner, H. (1987). Estructuras de la mente. La teoría de las múltiples Inteligencias. Ed. F.C.E. México.
- Herrmann, Ned. (1999). The Theory Behind the HBDI and Whole Brain Technology.

- Igoe, Tom. (2011). Making Things Talk. Canadá. P. 16-47.
- Igoe, Tom. y O'Sullivan, Dan. (1998). Physical Computing. P. 353-361.
- Ellis, SR. (1991) Nature and Origin of Virtual Environments: A Bibliographic Essay, Computing Systems in Engineering. P. 321-347.
- Kolb, D. (1981), Learning styles and disciplinary differences. In A. Chickering (Ed.), The Modern American College. San Francisco: Jossey-Bass. U.S.A. P. 2-25.
- Manual de estilos de Aprendizaje. (2001) Material Autoinstruccional para Docentes y Orientadores Educativos. Dirección de Coordinación Académica (SEP). Programa Nacional de Educación 2001-2006.
- Martín Iglesias, J. P., (2010). La pizarra digital interactiva (PDi) en educación, Anaya Multimedia, Madrid.
- Martínez, Frausto y Santillán Bonilla, Rosa Arellí. (2006). El uso del pizarrón interactivo en la escuela primaria P. 7-14.
- Massimo, Banzi. (2011). Getting Started with Arduino. 2nd. Edition. U.S.A. P. 53-71.
- Reid, J. (1996). Learning Styles: Issues and Answers. Learning Styles in the ESL/EFL Classroom. U.S.A.: Heinle & Heinle Publishers. P. 3-34.
- Sherman, W.R. y Craig, A.B.(2003). Understanding Virtual Reality. San Francisco, CA:Morgan Kaufmann.
- Sweller, J. (1988). Cognitive load during problem solving: Effects on learning, Cognitive Science. P. 12, 257-285.
- Tiffin, J. y Rajasingham, L. (1997). En busca de la clase virtual. La educación en la sociedad de la información. Ed. Paidós, Barcelona.
- VerLee, W.L. (1986). Aprender con todo el cerebro. Barcelona,España. Ed. Martínez-Roca.

Winn, W. (1993). A Conceptual Basis for Educational Applications of Virtual Reality. Reporte técnico no. TR-93-9. Human Interface Technology Laboratory, Washington Technology Center, University of Washington. P. 85-90.

Woolfolk A. (1996). Psicología Educativa, Ed. Prentice-Hall, México. P. 123-131.

Direcciones de internet consultadas

Puertos PCI. URL consultada:

<http://www.pcisig.com/home>

Interfaz ARDUINO. URL consultada:

<http://www.arduino.cc/es/>

Ayuda de Visual Basic MSDN. URL consultada:

<http://social.msdn.microsoft.com/Search/es-MX?query=user32&ac=8>

CAMTASIA Recording. URL consultada:

<http://www.techsmith.com/camtasia.html>

Manual de Estilos de aprendizaje. URL consultada:

http://www.dgb.sep.gob.mx/informacion_academica/actividadesparaescolares/multimedia/manual.pdf

Adaptación de los entornos virtuales a los estilos cognitivos de los estudiantes: un factor de calidad de la docencia virtual. URL consultada:

<http://biblioteca.universia.net/ficha.do?id=2010842>

ANEXOS

Anexo 1. Configuración de múltiples pantallas.

- Buscar y seleccionar panel de control, en apariencia y personalización seleccionar “Ajustar resolución de pantalla”.

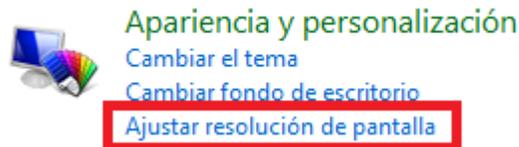


Figura 98. Apariencia y Personalización de Windows.

- En la opción “varias pantallas”, seleccionar la opción “extender estas pantallas”.

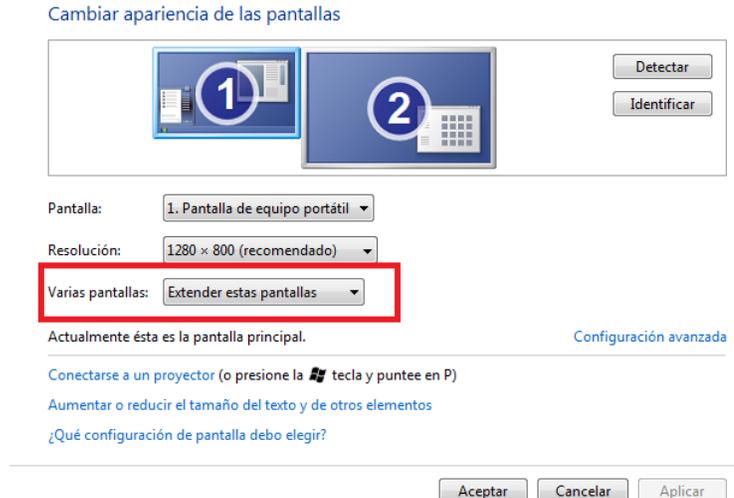


Figura 99. Configuración de pantallas de Windows.

- El proceso para tres pantallas es similar a los pasos anteriores, hasta “Resolución de pantallas”, donde se mostrarán 3 pantallas enumeradas del 1 al 3.

Anexo 2. Detección de los monitores conectados.

El programa contiene dos clases llamadas clsMonitor y clsMonitors.

- La clase clsMonitors, contiene una librería “user32”, esta librería usa varias funciones y métodos:

```
Private Declare Function GetSystemMetrics Lib "user32" (ByVal nIndex As Long) As Long
```

```
Private Declare Function GetMonitorInfo Lib "user32" Alias "GetMonitorInfoA" ( ByVal h As Long, M As tagMONITORINFO) As Long
```

User32.dll es una biblioteca de enlace dinámico (DLL) que implementa la biblioteca de clientes de API de usuario de Windows. Es un archivo principal para varias versiones del sistema operativo Microsoft Windows y contiene muchas de las propiedades del sistema operativo como la cantidad de monitores conectados y las características de cada monitor.

- Para obtener el número de monitores se instancia la clase “clsMonitors” y se usa el siguiente método para obtener el número de monitores conectados.

```
Set cMonitors = New clsMonitors → Instancia
```

```
cMonitors.Monitors.Count → Número de monitores
```

- La forma en que se obtiene la resolución actual de cada pantalla conectada es la siguiente:

```
Set cMonitor = New clsMonitor → Instancia
Sub ARMATE_CONFIGURACION_SISTEMA_ACTUAL()
    Dim I As Integer
    I = 1
    For Each cMonitor In cMonitors.Monitors
        With cMonitor
            MONITORES_SISTEMA_ACTUAL(I).NOMBRE = I
            MONITORES_SISTEMA_ACTUAL(I).Height = Alto
            MONITORES_SISTEMA_ACTUAL(I).Left = .Left
            MONITORES_SISTEMA_ACTUAL(I).Top = .Top
            MONITORES_SISTEMA_ACTUAL(I).Width = Alto
        End With
        I = I + 1
    Next
End Sub
```

Todos los datos obtenidos son en pixeles, para ello se creó una subrutina que llena toda la información obteniendo los datos de la clase clsMonitor e instanciándola.

Anexo 3. Formas MDI.

Visual Basic permite crear aplicaciones empleando una interfaz de múltiples documentos (MDI - Multiple Document Interface), o múltiples formularios. En una aplicación MDI, se pueden abrir varias ventanas hijas. Cada una de estas ventanas hijas (formularios) son iguales entre sí. En una aplicación MDI puede haber varias ventanas hijas, pero solo una ventana padre por aplicación. El formulario padre actúa como contenedor de los formularios hijos. Muchos procesadores de textos bajo Windows son buenos ejemplos de aplicaciones MDI.

Para crear una aplicación MDI se empezará a crear un nuevo proyecto y, se accede a Insertar/MDI Form. El nuevo formulario será el formulario padre. Para que un formulario sea un formulario hijo, se deberá cambiar su propiedad MDIChild y establecerla a True.

Anexo 4. Cómo detecta el tamaño de la forma (0) de cada monitor.

- Se utiliza la propiedad del formulario WindowState, que tiene los siguientes atributos.
 - 0 – Normal
 - 1 - Minimized
 - 2 – Maximized
- Se establece la propiedad 2 y automáticamente ajusta la primer forma en la primer pantalla, posteriormente se guardan los datos.
- Para el resto de las formas se recorre con la propiedad “forma.left” sumándole el ancho de la forma cero (paso anterior) y se establece la propiedad 2 de WindowState y se guardan los datos. Así se hace sucesivamente para el resto de las formas.

Anexo 5: Datos que contiene el archivo de configuración.

- Los datos que contiene el archivo de configuración son un arreglo de TYPEs:

```
Public TYPE MONITOR
    NOMBRE As Integer
    Left As Long
    Top As Long
    Width As Long
    Height As Long
End TYPE
```

```
Public FORMAS_CONFIGURADAS() As MONITOR → Arreglo
```

El tamaño del arreglo es determinado por el número de monitores conectados, por cada uno se guarda un registro de tipo TYPE Monitor.

Ejemplo 2 monitores:

- El tamaño del arreglo será de 2.
ReDim FORMAS_CONFIGURADAS(0 To CUANTOS_MONITORES)
Quedaría de esta forma:
ReDim FORMAS_CONFIGURADAS(0 To 1)
- Se guardarán dos registros con la información siguiente:
 - Para la forma MDI siempre será la posición cero "0".
FORMAS_CONFIGURADAS(0).Height = alto del MDI
FORMAS_CONFIGURADAS(0).Left = posición Y del MDI
FORMAS_CONFIGURADAS(0).NOMBRE = MDI
FORMAS_CONFIGURADAS(0).Top = posición X del MDI
FORMAS_CONFIGURADAS(0).Width = ancho MDI
 - Para la forma 1 que abarca la totalidad del monitor 1:
FORMAS_CONFIGURADAS(1).Height = alto de la FORMA 1
FORMAS_CONFIGURADAS(1).Left = posición Y de la FORMA 1
FORMAS_CONFIGURADAS(1).NOMBRE = FORMA 1
FORMAS_CONFIGURADAS(1).Top = posición X de la FORMA 1
FORMAS_CONFIGURADAS(1).Width = ancho de la FORMA 1
 - Para la forma 2 que abarca la totalidad del monitor 2:
FORMAS_CONFIGURADAS(2).Height = alto FORMA 2
FORMAS_CONFIGURADAS(2).Left = posición Y FORMA 2
FORMAS_CONFIGURADAS(2).NOMBRE = FORMA 2
FORMAS_CONFIGURADAS(2).Top = posición X FORMA 2
FORMAS_CONFIGURADAS(2).Width = ancho FORMA 2
- La forma en que se guarda el arreglo es la siguiente, se guarda de forma binaria:
Sub GUARDA_ARCHIVOS_CONFIGURACION()
Open PATH_CONFIGURACIÓN For Binary As #32
Put #32, , FORMAS_CONFIGURADAS
Close #32
End Sub
- La forma en que se abre y verifica si existe el archivo es la siguiente:
Function VERIFICA_Y_ABRE_SI_EXISTE_EL_ARCHIVO_DE_SETUP() As Boolean
If Dir(PATH_CONFIGURACIÓN) <> "" Then
Open PATH_CONFIGURACIÓN For Binary As #32
Get #32, , FORMAS_CONFIGURADAS
Close #32
VERIFICA_Y_ABRE_SI_EXISTE_EL_ARCHIVO_DE_SETUP = True
Else

```

        VERIFICA_Y_ABRE_SI_EXISTE_EL_ARCHIVO_DE_SETUP = False
    End If
End Function

```

Si existe el archivo lo abre y regresa un verdadero, en caso contrario solo regresa un falso.

Anexo 6: Archivo de configuración.

El módulo mod_CONFIRURACION_Y_FORMAS contiene lo necesario para redimensionar una forma

- Determina el archivo que se utilizará y lo abre, en caso de que no exista regresará un valor false.

```

Private Function
VERIFICA_Y_ABRE_SI_EXISTE_EL_ARCHIVO_DE_SETUP() As Boolean
    VERIFICA_Y_ABRE_SI_EXISTE_EL_ARCHIVO_DE_SETUP = False
    Dim PATH_CONFIGURACIÓN As String
    Dim directorio_donde_estan_las_configuraciones As String
    directorio_donde_estan_las_configuraciones = "c:"

    PATH_CONFIGURACIÓN = directorio_donde_estan_las_configuraciones & "\SETUP_MONITORES_" &
        CUANTOS_MONITORES & ".txt"
    If Dir(PATH_CONFIGURACIÓN) = "" Then Exit Function
    Open PATH_CONFIGURACIÓN For Binary As #32
    Get #32, , FORMAS_CONFIGURADAS
    Close #32

    VERIFICA_Y_ABRE_SI_EXISTE_EL_ARCHIVO_DE_SETUP = True
End Function

```

- El arreglo que contiene toda la información es de tipo TYPE. La dimensión del arreglo FORMAS_CONFIGURADAS() depende de cuántos monitores estén conectados.

```

Private TYPE T_UNA_FORMA_CONFIGURADA
    NOMBRE As Integer
    Left As Long
    Top As Long
    WIDTH As Long
    HEIGHT As Long
End TYPE
Public FORMAS_CONFIGURADAS() As T_UNA_FORMA_CONFIGURADA

```

Anexo 7: Preparación y dimensión de la forma MDI.

- Para colocar las dimensiones y posición de la forma MDI se invoca la siguiente subrutina. Se usa las propiedades de TOP, LEFT, WIDTH y HEIGH.

```

Private Sub PREPARA_LA_FORMA_MDI()

```

```

With FORMAS_CONFIGURADAS(0)
    MDI_MONITOR.Top = .Top // MDI_MONITOR.Left = .Left // MDI_MONITOR.WIDTH =
    .WIDTH // MDI_MONITOR.HEIGHT = .HEIGHT
End With
End Sub

```

Anexo 8: Preparación y dimensión de formas.

- Se invoca la siguiente subrutina para determinar cuántas formas se utilizarán, colocando cada una en su posición y tamaño.

```

Private Sub prepara_dimension_y_posicion_DE_LAS_FORMAS()
' PREPARA LA FORMA TOTAL (PARA EL MDI)
Call PREPARA_LA_FORMA_MDI
' AHORA "SET" TODAS LAS DEMAS FORMAS Y LES PONE SUS DIMENSIONES Y POSICION
Dim IND_FORMA_CONFIG As Integer
For IND_FORMA_CONFIG = 1 To CUANTOS_MONITORES
    Call PREPARA_LAs_FORMAS_DEL_MONITOR(IND_FORMA_CONFIG)
Next IND_FORMA_CONFIG
End Sub

```

- Para colocar la posición de una forma se invoca la siguiente rutina, indicando el número de forma (para el caso de 2 monitores, el índice es de 1 a 2, ésta subrutina coloca una forma en su posición correcta):

```

Private Sub PREPARA_LAs_FORMAS_DEL_MONITOR(ind_F_CONFIG As Integer)
Dim IND_MONI As Integer
IND_MONI = ind_F_CONFIG - 1
Dim altura_monitor As Integer, ancho_monitor As Integer, TOP_MONITOR As Integer,
Dim LEFT_MONITOR As Integer
OTRO_MONITOR(IND_MONI).ESTA_DISPONIBLE = True
OTRO_MONITOR(IND_MONI).ES_DE_OTRA_COMPUTADORA = False
With FORMAS_CONFIGURADAS(ind_F_CONFIG)
    altura_monitor = .HEIGHT
    ancho_monitor = .WIDTH
    TOP_MONITOR = .Top
    LEFT_MONITOR = .Left
End With
For ind_forma = 0 To 2
Set LA_FORMA_SHOW(IND_MONI, ind_forma) = New FSHOW
//Se omitió código, es usado para otros fines
Set la_forma_PDF_show(IND_MONI, ind_forma) = New fParaPDF
With la_forma_PDF_show(IND_MONI, ind_forma)
    .soy_de_monitor = IND_MONI
    .soy_su_forma = ind_forma
    .me_llamo_instancia_sucesion_numero = 0
    .Top = TOP_MONITOR
    .HEIGHT = altura_monitor
    .WIDTH = ancho_monitor

```

```

If ind_forma > 0 Then .WIDTH = ancho_monitor / 2
.Left = LEFT_MONITOR
If ind_forma = 2 Then .Left = .Left + ancho_monitor / 2
' de una vez le damos el tamaño al pdf top y left quedan iguales
.AcroPDF1.HEIGHT = .HEIGHT - 50
.AcroPDF1.WIDTH = .WIDTH - 1250
.PICBACK.Left = .WIDTH - 950 ' el picback mide como 600 de ancho
.PICBACK.Top = 90
End With

Next ind_forma
End Sub

```

Anexo 9: Preparación y dimensión de los objetos que contienen la forma base.

- Para posicionar correctamente los objetos, como PictureBox, mediaplayer, richtextbox, AcroPdf, las etiquetas, los botones del pizarrón, etc., se determina cual forma dimensionará sus objetos, se hace un cálculo para que siempre los objetos abarquen toda la forma y siempre estén centrados.

```

Public Sub REDIMENSIONA_LOS_OBJETOS_EN_TODAS_LAS_FORMAS()
Dim ALTURA_ETIQUETA_DESCRIPCION As Integer, ALTURA_HASTA_dESCRIPCION As Integer
ALTURA_ETIQUETA_DESCRIPCION = FSHOW.lbl_descripcion_del_item.HEIGHT
Dim altura As Integer, Ancho As Integer
For IND_MONI = 0 To CUANTOS_MONITORES - 1
For ind_forma = 0 To 2
With LA_FORMA_SHOW(IND_MONI, ind_forma)
altura = .HEIGHT
Ancho = .WIDTH
ALTURA_HASTA_dESCRIPCION = altura - ALTURA_ETIQUETA_DESCRIPCION
.lbl_descripcion_del_item.Top = ALTURA_HASTA_dESCRIPCION
.lbl_descripcion_del_item.Left = (.WIDTH - .lbl_descripcion_del_item.WIDTH) / 2
With .el_rectangulo
.Top = 10 // .WIDTH = Ancho // .HEIGHT = altura // .Left = 10
End With
With .rtfrtf
.HEIGHT = ALTURA_HASTA_dESCRIPCION - 500
.WIDTH = Ancho - 500 // .Top = 200 // .Left = 200
End With
With .RTBparaTXT
.HEIGHT = ALTURA_HASTA_dESCRIPCION - 500
.WIDTH = Ancho - 500 // .Top = 200 // .Left = 200
End With
With .PictureIMG
.HEIGHT = ALTURA_HASTA_dESCRIPCION - 500
.WIDTH = Ancho - 500 // .Top = 200 // .Left = 200

```

```

End With
With .Picture1
    .HEIGHT = ALTURA_HASTA_dESCRIPCION - 500
    .WIDTH = Ancho - 500 // .Top = 200 // .Left = 200
End With

'----- TODO DEL PIZARRON-----
With .Pic_PIZARRON
    .HEIGHT = ALTURA_HASTA_dESCRIPCION - 500
    .WIDTH = Ancho - 500 // .Top = 200 .Left = 200
End With
With .FRM_PAINT
    .Top = 1100 // .Left = Ancho - .WIDTH - 5500
End With
With .frm_edit
    .Top = 100 // .Left = Ancho - .WIDTH - 400
End With
With .frm_tools
    .Top = 100 // .Left = Ancho - .WIDTH - 5100
End With
With .FRM_WRITE
    .Top = 1100 // .Left = Ancho - .WIDTH - 5800
End With
'-----
With .OLE1
    .HEIGHT = ALTURA_HASTA_dESCRIPCION - 500
    .WIDTH = Ancho - 500 // .Top = 1000 // .Left = 800
End With
End With

Next ind_forma
Next IND_MONI
End Sub

```

Anexo 10: Oculta todos los objetos y despliega las formas hijas.

Para ocultar todos los objetos, se coloca la variable visible de cada objeto en false, hay que invocar la subrutina:

HACE_INVISIBLES_TODOS_LOS_OBJETOS y pasarle como parámetro la forma. Se usa la propiedad visible de cada objeto y se pone en falso.

```

Sub HACE_INVISIBLES_TODOS_LOS_OBJETOS(f As Form)
    With f
        .Pic_PIZARRON.Visible = False

```

```

.frm_tools.Visible = False
.frm_edit.Visible = False
.OLE1.Visible = False
.Picture1.Visible = False
.PictureIMG.Visible = False
.RTBparaTXT.Visible = False
.rtf.frf.Visible = False
.WMP1.Visible = False
.MMControl1.Visible = False
.RTBparaTXT.Visible = False
.RTFTEMPORAL.Visible = False
.lbl_descripcion_del_item.Visible = False

```

End With

End Sub

- Para desplegar todas las formas hay que invocar la siguiente subrutina y recorrer una por una las formas con el método show.

```

Sub DESPLIEGA_LAS_FORMAS_PARA_MOSTRAR_TAMANO()
  For IND_MONI = 0 To CUANTOS_MONITORES - 1
    For ind_forma = 0 To 2
      With LA_FORMA_SHOW(IND_MONI, ind_forma)
        If IND_MONI = 0 And ind_forma = 1 Then
          .LBLCONFIRMACION.Visible = True
        Else
          .LBLCONFIRMACION.Visible = False
        End If
        If ind_forma > 0 Then .Show
      End With
    Next ind_forma
  Next IND_MONI
End Sub

```

Anexo 11: PCI, "Interconexión de Componentes Periféricos"

Un Peripheral Component Interconnect (PCI, "Interconexión de Componentes Periféricos") consiste en un bus de ordenador estándar para conectar dispositivos periféricos directamente a su placa base. Estos dispositivos pueden ser circuitos integrados ajustados en ésta (los llamados "dispositivos planares" en la especificación PCI) o tarjetas de expansión que se ajustan en conectores.

El bus PCI permite configuración dinámica de un dispositivo periférico. En el tiempo de arranque del sistema, las tarjetas PCI y el BIOS interactúan y negocian los recursos solicitados por la tarjeta PCI. La especificación PCI cubre el tamaño físico del bus, características eléctricas, cronómetro del bus y sus protocolos. El bus PCI tiene tecnología "plug and play". Aparte de esto, el bus PCI proporciona una

descripción detallada de todos los dispositivos PCI conectados a través del espacio de configuración PCI.

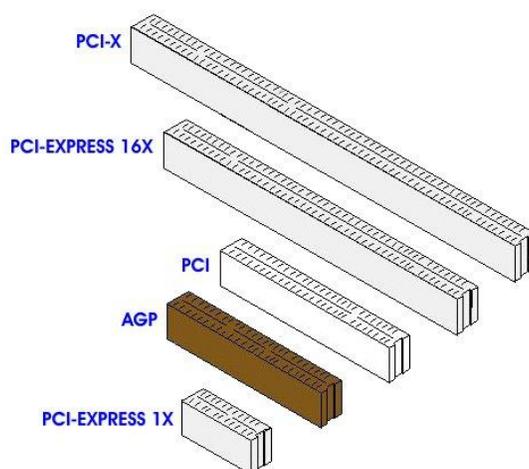


Figura 100. Conector PCI.

Anexo 12: Sonido Surround

El sonido surround, sonido envolvente o sonido 3D, se refiere al uso de múltiples canales de audio para provocar efectos envolventes a la audiencia, ya sea proveniente de una película o de una banda sonora.

La forma más simple de Surround es lo que se conoce como formato estéreo 3/2 y está compuesto por un total de 5 altavoces:

- Canal central. Situado en una posición central con respecto al canal izquierdo y derecho; en el mismo eje se encontraría el *sweet spot*, punto ideal en el que el oyente debería estar colocado.
- Canal izquierdo y canal derecho. A una distancia de 30°, con respecto al eje central que va del canal central al *sweet spot*. Es la misma posición que ocupan en el formato estéreo 2/0, por lo que mantiene la compatibilidad. En algunas ocasiones, es posible que el ángulo de separación sea aún mayor, teniendo como límite máximo los 35°.
- Dos canales traseros. Colocados en un ángulo de 110° con respecto al eje central, son los que permiten hablar de sonido envolvente, sin ellos, sería formato estéreo sin más. Los canales Surround no presentan demasiadas exigencias en cuanto a su colocación, pues la percepción humana no es muy buena cuando se enfrenta a sonidos procedentes desde la parte posterior.

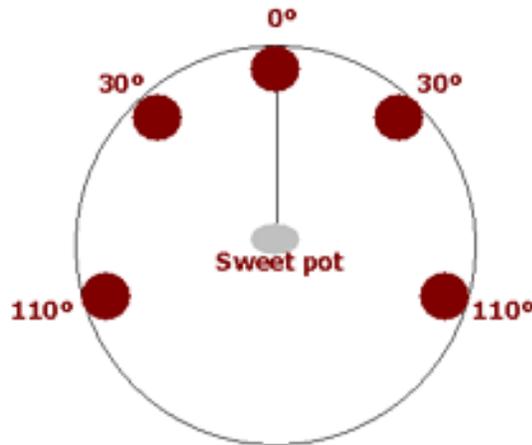


Figura 101. Sonido Surround.

Para poder crear correctamente la imagen sonora, el oído debe recibir la información procedente de cada uno de los 5 canales al mismo tiempo, por ello, los altavoces deben estar situados equidistantes con respecto al *sweet spot*. Esta equidistancia se logra considerando el sistema como un círculo imaginario, de forma que el *sweet spot* es el centro de la circunferencia, y los altavoces se ubican todos en el borde de la misma, distanciados del centro una longitud equivalente al radio del círculo.

Cuando esto no es posible, se introducirá un retardo electrónico en aquellos altavoces situados más próximos al oyente, hasta cuadrar el tiempo y que todas las señales lleguen al oído "en fase", es decir, al mismo tiempo.

En el plano teórico, a la hora de colocar los altavoces, habría que tener en cuenta que todos deberían estar situados a la misma altura, preferentemente a nivel del oído. Sin embargo, la presencia de obstáculos, a nivel práctico, hace que a veces se tengan que situar por encima o por debajo de esta altura ideal. Lo que hay que tener en cuenta en ese caso es que la desviación con respecto al eje horizontal, cuya referencia son los oídos del oyente, no sea superior a los 15°; a partir de ese límite, los sonidos son percibidos como procedentes de arriba o abajo.

Anexo 13: Descargar Sketch en la memoria del ARDUINO

Instalar la interface de ARDUINO, una vez instalado, ejecutarlo y configurar algunos parámetros.

- Conectar el ARDUINO con un cable USB.
- Si se está usando una placa ARDUINO Diecimila, hay que asegurarse de que la placa está configurada para alimentarse mediante la conexión USB. La

fuente de alimentación se selecciona con un puente ("jumper"), una pequeña pieza de plástico que encaja en dos de los tres pines situados entre los conectores USB y de alimentación de la placa. Comprobar que el puente esté conectando los dos pines más cercanos al puerto USB de la placa.

- Seleccionar el modelo del ARDUINO: tools/Board y posteriormente seleccionar nombre del ARDUINO, en este caso es el Diecimila, como se muestra a continuación.

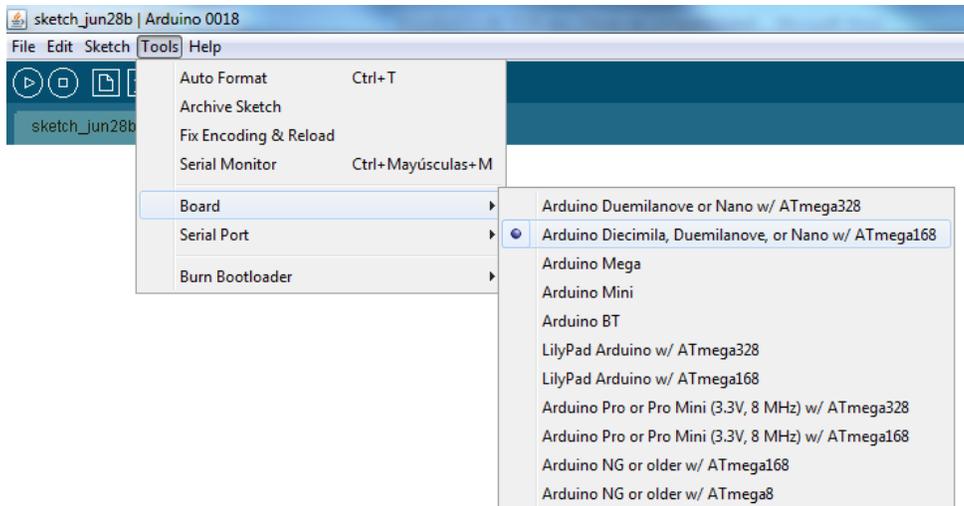


Figura 102. Selección del modelo de ARDUINO.

- Verificar el puerto de comunicación en el que se conectó el ARDUINO, seleccionar tools/SerialPort y el número de puerto (*Explicación Técnica Anexo 14*).

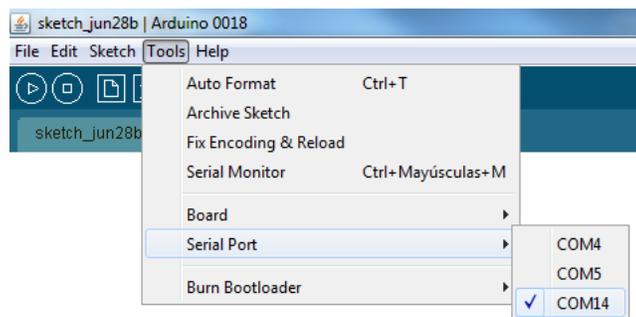


Figura 103. Propiedades del ARDUINO.

- El código que realiza toda la acción en el ARDUINO es el siguiente:

```
//Simple demo Arduino + Visual Basic 6 Serial Comm
int userInput=0; int RELE1=10; int LED1=3; int RELE2=11; int LED2=4; int RELE3=12; int LED3=5;
```

```

void setup()
{
  Serial.begin(9600);
  pinMode(RELE1,OUTPUT);
  pinMode(LED1,OUTPUT);
  pinMode(RELE2,OUTPUT);
  pinMode(LED2,OUTPUT);
  pinMode(RELE3,OUTPUT);
  pinMode(LED3,OUTPUT);
}
void loop()
{
  if(Serial.available())
  {
    int userInput=Serial.read();
    if(userInput=='1') // 1 to set the LED ON
    {
      digitalWrite(RELE1,HIGH);
      digitalWrite(LED1,HIGH);
      //Serial.write("HORELE1LA");
    }
    else if(userInput=='2')
    {
      digitalWrite(RELE2,HIGH);
      digitalWrite(LED2,HIGH);
    }
    else if(userInput=='3')
    {
      digitalWrite(RELE3,HIGH);
      digitalWrite(LED3,HIGH);
    }
    else if(userInput=='4')
    {
      digitalWrite(RELE1,LOW);
      digitalWrite(LED1,LOW);
    }
    else if(userInput=='5')
    {
      digitalWrite(RELE2,LOW);
      digitalWrite(LED2,LOW);
    }
    else if(userInput=='6')
    {
      digitalWrite(RELE3,LOW);
      digitalWrite(LED3,LOW);
    }
    else if(userInput=='7')

```

```

{
  digitalWrite(RELE1,HIGH);
  digitalWrite(LED1,HIGH);
  digitalWrite(RELE2,HIGH);
  digitalWrite(LED2,HIGH);
  digitalWrite(RELE3,HIGH);
  digitalWrite(LED3,HIGH);
}
else
{
  digitalWrite(RELE1,LOW);
  digitalWrite(LED1,LOW);
  digitalWrite(RELE2,LOW);
  digitalWrite(LED2,LOW);
  digitalWrite(RELE3,LOW);
  digitalWrite(LED3,LOW);
}
}
}

```

- Una vez generado el código en la interfaz, se verifica que el programa este correcto, oprimiendo el botón de VERIFY:



Figura 104. Opción Verify del ARDUINO.

Si hay errores en el código, la interfaz desplegará un mensaje de error.

- Para grabar el programa en la Memoria del ARDUINO, seleccionar la opción de UPLOAD, este proceso tarda varios segundos:



Figura 105. Opción Upload del ARDUINO.

Si es correcto desplegará la siguiente leyenda:

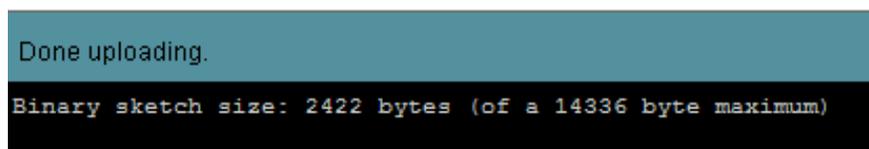


Figura 106. Mensaje de fin de proceso del ARDUINO.

En caso contrario desplegara un mensaje de error.

- Una vez hecho los pasos anteriores, el ARDUINO está listo.

Anexo 14: Detección del número de puerto del ARDUINO.

- En panel de control de Windows, seleccionar “Sistema y seguridad”.

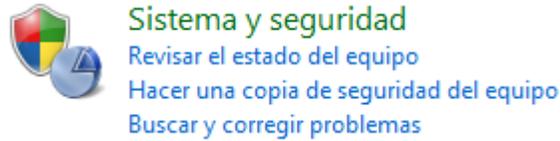


Figura 107. Sección del Control de Windows.

- En Sistema y seguridad, seleccionar “Administración de Dispositivos”.

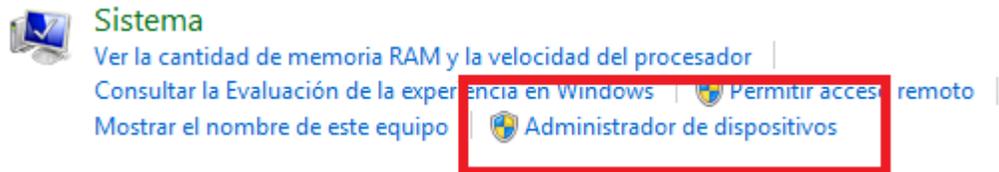


Figura 108. Sistema y seguridad de Windows.

- Ubicar en Puertos (COM Y LPT), el número del puerto COM del USB Serial (en este caso 14).

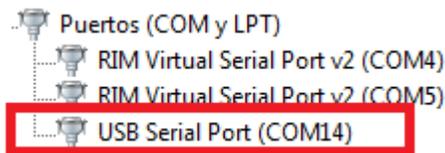


Figura 109. Lista de puertos conectados de Windows.

Anexo 15: Agregar el componente MSComm.

- En visual Basic seleccionar “Proyecto”, “Componentes” y buscar Microsoft Comm Control 6.0.

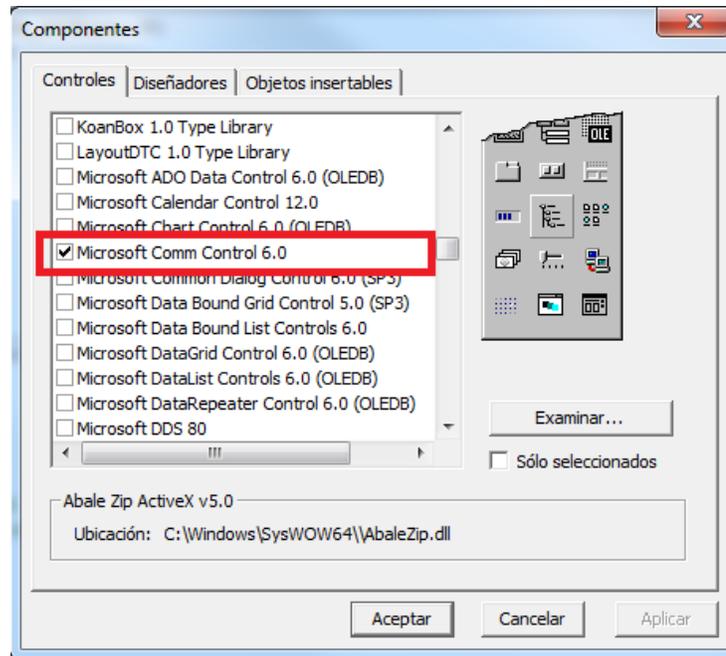


Figura 110. Componentes de Visual Basic.

- Posteriormente, en la barra de herramientas se agrega el siguiente componente:



- Para poder utilizarlo, solo hay que arrastrarlo a una forma y darle algunos parámetros.

Anexo 16: Apagar y prender pares de bocinas

Se creó un modulo llamado **mod_LAS_Bocinas**, que contiene todas subrutinas que serán usadas para las bocinas y el ARDUINO. Se creó una forma llamada **fbocinas** con el componente MSComm y un flex para asignar por forma los pares de bocinas.

- Hay una subrutina llamada LO_DE_LAS_BOCINAS, que se encarga de inicializar el arreglo de bocinas por forma en 0. Y hace la configuración del ARDUINO y muestra la forma de asignación de bocinas.

```
Sub LO_DE_LAS_BOCINAS()
    Dim mon As Integer, frm As Integer, par As Integer
    For mon = 0 To 3
        For frm = 0 To 2
            bocinas_por_Forma(mon, frm, 1) = 1
        
```

```

        bocinas_por_Forma(mon, frm, 2) = 0
        bocinas_por_Forma(mon, frm, 3) = 0
    Next frm
Next mon
tiene_arduino_instalado = inicializa_arduino(14)
If Not tiene_arduino_instalado Then
    cuantas_bocinas_usa = 1
    Exit Sub
End If
fbocinas.Show vbModal
End Sub

```

- Mostrando la pantalla **fbocinas**, se asigna un par de bocinas a una forma, utilizando el arreglo **bocinas_por_Forma(mon, frm, 1)** y se emplea la siguiente subrutina llamada **msfbocinasyformas**, para el cálculo del valor que asignará al arreglo:

```

Private Sub msfbocinasyformas_Click()
    Dim R As Integer, c As Integer
    With Me.msfbocinasyformas
        R = .Row
        c = .Col
        mnt = (R - 1) \ 3 * 1
        frm = R - 1 - 3 * mnt
        boc = c - 1
        bocinas_por_Forma(mnt, frm, boc) = 1 - bocinas_por_Forma(mnt, frm, boc)
        .TextMatrix(R, c) = ley_sino(bocinas_por_Forma(mnt, frm, boc))
        If UCase(Trim(.TextMatrix(R, c))) = "SI" Then
            .CellBackColor = azul
            .CellForeColor = blanco
            .CellFontBold = True
        Else
            .CellBackColor = blanco
            .CellForeColor = &H80000012
            .CellFontBold = False
        End If
    End With
End Sub

```

- Una vez creado el arreglo anterior, para apagar o prender el par de bocinas se aplica el siguiente código, haciendo el cálculo de la bocina que se prenderá, (el código del extent de un ítem de audio o video es 4):

```

If codigo_de_extent_DEL_ITEM = 4 Then
    For boc = 1 To cuantas_bocinas
        If bocinas_por_Forma(cual_moni, cual_forma, boc) = 1 Then
            enciende_bocina (boc)
        End If
    Next boc
End If

```

```
Else
  apaga_bocinas (boc + 3)
End If
Next boc
End If
```

Anexo 17: Compartir una carpeta

Windows 7, al igual que las anteriores versiones de Windows, permite compartir directorios con los equipos que se encuentran en la misma red de área local. Se pueden compartir o hacer una carpeta pública.

Para compartir una carpeta, tenemos que seleccionarla dar click derecho y seleccionar propiedades. Saldrá una ventana, seleccionar la pestaña “Compartir” y posteriormente en “Uso compartido avanzado”.

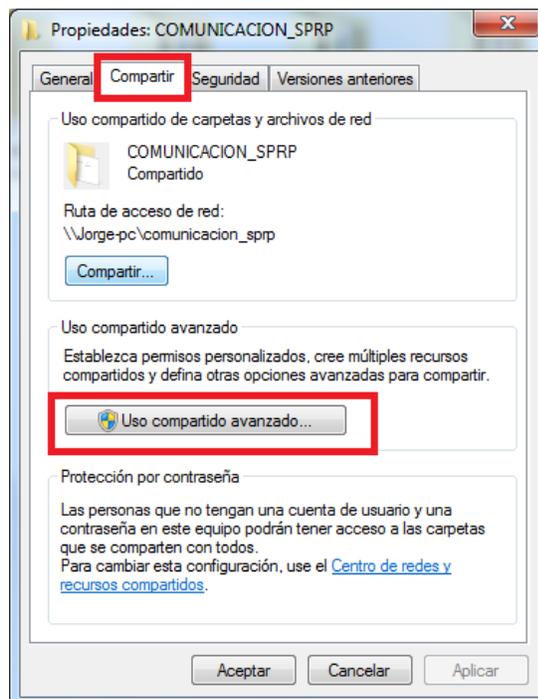


Figura 111. Propiedades de una carpeta en Windows.

- Saldrá una pantalla, seleccionar la opción “Compartir esta carpeta” y seleccionar “Permisos”.

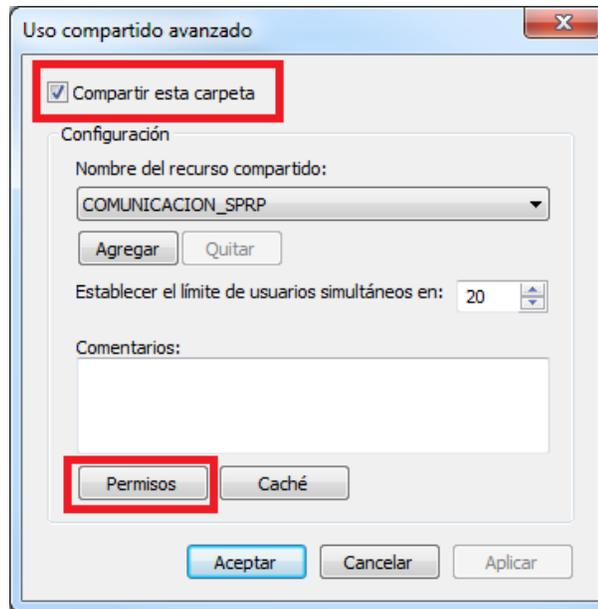


Figura 112. Opciones para compartir carpeta en Windows.

- Una vez seleccionado “Permisos”, aparecerá una pantalla, en la parte de “permitir”, seleccionar “Control total”, automáticamente se activarán las opciones de “Cambiar” y “Leer”. Finalmente seleccionar “Aceptar” en todas las ventanas hasta que no aparezca ninguna.

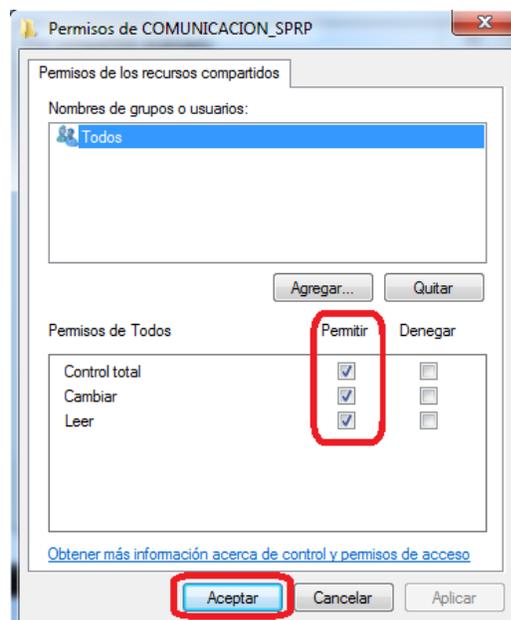


Figura 113. Opciones de permisos para una carpeta de Windows.

Anexo 18: Configuración de uso compartido avanzado

- Para cambiar los permisos del uso compartido de carpetas, necesitamos hacer algunas configuraciones básicas, ir a Panel de control\Redes e Internet\Centro de redes y recursos compartidos\Configuración de uso compartido avanzado, aparecerá un recuadro, seleccionar la opción “Público”.

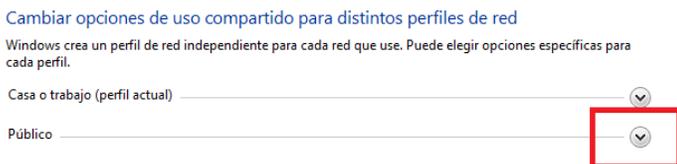


Figura 114. Opciones de uso compartido

- Activar o desactivar algunas opciones y guardar los cambios:

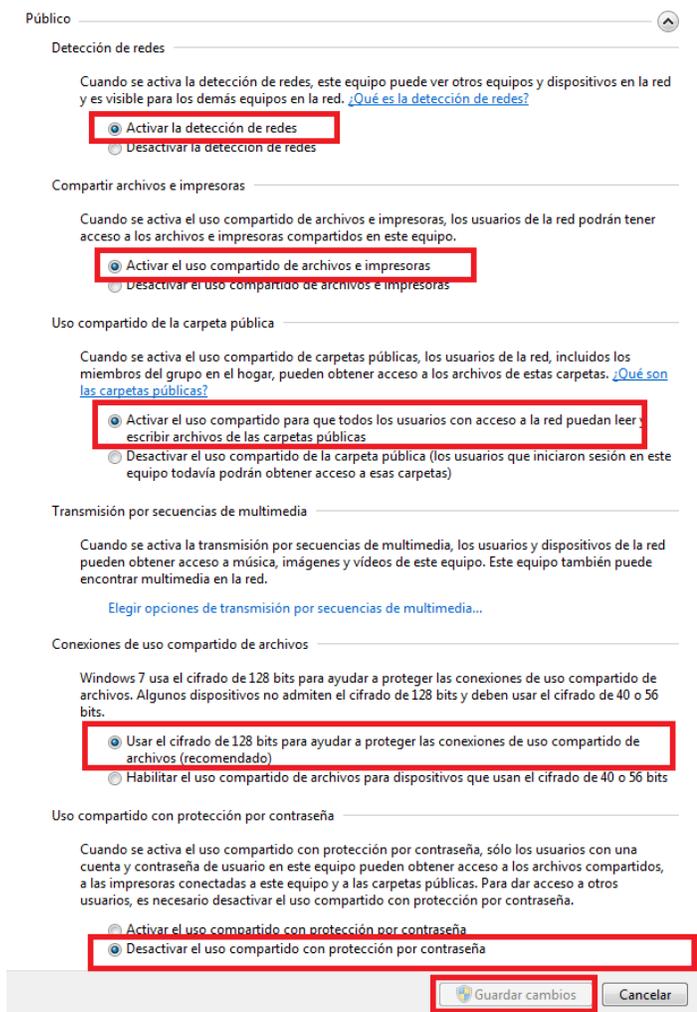


Figura 115. Opciones de carpetas públicas

Anexo 19: Averiguar el IP de red de una Computadora.

- Para averiguar la IP del equipo, abrir la ventana Símbolo del sistema, ubicada en inicio, seleccionar todos los programas, clic en todos los programas y seleccionar la opción “Símbolo del sistema”.

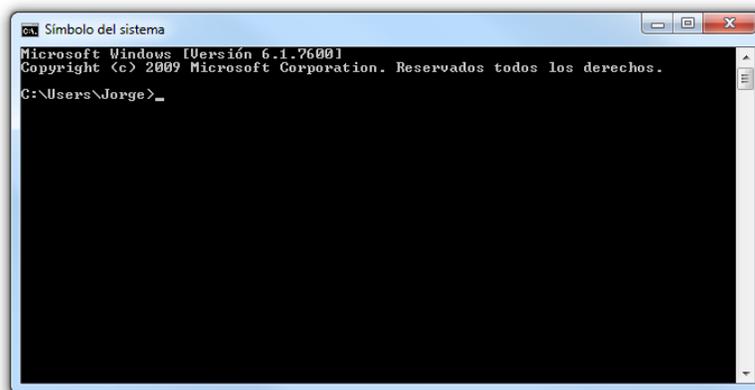


Figura 116. Símbolo del sistema

- Teclar ipconfig y darle enter, posteriormente aparecerá una etiqueta con la dirección IP:



Figura 117. Ejemplo de descripción IP

Anexo 20: Averiguar el nombre de equipo en la red.

- Para averiguar en nombre de equipo, dar Clic derecho en mi EQUIPO o PC y seleccionar propiedades.
- En la propiedades de la computadora aparecerá una etiqueta con el nombre de equipo:

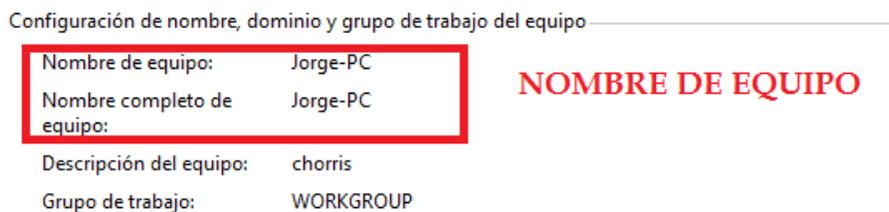


Figura 118. Ejemplo de nombre de equipo.

Anexo 21: Agregar el componente Winsock

- En visual Basic, seleccionar Proyecto, Componentes y buscar Microsoft Winsock Control 6.0.

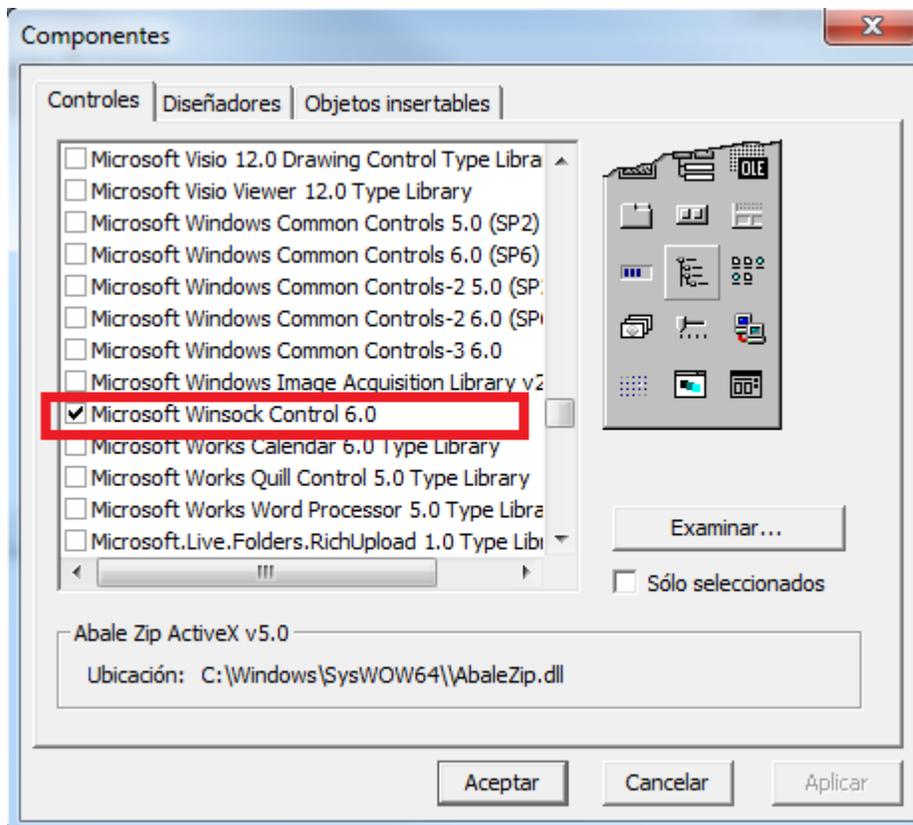


Figura 119. Componentes de Visual Basic.

- Posteriormente en la barra de herramientas se agrega el siguiente componente:



- Para poder utilizarlo, solo hay que arrastrarlo a una forma y darle algunos parámetros.

Anexo 22: Ejemplo de una estructura TYPE, para los mensajes de Entrada/Salida

- Existen varias estructuras de tipo TYPE. Por ejemplo, cuando se inicia la conexión, el tipo mensaje de regreso que manda la otra computadora

SPRPShow-Other(cliente/otra), a la computadora principal **SPRPBuildShow** (servidor/otra) es 22, se declara de la siguiente forma:

```
TYPE Tcadena_saludo_inicial_regreso
    long_esta_cadena As Integer
    QUE_ES As Integer ' en este caso vale 22
    Llego_bien As Integer
    CUANTOS_MONITORES As Integer
End TYPE
Private cadena_saludo_inicial_regreso As Tcadena_saludo_inicial_regreso
```

- Para poder llenar los datos de la estructura en la otra computadora **SPRPShow-Other**(cliente/otra), se le asigna el valor que le corresponde, de la siguiente forma:

```
With cadena_saludo_inicial_regreso
    .QUE_ES = 22 → Tipo de mensaje
    .Llego_bien = 1
    .CUANTOS_MONITORES = CUANTOS_MONITORES
End With
```

- Posteriormente se genera una cadena, concatenando cada uno de los valores y asignándole un tamaño a la parte concatenada. Al principio de la cadena, se le concatena el tamaño del mensaje y se manda el mensaje a la computadora principal, la cadena armada es caam:

```
With cadena_saludo_inicial_regreso
caam=Format(.QUE_ES,"00" & Format(.Llego_bien,"00" & Format(.CUANTOS_MONITORES,"00")
long_cadena_enviada = Len(caam)
caam = Format(long_cadena_enviada, "000000") & caam
Call MANDALE_MENSAJE(caam)
End With
```

- Una vez que llega el mensaje a la computadora del presentador **SPRPBuildShow** (servidor/otra), hay que descomponer el mensaje, sabiendo las longitudes se le indican las posiciones:

```
Dim posic As Integer, caam As String
caam = c
With cadena_saludo_inicial_regreso
    posic = 1
    .long_esta_cadena = Mid(caam, posic, 6)
    If Len(caam) <> .long_esta_cadena + 6 Then
        MsgBox "llego mal la cadena, mándala de nuevo"
        Exit Sub
    End If
    posic = posic + 6 /// .QUE_ES = Mid(caam, posic, 2)
    posic = posic + 2 .Llego_bien = Mid(caam, posic, 2)
    posic = posic + 2 . CUANTOS_MONITORES = Mid(caam, posic, 2)
```

End With

- Cuando se descompone una cadena, se verifica que las longitudes sean las mismas; si son correctas descompone la cadena; de lo contrario manda mensaje de error.

Anexo 23: Guardar el contenido de un PictureBox como archivo de imagen.

- Se definió una subrutina llamada “crea_un_archivo_con_el_picture”, que es la que se encarga de crear un archivo con el contenido de un picture box y almacenarlo como imagen.

```
Private Sub crea_un_archivo_con_el_picture()  
    If Len(Dir(nombre_archivo_picture_del_pizarron)) > 0 Then Kill (nombre_archivo_picture_del_pizarron)  
    Dim La_Imagen As IPictureDisp  
    Set La_Imagen = FF.Pic_PIZARRON.Image  
    SavePicture La_Imagen, nombre_archivo_picture_del_pizarron  
    'Folder.Temp.UBOOK.FILES  
    Set La_Imagen = Nothing  
End Sub
```

- Se declaro una variable La_imagen de tipo IPictureDisp, definiendo esta variable de este tipo y asignado el contenido (Imagen) del PictureBox que se desea guardar y con el método SavePicture se guarda lo que tiene la variable La_imagen a la ruta que se le especifique.

Anexo 24: Acciones de navegación

- Se definió una subrutina llamada “muestra_otro_pic_del_stack”, recibe como parámetro la acción del navegador.

```
Private Sub muestra_otro_pic_del_stack(naveg As String)  
    If Not armar_stack_de_pictures Then Exit Sub  
    Dim cual_muestro As Integer  
    Select Case naveg  
        Case "ANT"  
            izq = izq - 1  
            FF.Pic_PIZARRON.Picture = Nothing  
            FF.Pic_PIZARRON.Picture = stack_de_pics(izq).Picture  
            If izq < der Then FF.cmd_adelante_stack.Visible = True  
                FF.cmd_pic_final_stack.Visible = True  
            If izq = 1 Then  
                FF.cmd_atras_stack.Visible = False  
                FF.cmd_pic_inicio_stack.Visible = False  
            End If  
        Case "PRIM"  
            izq = 1
```

```

FF.Pic_PIZARRON.Picture = Nothing
FF.Pic_PIZARRON.Picture = stack_de_pics(izq).Picture
If izq < der Then FF.cmd_adelante_stack.Visible = True
FF.cmd_atras_stack.Visible = False
FF.cmd_pic_inicio_stack.Visible = False
FF.cmd_pic_final_stack.Visible = True
Case "SGTE"
    izq = 1 + izq
    If izq > der Then izq = der
    FF.cmd_atras_stack.Visible = True
    FF.Pic_PIZARRON.Picture = Nothing
    FF.Pic_PIZARRON.Picture = stack_de_pics(izq).Picture
    FF.cmd_pic_inicio_stack.Visible = True
    If izq = der Then
        FF.cmd_adelante_stack.Visible = False
        FF.cmd_pic_final_stack.Visible = False
    End If
Case "ULT"
    izq = der
    FF.cmd_atras_stack.Visible = True
    FF.Pic_PIZARRON.Picture = Nothing
    FF.Pic_PIZARRON.Picture = stack_de_pics(izq).Picture
    FF.cmd_adelante_stack.Visible = False
    FF.cmd_pic_final_stack.Visible = False
    FF.cmd_pic_inicio_stack.Visible = True
End Select
End Sub

```

- Esta rutina se posiciona en el Stack según lo que se requiera, atrás, adelante, inicio, final. Y carga el contenido del Stack al pizarrón.

Anexo 25: Obtiene ruta del archivo de una imagen prediseñada.

- Para obtener la ruta del archivo de una imagen prediseñada se diseñó un función que devuelve el path completo de una imagen prediseñada:

```

Public Function dame_sigte_objeto_predisenado() As String
' devuelve falso si no hubo
dame_sigte_objeto_predisenado = "YA NO HAY OBJETOS PREDIS"
ya_encontro_el_next_element = 0
ind_elemento_actual_del_arreglo = ind_elemento_actual_del_arreglo + 1
While ya_encontro_el_next_element <> 1 And ind_elemento_actual_del_arreglo <= NUMERO_DE_ELEMENTOS
If arr_elementos(ind_elemento_actual_del_arreglo).naturaleza_del_elemento = 1 Then
    If consigue_el_archivo Then
        dame_sigte_objeto_predisenado = lo_del_archivo_del_item.nombre_completo
    Exit Function
End If
End While

```

```
If ind_elemento_actual_del_arreglo > NUMERO_DE_ELEMENTOS Then se_acabo_esta_sucesion
End Function
```

Anexo 26: Determinar la posición de un objeto prediseñado

- Cuando se instancia un pizarrón, el constructor inicializa algunas parámetros y obtiene las reglas para insertar una imagen, estas reglas las contiene el pizarrón.

```
hil_desde = 1
hil_hasta = divisionesX
col_desde = 1
col_hasta = divisionesY
If .HILERA_UNICA > 0 Then
    hil_desde = .HILERA_UNICA
    hil_hasta = .HILERA_UNICA
End If
If .COLUMNA_UNICA > 0 Then
    col_desde = .COLUMNA_UNICA
    col_hasta = .COLUMNA_UNICA
End If
```

- Se invoca la rutina “determina_donde_pone_objeto_prediseñado” para calcular la hilera y columna, así poder obtener los datos de esa posición en particular, simplemente en posicionarse en el arreglo CUADRICULA” que contiene todos los datos del reticulado.

```
Private Sub determina_donde_pone_objeto_prediseñado()
    borrar_lo_que_tenia = False
    col_calculada = col_calculada + 1
    If col_calculada > col_hasta Then
        col_calculada = col_desde
        hil_calculada = hil_calculada + 1
    End If
    If hil_calculada > hil_hasta Then
        hil_calculada = hil_desde
        borrar_lo_que_tenia = True
    End If
End Sub
```