

Desarrollo de entornos para la creación y reproducción de Docusquemas

Development of Environments for the Creation and Display of Docuschemas.

Jesús Caudeli Tomé y Dr. José Jesús García Rueda
Universidad Carlos III de Madrid (España – UE)
jaudeli@inv.it.uc3m.es, rueda@it.uc3m.es
<http://bach.gast.it.uc3m.es/~jcaudeli/Proyecto.htm>

Resumen: El Docusquema representa un nuevo modelo para construir presentaciones multimedia destinadas al aprendizaje basándose en los principios del Aprendizaje Receptivo Significativo. Este modelo otorga mayor protagonismo a la imagen y el sonido frente al texto escrito tradicional. En este proyecto se ha definido un lenguaje para describir los Docusquemas utilizando XML, y se han desarrollado dos herramientas software utilizando el lenguaje Java. La función de la primera de ellas es generar y editar Docusquemas, mientras que la segunda es un applet que permite reproducirlos a través de Internet, integrando las presentaciones en páginas web.

Abstract: Docuschemas are a new model for the construction of multimedia, learning oriented presentations based on the Meaningful Receptive Learning. This model gives a more important role to image and sound, considering traditional written text a secondary option. In this project a language for the definition of Docuschemas has been defined using XML, and two software tools have been developed using the Java language. The first one's aim is to generate and edit Docuschemas, while the second is an a Java applet to display them in a Web browser, integrating the multimedia presentations into Web pages.

Palabras clave: Multimedia, Aprendizaje Receptivo Significativo,

Docuquemmas.

Key Words: Multimedia, Meaningful Receptive Learning, Docuschemas.

1. Introducción

La complejidad de los procesos de adquisición de conocimiento provoca que las iniciativas para modernizar la educación a través de la informática y las telecomunicaciones sean de índole muy diversa, teniendo que abordar simultáneamente varios problemas distintos pero interrelacionados. Éstos pueden ser, entre otros: la búsqueda de contenidos y la adecuación de éstos a los nuevos medios, la presentación al alumno de dichos contenidos [4], la conexión entre las diferentes unidades didácticas, el almacenamiento y difusión de los cursos, el seguimiento de los avances logrados por el alumno, etc.

Centrándonos en el problema de la presentación de contenidos multimedia, podemos dividirlo en dos apartados. Por un lado, se deben estudiar los procesos perceptivos y cognitivos del cerebro, y hacer uso de la información obtenida para presentar las ideas del modo más eficiente posible, de forma que se facilite lo máximo posible al alumno la adquisición de conocimientos y su memorización. Nosotros elegimos la teoría del Aprendizaje Receptivo Significativo desarrollada por Ausubel [1, 2] como punto de partida en el que basar nuestro modelo de presentación multimedia denominado Docuquema.

Por otra parte, una vez que se ha decidido cómo debería ser la presentación de la información a través del ordenador, debe estudiarse su viabilidad tecnológica. Debemos centrarnos en evaluar las diferentes soluciones tecnológicas existentes que nos permiten llevar a cabo la implementación del sistema buscado, y elegir aquella (o la combinación de varias) que más se adecue a nuestras necesidades. Una vez realizada la elección tecnológica, habrá llegado el momento de ponerse manos a la obra y construir el sistema con las herramientas adecuadas.

En este trabajo se ha abordado el segundo de los problemas descritos. Partimos de un modelo de presentación multimedia: el **Docusquema** [5], cuyos principios han sido publicados con anterioridad [6, 7, 8]. Éste modelo surge tras el análisis de algunos procesos cognitivos relacionados con el aprendizaje. Se trata de un modelo que desarrolla una teoría acerca de cómo debería ser una presentación multimedia que facilitara el aprendizaje y la construcción de un sistema de conocimiento en el alumno. El objetivo de este proyecto ha sido llevar a la práctica el modelo de presentación del Docusquema, construyendo las herramientas precisas para que el educador pueda crear presentaciones basadas en esta teoría y almacenarlas en un formato adecuado. También se proporcionará la aplicación que reproduzca las presentaciones en el ordenador del alumno, completando así el conjunto de soluciones necesarias para implementar el modelo del Docusquema.

2. Contexto tecnológico del proyecto

Estamos, o mejor, continuamos estando, en una continua explosión de la educación a través de Internet. En los últimos años han surgido multitud de cursos y títulos cuya docencia se imparte utilizando la red de redes como medio de transmisión. El número de alternativas resulta abrumador, así como la cantidad de tecnologías distintas que se han utilizado para construir estos cursos. Desde las clásicas páginas basadas en texto combinado con imágenes hasta animaciones generadas con Macromedia Flash [12, 13], pasando por cursos generados mediante herramientas de autor tipo Director, sitios web dinámicos que utilizan JavaScript, o applets de Java que permiten mayor complejidad en las interfaces. Y esto sólo por citar algunos recursos tecnológicos.

En la mayoría de los casos, el contenido multimedia de los cursos se limita para evitar largas descargas, dado el escaso ancho de banda de las conexiones domésticas. Sin embargo, se está produciendo un fuerte crecimiento en la contratación de líneas ADSL, acompañada del crecimiento moderado de otras tecnologías como el cable, Internet por satélite (en España, ofrece conexiones de banda ancha con retorno por satélite en lugar de la tradicional línea telefónica), LMDS, e incluso las conexiones a través de la red eléctrica (PLC – Power Line Communications) Este panorama permite

augurar un futuro prometedor en cuanto al ancho de banda medio disponible en los hogares, lo que permitirá que el contenido multimedia de los cursos se incremente para enriquecerlos.

Entonces, si el ancho de banda está aumentando y esto supondrá un incremento del empleo del multimedia en la teleeducación, se hacen necesarios modelos y herramientas que permitan crear presentaciones multimedia pedagógicamente efectivas.

3. Diseño y arquitectura de las aplicaciones

Tras un minucioso proceso de análisis y evaluación de distintas tecnologías aplicables en la realización de presentaciones multimedia [3, 7, 8], seleccionamos el lenguaje de programación de propósito general Java [15] por ser la que mejor se adaptaba a los requisitos de nuestro proyecto, por motivos principalmente de versatilidad y capacidades multiplataforma. Pero Java debía completarse con un modo eficiente de describir los Docusquemas que resultara sencillo de manipular, editar e intercambiar entre máquinas, llegando a la conclusión de que la tecnología que mejor cumple este propósito es XML [9]. Gracias a XML podemos definir nuestro propio lenguaje de marcas. Este lenguaje estará adaptado a las necesidades del Docusquema, y contendrá toda la información necesaria para describir una presentación que se ajuste a este modelo. De la combinación de las palabras *Docusquemas* y *XML* nace el lenguaje *DoX*. La estructura y normas básicas de *DoX* están detalladas en el apartado 4.

En los apartados posteriores se describen las dos herramientas software que completan nuestra solución. La primera de ellas es el reproductor de Docusquemas, al que hemos llamado *Docuplayer*. *Docuplayer* es un applet capaz de reproducir cualquier Docusquema descrito en un fichero que cumpla las normas del lenguaje *DoX*. Debido a su naturaleza, se integra en páginas HTML para que los usuarios puedan ver los Docusquemas desde sus casas sin necesidad de instalar ningún software específico.

Con el lenguaje *DoX* y el reproductor de Docusquemas ya disponemos de un sistema plenamente funcional. Aun así, hemos querido facilitar aún más el proceso de generación de Docusquemas nuevos, de modo que se ha desarrollado el *Docuwizard*.

Como su propio nombre indica, se trata de un asistente que es capaz de tomar la información proporcionada por el usuario y traducirla a un fichero de texto escrito en el lenguaje DoX. Las características de Docuwizard aparecen en el apartado 6, pero podemos adelantar que se trata de una aplicación independiente y no un de applet, como en el caso de Docuplayer, y que permite a los usuarios crear sus Docusquemas a través de una interfaz totalmente gráfica, sin necesidad de tener conocimiento alguno de DoX.

En la **¡Error! No se encuentra el origen de la referencia.** se muestra la relación entre todos los elementos que componen la arquitectura del sistema, y cómo se complementan para ofrecer una solución completa al diseño y reproducción de Docusquemas.

Mediante las herramientas de diseño de imágenes y animaciones, así como programas de edición de vídeo y sonido, se crean los materiales multimedia que habrán de formar parte de las presentaciones. Estos recursos también pueden obtenerse por otros medios, bien sea a través de Internet, CD-ROMs, etc.

Posteriormente, estos materiales multimedia se utilizarán para la composición de las presentaciones a través de Docuwizard, el asistente para la creación y edición de Docusquemas. Éste almacena la descripción de las presentaciones usando documentos XML escritos en el lenguaje DoX, que sirven como formato de intercambio entre esta aplicación y el applet reproductor (Docuplayer)

4. El lenguaje DoX: Docusquemas en XML

El lenguaje DoX representa la solución para la descripción de Docusquemas en un lenguaje de etiquetas sencillo y ajustado a sus necesidades. Durante la etapa de estudio de las tecnologías aplicables a este problema, se pudieron apreciar las ventajas de usar un lenguaje basado en XML como SMIL (“Synchronized Multimedia Integration Language”) [11]. Con la experiencia adquirida decidimos crear nuestro propio lenguaje de marcas personalizado, con el que se pudieran especificar totalmente las características propias de cada Docusquema sin necesidad de introducir toda la complejidad de SMILE.

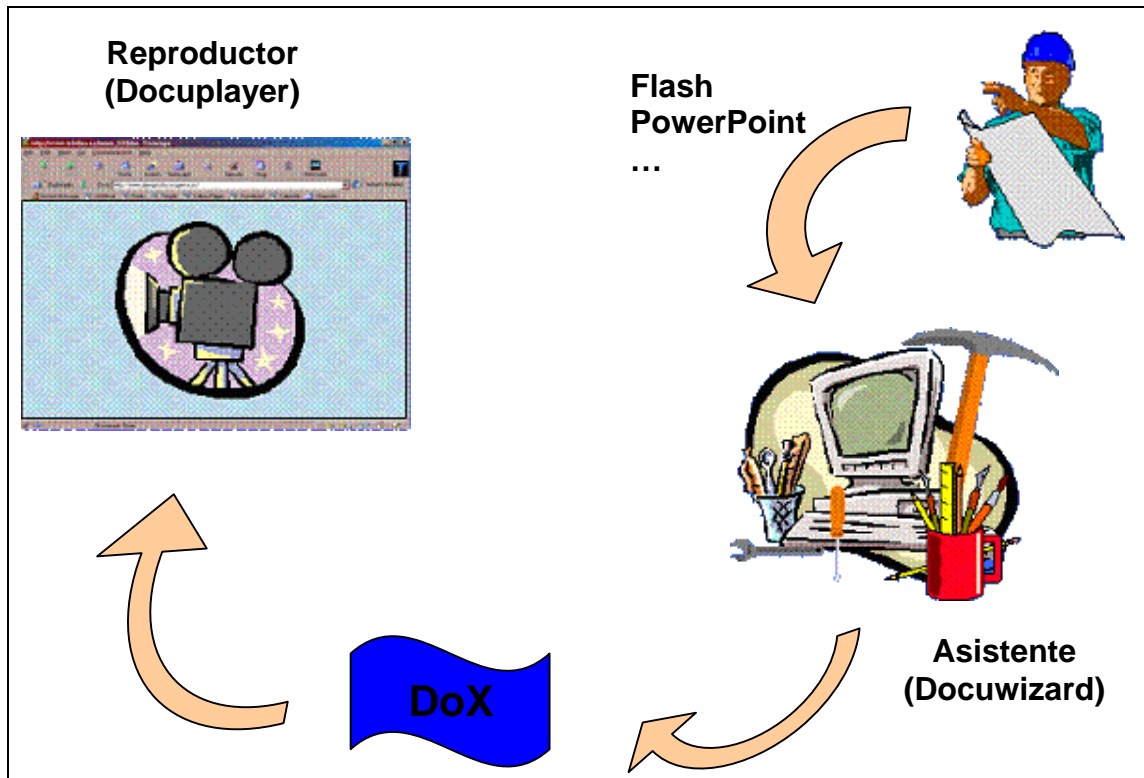


Figura 1: Relación entre los componentes del sistema

La decisión de usar un lenguaje de etiquetas basado en XML presenta notables ventajas: en primer lugar, se trata de una forma sencilla y abierta de realizar una descripción de contenidos. El uso de archivos de texto facilita su edición y visualización, y constituye una solución multiplataforma para distribuir los Docusquemas. Asimismo, la sencillez del lenguaje y su estructura jerárquica facilita el aprendizaje, y permite al creador basarse en ejemplos de otros Docusquemas para construir uno nuevo que se ajuste a sus necesidades. Por último, existen multitud de herramientas que permiten manipular ficheros XML, de modo que resulta sencillo construir un asistente que, partiendo de la información proporcionada por el usuario, genere un nuevo docusquema escrito en el lenguaje DoX.

A la hora de definir nuestro propio lenguaje tuvimos que estudiar qué características de los Docusquemas deberían quedar plasmadas en los ficheros. Partimos de la certeza de que la estructura básica de todas las presentaciones es la misma: un conjunto de conceptos distribuidos en la pantalla como etiquetas de texto, y cada uno de

ellos con un elemento multimedia asociado. Opcionalmente, los conceptos pueden contar con un fragmento de audio que actúe a modo de presentación y como enlace con el concepto anterior. Para cada uno de estos elementos (tanto las etiquetas de los conceptos como los recursos multimedia) es necesario especificar una serie de parámetros como posición, tamaño, volumen de reproducción, etc. También es necesario definir un orden de reproducción de los conceptos, así como añadir enlaces a determinados recursos de información complementaria con los que deberían contar todos los docusquemas, como son la versión escrita de la presentación, una lista de lecturas recomendadas, y otra de nodos que se presupone que deberían haberse visitado con anterioridad (los Docusquemas están concebidos para integrarse formando redes que desarrollen temas completos)

Todas estas características son propias de cada docusquema, y lo distinguen de los demás. Nuestro lenguaje debe abarcar, por tanto, toda la información referente a nombre y aspecto de los conceptos, orden de reproducción, y localización y características de los recursos multimedia. El comportamiento temporal, sin embargo, resulta común a todos los docusquemas. Una vez elaborada la lista ordenada de conceptos, el reproductor simplemente ha de llevar a cabo una reproducción secuencial de los mismos, mostrando los elementos asociados a cada uno de los conceptos en el orden adecuado y con pequeñas pausas entre ellos. Por este motivo, el lenguaje DoX no permite especificar características de sincronización al estilo de los bloques *<PAR>* y *<SEQ>* existentes en SMIL. Mientras que este último lenguaje debía estar abierto a la descripción de todo tipo de presentaciones, cada una de ellas con una estructura distinta de las demás, el nuestro únicamente debe ceñirse a la estructura secuencial de reproducción de los docusquemas. De este modo, no es necesario incluir ninguna información temporal, puesto que el reproductor de Docusquemas conoce la estructura común a todos ellos.

En el momento de definir el lenguaje DoX tuvimos que tomar determinadas decisiones sobre qué variables íbamos a considerar y a qué ámbito pertenecían. Por ejemplo, nos pareció adecuado permitir que cada elemento multimedia llevara especificado su volumen de reproducción, en lugar de utilizar un valor de volumen general para todo el Docusquema. De esta manera es posible utilizar fragmentos de sonido provenientes de distintos medios e igualar sus niveles. De igual modo, se

consideró necesario poder especificar el color de fondo de cada uno de los conceptos por separado, para que éstos pudieran integrarse adecuadamente con la imagen de fondo que se utilizara. Sin embargo, parece lógico que el documento con la versión escrita de la presentación sea único y se especifique en la sección de opciones generales, en lugar de estar compuesto por multitud de documentos pequeños asociados a cada uno de los conceptos. Estas decisiones han marcado la estructura del lenguaje de descripción de Docusquemas.

Según el tipo de recurso multimedia que se desee asociar a cada concepto, DoX ofrece cuatro opciones posibles, excluyentes entre sí:

- *video*: El recurso es un fichero de vídeo (con su propio audio incluido dentro del mismo fichero) En este caso es necesario especificar la posición y dimensiones de la ventana en que se mostrará la imagen, así como el volumen del audio durante la reproducción.
- *audio_imagen*: Se deben especificar por separado la localización del fichero de audio con su volumen correspondiente, y por otra parte la imagen con sus dimensiones. Ambos recursos se reproducirán simultáneamente, de modo que la imagen permanecerá en pantalla todo el tiempo que dure la reproducción del fragmento de audio.
- *audio_texto*: Caso similar al anterior, con la salvedad de que no se debe especificar la localización del archivo gráfico, sino que se puede escribir directamente el texto que aparecerá en la pantalla. Según las directrices del modelo de Docusquema, este texto no se refiere a la versión escrita completa del concepto actual, puesto que ésta se proporcionará en un fichero aparte. La posibilidad que ofrece este recurso es la de hacer aparecer en la pantalla una única frase, o a lo sumo un número pequeño de ellas, que se puedan leer de un vistazo y que complementen al audio que se reproducirá simultáneamente.

- *solo_audio*: Como su nombre indica, es el recurso que se utiliza cuando no deseamos que aparezca ningún elemento visual en la pantalla. Simplemente se escuchará el audio, que será el único responsable de transmitir la información de dicho concepto al estudiante.

5. El reproductor de Docusquemas: Docuplayer

En este proyecto se ha apostado por escribir nuestro propio programa para reproducir las presentaciones multimedia según el formato del Docusquema, lo cual es además una consecuencia de crear nuestro propio lenguaje de descripción (DoX). Partimos de la descripción de una presentación en un fichero de texto escrito en el lenguaje DoX, y necesitamos una aplicación que sea capaz de leer ese documento, extraer la información particular del nodo descrito, e integrarla en el flujo estándar de reproducción de un Docusquema. Esta herramienta, por tanto, no se puede utilizar como un reproductor multimedia general, sino que su comportamiento está adaptado a las normas de funcionamiento de los Docusquemas.

Existen ciertos prerequisites que determinan la arquitectura final de nuestra solución. En primer lugar, es necesario que la aplicación esté programada en un lenguaje que nos permita manipular flujos multimedia con facilidad, admitiendo el mayor número posible de formatos distintos. También sería deseable que nos proporcionara mecanismos para leer documentos escritos en XML y extraer la información almacenada en ellos de forma jerárquica y ordenada. Por último, si consideramos a Internet como la principal vía futura de distribución de contenidos educativos, debemos prestar atención a la integración de nuestra herramienta en la red como entorno habitual de trabajo. Por todos estos motivos se consideró a Java el lenguaje idóneo para escribir nuestra solución. La integración con la red es indiscutible, la gestión de ficheros XML se realiza de forma sencilla y estructurada gracias a la inclusión de DOM en la máquina virtual [18], y la manipulación de recursos multimedia se hace posible añadiendo las bibliotecas del JMF (“Java Media Framework”) como elemento adicional [17].

Existe cierta controversia en la comunidad de programadores a la hora de considerar JMF como solución para desarrollar las capacidades multimedia de Java. El origen de estos problemas reside en el letargo en el que Sun ha sumido a esta librería durante los últimos años. Con la crisis de las telecomunicaciones, la empresa ha atravesado momentos difíciles, tomando la decisión de centrarse en otros frentes que sumieron a JMF en un estado de semiabandono que ha desmotivado a muchos desarrolladores. Sin embargo, durante todo este tiempo Sun ha mantenido en numerosas ocasiones que el desarrollo de JMF no se ha abandonado totalmente, y que tiene proyectos de crecimiento futuro. De hecho, aunque muy poco a poco, en los últimos meses han aparecido revisiones de las bibliotecas para solucionar pequeños errores, y existe un amplio número de grupos de interés dispuestos a aportar su experiencia e incluso su código fuente para impulsar de nuevo el proyecto. En definitiva, la comunidad JMF está viva, aparentemente. Además, hoy por hoy sigue resultando la mejor alternativa abierta (el código fuente está disponible para los desarrolladores) que aporta soluciones completas en este campo. No se debe olvidar que Quicktime [16], por ejemplo, también posee sus propias bibliotecas multimedia para Java, aunque parece ser que éstas resultan algo inestables en comparación con JMF, provocando errores esporádicos que cuelgan la Máquina Virtual de Java. Además, resulta imprescindible que la aplicación Quicktime esté instalada en la máquina cliente que ejecuta el applet, y ésta sólo se encuentra disponible para Windows y Macintosh, con lo que se desvirtúa el carácter multiplataforma de Java y de nuestro programa.

Existe otra decisión de diseño aún por justificar, ésta menos controvertida que el tema de JMF. Se trata de la inclusión del reproductor en un applet en lugar de una aplicación independiente. Como se ha comentado con anterioridad, consideramos la distribución de contenidos por Internet como una prioridad para nuestra herramienta. El hecho de programar el reproductor como un applet evita que los estudiantes tengan que llevar a cabo complejos procesos de instalación para poder visualizar las presentaciones. Esto permite un gran ahorro en tiempo y costes, puesto que puede usarse prácticamente cualquier ordenador conectado a la red para reproducir los Docusquemas (únicamente sería necesario actualizar la máquina virtual de Java con las bibliotecas JMF, lo que resulta un proceso muy sencillo) Las ventajas que esto aporta en cuanto a crecimiento y expansión de nuestra solución educativa son importantes, sin olvidar la facilidad con que se podría actualizar el reproductor a una nueva versión, sin necesidad de hacer

cambios en los ordenadores cliente. Además de estas ventajas, la orientación hacia Internet nos ofrece mayores facilidades a la hora de obtener recursos multimedia distribuidos. Podemos localizar cualquier contenido a través de su URL e incluirlo en los Docusquemas con gran facilidad, permitiendo de este modo que los contenidos de la presentación no estén almacenados en un soporte físico que haya que distribuir por medios tradicionales, ni siquiera en un servidor centralizado, sino que pueden estar distribuidos por la Red.

El applet reproductor de docusquemas se muestra integrado en una página HTML. Basta con que los usuarios visiten la página de la lección a visualizar, y ésta comenzará a reproducirse automáticamente. Esto permite integrar totalmente los Docusquemas en la estructura de un sitio web, logrando que toda la complejidad del sistema quede oculta y que el alumno únicamente tenga la sensación de haber navegado entre páginas llenas de contenido. Sin embargo, también existe la posibilidad de percibir el Docuplayer como un reproductor independiente de los ficheros de Docusquemas. En este caso, el applet nos preguntará por la localización del fichero que deseamos cargar, que podrá estar dada por un URL genérico o una ruta local, pudiendo ser incluso una ruta relativa a la localización del applet. Es importante resaltar que, debido a las restricciones de seguridad que afectan a los applets, el reproductor únicamente podrá leer ficheros que se encuentren colgados públicamente en un servidor, o bien ficheros locales que se encuentren almacenados en subcarpetas hijas de aquella en la que se encuentra el código del applet.

Una vez que se ha cargado la presentación que se va a visualizar, se lleva a cabo la espera oportuna especificada en el fichero. Durante estos segundos de espera para que el alumno observe la estructura del esquema inicial, se mantienen a la vista las barras de botones superior e inferior de la aplicación (figura 2) Esto sirve para que el usuario, que se encuentra en un momento de observación del contexto, perciba el funcionamiento de la aplicación y tenga conocimiento de las posibilidades que ésta le va a ofrecer. Una vez finalizado el tiempo de espera, las barras de botones se ocultarán automáticamente y comenzará la reproducción. Una vez ocultas las barras, el Docusquema cobra protagonismo absoluto. Nada distrae la atención del alumno de la presentación, logrando una gran inmersión en el flujo de presentación de los conceptos. Sin embargo, el usuario ha visto los controles y sabe dónde están, de modo que cuando necesite

usarlos, instintivamente dirigirá el ratón hacia el lugar en el que se encontraban los botones, y éstos aparecerán automáticamente, logrando que el efecto de inmersión se obtenga sin sacrificar la funcionalidad del sistema.

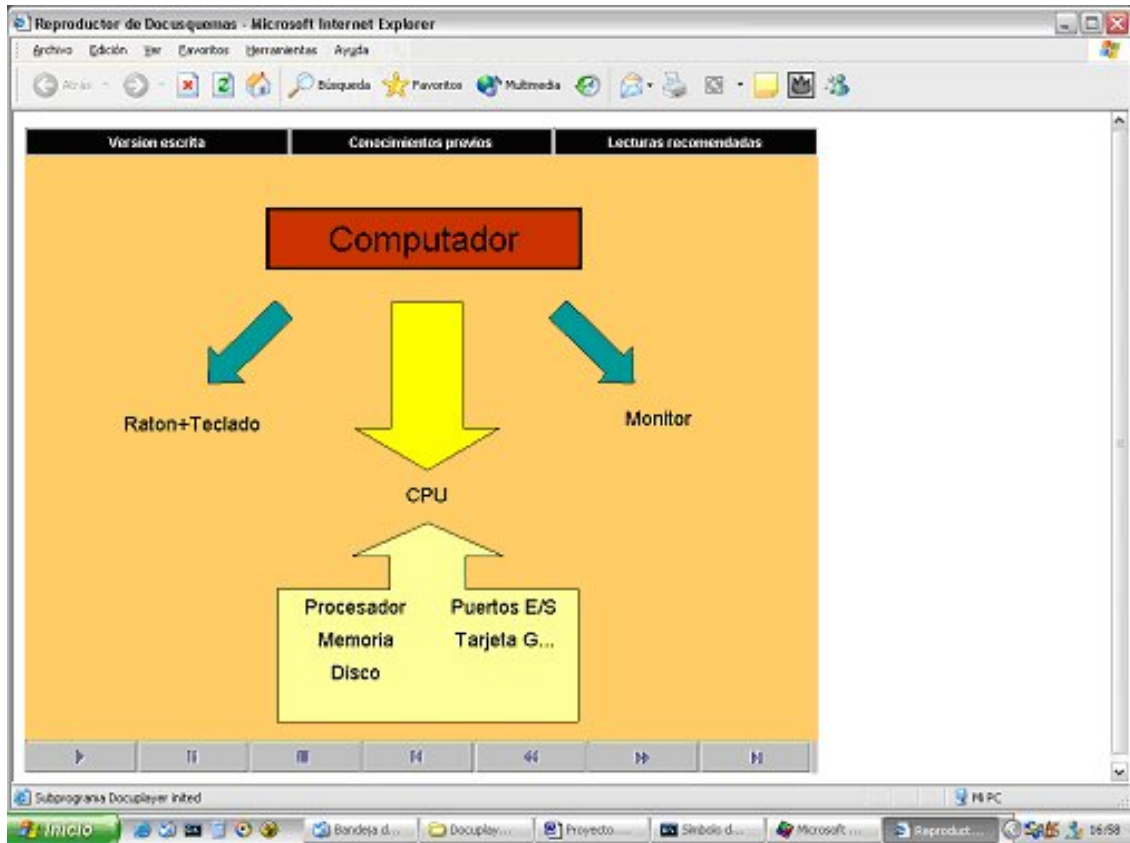


Figura 2: Ejemplo de Docusquema reproduciéndose en una página HTML gracias al applet Docuplayer

Como acaba de mencionarse, la barra de botones que se muestra en la parte superior contiene los enlaces a ficheros de documentación complementaria. Los tres tipos de complementos que deberían acompañar a los nodos son: la versión escrita de la lección que se presenta, una selección de lecturas recomendadas y los enlaces a los nodos que deberían haberse visitado con anterioridad [6]. Cada uno de los botones que se muestran en la barra corresponde a uno de estos tres elementos, y al pulsar sobre ellos se abrirá una nueva ventana del navegador que mostrará el documento elegido. Respecto a los formatos de ficheros que se pueden incluir en las presentaciones, estos están limitados por las capacidades del navegador de Internet para mostrarlos, y dado su grado de desarrollo actual podemos garantizar una gran variedad de formatos

disponibles. En cualquier caso, si no pudiera mostrar el contenido del fichero, el navegador ofrecería automáticamente la posibilidad de descargar el fichero al ordenador del usuario para que éste lo abriera con la aplicación adecuada, así que éste es un problema que no debe preocuparnos. Existe la interesante posibilidad de que la página de “nodos previos” incluya enlaces a dichos nodos, de modo que únicamente pulsando en el enlace adecuado visitaríamos el Docusquema correspondiente a la lección especificada.

La barra inferior, por otra parte, aglutina los controles de reproducción, tanto del Docusquema en general como del recurso multimedia que se esté mostrando en un momento dado, siendo la mayor parte de los controles comunes para ambos propósitos. Según el modelo del Docusquema, podemos pulsar sobre cualquier concepto para comenzar a reproducir todos los elementos asociados a él, y si el concepto pulsado es el principal (título), comenzará la reproducción del Docusquema completo. Si en cualquier momento deseamos hacer una pausa, reanudar la reproducción o simplemente detener la ejecución del Docusquema, podremos hacerlo desde los botones de la barra inferior. Respecto a los botones de avance y retroceso, se ha hecho una distinción necesaria. Existen dos tipos de botones diferenciados: el primero de ellos hace avanzar rápidamente el recurso multimedia que se esté visualizando, o bien lo hace retroceder hasta el principio. Los otros dos botones, en cambio, sirven para navegar entre conceptos, saltando de uno a otro secuencialmente. De este modo el usuario puede moverse por el Docusquema con total libertad, distinguiendo entre los bloques de la presentación (conceptos) y los recursos asociados a cada uno de ellos.

Se ha implementado además una funcionalidad extra a uno de los controles. En principio, cuando un usuario pulsa con el ratón sobre un concepto, se reproduce únicamente dicho concepto, con su audio de introducción en el caso de que lo tenga, seguido del elemento multimedia asociado al concepto, y la reproducción finaliza en este punto. Sin embargo, si durante esta presentación el usuario pulsa sobre el botón de avance o retroceso entre conceptos, el programa automáticamente supondrá que el alumno desea ver el resto de los conceptos, por lo que entrará en el modo de reproducción completa, y cada vez que un concepto finalice comenzará con el siguiente. De este modo ofrecemos la posibilidad de comenzar el Docusquema a partir de un

concepto intermedio, y a partir de ahí mostrar secuencialmente todos los conceptos que restan hasta llegar al final de la presentación.

Los únicos controles que sí se han duplicado son los de avance y retroceso. Al igual que en los reproductores de discos de música se distingue entre la acción de avanzar dentro de la misma canción y la de saltar a la siguiente pista, nosotros mantenemos esa distinción, permitiendo al usuario que se mueva dentro del recurso que esté viendo, o bien que salte a los conceptos anterior o siguiente. De este modo, Docuplayer ofrece una única barra de controles de aspecto y funcionalidad similar a la presente en cualquier reproductor de CD convencional, lo cual facilita la adaptación a la herramienta y hace más amigable el entorno de trabajo.

6. El asistente para la creación de Docusquemas: Docuwizard

Podría pensarse que la especificación de un lenguaje para describir los Docusquemas, y la programación de un applet capaz de reproducir los ficheros escritos en ese lenguaje conforman por sí solos una solución completa al modelo de funcionamiento del Docusquema. Es cierto que puede utilizarse un editor de textos para escribir tantas presentaciones como queramos de una forma rápida y fácil, y que la simplicidad del lenguaje DoX facilita su aprendizaje en un periodo breve de tiempo. Si se proporcionan además algunos ejemplos de Docusquemas que cubran el espectro de posibilidades distintas que pueden darse, resultaría sencillo utilizarlos para crear nuevos nodos a partir de ellos. Simplemente con esto quedaría cubierto el objetivo inicial de proporcionar una plantilla que facilitase la escritura de nuevos Docusquemas, superando las dificultades que conlleva comenzar desde cero.

El propósito de este proyecto era, sin embargo, acercar el modelo de los Docusquemas al mayor número de usuarios posible, con independencia de su nivel de formación informática y de su habilidad al teclado. Para lograr este propósito, no podemos pretender que los creadores de Docusquemas realicen todo su trabajo escribiendo líneas de XML como si fueran programadores. Resulta imprescindible contar con una aplicación que muestre la parte invariante del Docusquema (el esquema

principal, con los conceptos distribuidos en sus posiciones respectivas) conforme se lleva a cabo la creación o edición de un nodo completo.

La función por tanto de la herramienta de asistencia para la creación de Docusquemas es facilitar aún más el proceso de creación de presentaciones. De este modo, entre sus objetivos estará la ocultación del código del lenguaje DoX al usuario, la reducción del tiempo necesario para desarrollar un nodo partiendo de cero, y la presentación en tiempo real del aspecto del esquema en la pantalla. No es necesario que realice complejas funciones ni que implemente una interfaz de última generación. Se trata de realizar un programa intuitivo, de manejo sencillo y rápido aprendizaje, pero que permita modificar todas las características de los Docusquemas y observar el resultado.

La creación de esta herramienta favorece que el educador tenga la posibilidad de eludir los ficheros de texto, creando sus nodos a través de una interfaz gráfica sin tener que aprender las características del lenguaje DoX. Pero esto no significa que, posteriormente, una vez que se haya familiarizado con la estructura de los Docusquemas, quizá quiera dar un paso más, y atreverse a abrir un fichero de descripción en un editor de textos para realizar pequeñas modificaciones a mano.

El programa se ha planteado como una aplicación independiente, en lugar de ser un applet integrado en una página web. Mientras que el Docuplayer está pensado para ser utilizado en un elevado número de máquinas distintas con una extensa distribución geográfica, Docuwizard sólo necesita estar instalado en unos pocos ordenadores, los de los profesores autores de las distintas presentaciones. También se han descartado los applets porque presentan grandes limitaciones para leer y escribir en el sistema de ficheros local, y esta capacidad resulta imprescindible en nuestra aplicación. Los usuarios deben tener la posibilidad de crear Docusquemas con los recursos que tengan almacenados en su ordenador, y al acabar guardar el fichero DoX en su disco duro. Al ejecutarse como una aplicación, podemos mostrar cuadros de diálogo para abrir y guardar ficheros, y puesto que gran parte de la descripción de nodos consiste en especificar las localizaciones de los archivos multimedia, abrir una ventana de navegación de ficheros supone un ahorro de tiempo considerable frente a la alternativa de teclear la ruta completa hasta el fichero.

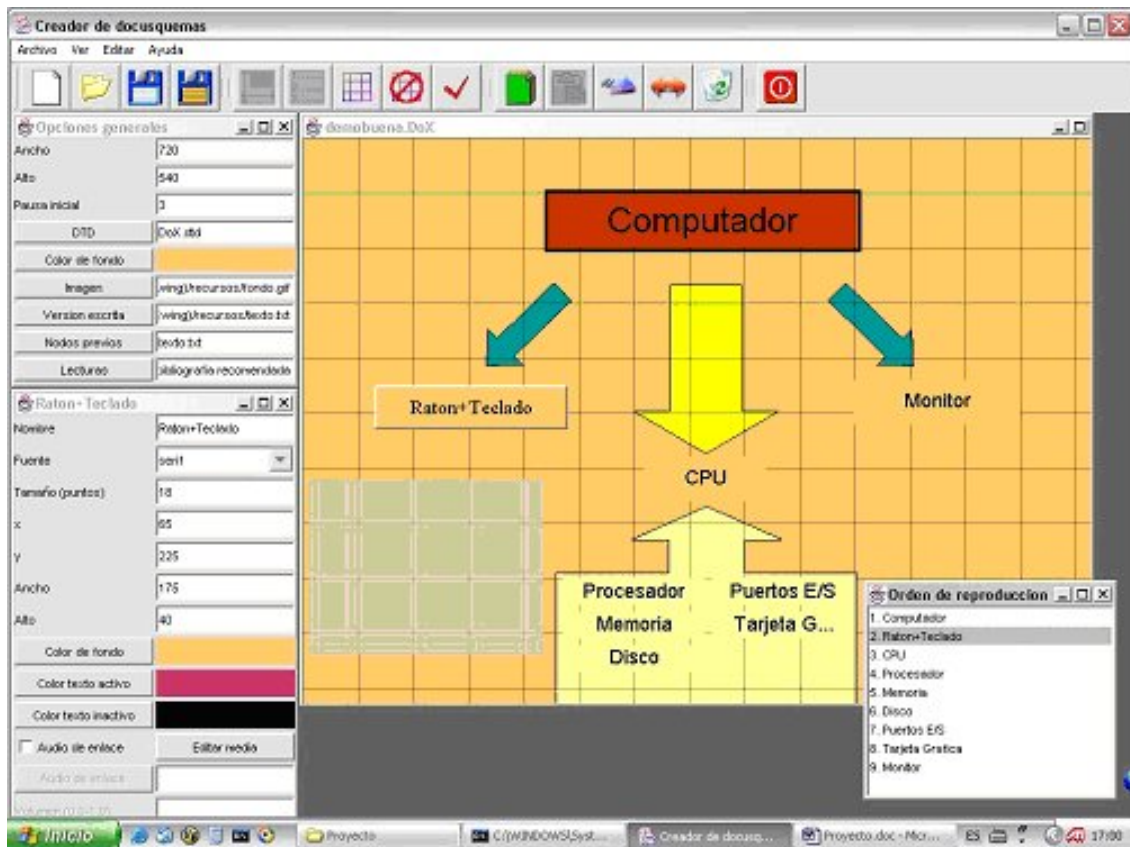


Figura 3: Pantalla principal de Docuwizard

En la **¡Error! No se encuentra el origen de la referencia.** se puede ver el aspecto que presenta la aplicación durante la edición de un Docusquema. A primera vista se aprecian los bloques principales:

- Una ventana de previsualización, donde se puede ver el aspecto que presentará el Docusquema en pantalla, y apreciar en tiempo real los cambios que se realicen en la imagen de fondo, los colores, tamaños y posiciones de los conceptos, etc. Cuando se selecciona un concepto pulsando sobre él, aparecerá un recuadro sombreado que mostrará el área ocupada por el recurso multimedia asociado a dicho concepto. De este modo no es necesario recurrir al Docuplayer para saber si un vídeo tapaná determinada zona de la pantalla cuando se ejecute el concepto correspondiente. Sobre la imagen de fondo hay dibujada una cuadrícula, su función es orientar al usuario en la distribución de los recursos sobre la

pantalla. La rejilla está formada por cuadros de 50x50 píxeles, de modo que con un simple vistazo se puede estimar el tamaño del Docusquema y la posición y dimensiones de cada concepto. El usuario puede elegir el color de la cuadrícula para que resalte sobre el fondo, y también puede hacerla desaparecer si lo desea, a fin de trabajar más cómodamente. Esta ventana no es únicamente un tapiz en el que se muestra el Docusquema, sino que tiene un papel activo, puesto que se puede pulsar sobre cualquiera de los conceptos para seleccionarlo y modificar sus características.

- Una ventana de opciones generales en la parte superior izquierda. Aquí se pueden configurar parámetros como el tamaño del Docusquema, el color de fondo, la imagen que servirá de tapiz, y las localizaciones de los archivos con información complementaria, así como la posición de la DTD que se usará para validar el fichero DoX resultante (lo habitual es que se incluya la DTD como un fichero en el mismo directorio que el que se está editando) Los cambios en las opciones de aspecto se verán reflejados inmediatamente en la ventana principal. Para elegir el color de fondo aparece un cuadro de diálogo de selección de color. De este modo se puede escoger el color exacto que se desea sin necesidad de conocer sus componentes RGB, y el programa se encarga de hacer la traducción automáticamente. Es muy importante señalar que la ventana de selección de color que proporciona Java no permite especificar el grado de transparencia (canal alfa) Por lo tanto, si se desea utilizar transparencias se deberá modificar personalmente el valor de la componente “a” de los colores, abriendo el fichero DoX correspondiente con un editor de textos. Respecto a los archivos, se puede especificar su localización de dos maneras distintas: escribiendo directamente la URL o la ruta en el recuadro disponible, o bien pulsando el botón correspondiente a la opción que se desea modificar, que abre un cuadro de diálogo de “Abrir archivo” en el que podemos navegar por el sistema de ficheros del ordenador y elegir el que deseamos. Si el archivo seleccionado se encuentra en el mismo directorio que el fichero DoX que estamos editando, o bien en alguno de sus subdirectorios, el programa automáticamente escribirá la localización

del fichero como una ruta relativa, en lugar de almacenar la ruta completa desde el directorio raíz.

- En la parte inferior izquierda aparece la ventana de opciones de concepto. En ella se muestran las características generales del concepto que está seleccionado actualmente. El funcionamiento es similar a la ventana de opciones generales, con unos campos para especificar la posición y dimensiones, tres selectores de color, y una lista desplegable para elegir el tipo de fuente. En la lista aparecen únicamente los tipos de fuente básicos de Java, los que garantizan el mismo aspecto en todas las plataformas para asegurarnos de que el resultado final en cualquier máquina será el que nosotros deseamos. En la parte inferior encontramos una casilla que cuando es seleccionada activa los campos correspondientes al audio de introducción opcional. De este modo, podemos activar o desactivar este recurso de forma rápida y sencilla. También existe un botón que muestra una nueva ventana con las opciones particulares del recurso multimedia asociado al concepto.
- La ventana de opciones de medios muestra los cuatro tipos de recursos multimedia que podemos incluir en los Docusquemas. En función del tipo que escojamos, en la parte derecha aparecerán unos campos u otros con las opciones por defecto, que nosotros podremos editar. Podemos cambiar entre los cuatro tipos distintos de medios viendo las opciones posibles sin perder los datos originales que tuviera el concepto.

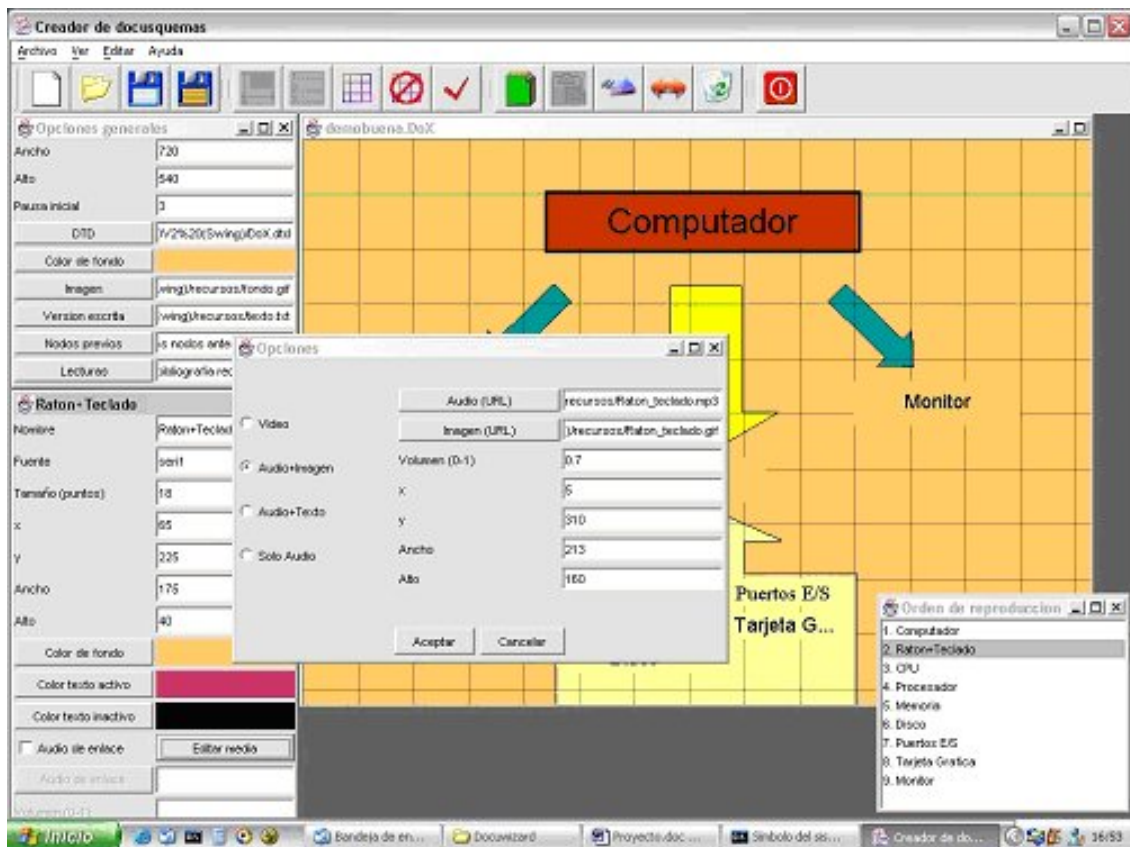


Figura 4: Docuwizard con la ventana de opciones de medios abierta

- En la parte inferior derecha aparece una ventana con la lista de todos los conceptos colocados por orden de reproducción. Esta lista nos permite conocer la secuencia de ejecución de los distintos conceptos, y pulsando sobre ella también podemos seleccionar el concepto que deseamos editar. Existe otra una forma alternativa de seleccionar los conceptos, y es pulsando directamente sobre ellos en la ventana principal. Cuando estemos modificando las características de un concepto, éste se resaltaré con un recuadro para que exista en todo momento un indicador del elemento con el que está trabajando, proporcionando así una realimentación necesaria para que el usuario no se pierda. Además aparecerá un recuadro sombreado que mostrará el área ocupada por el recurso multimedia asociado a dicho concepto. En la figura 4 se puede ver resaltado el concepto “Ratón+Teclado”, con sus opciones visibles desplegadas en la ventana de la izquierda.

- Finalmente, en la parte superior de la ventana de la aplicación aparecen las barras de herramientas y de menús. Siguiendo el carácter sencillo e intuitivo de la interfaz, se han diseñado unos botones grandes, con dibujos expresivos y llenos de colorido, que muestran una etiqueta que explica su función cuando se posa el ratón sobre ellos. Estos botones agrupan prácticamente todas las funciones del editor de Docusquemas, y se encuentran separados en tres grupos: el primero para las operaciones referidas al trabajo con archivos (crear nuevo, abrir, guardar, guardar con nombre nuevo), el segundo para modificar el entorno de trabajo (mostrar la ventana de opciones, la lista de conceptos, cambiar el color de la cuadrícula o hacerla desaparecer y redibujar la ventana principal), y finalmente el tercero aglutina las operaciones que se pueden realizar sobre los conceptos, como copiar, pegar, insertar un concepto nuevo, cambiar el orden de reproducción y eliminar un elemento. Todas estas operaciones se pueden llevar a cabo también desde los menús situados sobre la barra de botones.

Durante el desarrollo de la aplicación, en todo momento se ha buscado la facilidad de uso y la agilidad en la creación de nuevos Docusquemas. Prácticamente toda la información se encuentra a primera vista y se puede modificar con muy pocas pulsaciones de ratón. Cada vez que se introduce un nuevo dato en un campo de texto pulsando la tecla ↵ o el botón de “Aceptar” en un cuadro de diálogo, la ventana de previsualización se refresca para reflejar las modificaciones. Aun así el usuario tiene la posibilidad de forzar la actualización en un momento determinado, bien pulsando sobre el botón correspondiente de la barra superior, o bien mediante una combinación de teclas (CTRL+F5)

Cada vez que se introduce un nuevo dato, el programa comprueba que éste es válido (por ejemplo asegurando que el usuario ha introducido una cifra en la casilla de número de orden en lugar de una letra, y que el número se encuentra dentro de los límites permitidos) De este modo los errores se van notificando en el momento en que se producen, en lugar de dejar que se acumulen y que aparezcan todos en el momento de salvar el fichero.

La interfaz basada en ventanas independientes dentro de un marco general proporciona flexibilidad para personalizar el espacio de trabajo. En todo momento las distintas ventanas se pueden mover, redimensionar, minimizar, o cerrar totalmente para recuperarlas posteriormente. De este modo el usuario puede configurar el entorno de trabajo a su gusto, redundando en una mayor comodidad de uso. Sin embargo, se ha optado por mantener las distintas subventanas dentro de la ventana principal de la aplicación, para lograr un efecto de unidad y sencillez. A menudo los usuarios se sienten abrumados cuando ven desplegarse frente a ellos multitud de ventanas independientes, piensan que el programa será difícil de manejar y tienden a rendirse rápidamente.

7. La cuestión del ancho de banda

Al desarrollar el modelo del Docusquema nos ubicábamos en una situación teórica ideal en la que la tecnología no era un factor limitante. En base a este presupuesto se definió una teoría sobre la estructura ideal que deben tener las presentaciones para que resulten pedagógicamente efectivas. Sin embargo, ya en su momento éramos conscientes de que la implementación del Docusquema en el mundo real podría adolecer de ciertas limitaciones que lo alejaran del modelo teórico.

Nuestras presentaciones se componen de multitud de recursos multimedia cuya descarga a través de Internet puede resultar excesivamente prolongada, dada la velocidad de las conexiones actuales presentes en los hogares, y resulta necesario establecer pautas que mitiguen ese inconveniente.

Hay que tener en cuenta que las conexiones de banda ancha en los hogares están creciendo de forma explosiva a través del ADSL principalmente, pero sin olvidar otros medios como el cable, acceso por satélite, LMDS y PLC, tal y como se ha comentado anteriormente. Esta situación nos permite augurar un futuro en el que los Docusquemas puedan estar dotados de gran riqueza de medios, sin que el tiempo de descarga suponga un factor limitante.

Sin embargo, existen ciertas técnicas que pueden reducir el tamaño de los Docusquemas mientras esperamos a que nuestro sueño de la banda ancha omnipresente se haga realidad:

- Los GIFs animados acompañados de un fichero de audio resultan una alternativa barata a los vídeos. Su realización es más sencilla, acercándolos a las posibilidades de usuarios no versados en la edición de vídeo, y el consumo de ancho de banda se reduce de forma abrumadora.
- Codificación de los fragmentos de audio en el formato MP3. Docuplayer los reproduce sin dificultad, y podemos lograr tasas de compresión de 7 a 1 e incluso mayores.
- Utilización de animaciones generadas en Flash para las presentaciones parciales de los conceptos. Al igual que los archivos en GIF animados, suponen una alternativa ligera (en términos de ancho de banda) a los vídeos. Actualmente, el JMF no soporta versiones de Flash superiores a la 2.0, lo cual limita las posibilidades de uso de este recurso. Sin embargo, y dada la demanda de la comunidad de desarrolladores de JMF, todo apunta a que esta será una de las prioridades en el desarrollo de futuras versiones de esta API.
- Una adecuada elección en general de los formatos de codificación de los medios supondrá una mejora de la eficiencia que redundará en un menor consumo de ancho de banda.

8. Distribución de las herramientas

Las aplicaciones se han hecho disponibles públicamente a través de una página web [19], junto con un Docusquema de ejemplo que permitiera comenzar a trabajar con ellas sin partir de cero. La disponibilidad de estas aplicaciones se ha publicitado en diversos foros de comunidades interesadas en el ámbito de la teleeducación, entre ellos EDUDIST, E-LEARN y la Cátedra UNESCO de Educación a Distancia (CUED).

Los comentarios recibidos a través de estos medios tienen un carácter general de felicitación por la evolución lograda en el tema de los Docusquemas, tanto por el modelo teórico como por las herramientas desarrolladas. El haber llevado a cabo una implementación práctica supone un gran salto adelante.

Es especialmente destacable la experiencia llevada a cabo en dos universidades mejicanas, donde en un par de cursos de grado se probó tanto el modelo de Docusquema como las herramientas para elaborarlos, resultando en una gran aceptación por parte tanto de los docentes como de los alumnos, que a la postre revirtió en una sensible mejora del proceso y los resultados de aprendizaje [10].

9. Conclusiones y trabajos futuros

Hemos partido de un modelo para la elaboración de cursos *online* y nos hemos centrado en su dimensión expositiva. Partiendo de un modelo para la elaboración de presentaciones multimedia especialmente orientadas a un aprovechamiento más integral de las capacidades cognitivas de los estudiantes, a fin de lograr mejores resultados de aprendizaje, modelo denominado “Docusquema”, se ha desarrollado un conjunto de herramientas y lenguajes que, empleados conjuntamente, proporcionan una solución para la implementación del modelo, utilizando para ello las tecnologías disponibles en la actualidad.

Disponemos ahora, por tanto, de unas herramientas que permiten presentar contenidos multimedia con facilidad, ubicándolos libremente en la pantalla y controlando el flujo de la reproducción. La presentación se ejecuta de forma sincronizada, evitando la superposición de medios y realizando las pausas oportunas.

Por otra parte, la forma en que se han implementado los Docusquemas permite la importación de recursos multimedia capturados y desarrollados utilizando herramientas externas, y facilita la interacción del usuario con estos recursos, puesto que puede escoger que se reproduzca el medio que desea, y manipularlo a su antojo durante la exposición.

En todo momento se ha perseguido la facilidad de uso como factor crítico en el desarrollo de las aplicaciones, y no sólo se proporciona un ejemplo de guía para la escritura de Docusquemas, sino que se ha implementado un asistente que facilita tanto la creación de nuevos nodos como la edición de los existentes.

Finalmente, la solución implementada es totalmente multiplataforma, y la distribución de Docusquemas se ha pensado para su total integración con Internet, a través de applets que manipulan documentos XML.

Llegados a este punto, resulta satisfactorio comprobar los resultados de nuestro proyecto. Este hecho, sin embargo, no conduce a dar por finalizado nuestro trabajo, sino que plantea nuevos objetivos, líneas de desarrollo que completen y mejoren el sistema actual, y proyectos de trabajo futuro que extiendan el empleo de Docusquemas en el mundo de los cursos *online*. Estos nuevos objetivos se plantean en las siguientes líneas de trabajo:

- El Docusquema comprende tan sólo la Dimensión Expositiva (intranodo) de un modelo de generación de cursos *online* más complejo [5]. Ahora corresponde llevar a cabo la integración con la Dimensión Estructural (internados) del modelo, de modo que una y otra se complementen conformando una solución completa.
- Por otra parte, seguimos realizando experimentos adicionales con Docusquemas, que ahora podemos desarrollar con mayor facilidad. En estas experiencias en entornos de aprendizaje reales se analizan de forma completa y rigurosa las impresiones tanto de los profesores como de los alumnos.

Bibliografía

1. Ausubel, D. (1963); **The Psychology of Meaningful Verbal Learning**. Grune & Stratton. Nueva York, 1963.

2. Ausubel, D. (1978); **In defence of advance organizers: A reply to the critics.** Review of Educational Research, nº48, pp. 251-257. 1978.
3. Caudeli Tomé, J. (2001); **Estudio sobre las posibilidades técnicas de implementación del Docusquema** (Trabajo dirigido) Departamento de Ingeniería Telemática. Universidad Carlos III de Madrid. 2001.
4. Dunlop, M. and Scott, D. (2001); **An Examination of the Impact of Aspects of Online Education Delivery on Students.** Actas de AusWeb01, 21-25 de abril, Coffs Harbour, Australia.
5. García Rueda, J.J. (2002) **Modelado de Experiencias Educativas en la World Wide Web** (Tesis Doctoral). Departamento de Ingeniería de Sistemas Telemáticos. Universidad Politécnica de Madrid. 2002.

6. García Rueda, J. J. y Sáez Vacas, F. (2001); **The way of significant innovation: When Guttenberg became nonlinear**; Actas de NAWeb 2001: The Web-Based Learning Conference; Universidad de New Brunswick, 13-16 de Octubre 2001, Fredericton (Canada).
7. García Rueda, J. J.; Caudeli, J. y Sáez Vacas, F. (2002); **Conocimiento Declarativo en un Sistema Hipermedia: Creación de Nodos para Aprendizaje Receptivo Significativo. Congreso Interacción Persona-Ordenador.** Universidad Carlos II de Madrid. 2002.
8. García Rueda, J. J.; Caudeli, J. y Sáez Vacas, F. (2002) **Meaningful Reception Learning in a Multimedia Context: Colourful Cognition in Action.** 4th International Conference on New Educational Environment. Lugano (Suiza). 2002.
9. Pitts, N. (1999); **XML in record time.** Anaya Multimedia. 1999.
10. Franco Espinosa, C.; García Rueda, J.J. y Román Julián, R. (2004); **DOCUSCHEMAS: EXPERIENCING WITH A MULTIMEDIA TOOL FOR SUPPORTING HIGHER EDUCATION**; en las actas de la International Conference on Education IADAT-e2004, Innovation, Technology and Research in Education; del 7 al 9 de julio de 2004; Bilbao (España).

Sitios web

11. Foro XML-ES de la Universidad Carlos III de Madrid: www.uc3m.es/~xml
12. Synchronized Multimedia Integration Language (SMIL): www.w3.org/TR/smil10
13. Flash: www.flash.com
14. Macromedia Inc.: www.macromedia.com

15. Página oficial de Java de Sun: www.java.sun.com
16. Quicktime: www.quicktime.com
17. Java Media Framework (JMF): www.java.sun.com/jmf
18. XML en Java: <http://www.java.sun.com/xml>
19. Página oficial de los Docusquemas :
<http://bach.gast.it.uc3m.es/~jcaudeli/Proyecto.htm>