

Patrones de diseño en un Sistema Multimedia para la Educación

J.A. Ortega, J.M. Márquez, J. Torres, M. J. Escalona

Dpto. de Lenguajes y Sistemas Informáticos
Facultad de Informática y Estadística
Universidad de Sevilla
ortega@lsi.us.es

Resumen.

Diseñar e implementar sistemas multimedia para la educación no es una tarea sencilla. No sólo por el trabajo que conlleva sino por el cuidado y la dedicación que hay que prestar por encima de todo al diseño de la aplicación, de forma que la función pedagógica de la misma no se vea mermada lo más mínimo por un diseño poco funcional que dificulte el aprendizaje del alumno. En este artículo mostraremos como un conjunto de estrategias (patrones) que pueden sernos útiles a la hora de diseñar y construir nuestra aplicación, algunas de nuestras aproximaciones en un sistema multimedia para la educación basado en ventanas.

1. Introducción

En los últimos años existe una tendencia a considerar el desarrollo hipermedia con un enfoque de proceso de ingeniería (del software), por lo que ya se han propuesto diferentes metodologías que asumen la orientación a objetos como paradigma de diseño, como OOHDM [Schwabe96].

Cabe destacar el interés por la representación de la navegación mediante un modelo de clases, como queda patente en NDT [Escalona02], donde se presenta un proceso para realizar la especificación de requisitos, el análisis de la navegación y el diseño de la misma. Con NDT se consigue un modelo navegacional a partir del cual se podrían aplicar las propuestas de otras metodologías.

El diseño inicial de un sistema multimedia para la educación debe conjugar una serie de principios psicopedagógicos con el fin de conseguir la mayor optimización didáctica posible del sistema. Hay que tener en cuenta cualidades tan importantes como la amigabilidad, comprensibilidad, navegabilidad, actualidad o interactividad

[Astigarraga99], evitando siempre caer en el barroquismo, que suele ser la causa principal del llamado *analfabetismo funcional*.

Como ejemplo vamos a mostrar el uso de estos patrones en Eduka [Márquez02], un sistema para educación semipresencial realizado íntegramente en Java. Comentaremos algunos patrones de diseño ya introducidos en [Lyardet98] y su aplicación a un sistema basado en navegación por ventanas.

2. Navegación a través de la información

Los patrones de esta sección, encaran problemas relativos a la organización y acceso a la información.

2.1 Nodo como unidad simple de información.

Objetivo: ¿Qué constituye una unidad de información con respecto a la navegación?

Desarrollo: Un nodo de información deberá constituir por sí mismo un conjunto de información con sentido propio para un conjunto de usuarios que llevan a cabo una tarea concreta.

2.2 Contexto navegacional

Objetivo: ¿Cómo organizar la información? ¿Se deben establecer relaciones entre el estado de navegación y la forma en que puede ser presentada la información?

Desarrollo: A menudo surgen multitud de caminos que llevan a la misma información (mismo nodo de información). La solución pasa por definir características peculiares, que enriquezcan la interfaz del objeto que está siendo visitado. El contexto navegacional esta compuesto de un conjunto de nodos de información y enlaces. Los nodos son decorados con información adicional sobre el contexto particular y enlaces adicionales.

2.4 Factoría

Objetivo: ¿Como podemos presentar eficientemente la información que necesita el usuario de forma que ésta pueda ser correctamente comprendida?

Desarrollo: Mostrar de alguna forma la información relacionada en el mismo nodo, por ejemplo con ventanas pop-up (emergentes) que pueden ser desactivadas fácilmente o bien reservando un espacio del nodo para estos menesteres.

2.4 Referencia Activa

Objetivo: ¿Como podemos proporcionar una referencia permanente sobre el estado actual de la navegación, combinando una herramienta de orientación con una forma fácil de navegar en un conjunto relacionado de nodos, en el mismo nivel de abstracción?

Desarrollo: Una buena solución es diferenciar al índice y los nodos de información en objetos diferentes, ambos visibles al mismo tiempo. Una modificación el objeto que actúa de índice provocará la actualización del objeto que contiene al nodo de información, cargando el nuevo nodo.

3. Organización de la capa de presentación

3.1 Separación entre información e interacción

Objetivo: Diferenciar entre controles que efectúan tareas diferentes, especialmente entre controles que reciben información por parte del usuario: controles de entrada, y controles que proporcionan información: controles de salida.

Desarrollo: Agrupar los controles para separar los de entrada de los de salida. Se permitirá que los controles de entrada, puesto que requieren mucho más la participación del usuario, se mantengan mezclados con la información en aquellos casos en los que sea necesario, mientras que en ningún momento los controles de salida se mezclarán con la información.

3.2 Agrupación funcional

Objetivo: ¿Como reconocer diferentes tipos de controles en la interfaz de una forma rápida?

Desarrollo: Definir reglas de diseño de interfaz gráfica, en las que se especifiquen los criterios de agrupación funcional y la separación entre controles de funcionalidad semejante, el tamaño de éstos, aspecto... así como la disposición de los diferentes grupos de controles y la separación entre grupos con funcionalidad diferente.

3.3 Información bajo demanda

Objetivo: ¿Cómo organizar la interfaz de forma que los datos, la información, se presente tal y como queremos?

Desarrollo: En numerosas ocasiones cuando diseñamos, lo hacemos de tal forma que todos los controles, enlaces y demás miembros de la interfaz se visualicen correctamente en la pantalla. Pero no podemos controlar que el usuario final lo perciba tal y como lo hemos diseñado. Ante problemas de esta índole, tenemos dos alternativas: diseñar la interfaz de nuestra aplicación en un tamaño menor o bien incluir una serie de controles que nos permitan variar el aspecto de la información que se está presentando: zoom-in, zoom-out, mostrar sólo textos, realzar imágenes... Estos controles responden a la filosofía "Lo que ves es lo que necesitas" (*What you see is what you need*). La activación de estos controles no produce navegación ni abre nuevos nodos de información, simplemente hacen que la visualización, lo que percibe el usuario, varíe a petición del mismo.

4. Nuestras aproximaciones

4.1 Nodo como unidad simple de información

En Eduka, la información proviene de dos fuentes distintas: por lado de la información sobre los usuarios, el sistema y documentos multimedia, que es extraída de un sistema de bases de datos; y por otro, de la videoconferencia en tiempo real que se obtiene "al instante" y que no está almacenada en ningún sitio. Eso es lo que constituye nuestra base para diseñar los nodos de información. En Eduka, se ha denominado a los nodos de información como "ventanas hoja", puesto que a partir de ella no podemos llegar a ninguna otra ventana de información. Cada ventana hoja responde a un tópico concreto, por ejemplo, "propiedades de configuración de la transmisión de vídeo y audio en tiempo real" como se muestra en la figura.

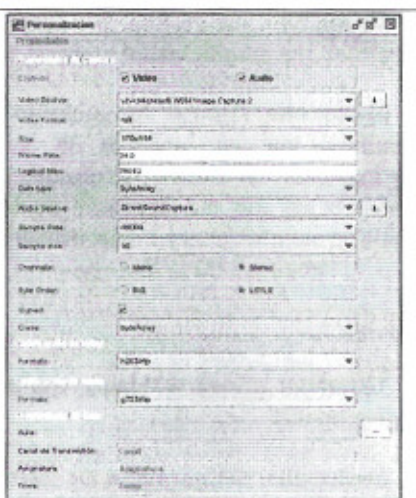


Figura 1. Ventana hoja de personalización de propiedades de transmisión

4.1 Contexto navegacional

En Eduka se modela este patrón como "ventanas nodo" (no confundir con nodos de información). Las ventanas nodo agrupan un conjunto de *ventanas hoja* accesibles a través de una barra de menú. Estas ventanas hoja se abren en el interior de la ventana nodo facilitando así al usuario la asociación de los conceptos con el contexto en el que están relacionados.

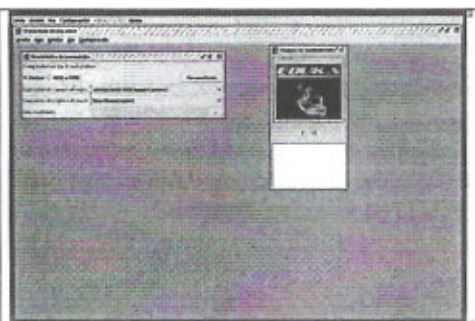


Figura 2. Ventana nodo de Transmisión de una clase. En ella podemos apreciar internamente dos ventanas hoja: propiedades predefinidas y ventana de monitorización

Un buen ejemplo disponible para páginas web HTML, que además se combina con una interfaz gráfica muy amena y lúdica, lo constituye la página The Village [WSI02], de Wall Street Institute, que usa la metáfora de un pueblo para asociarlo a la funcionalidad requerida para la enseñanza del inglés. En la página de The Village, cada una de las tareas a realizar (ejercicios de gramática, vocabulario, chat, lecturas, ...) representan los contextos navegacionales, ordenados independientemente por temas o niveles que se asociarían con cada uno de los nodos de información de cada contexto.

4.7 Información bajo demanda

Un ejemplo de aplicación en Eduka de este patrón puede verse en la visualización de la videoconferencia. En tiempo real y mientras se está recibiendo las tramas de audio y vídeo, el alumno puede: aumentar la ventana de recepción de vídeo, re-escalándose así la imagen (*zoom-in*), disminuirla (*zoom-out*), elegir recibir únicamente la señal de audio para ahorrar ancho de banda, elegir recibir únicamente la señal de vídeo o recibir las dos. Todo sobre el mismo nodo de información que constituye la videoconferencia en tiempo real.

5. Conclusión

Hemos conseguido facilitar la comprensibilidad y el acceso de la información mediante un sistema multimedia complejo cuyo coste de desarrollo es muy alto en comparación con sistemas hipertextuales más tradicionales. Con la aplicación en el diseño de nuestro sistema del conjunto de patrones que se muestran en este artículo se ha conseguido un elevado nivel de eficiencia, una rápida familiarización de los

alumnos con el sistema de navegación empleado, y una clara diferenciación entre navegación e información gracias a al concepto de ventana en sí.

Esperamos que con este trabajo los autores de aplicaciones multimedia para la enseñanza puedan ver un enfoque distinto de este tipo de aplicaciones y como los patrones de diseño puede ayudar no sólo a solventar sino también a identificar problemas que comúnmente se presentan en estas aplicaciones. Creemos que este tipo de sistemas puede compensar el coste de desarrollo con unos niveles muy altos de eficacia en la tarea educativa.

Referencias

- [Astigarraga99] E. Astigarraga, X. Carrera: "Formación en tecnología básica mediante nuevas tecnologías". EDUTEC-99. Las Nuevas Tecnologías para la mejora educativa. Ed. Kronos, 2000, p 112.
- [Escalona02] M. J. Escalona, M. Mejías, J. Torres y A. Reina: "NDT: Una técnica para el desarrollo de la navegación". Artículo para el 5º Workshop iberoamericano de Ingeniería de Requisitos y Ambientes Software. La Habana (Cuba). Abril de 2002.
- [Lyardet98] F. D. Lyardet, G. H. Rossi, D. Schwabe: "Using Design Patterns in Educational Multimedia applications", ED-MEDIA'98.
- [Márquez02] J.M. Márquez. "Eduka, primer estudio de aproximación a un sistema de educación semipresencial". Proyecto Fin de carrera para la obtención del título de Ingeniero en Informática. Departamento de Lenguajes y Sistemas Informáticos. Escuela Técnica Superior de Ingeniería Informática. Universidad de Sevilla. Junio 2002.
- [Rossi96] G. Rossi, D. Schwabe: "The Object-Oriented Hypermedia Design Model". <http://www-lifia.info.unlp.edu.ar/~fer/oohdnv/>
- [Schwabe96] D. Schwabe, G. Rossi, y S. D. J. Barbosa. "Systematic Hypermedia Application Design with OOHD". Hypertext'96 (HT96). Washington, Marzo 1996.
- [SUN01] Sun Microsystems Inc. "Java Look and Feel design guidelines. Second Edition". <http://java.sun.com/products/jlfd2/book/index.html> Febrero 2001.
- [WSI02] Wall Street Institue. "The Village, the Wall Street Institute Student Community". Extranet mundial: <http://www.e-wsi.com>