

## Marco conceptual de ayuda al aprendizaje basado en la colaboración y el conocimiento

**Manuel E. Prieto**  
**Ramón Manjavacas**  
**Miguel A. Laguna**

### Sinopsis

*Una característica deseable de los sistemas de ayuda al aprendizaje colaborativo es que deben proporcionar soporte para que los estudiantes resuelvan problemas nuevos, utilizando experiencias anteriores mientras intercambian información y conocimiento con el resto de los miembros del grupo. Se ha podido comprobar que, durante las interacciones, tienen lugar diálogos y comportamientos que, en principio no contribuyen a mantener un ambiente de colaboración adecuado a la resolución de problemas.*

*En nuestra investigación deseamos resolver ciertos problemas que surgen durante las sesiones sincrónicas de aprendizaje, utilizando resultados de los sistemas de trabajo en grupo y modelos de representación del conocimiento. El aprendizaje entendido como el proceso de perfeccionamiento de las habilidades de solución de problemas procedimentales. El conocimiento considerando los diversos aspectos que tienen lugar durante la interacción de los alumnos con el sistema.*

*Se ha desarrollado un Modelo de Estudiante Simulado (MES) el cual intenta detectar e impedir la ocurrencia de situaciones negativas en las sesiones de resolución de problemas de programación.*

*El presente trabajo se centra en las dos direcciones principales de desarrollo actual de la investigación: por una parte se intenta formalizar el MES, y por otra se diseña un marco conceptual y una herramienta genérica de desarrollo de aplicaciones de aprendizaje colaborativo.*

*Términos claves: <Programas informáticos> <modelos educativos> <trabajo en equipo> <proceso cognoscitivo> <solución de problemas> <aprendizaje>*

### Abstract

*Collaboration can be useful in procedural learning, but sometimes situations arise that hinder desired results. Online collaboration tools can be utilized for significant change in adapting student-centred approaches. We attempt to solve some Computer Aided Learning problems with knowledge and collaboration approaches. Learning understood as the processes to improve problem solving or procedural (intellectual) abilities. Knowledge*

---

*considered not only in the sense of object-matter symbolic processing, but generally in any way of meaning acquisition during human-computer interaction.*

*We proposed to add a Simulated Student in the learner's group in order to avoid some of these undesired situations. We describe the main concepts of the Simulated Student Model (SSM) that has been implemented in a concrete tool to help college students to develop good habits in programming.*

*Actually two leading lines emerged in our work. One is related with the definition of a framework, which extends and generalizes the SSM. The second focuses in the construction of a formal model of the SSM. This paper describes the work in progress in both directions*

*Key terms: < Computer programs > < educational models > < teamwork > < cognitive processes > < problem solving > < learning >*

### Introducción

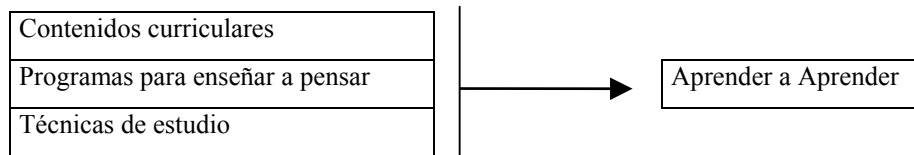
Construir conocimiento es un proceso de negociar significados. En una sesión de aprendizaje seguramente todos van a aprender algo, pero lo harán de manera distinta, porque depende de la estructura previa de cada uno de los sujetos. Esa estructura está formada por sus conocimientos, por sus experiencias, por una conformación que hace que el sujeto sea único y se relacione con el aprendizaje de manera distinta (Chistik, 2001).

Por otro lado los computadores no pueden considerarse simplemente como herramientas, sino como medios para extender la capacidad intelectual de los estudiantes. Cuando los estudiantes utilizan tecnología como herramienta de productividad o

comunicación, están relacionados con un nivel superior de pensamiento para tomar decisiones sobre como reunirse, organizarse, analizar y compartir /-\*-información. Al utilizar esta tecnología para llevar a cabo tareas desafiantes, estos mismos estudiantes toman la iniciativa de diseñar su propio proceso de enseñanza y regulan su ritmo de aprendizaje.

### Estrategias cognitivas, intervención educativa y meta-aprendizaje

En algunos Sistemas Educativos como el español, se sitúa a los programas destinados a enseñar a pensar en el núcleo del aprender a aprender, junto con los contenidos curriculares y las técnicas de estudio (Ministerio de Educación y Ciencia, 1992).



El objetivo de los programas cognitivos es la adquisición, por parte de los alumnos, de estrategias cognitivas que faciliten sus modos de seleccionar, adquirir, organizar o integrar nuevos conocimientos.

Las intervenciones educativas que persiguen el éxito académico y el apoyo ante las dificultades de aprendizaje se deducen del modelo de Brown adaptado por Repetto el cual incluye cuatro fuentes de influencia en la situación de aprendizaje:

1. las características del aprendiz (aptitudes, rasgos de la personalidad, conocimientos previos y condiciones socioculturales);
2. la naturaleza de los materiales por ser aprendidos (Modalidad visual o lingüística, estructura física, estructura psicológica, dificultad conceptual, orden y secuencia de los materiales, metodología didáctica apropiada ...);
3. las tareas criterios (reconocimiento, recuerdo, transferencia ... );

4. las tareas de aprendizaje que el aprendiz aplica (atención, estructuración, elaboración, solución de problemas, memorización, comprensión).

Los modelos de aprendizaje como el de Brown-Repetto, constituyen un marco conceptual conveniente para la realización de los sistemas computarizados de apoyo al aprendizaje.

Algunos autores sugieren que la interacción social puede ser una forma de aumentar la motivación y obligación personal en el aprendizaje. Esto se debe a que al trabajar en equipo el error de una persona repercute en el trabajo de los demás por lo que los alumnos realizan un mayor esfuerzo por hacer las cosas bien, y este se suele ver recompensado por un aumento de las habilidades cognitivas. Una colaboración eficiente es una situación en la cual un alumno puede aprender de otros alumnos mientras que él realiza una tarea que demanda o precisa la aplicación de elementos de conocimiento que poseen

los otros alumnos, y este conocimiento se considera asequible para su asimilación (Ayala y Yano, 1994).

Para nuestro grupo de investigación es muy importante el problema de la adquisición del conocimiento en función del desarrollo de tutores/entrenadores inteligentes, que incorporan representaciones basadas en reglas del dominio objeto de estudio, estos requieren que el autor construya un sistema experto, una tarea que usualmente involucra gran cantidad de programación y/o habilidades de ingeniería del conocimiento (Murray, 1997).

#### Evolución de nuestras investigaciones Modelo del Estudiante Simulado

El Estudiante Simulado ha sido el primer intento dentro de nuestro grupo para obtener un modelo que soporte el concepto de Zona de Desarrollo Próximo (de Lev. S. Vygotsky) en el aprendizaje procedimental en condiciones de colaboración síncrona y distribuida. Se ha desarrollado una implementación de este sistema llamada HabiPro con el objetivo de ayudar a los estudiantes de programación orientada a objetos (Vizcaíno, 2001).

La aplicación considera dos tipos principales de actividad: la colaboración y la comunicación. Cada actividad se encuentra en la interfaz en forma de áreas de trabajo. En la primera se presentan problemas relativos a la programación con el lenguaje Java, y están contenidos los aspectos de control de piso, control de presencia y mecanismos de votación. En la

segunda los estudiantes pueden comunicarse mediante un chat.

El Estudiante Simulado representa a un agente virtual que colabora con los usuarios del sistema y que intenta conducir y ayudar al máximo en la tarea de aprendizaje. Es un agente activo que participa en la colaboración de igual manera como lo haría otro estudiante. Durante la resolución de ejercicios, pueden ocurrir situaciones negativas como conversaciones no relacionadas con la resolución del ejercicio, alumnos que no interactúan o alumnos que se encuentran desorientados. El Estudiante Simulado tiene la capacidad de detectar estas situaciones y de actuar para evitar sus efectos negativos. La estructura del Estudiante Simulado se compone de una base de conocimientos que contiene un modelo de cada estudiante y un modelo por cada grupo de estudiantes, así como información acerca de la materia a tratar y las reglas que describen los roles, los mecanismos de intervención y las formas externas de actuación.

#### Descripción del Modelo

El modelo del Estudiante Simulado tiene tres componentes principales: Modelo del Estudiante individual (SM), Modelo del Grupo (GM), y Modelo del Comportamiento del Estudiante Simulado (SSBM). También tiene dos módulos complementarios: la Interface y el Administrador de la Información (ver Figura 1).

Figura 1. Configuración del Modelo del Estudiante Simulado.

#### HabiPro

HabiPro (Hábitos de Programación) es una aplicación Cliente-Servidor para desarrollar buenos hábitos de programación. Es un sistema de aprendizaje colaborativo, síncrono, distribuido, en el cual, los estudiantes aprenden a comprender y depurar programas, a desarrollar buenos estilos de programación y también pueden resolver problemas en forma colaborativa.

En el desarrollo de programas, HabiPro permite a los estudiantes: encontrar y corregir errores en un programa, predecir el resultado de un programa y completar un programa. HabiPro puede proporcionar pistas, mostrar mensajes de compiladores e incluso mostrar un ejercicio similar corregido, permitiéndole al estudiante desarrollar un buen estilo de programación eligiendo nombres adecuados para las variables, métodos, clases, etc. y comparando sus programas, (ver Figura 2).

### Arquitectura de HabiPro

El Cliente esta formado por cuatro módulos: Interface, Administrador de la Información, Conocimiento Colaborativo, Modelo del Estudiante y Marco de Comunicación. El Servidor lo integran tres

componentes: el Modelo del Comportamiento del Estudiante Simulado, el Modelo de Grupo y el componente Problemas que está subdividido en Problemas y Respuestas, además del Marco de Comunicación .

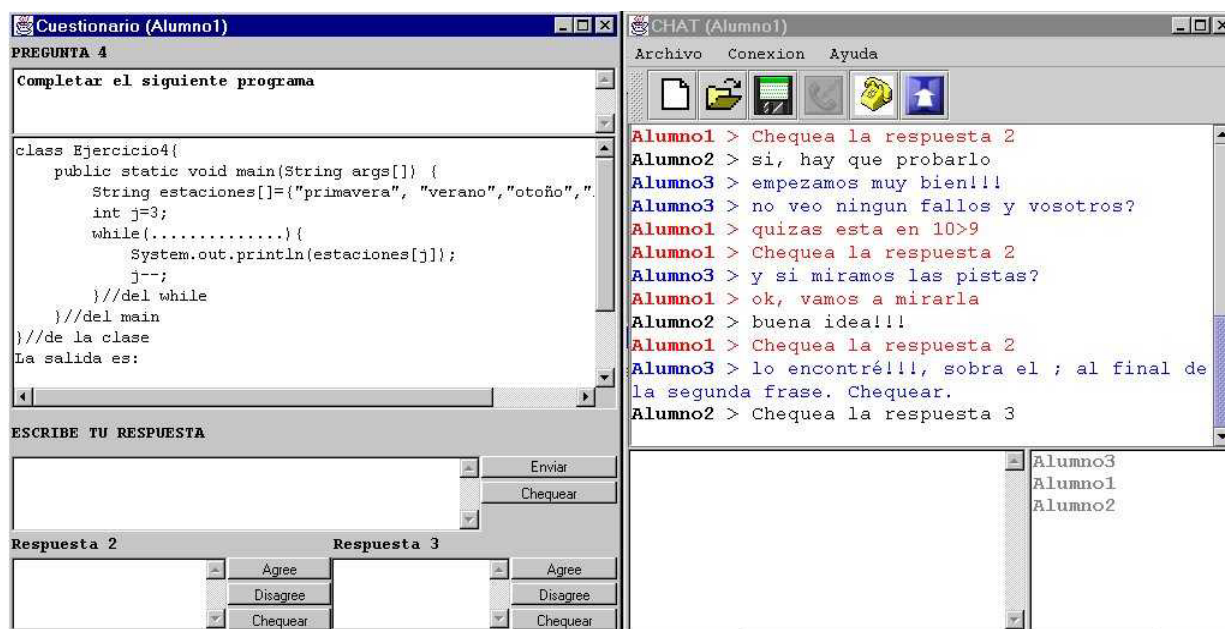


Figura 2. Interfaz de HabiPro.

### Líneas de investigación actuales Marco conceptual para la construcción de sistemas de Ayuda al Aprendizaje basados en la Colaboración y el Conocimiento

El Marco conceptual contiene una definición general de componente, sus atributos y eventos así como las relaciones entre componentes. El objetivo es potenciar el aprendizaje de habilidades que ayuden a la resolución de problemas sobre la materia que esté

desarrollando y que estas habilidades puedan convertirse en "hábitos".

El concepto de "ayuda" parte de la idea de tutoriales guiados, que acompañan en cualquier momento a un usuario a solventar una duda surgida en su camino de descubrimiento de nuevos conocimientos, hasta la idea de la existencia en estos sistemas de la figura del "Estudiante Simulado" (Vizcaíno, 2001). Esta figura representa a un agente virtual que colabora con los usuarios del sistema y que

intenta conducir y ayudar al máximo en esa tarea de aprendizaje. Por otra parte, en la tesis doctoral (Vizcaino, 2001), se ha ofrecido un sistema de ayuda al aprendizaje basado en la colaboración y el conocimiento.

Debido al carácter informal de la solución propuesta para el Estudiante Simulado y a la especificidad de la herramienta creada, se ha planteado la necesidad de la definición de un marco de trabajo general que extienda y generalice el desarrollo de sistemas de ayuda al aprendizaje basados en la colaboración y el conocimiento.

Una de las características más relevantes de este marco será la concepción de este tipo de sistemas mediante la integración de componentes software. Esto es además, una de las nuevas tendencias surgidas dentro del campo del CSCL y CSCW (CSCW & Groupware, 2002), las cuales conciben el desarrollo de sus sistemas a partir de arquitecturas software basadas en componentes.

El desarrollo de software basado en componentes (CBSD – Component Based Software Development) es una de las disciplinas importantes dentro del campo de la Ingeniería de Software (CBSD – CBSE, 2002).

La tecnología CBSD tiene como objetivo la construcción de grandes aplicaciones software mediante la integración de componentes. De esta forma, se potencia la flexibilidad y mantenibilidad de los sistemas y se reducen los costes del desarrollo y mantenimiento del software.

Uno de los requerimientos clave en la construcción de un componente es la definición de una interfaz pública que permita comunicarse con el resto de componentes con los que necesite interactuar.

Las cuatro actividades principales que caracterizan el desarrollo basado en componentes (Brown y Wallnau, 1996) son:

- evaluación: consiste en buscar componentes que se ajusten al problema a resolver, evaluar la madurez de los mismos mediante criterios propuestos por la ISO (Information Technology..., 1991) o IEEE (IEEE recommended..., 1993) para dicho cometido y elegir el más apropiado;

- adaptación: consiste en realizar las operaciones necesarias para adaptar el componente seleccionado a las necesidades, puesto que en la creación del mismo podrían no haberse asumido todos los requerimientos actuales. Esa adaptación será posible según el grado de accesibilidad a la estructura interna del componente, presentándose en grado como (Valetto y Kaiser, 1995): caja blanca (acceso completo al código fuente), caja gris (acceso restringido proporcionando un lenguaje de extensión o una interfaz de programación de aplicación (API)) o caja negra (solo acceso a las operaciones que proporciona);
- ensamblar en el sistema: los componentes deben ser integrados a través de una infraestructura bien definida que proporcione la unión e integración;
- evolución del sistema: mantenimiento del sistema mediante la actualización de componentes.

Uno de los aspectos relacionados con esta tecnología es la ausencia de lenguajes estándar de composición de componentes, existiendo en algunos casos, lenguajes específicos para una plataforma, o en otros casos, lenguajes genéricos que se encuentran en desarrollo. Este aspecto limita la adopción de un proceso de desarrollo basados en componentes y aumentan los costes asociados a ellos.

La elección de un lenguaje de composición de componentes apropiado que favorezca el desarrollo basado en componentes será también uno de los aspectos considerados en este proyecto.

Pretendemos integrar un marco conceptual de trabajo para la construcción de sistemas de ayuda al aprendizaje basados en la colaboración y el conocimiento, tomando en cuenta tanto las posibles categorías de herramientas que pueden integrarse en sistemas de esta clase, como las nuevas tendencias relacionadas con la Ingeniería de Software dentro de este campo de estudio.

Este entorno de desarrollo será lo suficientemente abierto para permitir la introducción de componentes software que añadan nuevas funcionalidades para futuros sistemas o de nuevas versiones mas actualizadas de componentes ya

existentes en el entorno. La extensión se hace definiendo un campo conceptual del que se hará una implementación en XML.

Esta herramienta será construida para un entorno Windows 2000 y posteriores y su implementación se llevará a cabo mediante el entorno de desarrollo Microsoft Visual Studio.Net Beta2. Este entorno de desarrollo rápido favorece la construcción de aplicaciones software que presentan una arquitectura basada en componentes.

#### Agente Virtual de Ayuda a la Colaboración y el Aprendizaje

Como parte de nuestra línea de investigación se trabaja en dotar al Estudiante Simulado de funciones adicionales y de reforzar las existentes, a través de patrones de diseño y diseño basado en componentes. Se trata de incrementar las capacidades del agente artificial que interactúa en la colaboración sincrónica durante las sesiones en las que los alumnos reales intentan desarrollar el aprendizaje procedimental (saber hacer). Consideramos ayudar a los estudiantes a desarrollar conocimiento, habilidades y destrezas, creando un ambiente de aprendizaje interactivo, multidisciplinario y explorativo, orientando nuestra aplicación al aprendizaje centrado en el estudiante.

En este modelo centrado en el estudiante, ellos son responsables de su propio aprendizaje y están motivados a utilizar la tecnología para seguir caminos personalizados de conocimiento (Gates, 2001).

Los objetivos establecidos son:

1. formalizar y generalizar el Concepto de Estudiante Simulado, para que sea independiente del contexto en el que se aplique; hacer su comportamiento programable; generalizarlo mediante la utilización de patrones de diseño, para que pueda ser aplicado a cualquier tipo de ambiente colaborativo;
2. crear un componente software del Agente que sea independiente del contexto, configurable y adaptable a cualquier entorno de aprovechamiento;

3. construir una aplicación software que permita observar el comportamiento del Agente;
4. la elección de un lenguaje de composición de componentes apropiado que favorezca el desarrollo basado en componentes.

Para la formalización y generalización el Concepto de Estudiante Simulado se hará uso del Proceso Unificado de Modelado y de la Teoría de Patrones de Diseño (Vlissides, Helm, Johnson y Gamma, 1995).

Las etapas a seguir en este proyecto son:

1. análisis del Modelo del Estudiante Simulado para comprender su naturaleza y abstraer los conceptos genéricos;
2. establecer el sistema basado en el conocimiento que contendrá el Agente para la realización de tareas de análisis, identificación de situaciones y actuaciones procedentes;
3. especificación de los patrones de diseño del Agente;
4. desarrollar el proyecto con base de un ciclo de vida en espiral. Los ciclos completos que se obtengan en las etapas iniciales concluirán en prototipos del Agente de manera que se vaya añadiendo más funcionalidad;
5. verificar y validar el Agente al final de cada ciclo, para iniciar el siguiente con nuevos requisitos. Esta etapa puede realimentar la especificación del patrón.

Para la elaboración del componente agente y para la aplicación del ejemplo se utilizará la plataforma Microsoft Visual Studio.NET Beta2. Este entorno de desarrollo rápido favorece la construcción de aplicaciones software que presentan una arquitectura basada en componentes.

Para las tareas de rápido prototipaje y la construcción del sistema basado en el conocimiento se pretende utilizar herramientas declarativas como el lenguaje Prolog (Bratko, 1991). No es objetivo del Agente el procesamiento del lenguaje natural. Sin embargo el Estudiante Simulado debe extraer información de la conversación y debe participar en la misma. Para ello se usarán métodos de procesamiento práctico del lenguaje natural (Russell y Norving, 1996)



junto con otras técnicas de inteligencia artificial como lógica borrosa.

Por último, se utilizarán las librerías de manejo de archivos XML para el almacenamiento de información. El uso de este estándar facilitará la comunicación con otras aplicaciones y/o agentes en caso que fuera necesario.

#### Trabajos Futuros

Nuestro objetivo es contribuir a que los estudiantes puedan tener un mayor control sobre el proceso de aprendizaje y lograr la optimización de los resultados obtenidos. Según investigaciones recientes, el nivel de retención e internalización de los conceptos resulta favorecido cuando los estudiantes utilizan más de un sentido durante el proceso de aprendizaje. Para ello en nuestro grupo de investigación nos hemos abocado a la tarea de revisar el estado del arte de la colaboración en el aprendizaje para “fabricar” un entorno de desarrollo que contenga un repositorio de componentes y un lenguaje de especificación de construcción de componentes, al menos para aplicaciones cliente-servidor que sirva para crear aplicaciones de Ayuda al Aprendizaje Colaborativo.

Habrán 6 categorías o tipos de componentes para:

1. la comunicación;
2. la colaboración;
3. la organización del trabajo en grupo;
4. procesamiento de datos multimedia;
5. el manejo del conocimiento;
6. promoción del aprendizaje.

Una base colaborativa mediante el uso de una interfaz única común para todas las aplicaciones, que permita disponer de los siguientes recursos:

- alta resolución de imágenes estáticas y en movimiento, incluyendo fotos, diapositivas y videos;
- plug-ins que adicionan recursos y visualizaciones al browser de navegación en la web;
- sonidos, voz y música;
- interactividad síncrona con recursos de texto, voz e imágenes;

- estructura hipermedia que combina todos los recursos y formatos de media en la base digital.

Con las herramientas de comunicación colaborativas del correo, foro, chat y video conferencia se desarrollan actividades de grupo y debate que aportan un importante elemento de socialización en el aprendizaje colaborativo. Cada uno de estos componentes le aportará al estudiante una parte cada vez más importante de su tiempo a crear, manejar o proyectar información.

Se va a construir un tipo de ayuda al aprendizaje dependiendo si el alumno es un sujeto activo o pasivo, proporcionando un dinámico intercambio de conocimientos e información entre todos los miembros. Los estudiantes no deben sentirse solos durante el proceso de aprendizaje, así que la retroalimentación y el refuerzo frecuente y dirigido es importante. El estudiante en todo momento podrá recurrir a variados sistemas de ayuda: tutor, grupo o recursos de apoyo. El más importante es el tutor, con quien se podrá contactar en tiempo real (en modo texto a través de chat) o en diferido (a través de correo electrónico). Nuestro agente virtual evalúa el progreso del alumno, analiza la producción individual y colectiva, así como su profundidad, estimula la participación colaborativa y establece reglas de comportamiento.

#### Conclusiones

El aprendizaje colaborativo no es tanto una técnica sino una arquitectura, porque aplica numerosos componentes entrelazados para incentivar un ambiente de aprendizaje. Este aprendizaje se basa en la asistencia de facilitadores capaces de organizar el contenido inteligentemente y poseer la suficiente adaptabilidad para aportar las soluciones creativas necesarias y ayudar a los estudiantes para desarrollar habilidades en el uso de tecnologías. Hay que dar más importancia al aprendizaje y por tanto es imprescindible mejorar nuestro conocimiento sobre los procesos de aprendizaje y de representación del conocimiento en las personas.

La construcción de sistemas colaborativos para el aprendizaje requiere un conocimiento

interdisciplinar, puesto que es necesario saber que factores influyen en el aprendizaje y que factores influyen en la dinámica de trabajo en grupo. La finalidad de nuestro trabajo es desarrollar ambientes de trabajo fáciles de usar, útiles y que sobre todo, fomenten el aprendizaje y las habilidades sociales.

Este proyecto pretende definir un marco conceptual para la construcción de Sistemas de Ayuda al Aprendizaje Basados en la Colaboración y el Conocimiento. En la especificación de este marco se consideran tanto las posibles categorías de herramientas que pueden integrarse en sistemas de esta clase, como las nuevas tendencias relacionadas con la Ingeniería de Software dentro de este campo de estudio.

El objetivo final de este marco conceptual de trabajo es extender y generalizar la tarea de construcción de sistemas de este tipo como impulsor tecnológico y proveedor de infraestructura para la

administración del aprendizaje y del conocimiento mediante la utilización de todos los componentes de la tecnología desarrollando modelos eficaces de aprendizaje y tecnología para facilitar e incrementar el acceso a la educación y a la capacitación a nivel individual y grupal.

Como resultado práctico se aportará un Entorno de Desarrollo para sistemas de este ámbito de estudio, que refleje la filosofía especificada en el marco referido. En la descripción de la solución en los patrones de ese catálogo o lenguaje de patrones, debe aparecer código (componentes) en la especificación XML.

Este Entorno de Desarrollo será lo suficientemente abierto para permitir la introducción, ya sea de componentes software que añadan nuevas funcionalidades para futuros sistemas, o bien de nuevas versiones más actualizadas de componentes ya existentes en el entorno.

### Referencias

- Ayala, G. and Yano, Y. (1994). Design Issues in a Collaborative Intelligent Learning Environment for Japanese Language Patterns, Educational Multimedia and HyperMedia. Vancouver, Canada: AACE
- Ayala, G. and Yano, Y. (1995). Interacting with a Mediator Agent in Collaborative Learning Environments. Symbiosis of Human and Artifact: Future Computing and Design for Human-Computer Interaction, 20A, y. Anzai, K. Ogawa and H. Mori (eds.). Advances in Human Factors/Ergonomic, Elsevier Science Publishers.
- Bratko, I. (1991). Prolog Programming for A.I., Addison Wesley.
- Brooks, F. Jr. (1987). No Silver Bullet: Essence and Accidents of Software Engineering, Computer 20, 4. Abril.
- Brown, A. and Wallnau, K. (1996). Engineering of Component-Based Systems, 7-15. Component-Nased Software Engineering: Selecte Papers from the Software Engineering Institute. CA: IEEE Computer Society Press. Los Alamitos.
- CBSD – CBSE. (2002). The Software Engineering Institute (SEI) – Carnegie Mellon University. Encontrado en: [http://www.sei.cmu.edu/str/descriptions/cbsd\\_body.html](http://www.sei.cmu.edu/str/descriptions/cbsd_body.html).
- Chistik, M. (UBA). (2001). Construir estrategias de enseñanza. Encontrado en: <http://www.competir.com>.
- CSCW & Groupware. (2002). Encontrado en <http://www.cs.tcd.ie.Sotirios.Terzis/CSCW.html>.
- Gates, B. (2001). Sistema Nervioso Digital para la Educación., Microsoft Corporation. Encontrado en: <http://www.microsoft.com/latam/educacion/dns/>.
- (1993). IEEE Recommended Practice on the Selection and Evaluation of CASE Tools (IEEE std. 1209-1992). NY: Institute of Electrical and Electronic Engineers, New York.
- Information Technology – Software Produc Evaluation – Quality Characteristics and Guidelines for their Use. International Standards Organization/ International Electrochemical Commission, Geneve, Zwitterland. 1991.
- Murray, T. (1997). Expanding the Knowledge Acquisition Bottleneck for Intelligent Tutoring System. IJAIED97. Encontrado en: <http://www.cs.umass.edu/~tmurray>.
- Russell, S. and Norvig, P. (1996). Inteligencia Artificial, Un enfoque moderno. Prentice-Hall Hispano Americana.
- Valetto, G. and Kaiser, G. E. (1995). Enveloping Sophisticated Tools into Computer-Aided Software Engineering Environments. Proceedings of 7<sup>th</sup> IEEE International Workshop on CASE. Toronto, Ontario, Canada, July 10-14, 1995. Ca: IEEE Computer Society Press. Los Alamitos.
- Vizcaíno, A. (2001). Enhancing collaborative learning using a simulated student agent. – UCLM.
- Vlissides, J., Helm, R., Johnson, R., and Gamma, E. (1995). Design patterns. Addison Wesley.
- Wan, D. and Johnson, P. (1994). Experiences with CLARE: a computer-supported collaborative learning environment. Human-Computer Studies. (Eds). Academic Press.
- Wirfs-Brock, R., Wilkerson, B. y Wiewer, L. (1990). Designing Object-Oriented Software. Prentice Hall.