

Hacia la Nueva Generación de Sistemas de Aprendizaje Basado en la Web

Towards the New Generation of Web-Based Learning Environments

Leonid B. Sheremetov¹ y Vladimir L. Uskov²

¹Instituto Mexicano del Petróleo, Programa de Investigación en Matemáticas Aplicadas y Computación

Eje Central Lázaro Cárdenas No. 152, Col. Sn. Bartolo Atepehuacan, C.P.07730, México D.F.

Centro de Investigación en Computación-IPN, Laboratorio de Agentes

Juan de Dios Bátiz s/n esq. Miguel Othón de Mendizabal, CP. 07738, México, D.F.

²Facultad de Ciencias de Computación y Sistemas de Información del Centro de Investigación de Tecnologías de Educación

Basadas en Internet, Universidad de Bradley, Illinois

1501 W. Bradley Avenue, Peoria IL 61625, USA

e-mail : sher@cic.ipn.mx, uskov@bradley.edu

Artículo recibido en septiembre 26, 2001; aceptado en enero 23, 2002

Resumen

El artículo presenta varios enfoques modernos al desarrollo y organización de sistemas de educación y capacitación basadas en la Web. Se describen dos prototipos de sistemas de educación y capacitación virtual, EVA (Espacios Virtuales de Aprendizaje) e INTERLABS, contruidos sobre las tecnologías avanzadas de información, servlets de Java, streaming de audio y video, respectivamente. Se presentan agentes asistentes personales y una infraestructura multiagente como el centro aglutinador de la integración de los dos ambientes en un sistema de educación y capacitación de nueva generación.

Palabras clave: educación basada en la Web, clases vía Web, tecnologías de información, agentes de software.

Abstract

The article describes several modern approaches to a design, development and organization of Web-based learning and training systems of a new generation. Two prototypes of such systems are described, particularly, EVA (Virtual Spaces of Learning) and INTERLABS. Those prototypes are built on the advanced information technologies such as Java servlets and streaming multimedia technologies, respectively. Personal assistant agents and multiagent infrastructure as agglutinative tools of integration of those two virtual learning environments are described in details.

Key words: Web-based education, Web-lecturing technology, information technologies, software agents.

1 Introducción

Hoy en día, el mercado de servicios educativos en la red mundial Internet y los sistemas de cómputo más modernos de educación y capacitación virtual (SECV), se desarrollan en forma impetuosa. Conforme al reporte de la Asociación Internacional de Tecnologías de la Información, en 1995 el mercado mundial de sistemas de educación virtual ha sido estimado aproximadamente en 19 millones de dólares estadounidenses. Conforme a las presentaciones en el Congreso Mundial de tecnologías de Web (Proceedings of the WebNet, 2001), el mismo mercado mundial se proyecta a alcanzar el nivel de 54.1 mil millones de dólares estadounidenses para el año 2005.

El desarrollo de los primeros SECV empezó prácticamente al mismo tiempo que la Web misma, en 1992. Los primeros esfuerzos se concentraron en el desarrollo de los cursos en línea extendidos con una colección de herramientas cumpliendo diferentes funciones de enseñanza virtual. Entre las herramientas de mayor difusión se pueden mencionar las siguientes:

1. correo electrónico y el envío automático a grupos de usuarios (alumnos), como Hypernews, NewsXpress, Newswatcher, Agent, Gravity, Eudora, Pegasus, etc.;
2. las paginas Web estáticas en el formato HTML utilizando los convertidores de texto a html como RTFtoHTML, Wp2x, Latx2html, Hyperlatex, Txt2html, Webify, o verificadores WebLink, HTML Validation Service, and Net Mechanic;
3. herramientas de intercambio de mensajes como iChat, HyperChat, Palace, TeleCafe, Active Worlds y, posteriormente, audio y video conferencias como VDophone, Internet Phone, WebPhone, CU-SeeMe, Big Picture, VideumConfPro;

4. herramientas de almacenamiento e intercambio estructurado de información y archivos, tales como Fetch, WS-FTP, CuteFTP;

La desventaja principal de sistemas de este tipo, que algunos autores llaman los SECV de la primera generación, es la ausencia de integración e interacción entre los componentes. Como consecuencia, en la segunda fase, se desarrollaron diferentes sistemas de Administración de Cursos en línea que integren estas herramientas en un entorno de educación virtual.

Esta integración permite realizar las funciones de planificación y administración del proceso de aprendizaje, soporte de configuración y acceso a los materiales educativos, tareas y asignaciones (que incluye autenticación de los usuarios e instalación automática de los componentes de software necesarios en la maquina del cliente entre otras), evaluación de conocimiento y avances de alumnos por medio de generación y aplicación automática de exámenes, y comunicación entre los integrantes de cursos (alumnos y profesores en forma síncrona y asíncrona). Actualmente, la gran mayoría de las instituciones educativas de EE.UU y México ofrecen varios cursos en línea y los planes de estudios enteros sobre Internet utilizando las herramientas de educación y capacitación de la segunda generación tales sistemas como WebCT, Learning Space, BlackBoard, Virtual-U, CourseInfo, etc. (Proceedings of the WebNet, 2001). Sin embargo, *E-learning* en sistemas de este tipo sigue siendo básicamente *E-reading* debido al uso escaso de los medios audio-visuales de adquisición de conocimientos.

Recientemente, las universidades principales de EE.UU, como la universidad de Stanford, la Universidad de Berkeley, la universidad Carnegie-Mellon y la universidad de Michigan (Ann-Arbor), desarrollan e implementan la nueva tecnología de la educación y capacitación, para ser exactos, una nueva tecnología de enseñanza - la Tecnología de clases vía Web (*Web-Lecturing Technology*) activa (Uskov V., 2001), (Brusilovsky P., 2000). En lugar de una clase tradicional cara-a-cara esta tecnología proporciona una clase impartida por un maestro virtual. *Web-Lecturing* esta basada en uso activo de a) tecnologías de multimedia (de audio, de vídeo, y de animación), y b) los canales de alta velocidad de transmisión de la información sobre Internet.

Simultáneamente, se esta trabajando sobre los sistemas de educación y capacitación conceptualmente contruidos sobre la base de un paradigma *de educación activa* (Uskov V., 2001). El aprendizaje activo o aprendizaje centrado en el alumno como un paradigma moderno de aprendizaje significa que el aprendiz, sus exigencias y su comodidad están en el centro de educación. Si un aprendiz tiene un ambiente de aprendizaje cómodo, entonces su educación tiene una eficacia máxima. Las principales ventajas de aprendizaje activo son:

- 1) La responsabilidad del estudiante e iniciativa para promover la propiedad del aprendizaje y las habilidades transferibles,
- 2) Las estrategias intencionales de aprendizaje, los métodos explícitos de aprendizaje, la reflexión de procesos de aprendizaje para el desarrollo de habilidades de meta cognición,
- 3) Las tareas y proyectos de solución de problemas dirigida por metas genera los productos de aprendizaje de valor,
- 4) Los maestros como *facilitadores*, entrenadores y guías, y no como fuentes de conocimiento, que estimula discusión entre maestros y aprendices,
- 5) Los contextos por aprender, tomados de problemas del mundo real, las estrategias de valoración auténticas para evaluar habilidades del mundo real, y
- 6) El aprendizaje en forma cooperativa y en equipos.

La introducción del paradigma de aprendizaje activo en la ECV implica la búsqueda de nuevas tecnologías de información para su apoyo. Participando en los principales congresos sobre las tecnologías de información en educación, celebrados en 2001, tales como ED-MEDIA-2001, WEBNET-2001, IEEE/ASEE FIE-2001, TELEMÁTICA-2001, etc. (Proceedings of the WebNet, 2001), (Montgomerie C. and Viteli J., 2001), (Proceedings of the IEEE/ASEE, 2001), (Proceedings of the TELEMÁTICA, 2001), (Proceedings of the DOE, 2001), (Proceedings of the e-Learning Conference, 2001), los autores del presente artículo han marcado una tendencia principal en la construcción de SECV que consiste en la integración de:

1. Los ambientes interactivos de aprendizaje contruidos sobre la base de clases vía Web (*Web lecturing*), la reproducción de clases en tiempo real (*video-based messaging*) con el uso activo de streaming de audio y vídeo (*multistreaming*).
2. Los ambientes de interacción basados en juegos interactivos y mundos virtuales distribuidos, contruidos sobre objetos de uso múltiple.
3. Los ambientes inteligentes contruidos en base a los principios y la tecnología de agentes inteligentes, redes de suministro de conocimiento (*Knowledge Supply Chain Networks*), reconocimiento y generación del habla para las interfaces flexibles hombre-maquina.
4. Tecnología de portales que permiten adaptar y personalizar la información almacenada en la Web y actualizarla automáticamente.
5. Lenguajes de programación y especificación de aplicaciones para Internet e Internet2, tales como *eXtensible Markup Language* (XML), *Synchronized Multimedia Integration Language* (SMIL) y SCORM

(Sharable Content Object Reference Model), desarrollados por el World Wide Web Consortium y Co-Lab de Aprendizaje Distribuido Avanzado (*Advanced Distributed Learning - ADL Co-Lab*) respectivamente (<http://www.w3.org/TR/REC-smil/>, <http://www.w3.org/XML/>).

6. Tecnología de Web semántica basada en la integración de los fragmentos aislados de la Web en un sistema global en base a una forma común de acceso a datos para los usuarios y las aplicaciones educativas Web.

Las razones mencionadas anteriormente han motivado a los autores de este trabajo y a los grupos de investigación liderados por ellos comenzar en 2000 las investigaciones en el campo de la integración de las tecnologías modernas de educación y capacitación basadas en Internet, y también sobre la creación de los sistemas de educación de una nueva generación. Este trabajo está basado en las experiencias en el desarrollo de los ECV EVA e INTERLABS, así como en el desarrollo de cursos en línea de educación y capacitación y planes de estudios basados en ellos. Su foco principal es mostrar la manera de diseñar y desarrollar sistemas de aprendizaje virtual inteligentes, basados en el conocimiento y en la tecnología de agentes.

El presente artículo está organizado de la siguiente manera. La primera parte se divide en dos secciones en las que se presentan las arquitecturas de dos sistemas de educación virtual, EVA e INTERLABS. En la segunda parte, se describen una infraestructura y el uso de la tecnología de agentes como el centro aglutinador de la integración de los dos ambientes en un sistema de ECV de nueva generación. Finalmente, se presentan las conclusiones y planes de trabajo futuro.

2 Antecedentes

2.1 Sistema de Enseñanza Virtual EVA

Los sistemas de ECV de la segunda generación son sistemas de enseñanza asincrónica que utilizan para la personificación de la interface del ambiente para cada usuario, la generación dinámica de las páginas Web en base a la información almacenada en la base de datos (BD). De esta misma forma está implementado el núcleo del ambiente EVA (Espacios Virtuales de Aprendizaje) desarrollado en el Laboratorio de Agentes del Centro de Investigación en Computación del IPN (<http://eva.cic.ipn.mx>), (Sheremetov L. and Quintero R, 2001). Las principales funciones de EVA son:

- 1) Planificación de trayectorias de aprendizaje de alumnos,
- 2) Generación de materiales en forma personalizada para su estudio en modo asincrónico,
- 3) Soporte de actividades de aprendizaje en forma individual y en grupos,

- 4) Monitoreo de avances de alumnos,
- 5) Experimentación en laboratorios virtuales, etc.

La realización de estas funciones se garantiza a) por medio de la organización conceptual del ambiente en cinco espacios de aprendizaje y b) por la generación dinámica del ambiente de aprendizaje con las tecnologías de Java y agentes. El ambiente está estructurado en cinco espacios; cada espacio en realidad es un conjunto de herramientas que permiten el desarrollo de diferentes actividades educativas (fig. 1).

1. Espacio de conocimiento. El conocimiento del dominio de aprendizaje está estructurado y agrupado por libros electrónicos llamados POLIlibros, los capítulos de los cuales están constituidos por unidades de material didáctico (UMD). Los POLIlibros se generan automáticamente por el sistema según la trayectoria de aprendizaje del alumno.
2. Espacio de colaboración. Cuenta con herramientas para la colaboración como son cuartos de charla (*chats* sincrónicos), foros de discusión (asíncronos) y teleconferencias (sincrónicos) entre otras. Estas herramientas son dinámicas y personalizadas, esto mediante el acceso a la base de datos (BD) de EVA.
3. Espacio de consultoría. Agrupa las herramientas de interacción entre el alumno y sus profesores.
4. Espacio de experimentación. Cada POLIlibro tiene asociado un espacio de experimentación donde se pueden realizar simulaciones, trabajar en mundos virtuales, entre otros para consolidar los conocimientos construidos en otros espacios.
5. Espacio personal. En el prototipo de la versión 1.0 se agregó este quinto espacio donde el usuario puede acceder a la información personal: sus páginas personales, avances académicos, tareas, aplicaciones personales, configurar agentes asistentes, etc.

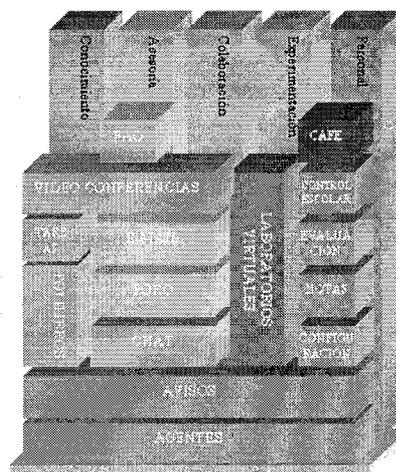


Figura 1. Distribución de las herramientas en el ambiente del alumno de EVA

El ambiente de EVA es un middleware escrito en Java utilizando servlets para la generación del entorno virtual de aprendizaje personalizado, mediante el acceso a la BD. Este *middleware* utiliza una arquitectura de tres capas. La información de EVA se almacena en una base de datos. El uso de los *drivers* JDBC (*Java Data Base Connectivity*) permite conectar el ambiente de EVA con cualquier manejador de BD. La generación del contenido dinámico del ambiente se hace por medio de una petición del *EVA* (usuario de EVA) utilizando un navegador. Esta petición es direccionada al servidor y procesada por el servlet de JAVA, éste a su vez se conecta a la base de datos mediante JDBC (o el puente *jdbc-odbc*) y accesa a la BD. Por último, una vez recuperada la información de la base de datos ésta se envía de nueva cuenta al servlet que a su vez con esta información genera una página en tiempo real.

Cabe mencionar, que la conexión con una BD es uno de los problemas más importantes en el desarrollo de ambientes de aprendizaje basado en la Web. Resulta que en general las aplicaciones de Web llevan una carga de conexión con la BD más alta y, al mismo tiempo, menos predecible, que las aplicaciones sin conexión a la red. Frecuentemente, se gastan mucho más recursos en la realización de estas conexiones y desconexiones, que las que se requieren directamente para la interacción con la BD. Por otro lado, las conexiones de usuarios con la BD normalmente son más cortas debido a la naturaleza del trabajo en Internet (cambio frecuente de servidores).

La versión actual de EVA utiliza la técnica de *pool* de conexiones que permite controlar recursos de aplicaciones de Web. Esta técnica permite distribuir la carga de las solicitudes de varios usuarios estableciendo un *pool* de conexiones que los servlets del usuario pueden usar. En otras palabras, cada solicitud de usuario lleva sólo una parte de proceso de conexión / desconexión en él. Después de que los recursos iniciales están agotados, para hacer una nueva conexión con el *pool*, la carga adicional ya no es importante porque durante el trabajo se usan repetidamente las conexiones que ya existen.

Los servlets usan el *pool* de conexiones de la siguiente manera. Cuando el usuario realiza una solicitud a un servlet, el último usa una conexión del *pool* que ya existe (y que está libre). Cuando la solicitud está satisfecha, el servlet devuelve la conexión con el *pool* para su uso por otro servlet. Es importante subrayar, que un servlet no se dirige directamente al *pool* de conexiones. Para obtener la conexión, se usa el administrador del objeto (*Object Manager*) en lugar de él, que automáticamente proporciona funciones de servicio de comunicación.

El *pool* de conexiones también permite controlar el número de conexiones paralelas con el servidor. Es muy útil, sobre todo si de acuerdo con la licencia del servidor que se usa con el ambiente de aprendizaje, existen las restricciones en el número de usuarios. En este caso, es posible crear un *pool* para el servidor de datos y establecer

el parámetro máximo de conexiones igualándolo al número máximo de los usuarios permitidos según la licencia del software.

El prototipo de la primera versión de EVA está implementado sobre la plataforma Windows, contenedor de servlets *Tomcat*, servidor de material WEB Apache Web Server y finalmente Access o Oracle como manejador de la BD. Actualmente se implementó una versión experimental sobre la plataforma de Linux, como sistema operativo, Mysql como manejador de base de datos con Mysql-JDBC como fuente de datos JDBC.

Actualmente, la primera versión comercial de EVA a cargo de la Subdirección de Investigación Aplicada del CIC está en la fase de desarrollo. En el Laboratorio de Agentes se está experimentando con la versión prototipo que se utiliza como núcleo sobre el cual se integran diferentes sistemas multiagente para extender la funcionalidad del ambiente (ver la sección 3).

2.2 Sistema de Enseñanza INTERLABS

Hoy en día existen varios sistemas de educación y capacitación de la tercera generación, basados en el uso activo de la red Internet, por ejemplo sistemas de *Stanford Online*, JTL, MANIC, CALAT, AOF, Sync-O-Matic 3000, WLS, MStar, Classroom 2000, KMi Stadium, CA309, entre otros (Uskov V., 2001),(Brusilovsky P, 2000).

Las principales y nuevas funciones de los sistemas de educación y capacitación de la tercera generación en comparación con los sistemas de segunda generación son:

- 1) Posibilidad de visualizar fragmentos de los cursos de capacitación en videos y reproducir los fragmentos de audio en tiempo real (*streaming* de vídeo y de audio) y la transmisión de nuevas partes de los cursos en línea desde el servidor en el momento de interacción del cliente y el usuario,
- 2) Posibilidad de visualizar las diapositivas de las presentaciones de un curso en línea (por ejemplo, las laminas de Microsoft PowerPoint) y su sincronización temporal con los fragmentos de video y de audio,
- 3) Posibilidad de controlar el *streaming* de vídeo y de audio, y las presentaciones de diapositivas, por ejemplo, implementando "saltar adelante", "saltar atrás", "Reproducir", "Pausa".

El sistema de enseñanza en línea INTERLABS (fig. 2) está desarrollado en el Centro de Investigación de Tecnologías de Educación basadas en Internet en la Universidad Bradley, E.U (<http://interlabs.bradley.edu>). Este sistema es un típico representante de la tercera generación de los sistemas de educación y capacitación sobre Internet, y tiene todas las funciones estándar mencionadas anteriormente. Las

funciones adicionales únicas del sistema INTERLABS son las siguientes:

- 1) Posibilidad de organizar las videoconferencias por tipos "par-a-par (uno-a-uno)" y "uno-a-muchos" sobre el protocolo IP,
- 2) Posibilidad de organizar las conferencias de audio,
- 3) Posibilidad de organizar las conferencias de datos (el intercambio automático de datos entre programas que trabajan en el sistema) sobre el protocolo IP,
- 4) Posibilidad de realizar trabajo virtual simultáneo de varios clientes con el mismo programa y con acceso remoto (aplicaciones compartidas),
- 5) Posibilidad de organizar las discusiones electrónicas y conexiones a grupos existentes de "discusiones por intereses",
- 6) Posibilidad de bajar unidades, fragmentos y cursos enteros de capacitación desde el servidor del Centro, y unas otras funciones.

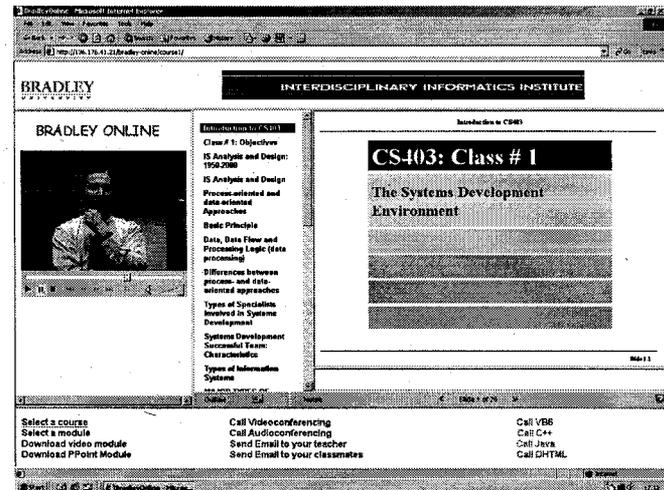


Fig. 2. Interface del Sistema de Enseñanza INTERLABS

Para la transmisión de audio y video se utiliza la tecnología de *streaming*. En el lado del servidor INTERLABS, para la compresión de video se utiliza el formato de Microsoft MPEG4V3 (conocido también como ASF) con las siguientes características de transmisión: 100kbps, 15 *frames* por segundo, formato de video 320x240 píxeles y color de 16-bit. Para la compresión de archivos de audio se utiliza el formato MP3 con la compresión a 22.1 kHz, que se transmite bajo la velocidad de 50kbps. Para las presentaciones electrónicas, la sincronización entre los archivos de video y presentaciones de PowerPoint se realiza utilizando *Advanced Indexer* de Microsoft. Los archivos *script* para *Advanced Indexer* se generan por un programa desarrollado por los autores de INTERLABS. Para generar los cuartos de charla (*Chat*

rooms) en el lado del cliente se utiliza Jpilot JAVA IRC applet o mIRC Windows IRC (como alternativa a Jpilot). En el lado del servidor se utiliza la herramienta ircd2.9.5. El audio y video se transmite sobre el protocolo TCP/IP.

En el lado del cliente, el sistema INTERLABS puede funcionar tanto en el modo síncrono, como en modos asíncronos. En el modo síncrono (el modo en línea) se usan Microsoft Windows Media Player Active X control, Microsoft PowerPoint Player, Java Script, marcos y Microsoft NetMeeting SDK, entre otros. En el modo asíncrono (el modo autónomo) se usan Microsoft Windows Media Player Active X control, Java Script, Microsoft PowerPoint, Microsoft Visual Basic y sistemas de VirtualDub (para digitalización de video y audio).

Para integrar las herramientas externas en la interface de INTERLABS se utilizan las capacidades de controles ActiveX (fig. 3).

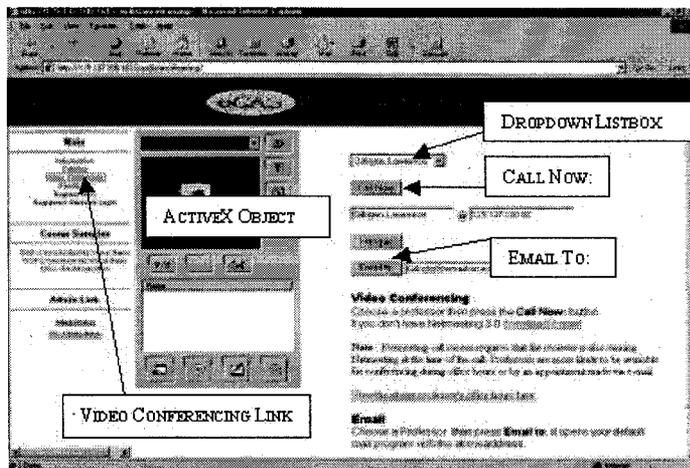


Fig. 3. Interface del Sistema de Enseñanza INTERLABS con la herramienta NetMeeting integrada

Cabe destacar que una de las principales ventajas del sistema INTERLABS es su alta portabilidad. INTERLABS puede trabajar para varios navegadores de Internet, y también puede interactuar con los sistemas de la segunda generación. En la primera fase, se llevaron a cabo los experimentos de integración de algunos clases en línea impartidos con INTERLABS con la funcionalidad de administración de cursos brindada por la herramienta BlackBoard 5.01.

2.3 Motivación de la Investigación

Según la experiencia del uso de ambos sistemas, éstos reúnen algunas características que habilitan la introducción del modelo de aprendizaje activo. EVA implementa las funciones de administración de cursos ofreciendo múltiples herramientas para instructores, aprendices, administradores, etc. Sin embargo, los experimentos realizados con la versión-prototipo de EVA han demostrado también

problemas típicos para los sistemas de segunda generación, como son: la flexibilidad e inteligencia restringidas del ambiente, o la interacción limitada entre los usuarios.

Por otro lado, el INTERLABS tiene herramientas múltiples para comunicación directa "estudiante-estudiante" y "estudiante-profesor". Por consiguiente, la integración de los dos sistemas llevaría a la situación "ganar-ganar". Sin embargo, una simple combinación de dos sistemas no permite crear los ambientes capaces de satisfacer los requerimientos de sistemas de aprendizaje activo y se requiere del desarrollo de nuevas tecnologías que permitan la construcción de sistemas abiertos, cooperativos e inteligentes.

Dos puntos son claves para el desarrollo de SECV de la nueva generación. Primero, se requiere de las soluciones estándares que permitan no solo resolver el problema de integración de dos sistemas particulares, sino también den las soluciones que se pueden utilizar como marco de referencia para la construcción de SECV abiertos. En este aspecto, se requiere del uso de especificaciones técnicas abiertas y de lenguajes de programación para apoyar el aprendizaje distribuido. Uno de los enfoques estándares y las herramientas se están desarrollando en el marco del modelo de referencia de objetos de contenido compartido SCORM (*Sharable Content Object Reference Model*), cuyo propósito es asegurar el acceso a educación de alta calidad y materiales de capacitación los cuales pueden ser personalizados a las necesidades de los alumnos individuales y puestos a disposición en el lugar y en el momento que sean requeridos. El uso de lenguajes XML y SMIL como lenguajes de especificación de componentes permite que los componentes generados sean compatibles con la plataforma de EDUCOM/IMS. Por las limitaciones del espacio, este aspecto está fuera de consideración de este artículo, algunos problemas de desarrollo de componentes de aprendizaje reutilizables para EVA se presentan en (Peredo R. And Sheremetov L., 2001).

Por otro lado, se requiere del uso activo de las tecnologías de inteligencia artificial que permiten incrementar la eficiencia de enseñanza virtual. Una de estas tecnologías es la tecnología de agentes (Singh M. P.,), (Jennings N.R., 2000).

3 Una Infraestructura de Agentes para Ambientes de Aprendizaje

Los enfoques modernos de desarrollo de sistemas de ECV son desarrollados principalmente dentro del marco de los problemas de inteligencia artificial, específicamente en sus áreas orientadas a la enseñanza cooperativa en un ambiente distribuido multiagente. El concepto del agente es uno de los conceptos clave en sistemas modernos de educación. Dentro del marco de este artículo, el concepto *del agente* se considera como el proceso de software con el conocimiento

limitado acerca de un dominio del problema, capaz de alcanzar sus objetivos en cooperación con otros agentes y usuarios.

En un ambiente de ECV, los agentes se agrupan en sistemas multiagente (SMA) los cuales pueden jugar un papel de 1) profesores virtuales, 2) estudiantes virtuales, 3) los ayudantes virtuales que asisten a los estudiantes en el aprendizaje, 4) los administradores virtuales de cursos, etc. (Hayes-Roth B, 1995), (Sheremetov L. and Nuñez G., 1999). Estos agentes se usan ampliamente en Ambientes Virtuales de Aprendizaje para lograr las siguientes metas:

- Para proporcionar el apoyo siempre disponible a los estudiantes;
- Para reducir la carga en los tutores de contestar problemas comunes;
- Para reutilizar el conocimiento generado de las discusiones en las generaciones anteriores;
- Para animar a los estudiantes a usar la tecnología y participar en las actividades dentro del ambiente.

En el contexto del presente trabajo el uso de agentes pretende alcanzar dos fines: a) implementar varias funciones del ambiente por medio de agentes (como las citadas anteriormente) y b) utilizar una infraestructura multiagente como un *middleware* que habilite la integración de dos diferentes ambientes de aprendizaje. Para lograr este objetivo se desarrolló una infraestructura multiagente que se presenta a continuación.

3.1 Arquitectura Multiagente con Capas

La Figura 4 representa una arquitectura con capas para un SECV multiagente que consiste de cuatro diferentes tipos de agentes: asistentes personales, agentes del SECV, agentes de soporte y plataforma del agentes.

Los Agentes Asistentes Personales son una clase de agentes inteligentes que actúan semi-autónomamente en nombre de un usuario, modelándole su interés y proporcionando servicios al usuario u a otra persona y otros asistentes que lo requieran. Estos agentes por lo general se instalan en la maquina del cliente y funcionan en forma local interactuando con el software legado de escritorio (paquetes de correo, navegadores Web) y forman parte de diferentes SMA interactuando con otros agentes del ambiente virtual.

Los agentes del ambiente virtual son agentes de propósito general que extienden la funcionalidad de sistemas tradicionales de Administración de Cursos con un conjunto de funciones "inteligentes" que permiten el modelado del estudiante, la generación automática del plan de estudios y la asesoría automática, entre otras.

El propósito de tales funciones es la mejora de la efectividad de aprendizaje basada en la adaptación del material didáctico a las habilidades y preferencias del estudiante. Estos agentes se instalan en una máquina que funciona como plataforma de agentes pero actúan en interacción directa con el ambiente de aprendizaje. Los agentes de este tipo pueden formar parte de diferentes SMA. Cada usuario del sistema obtiene una instancia del agente e interactúa con él a través de un agente de interface.

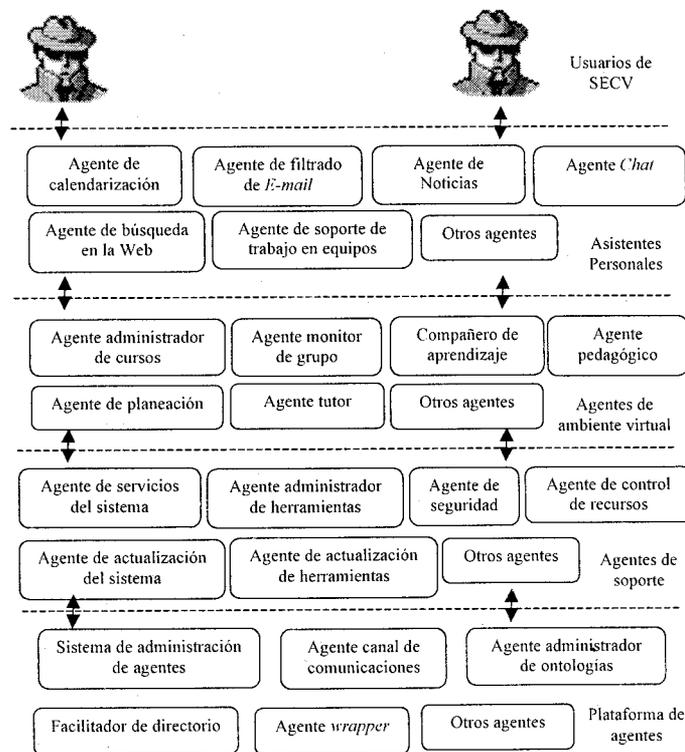


Figura 4. Arquitectura multiagente con capas de SECV

Los agentes de soporte forman un SMA de administración del sistema y se encargan de actualizar automáticamente el ambiente y sus herramientas, así como proveer la seguridad del mismo.

Los SMA representan sistemas de cómputo complejos distribuidos, para el funcionamiento de los cuales se requiere de un ambiente especial del soporte de su operación que proporciona contextos lógicos y temporales para la creación, la operación y la destrucción de agentes orientados a la solución de problema del dominio de aplicación (Sheremetov L. and Contreras M., 2001). A este ambiente se le llaman Plataforma de Agentes (*Agent Platform - AP*). Los agentes de este tipo complementan la infraestructura multiagente de un ambiente de aprendizaje. Estos agentes juegan un papel de un núcleo aglutinador del ambiente de aprendizaje.

3.2 Plataforma de Agentes

En la etapa actual de investigación, se ha desarrollado la *plataforma de agentes y componentes (Component Agent Platform - CAP)*. La plataforma de agentes consiste de los siguientes componentes generales:

1. Sistema de Administración de Agentes (*AMS - Agent Management System*) maneja el ciclo de vida de los agentes físicamente instalados sobre CAP. Para estos objetivos el agente administrador mantiene permanentemente una lista de todos los agentes residentes en la CAP. Esta lista incluye, como mínimo, el identificador único del agente y su dirección sobre CAP.
2. Agente Facilitador de Directorio (*Directory Facilitator - DF*) es el gerente de dominio de aplicación que es responsable del registro de servicios (roles) de agentes de aplicación y de soporte. Este agente habilita la comunicación entre agentes tipo "par-a-par" y maneja una lista de los servicios proporcionados por los agentes, y características adicionales, como el tipo, un estado y un contenido del agente. Para cada tipo de servicio esta guardado su nombre, tipo, y la ontología (del dominio al cual pertenece el servicio). Los agentes se ponen en contacto con el DF para el registro de los servicios o para la recuperación de documentos sobre los servicios dados por otros agentes. El grupo de agentes registrados en el DF es llamado *el dominio de agentes*. Así, para cada ambiente de educación virtual es necesario que exista como mínimo un DF.
3. Canal de Comunicaciones entre Agentes (*Agent Communication Channel - ACC*) habilita la comunicación física entre agentes, incluyendo a los agentes auxiliares de la plataforma (para la definición de la comunicación a nivel lógico se usa el DF). Así, cada agente debería tener el acceso como mínimo a un ACC. Los canales de comunicación suministran la entrega de mensajes no sólo dentro del marco de una plataforma lo hacen también entre varias plataformas. La computadora-cliente de cada usuario en línea del servidor de cursos tiene un ACC, proporcionando la comunicación entre los agentes asistentes personales del usuario con los agentes auxiliares del ambiente.
4. El Agente de Envoltura (*Wrapper Agent - WA*) provee la conversión de los formatos de mensajes entre agentes y otro software (bases de datos, paquetes de aplicación etc.).
5. El Agente de Administración de Ontologías (*Ontology Management Agent - OM*) provee la conectividad entre servidores de cursos en línea a nivel de conocimiento. Para la operación del sistema se usan las siguientes ontologías: a) de servicios (el modelo de estudiantes, el modelo de las estrategias de aprendizaje, etc.), b) de dominios de aprendizaje, c) de la administración y

mantenimiento de los cursos, d) de roles (estudiante, profesor, monitor de grupo, gerente de curso, etc.).

La CAP proporciona al usuario un conjunto de herramientas de administración y desarrollo de los SMA. Primero, todos los agentes de aplicación están basados en la plantilla *BasicAgent*. Esta plantilla implementa las funciones de interacción de agentes con el ambiente en el que se desenvuelven a través de mensajes de lenguaje de comunicación ACL (*Agent Communication Language*) para comunicarse con otros agentes o a través del uso de sensores y efectores (que pueden ser desde una interface gráfica para el usuario hasta sensores que monitorean el mundo físico).

Los sensores y efectores, son objetos que el usuario debe de implementar y añadir a su agente al diseñarlo, cada sensor corre en su propio hilo de ejecución y puede dar inicio a la ejecución de acciones mediante los efectores o iniciar una tarea del agente. Los efectores representan acciones básicas que el agente puede efectuar para modificar el ambiente en el que se desenvuelve. Estas acciones pueden ser invocadas desde las tareas del agente o disparadas directamente por los sensores (lo que daría como resultado un comportamiento reactivo).

La otra forma de interactuar con el ambiente (en concreto con otros agentes que se desenvuelven en él) es a través del envío de mensajes ACL. Cuando se recibe un mensaje este es analizado por el manejador de conversaciones, que decidirá como debe de procesarse ese mensaje. Si el mensaje pertenece a una conversación existente se le entregará a esta para su procesamiento, en caso contrario podría iniciarse una nueva conversación.

En la implementación de esta plataforma, los mensajes de ACL se ponen en código usando *eXtensible Markup Language* (XML) como una representación sintáctica. En general, cada agente sólo usa un subconjunto del ACL. Este subconjunto puede ser diferente de los subconjuntos usados por otros agentes. Puesto que el análisis sintáctico (*parsing*) del lenguaje puede ser una tarea bastante complicada, se utiliza el XML DOM de Microsoft para analizar y construir los mensajes de XML. En la CAP se implementaron dos de los lenguajes más usados de comunicación entre agentes: el lenguaje ACL de la FIPA y el Lenguaje de Consulta y Manipulación de Conocimiento (*Knowledge Query and Manipulation Language - KQML*) (Finin T. et al, 1997), (<http://www.fipa.org>).

El uso de XML para codificar el lenguaje de contenido de mensajes da varias ventajas y facilita la integración de la CAP en ambientes del software reales. Otra ventaja de usar XML es que agentes pueden aprender modificando su lenguaje en el proceso de ejecución (*on the fly*). Por ejemplo, es posible extender la lista común de parámetros de mensajes ACL en el parámetro del sobre (*envelope*). Para que cualquier agente con la lista extendida puede informar a sus pares sobre los nuevos parámetros usados

para que ellos modifiquen los *Document Type Definition* (DTD) correspondientes extendiendo la sintaxis del lenguaje. La similitud de la estructura de los dos lenguajes utilizados para la comunicación facilita la implementación y el uso de los archivos DTD para ACL y KQML, que depende del parámetro en el campo "*language*" del primer mensaje de cada conversación.

La aplicación de agentes de la envoltura también es simplificada usando el lenguaje de XML. Dos diferentes plantillas de *BasicAgent* casi con la misma funcionalidad pero en base a tecnologías diferentes están desarrolladas. Una de ellas es un control de ActiveX y la otra es una clase de Visual FoxPro (VFP) guardada en una biblioteca de clases visuales (VCL).

3.3 Agentes Asistentes Personales

Como el siguiente paso, se desarrolló un sistema multiagente de asistencia personal (*Multiagent System of Personal Assistance - MSPA*) que incluye la colección de agentes inteligentes y de agentes de apoyo. El MSPA funciona sobre la CAP. A la fecha, los asistentes desempeñan varios roles como los que se describen mas adelante. Cada rol se agrega como un control al núcleo del agente básico, lo anterior permite simplificar bastante la configuración del agente por su usuario.

- 1) El Agente Monitor de Grupo a) implementa un ambiente de apoyo a la solución cooperativa de problemas, b) ayuda a los tutores en la coordinación de las operaciones de los miembros de grupo, c) puede definir el problema para discutir, dar privilegios en la aceptación de las contribuciones, propone soluciones, entre otros. Ayuda a los estudiantes, regulando su participación en la discusión, mostrando mensajes de coordinación, etc.
- 2) El Agente de Agenda Personal a) implementa el sistema de noticias y de mensajería asincrónica entre profesores y estudiantes, incluyendo la conectividad entre los usuarios certificados sobre varios servidores, b) habilita la transferencia de todos los mensajes acerca del usuario concreto en su agenda electrónica, realizada como una aplicación independiente en el ambiente de la maquina-cliente.
- 3) El Agente Chat proporciona a los usuarios el apoyo a la discusión estructurada y síncrona, incluyendo la conectividad entre los usuarios certificados sobre varios servidores.

Cada usuario de los sistemas de educación y capacitación EVA e INTERLABS puede activar un agente asistente según el papel social que juega en el ambiente cooperativo, como tutor o estudiante. Este rol puede variar, por ejemplo en la solución de problemas en grupo, el rol del líder de grupo puede ser desempeñado por uno de sus participantes.

Los CAP y MSPA juegan un papel clave en la integración de ambos sistemas de educación virtual en una "universidad virtual". Para cualquier agente asistente (maestro o estudiante) un DF busca los servicios proporcionados por otros agentes y otros asistentes. El Agente *Wrapper*, a su vez, suministra el acceso a la base de datos de los sistemas con el propósito de consultar y modificar la base de una manera explícita. Leyendo los datos del usuario de la base de datos del servidor apropiado, se establece con la ayuda de agentes auxiliares de AP la conexión con aquellos usuarios que forman su grupo de trabajo, pero ya independientemente del lugar físico de su registro. Así, hay un espacio de estudio virtual distribuido - el *Campus Virtual* o la *Universidad Virtual*.

El ambiente de trabajo cooperativo soporta diferentes escenarios de interacción. El más sencillo es un ambiente de *Chat* que permite generar dinámicamente los grupos de usuarios dependiendo de su avance y establecer comunicación entre sus asistentes personales (fig. 5).

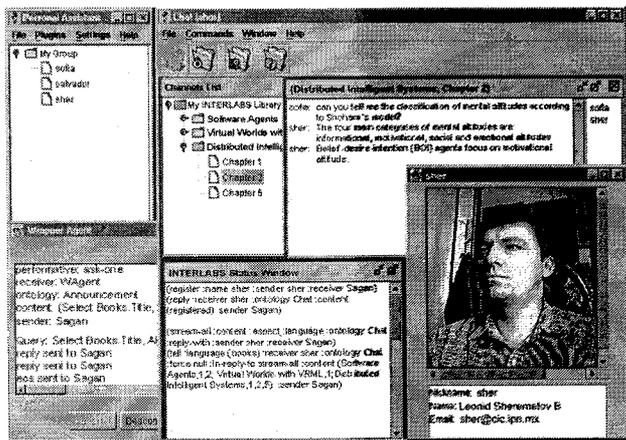


Fig. 5. La interface del ambiente de apoyo personal de operación de grupo

Como se muestra en la figura 5, el agente asistente del usuario (sher) establece los canales de comunicación con otros integrantes del grupo (sofia, salvador) que estudian el capítulo 2 del POLIlibro "Sistemas Inteligentes Distribuidos" (ventana "Channels list"). Esta lista es dinámica, se genera en base a la información almacenada en la BD de EVA. Para hacerlo, se establece la comunicación entre el agente administrador (llamado Sagan) y el agente *Wrapper* de la BD (llamado WAgent). Esta comunicación se visualiza en ventanas auxiliares de MSPA (*Status Window*, *Wrapper Window*), y para su codificación se utiliza KQML. Una vez establecida la comunicación entre los participantes del grupo, se inicia la sesión de trabajo.

Otra parte del ambiente colaborativo proporciona soporte para la resolución de problemas en equipos formados por estudiantes y agentes. Este ambiente consiste de dos tipos de agentes: el Monitor de Grupo y Participante o Compañero Virtual. Estos agentes administran sus bases de

conocimiento (*Knowledge Base* - KB). Esto es, la comunicación con otros agentes en la parte del contenido de mensajes es con respecto a su base de conocimiento por ejemplo cuestiones acerca de lo que su KB contiene, solicitudes para agregar o borrar declaraciones de la KB o solicitudes para usar el conocimiento contenido en la KB para rutear mensajes al agente apropiado. Para la representación del modelo de usuario y la planificación de las actividades del agente, se utiliza la descripción del comportamiento del agente en base a reglas codificadas en Jess (*Java Expert System Shell*) (Friedman- Hill E. J., 1998).

El Agente Monitor de Grupo mantiene el modelo de conocimiento compartido de un grupo y lo compara con un modelo del problema almacenado en la base de conocimiento que contiene los objetivos, conceptos, actividades, etc. que caracterizan el grupo. Su conducta es guiada por un conjunto de reglas de conversación independientes del dominio que refieren a las interacciones entre los miembros del grupo. Durante la sesión de grupo, el agente Monitor mantiene el estado del problema actual (espacio de conocimiento compartido de grupo) y la bitácora de todas las contribuciones de cada participante. El también coordina las intervenciones de los participantes y sugiere las estrategias de resolución de problema (ver la ventana "Agente Asistente" en la fig. 6).

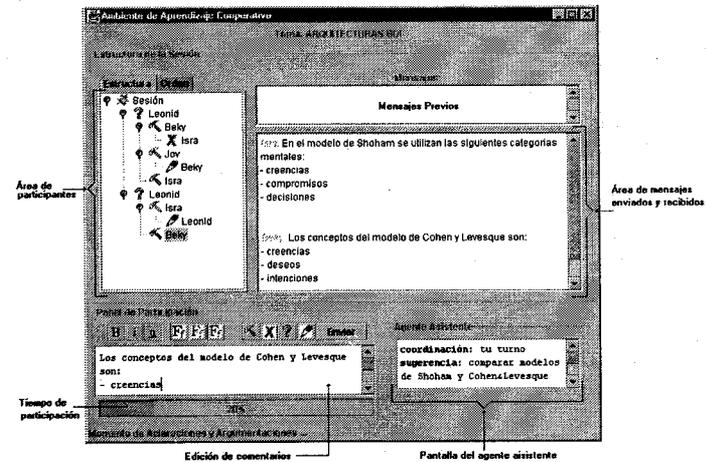


Fig. 6. La interface del usuario del ambiente colaborativo

Otra tarea del agente Monitor de Grupo es el apoyo de interacción entre los miembros de grupo para la solución de una tarea. Para la solución de esta tarea se utiliza un agente adicional - el Participante Virtual o Compañero de Aprendizaje (*Learning companion*). El estado del problema se usa para cambiar el modo de la interacción seleccionando de su base de reglas una de las técnicas de monitoreo. Este resulta en el cambio de conducta del compañero de aprendizaje del líder de grupo (compañero fuerte) a través de un compañero débil a un observador pasivo.

Actualmente se está trabajando en las interfaces visuales y animadas del agente asistente implementadas en base al Agente de Microsoft (también es un control de ActiveX). Este agente tiene la finalidad de proporcionar orientación y explicación, otorga facilidades para que la información requerida llegue al usuario de una forma entendible y sencilla, emulando la forma tradicional de comunicación como lo es a través del habla. Tales interfaces hacen un sistema de ECV más atractivo y fácil al utilizarlo. Especialmente, en las actividades de aprendizaje que se realizan durante la estancia en una sesión en el sistema con las cuales se promueven los objetivos de aprender. Con la implementación de ellos se busca que el alumno navegue a través de un sistema realista, expresivo y atractivo lo cual generara un agradable espacio de estudio. Actualmente también se está experimentando con los prototipos de otros agentes, como son: *Tutor artificial*, *Examinador*, *Gerente del curso*, *Agente de Búsqueda en el curso*, entre otros.

4 Conclusiones

Entre enero y marzo del año 2001, 14 profesores de tres universidades del estado de Illinois (EE.UU) desarrollaron 19 cursos en línea sobre INTERLABS. Los profesores tenían los niveles completamente diferentes de preparación en el campo de ciencias de la computación y tecnologías de información. La versión de demostraciones del sistema INTERLABS y fragmentos de los 14 cursos de capacitación en línea se encuentran disponibles en el sitio del Centro de Investigación de Tecnologías de Educación basadas en Internet de la universidad Bradley (<http://interlab.bradley.edu>).

La experiencia de uso del sistema INTERLABS, y también los experimentos realizados con la versión-prototipo de EVA (<http://eva.cic.ipn.mx>) en el Programa de Postgrado virtual en el Centro de Investigación en Computación del Instituto Politécnico Nacional de México durante los años académicos 2000-2001, han mostrado un buen rendimiento y la fiabilidad de los sistemas de educación y capacitación en general, y sus sistemas multiagentes en particular, la interacción confiable y estable de varios agentes con el software tradicional que forma la base de los sistemas presentados en el artículo. Ambos sistemas de educación virtual han llamado la atención de los profesores y estudiantes de las dos universidades.

En la segunda fase de nuestros experimentos, se habilitó desde el entorno de EVA el acceso a los alumnos de EVA a algunas clases en línea impartidos con INTERLABS (fig. 7), así mismo el acceso de los alumnos a los servicios de asistentes personales. En los últimos experimentos conducidos en Enero del 2002, de 117 alumnos que no eran del área de ciencias de la computación, el 79% prefieren usar la tecnología de *Web-lecturing* integrada en el ambiente de administración de cursos que los cursos estáticos y pasivos de la 2da generación.

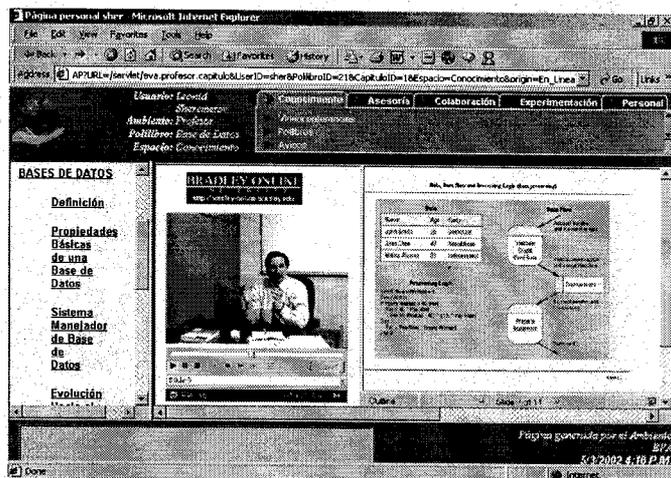


Fig. 7. La interface del usuario del ambiente EVA con el sistema INTERLABS

La primera lección aprendida de la integración de los dos sistemas es la necesidad de cambiar la arquitectura cliente-servidor por las limitaciones que ésta implica, en la que el mantener al día la lógica de la aplicación con las reglas del negocio se ha vuelto una tarea titánica. Se ha propuesto la migración a una arquitectura multicapa (*Multi-tier*) y se ha planteado la necesidad de que se mejore la escalabilidad, modularidad y seguridad del sistema. Se ha encontrado que el uso de la tecnología de Servicios Web XML brindaría un interesante punto de partida para el rediseño de esta aplicación.

Los sistemas de aprendizaje activo abren nuevos horizontes en la creación de ambientes de educación virtual utilizando la tecnología de sistemas multiagentes. Esto debido a un gran potencial incorporado en la tecnología de agentes de software que son programas activos. El uso de la tecnología de agentes permite:

1. realizar diferentes modelos pedagógicos y metodológicos construidos sobre varias teorías de educación,
2. construir el modelo adaptativo de estudiantes,
3. realizar varias:
 - a. técnicas de educación y capacitación,
 - b. formas de representación de conocimiento,
 - c. técnicas de interacción de la operación de grupo,
 - d. modelos de evaluación de individuo y la operación de grupo de alumnos, divididos en el tiempo y en el espacio.

Las principales innovaciones presentadas en el presente artículo pueden encontrarse en la naturaleza distribuida de la aplicación y en la flexibilidad del enfoque de agentes (se han definido diferentes agentes y se han cooperado para resolver tareas de enseñanza). Investigaciones activas sobre la creación de comunidad virtual de los estudiantes reales y

virtuales están en proceso. Los siguientes agentes de sistemas INTERLABS y EVA ya están en la fase de depuración:

1. En el agente Examinador se están agregando las reglas diagnósticas para la construcción automática del modelo de estudiante en base a su progreso académico, comportamiento, el desarrollo de las habilidades de auto-capacitación, etc.;
2. En el agente Administrador de Curso se están agregando las funciones de construcción automática de programas y planes de estudio y la generación de informes sobre el progreso académico de alumnos en base a los datos personales;
3. En el agente Canal de Comunicación se están agregando las funciones de entrega de audio y documentos de vídeo por demandas del usuario, uso compartido de documentos en Internet, etc.;
4. En el Agente de Búsqueda y Filtrado en la Web se están agregando las funciones de exploración automática de los fragmentos requeridos de información y de los multimedia en Internet.

Los autores esperan, que después de la primeras pruebas sobre la base de los dos sistemas comenzarán el uso activo de los cursos en línea dentro del marco de instituciones educativas virtuales correspondientes y, por otra parte, la integración de los dos sistemas en un ambiente de educación virtual de nueva generación entrará en su fase final.

Agradecimientos

A los autores de este artículo les gustaría agradecer la Fundación Nacional de la Ciencia (National Science Foundation - NSF, E.E.U.U.) por el apoyo parcial para este trabajo dentro de los Proyectos NSF CCLI-EMD #DUE-0196015 (2000-2003) y #DUE-0002219 (2000-2001). El apoyo parcial para este trabajo también fue proporcionado por el CONACyT (México) dentro del proyecto 31851-A (2000-2002) e IPN dentro del proyecto 980739 (1998-2001). Los autores quisieran agradecer al Sr. Alexander Uskov, estudiante de CS/IS en la Universidad de Bradley, y a los profesores y alumnos del Laboratorio de Agentes del CIC-IPN por la colaboración en el diseño y desarrollo del software de INTERLABS y EVA respectivamente. Agradecimientos particulares a Miguel Contreras por la implementación de la plataforma de agentes.

Referencias

Proceedings of the Webnet-2001 World Conference on the Internet and WWW, Orlando, Florida, October 2001.

Uskov, V. Development of Web-Based Instructional Tool and Online Educational Materials of the Third Generation. Proceedings of the WebNet-2001 World Conference on the Internet and World Wide Web, Orlando, Florida, October 23-27, 2001.

Peter Brusilovksy. Web Lectures: Electronic Presentations in Web-Based Instruction. Syllabus, N 1, 2000, pp. 18-23.

Craig Montgomerie, Jarmo Viteli (Eds.). Proceedings of ED-MEDIA 2001 World Conference on Educational Multimedia, Hypermedia & Telecommunications, Tampere, Finland; June 25-30, 2001, AACE.

Proceedings of the IEEE/ASEE Annual Frontiers in Education 2001 Conference, Reno, Nevada, October 2001.

Proc. of the International Conference TELEMATICA-2001, St. Peterburg, Russia, June 18-21, 2001.

Proceedings of the XX Conference on Distance and Open Education, Dusseldorf, April 3-5, 2001.

Proceedings of the e-Learning conference, Washington, DC, April 7-9, 2001.

W3 Consortium: Extensible Markup Language (XML) and Synchronized Multimedia Integration Language (SMIL) Specifications, URL: <http://www.w3.org/TR/REC-smil/>, <http://www.w3.org/XML/>.

IMS *Global Learning Consortim*, URL: <http://www.imsproject.org/>.

Advanced Distributed Learning Co-Lab, URL: <http://www.adlnet.org>.

EVA, URL: <http://eva.cic.ipn.mx>.

Sheremetov, L. & Quintero, R. EVA: Collaborative Distributed Learning Environment Based in Agents. In Proc. of the World Conference ED-MEDIA '2001, Tampere, Finland, June 24-30, 2001. CD ROM.

INTERLABS, URL: <http://interlabs.bradley.edu>

Peredo, R. & Sheremetov L. Arquitectura de Componentes de Software Programables Reutilizables Inteligentes para Materiales Multimedia en EVA, En (Sossa, Freeman & Vizcaíno, eds.) Memorias del Congreso Internacional de Computación, CIC'2001, México, nov. 14 - 16, pp. 461-472.

Singh, M. P. *Multiagent Systems, A Theoretical Framework for Intentions, Know-How, and Communications*, Springer-Verlag, Lecture Notes in Artificial Intelligence, Volume 799.

Jennings, NR. On agent-based software engineering. Artificial Intelligence, 2000, 117:277-296.

Barbara Hayes-Roth. An Architecture for Adaptive Intelligent Agent Systems. *In Artificial Intelligence*, Vol. 72, 1995, pp. 329-365.

Chan, T.W. (1996) *Learning Companion Systems*, Social Learning Systems, and Intelligent Virtual Classroom, In Proc. of the World Congress on AI in Education, 1996, Vol. 7, No. 2, 125-159.

Gordon, A. and Hall, L. (1998) *Collaboration with Agents in a Virtual World*. In Proc. of the Workshop on Current Trends and Artificial Intelligence in Education, 4 World Congress on Expert Systems, Mexico-city, Mexico, 1998.

Dillenbourg, P., Jermann, P., Schneider, D., Traum, D. & Bui, C. (1997), The design of MOO Agents: Implementations from an Empirical CSCW Study, In B. Du Boulay and R. Mizoguchi (Eds.) *Artificial Intelligence in Education*, IOS Press, 15-22.

Capuano, N., M. Marsella and S. Salerno, *ABITS: An Agent-based Intelligent Tutoring System for Distance Learning*, In Proc. of the Fifth International Conference on ITS, Montreal, Québec, Canada, June 19-23, 2000.

Sheremetov, L and Núñez, G., Multi-Agent Framework for Virtual Learning Spaces. *Int. Journal of Interactive Learning Research*, 10 :301-320, 1999.

Sheremetov, L. & Contreras, M. Component Agent Platform. *In Proc of the Second International Workshop of Central and Eastern Europe on Multi-Agent Systems, CEEMAS'01*, Cracow, Poland, 26-29 of September, 2001, pp. 395-402.

T. Finin, et al. *KQML as an Agent Communication Language* ", in *Software Agents*, AAAI/MIT Press, 1997.

FIPA ACL Specification, Foundation for Intelligent Physical Agents, URL: <http://www.fipa.org>.

Friedman-Hill, E. J. (1998), Jess, The Java Expert System Shell, Version 4.3, Technical report SAND98-8206 Sandia National Laboratories, Livermore, CA, <http://herzberg.ca.sandia.gov/jess>.



Leonid Borisovitch Sheremetov nació en San Petesburgo, Rusia. Es egresado de la Universidad Estatal Técnica de Marina de San Petesburgo (Leningrado), Federación Rusa. Obtuvo el grado de Doctor en Ciencias del Instituto de Informática y Automatización de San Petesburgo, preteneciente a la Academia Rusa de Ciencias en 1990. Sus áreas de investigación y desarrollo son: *Sistemas Multiagentes e Inteligencia Artificial Distribuida, Educación a Distancia y Sistemas de Apoyo a la Toma de Decisiones*. Actualmente es investigador del Insituto Mexicano del Petróleo y jefe del Laboratorio de Agentes del Centro de Investigación en Computación del IPN. Es líder de varios proyectos nacionales e internacionales. Es miembro del SNI-México e investigador del Nivel Superior de la Federación Rusa y autor de más de 100 publicaciones



Vladimir Lvovich Uskov nació en Moscú, Rusia. Es egresado de la Universidad de Aviación de Moscú. Obtuvo el grado de Doctor en Ciencias de la Computación en 1986. Sus intereses de la investigación incluyen la *Tecnología de clases vía Web, Educación basada en la Web, Educación a Distancia, Sistemas Tutores Inteligentes e Ingeniería de Software*. El Dr. Vladimir es profesor titular de Informática y Sistemas de Información en la Universidad de Bradley, USA y es también director del Instituto de Investigación InterLabs. Es líder de varios proyectos de la NSF y autor de más de 150 publicaciones.

