



[www.cibereduca.com](http://www.cibereduca.com)



**V Congreso Internacional Virtual de Educación  
7-27 de Febrero de 2005**

## **Aplicaciones de Agentes Pedagógicos en Entornos Virtuales para la Enseñanza.**

Brunny Troncoso Pantoja  
[brunny.troncosopantoja@gmail.com](mailto:brunny.troncosopantoja@gmail.com)  
Universidad Politécnica de Madrid  
Universidad del Bío Bío – Concepción -Chile

### **Resumen:**

Un Agente Pedagógico (AP) es una aplicación de agentes de software en entornos de aprendizaje de algún dominio de conocimientos específico. Esencialmente, un AP actúa como la personificación de un tutor en un entorno virtual de aprendizaje, entorno que puede ser textual, bidimensional o tridimensional (de escritorio o de inmersión). La personificación de este tutor puede o no ser humana, pero resulta vital que posea características antropomorfas que permitan la comunicación verbal y no-verbal entre el tutor y el estudiante, de manera de motivar a éste, de capturar su atención y de facilitar la comprensión de conceptos, logrando así el cumplimiento de los objetivos educacionales. Se considera, además, que un AP es inteligente en el mismo contexto de los sistemas tutoriales inteligentes: por la utilización de técnicas de Inteligencia Artificial (IA) para la representación de conocimiento y la explicación de su razonamiento.

Esta ponencia comenzará entregando una visión general de lo que es un AP, partiendo desde su definición y dando una breve mirada histórica a su desarrollo, para luego llegar a sus características distintivas y a sus posibilidades de aplicación en la enseñanza. Entregará también una visión de las experiencias de aplicación pedagógica de agentes desarrolladas por

algunos centros específicos de investigación, señalando las ventajas de su utilización y sus dificultades asociadas.

Aún cuando gráficamente sólo se vea un tutor, generalmente bajo esta representación opera un conjunto especializado de agentes que interactúan entre sí y con el entorno que les rodea, con arquitecturas de software específicas, estableciendo un sistema multi-agente que determina el comportamiento del agente pedagógico de acuerdo a los diferentes parámetros involucrados. Para ilustrar estos conceptos, se explicará la clásica arquitectura de STI y una arquitectura basada en agentes propuesta para sistemas educativos en entornos virtuales de entrenamiento.

## **Introducción**

Se ha visto que el avance científico tecnológico, a la par que concentra la atención en un ámbito cada vez más reducido del conocimiento humano (especialización científico-tecnológica) tiende a unir y aplicar en pos de sus objetivos a diversas y variadas áreas de conocimiento. Surgen así especializaciones como ingeniería genética o informática médica, por nombrar algunas.

Haciendo un poco de historia, con el surgimiento de los primeros ordenadores nace también la idea de aplicarlos para la educación y el aprendizaje (junto a otras muchas áreas del conocimiento), así proyectos como PLATO, CCC, NDPCAL o TICCIT marcaron un camino inicial en la década de los 60, y con la llegada de los microcomputadores, asentaron el desarrollo de los llamados sistemas tutoriales que, luego de incorporar técnicas de inteligencia artificial para el desarrollo de sus actividades, comenzaron a llamarse sistemas tutoriales inteligentes (STI).

En paralelo a este desarrollo de software educativo y técnicas de inteligencia artificial, la industria tecnológica se desarrollaba exponencialmente, permitiendo la aplicación de nuevas y mejores herramientas al desarrollo de las aplicaciones. De esta forma se ha pasado de imágenes bidimensionales a entornos virtuales 3D, ya sea de escritorio o de inmersión, con nuevas y mejores herramientas de autoría que facilitan su utilización en el desarrollo de nuevos entornos.

Por otro lado, la irrupción de los agentes software, el mayor y más reciente aporte de la inteligencia artificial al desarrollo de software, no ha sido ajena a los sistemas tutoriales inteligentes y ha generado otra serie de cambios, tan profundos que algunos investigadores comienzan a llamar a la unión de todos estos elementos Sistemas Educativos Inteligentes (SEI) como un nombre genérico para representar a estas aplicaciones actuales, en que el tutor-agente se “personifica” (toma forma humana, animal, de caricatura u otra) interactuando con el alumno y con otros agentes del entorno virtual (textual, 2D, 3D de escritorio o inmersión) de aprendizaje -entrenamiento.

Sumado a todo esto están los avances que se han logrado en psicología de la educación, estrategias pedagógicas y teorías de la comunicación, que han venido a incorporarse al desarrollo de estas aplicaciones y actúan realimentándose con nuevos resultados y desafíos generados por éstas a su vez.

Llegados a este punto, el desarrollo de aplicaciones de software educativo obliga a conjugar el trabajo de equipos multidisciplinarios, especialistas en diversas áreas de computación gráfica, redes, educación, etc. colaborando en pro de un objetivo común: el desarrollo de un SEI específico. En términos de desarrollo de software, estas aplicaciones incorporan características de STI, de agentes inteligentes y personificados, de entornos virtuales, reconocimiento de lenguaje natural y otros más; intentan, además, aplicar estrategias pedagógicas dinámicas e inteligentes, ser adaptativas y aplicar principios psicológicos ya conocidos para generar

emociones y motivar al estudiante.

El fin último es mejorar el aprendizaje de un dominio de conocimiento a través del uso de la tecnología. Para lograr esto, se busca generar un entorno de aprendizaje lo más natural posible para el estudiante, que reproduzca (no necesariamente con realismo) las condiciones y características del entorno que se desea enseñar (desde simbiosis en mundos submarinos a operación de plantas nucleares), en que la interacción entre tutor-alumno (o entrenador-alumno) se establezca lo más similar posible a lo que sería una buena instrucción en un entorno real. Lograr esta “naturalidad” en el entorno y en la interacción con el tutor es una labor compleja, para la que hace falta la aplicación de las técnicas de desarrollo de software ya descritas.

### Visión General de los agentes

La característica común a los agentes es el ejercicio de la acción; un agente actúa y produce un efecto. Ellos pueden ser humanos o bien, animales o “cosas”. Una clasificación primaria de agentes sería entonces la de distinguir entre agentes humanos o animales y sintéticos o no naturales. Entre estos últimos es necesario distinguir entre agentes robots, que habitan e interactúan en entornos reales, y agentes software, que habitan en un entorno artificial o virtual, es decir en un sistema de computación.

Los agentes son autónomos, lo que implica independencia del ser humano. Un agente autónomo puede ser usado como modelo de sistemas biológicos, humanos o animales. Esto es una situación nueva: agentes autónomos que se desenvuelven sin intervención humana, poseen sensores para percibir el ambiente que los rodea y ejecutan acciones cuando éste cambia.

En Franklin & Graesser(1996) se analizan variadas definiciones de agentes, la mayor parte dedicada a agentes software, hasta llegar a una definición formal de agente de software lo suficientemente general para englobar a los agentes ya existentes y a sus comportamientos asociados, que pudiera responder cuál es la esencia de ser agente, qué lo diferencia de un programa, y desde esta definición básica permitir la incorporación de conceptos para la especialización de los agentes. Ellos definen un agente autónomo como un sistema ubicado dentro de un entorno, formando parte del mismo, capaz de percibirlo y actuar en él en el tiempo, en persecución de sus propios objetivos generando un efecto que percibirá a futuro en su propio ambiente. Otros autores del área coinciden, en general, con esta definición, aunque luego la especifican incorporando a la discusión los sistemas multi-agentes.

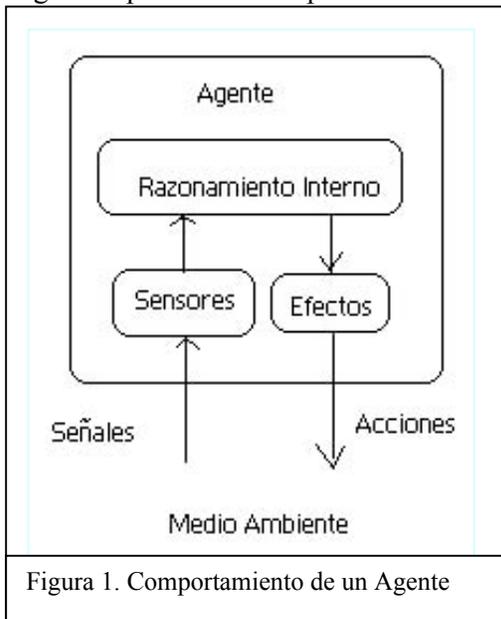


Figura 1. Comportamiento de un Agente

Dowling(2002) define un agente como un objeto que puede ser visto con una meta u objetivo, capaz de resolver autónomamente problemas a través de la interacción, colaboración, competición, negociación u otra relación con el medio que le rodea o con otros agentes. Señala además que los agentes software constituyen la aplicación de crecimiento más rápido de la Inteligencia Artificial y es un aspecto fácilmente identificable en diversos ambientes y formas de computación.

¿Cómo distinguir entre un programa tradicional y un agente? Un programa tradicional recibe una entrada (desde este punto de vista, puede estar

“percibiendo” su ambiente), procesa la información y genera una salida (al ambiente), pero es incapaz de percibir cómo se modifica su ambiente por la salida generada, siendo esta una de las principales diferencias entre un programa tradicional y un agente. Otra diferencia importante es que un programa “vive” sólo cuando se le llama a ejecución, quedando inactivo el resto del tiempo, en tanto un agente está permanentemente activo y atento a su entorno. Franklin & Graesser(1996) señalan enfáticamente que todos los agentes software son programas, pero no todos los programas son agentes software.

En la mayor parte de los dominios, sin embargo, el agente sólo tiene un control parcial del entorno compartiendo el control con otros agentes, usuarios u otras estructuras de control. Los entornos en general son no-deterministas, por lo que una misma acción de un agente puede tener distintos efectos dependiendo de otros parámetros del entorno.

Los agentes autónomos poseen algunas propiedades: reactividad, autonomía, orientación a la meta, continuidad temporal, comunicatividad, aprendizaje, movilidad, flexibilidad, veracidad, inteligencia entre otras (Franklin & Graesser(1996), Giraffa & Viccari(1998), Fernández et al(2004)). Franklin & Graesser(1996) señalan que las cuatro primeras características listadas las posee todo agente de software, y es la presencia (o ausencia) de las siguientes la que genera una taxonomía “natural” para ellos; un agente puede ser inteligente, móvil y de aprendizaje, por ejemplo. Esto sin embargo, no elimina otras clasificaciones posibles de acuerdo a su función o utilidad principal; por ejemplo agente de planificación y búsqueda. Aún más, sigue siendo posible el clasificarlos de acuerdo a otros criterios; por ejemplo por sus estructuras de control en: planificadores, adaptativos y reguladores; o por sus mecanismos de control en: algorítmicos, basados en reglas, planificadores, difusos, de redes neuronales o aprendizaje automático; su arquitectura interna, su capacidad de percepción del entorno, y otros criterios darían lugar a otras clasificaciones adicionales. Desarrollar una taxonomía de agentes que sea suficientemente generalizable es un trabajo difícil, ya que ésta debe ser suficientemente abstracta para cubrir estas posibilidades.

Una taxonomía general de los distintos tipos de agentes, de entre varias existentes, es presentada por Pfeifer(1999) :

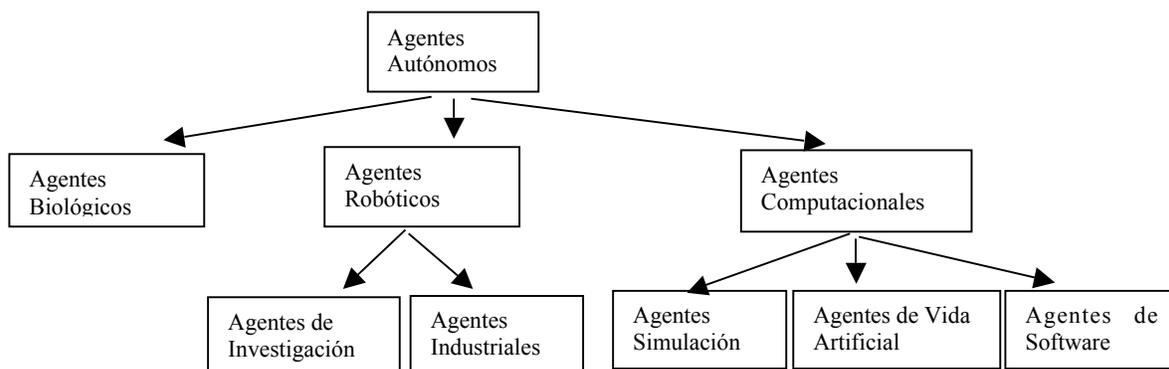


Figura 2, Taxonomía de Agentes

Desde esta clasificación, si un agente de software desarrolla una función específica, sería ésta la que lo caracterizaría, existiendo entonces agentes comunicacionales, agentes de información, agentes planificadores, etc.

En general, los agentes no actúan solos, sino como una sociedad de agentes o un sistema multiagente, en el que las tareas se dividen repartiendo la responsabilidad entre los agentes integrantes, que se especializan en alguna labor; de esta manera, el sistema se torna más flexible permitiendo una eficiente gestión de recursos. Esta sociedad de agentes habita un entorno que ha integrado en sí a un conjunto de objetos, algunos de ellos manipulables por los agentes (pueden ser percibidos, creados, destruidos o modificados por ellos); los agentes constituyen, por tanto, las entidades activas del sistema y existe un conjunto de relaciones y operaciones que hacen posible que los agentes perciban, produzcan, consuman, transformen y manipulen objetos del entorno; asociado a todo esto, existen operadores que representan la aplicación de las operaciones sobre el mundo virtual y la reacción de éste al ser modificado.

La habilidad social o de comunicación de los agentes surge entonces en el diálogo para la asignación de tareas a ser realizadas, para el desarrollo de trabajos comunes, para la coordinación y negociación de actividades; éste diálogo se desarrolla en un lenguaje propio, generalmente: KQML (Knowledge Query and Manipulation Language) o FIPA ACL (Foundation for Intelligent Physical Agents)(<http://www.fipa.org>, <http://www.cs.umbc.edu/kqml>).

Por otro lado, el desarrollo de sistemas multiagentes es una tarea compleja, para la que se plantean diferentes arquitecturas, entornos de desarrollo (como ZEUS, AgentBuilder) y plataformas de ejecución (JADE, FIPA-OS, Jackal, OAA entre otras). La tendencia general es aplicar una mezcla de técnicas y metodologías ya existentes (orientación a objetos, sistemas expertos, inteligencia artificial, etc.), dado que, aunque existen algunas metodologías específicas disponibles, éstas son aún de ámbito académico. En torno a las plataformas de agentes; JADE es actualmente la más usada, está basada en Java y presenta una serie de utilidades para el desarrollador (<http://sharon.cselt.it/projects/jade/>), FIPA OS <http://fipa-os.sourceforge.net/> (Fernández, 2004, b).

### **Agentes Pedagógicos**

Los agentes pedagógicos son agentes inteligentes, y como su desempeño es en un entorno virtual, suelen denominarse también agentes virtuales, por lo que al hablar de agentes pedagógicos muchas veces se habla también de agentes pedagógicos virtuales inteligentes.

Aunque un agente pedagógico no necesariamente debe poseer forma humana, diversos estudios han demostrado que sí debe poseer características antropomórficas, es decir, se le deben poder atribuir características habituales del comportamiento humano. Un agente pedagógico virtual habita un entorno virtual que es el medioambiente en el que se desenvuelve; al ser inteligente posee las características asociadas a un agente inteligente: reactividad y autonomía en su actuar, capacidad de tomar decisiones en pro del logro de sus objetivos, entre otras.

Como primer antecedente histórico, los libros cuentan historias de cabezas parlantes pertenecientes a Roger Bacon, a Carlo Magno, al maestro de Tomás de Aquino, al obispo Grosseteste y al Papa Silvestre II, entre otros. Por otro lado, en el siglo XVI, el gran médico y alquimista Paracelso daba la receta para la creación de un homúnculo, un diminuto ser humano viviente.

Se dice que, hacia 1550, Eliyah de Chelo creó un hombre artificial, llamado Golem. Golem es una palabra talmúdica que se refiere a todo lo sin forma, como un embrión o la masa informe del polvo con que Yaveh creó a Adán. En 1580 nació otra leyenda de creación de un Golem,

según la cual el rabino de Praga creó un Golem para defender a los judíos de la ciudad contra los gentiles; estos Golems, seres artificiales, eran dotados de comportamiento autónomo (“vida”) a través de una invocación mágica y por un sello que llevaban en la frente. Estas leyendas de creación son importantes porque la idea del hombre fabricado por el hombre queda separada de un trasfondo técnico.

Remontándose en la historia, se podría considerar que los primeros hombres virtuales fueron los autómatas construidos en los siglos pasados. Desde la Grecia antigua existe la noción de autómatas: en efecto el término "automatos" significaba "que se mueve por si mismo" y era sinónimo de misterio y fascinación. En la historia de los autómatas, durante el siglo XVIII, se destacan la creación de piezas excepcionales: los andróides. Estos autómatas con formas humanas son el resultado del genio y la habilidad de creadores magistrales como Vaucanson, en Francia, y Jaquet-Droz en Suiza.

Los primeros hombres mecánicos fueron las figuras móviles de los campanarios de fines de la edad media. Encima de la Piazza San Marco hay dos grandes figuras que golpean una campana para dar las horas. La utilización de las máquinas era en cierta forma una magia disfrazada, una manera de hacer aceptable lo que todavía era imposible. Se decía que Regiomontano construyó un águila artificial que voló para saludar al emperador Maximiliano cuando éste se acercó a Nuremberg en 1470; se supone que Leonardo da Vinci construyó un león artificial en 1500 para el rey de Francia, Luis XII. Se dice que hacia 1640, Descartes creó un autómatas que guardaba en una caja y transportaba siempre consigo, hasta que en un viaje un capitán le obligó a lanzarlo a la mar.

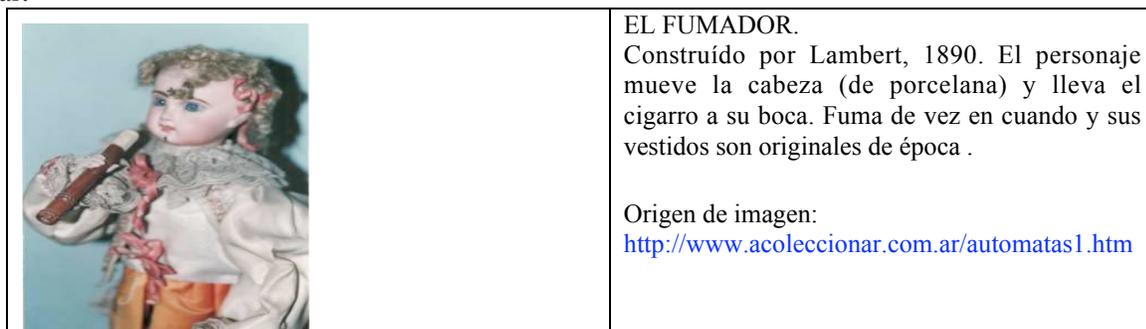


Figura 3. El Fumador

Al intentar relacionar estos intentos con la creación de humanos virtuales en nuestras modernas aplicaciones debemos remontarnos a la creación de las primeras computadoras. Aunque las computadoras digitales están sustentadas en conceptos desarrollados en siglos pasados, es entre los años 30 y los 40 cuando quedan realmente disponibles para los investigadores. Se dice que la primera persona que concibió el uso de la computadora más allá de sus aplicaciones militares, como una herramienta para transformar el pensamiento y la actividad creativa del hombre, fue Vannevar Bush en 1945. En su trabajo “As we may think” (<http://www.theatlantic.com/unbound/flashbks/computer/bushf.htm>), describe las dificultades que el hombre tiene para conseguir, revisar, manejar y transmitir los resultados de su investigación. Para resolver este problema él inventa MEMEX, un dispositivo con aspecto de escritorio, con teclado y conjunto de botones y manillas, donde se podía guardar libros, registros y comunicaciones para consultarlos de manera rápida y flexible a través del uso de microfilms.

Memex fue la primera versión del computador personal. Bush no sólo fue un visionario de la

aplicación de la computadora para almacenamiento y recuperación de información, y el valor del indexamiento asociativo en esa actividad, sino también anticipó la naturaleza multimedia del uso del computador en el futuro.

Es entre los años 60 y 70 que cuatro tendencias independientes comenzaron a trabajar en el área de las interfaces, generando la disciplina que se extendió y desarrolló a partir del año 1982, año en que se celebró la primera conferencia que trataba el tema, la "National Bureau of Standards Conference on Human Factors in Computer Systems". Estas cuatro tendencias fueron: prototipos y desarrollo iterativo proveniente de la ingeniería de software; factores humanos y psicología proveniente de sistemas de computación; software de interfaz usuario proveniente de computación gráfica y modelos; y teorías y marcos de trabajo proveniente de la ciencia cognitiva. (Caroll, 2001).

El desarrollo de prototipos fue originado inicialmente por la llamada "crisis del software" de los años 60, en la que el repentino y rápido desarrollo del hardware exigía un desarrollo de aplicaciones software mayor al disponible. Como resultado de esta crisis surgió la disciplina de la ingeniería de software, los métodos y técnicas de desarrollo estructurado y también el ver al diseño como una actividad iterativa, así el uso de prototipos para ir "visualizando" el sistema y el cumplimiento de sus requisitos fue sentando las bases del desarrollo de interfaces hombrecomputador.

La crisis del software, además, intensificó el interés en estudiar el aspecto humano de la programación, comenzando a ver a ésta como un área de la psicología que involucra resolución de problemas y manipulación de símbolos. Para finales de los 80, el grupo de investigación en psicología de l software estaba ya consolidado, dedicando su estudio a temas relacionados con la forma en que las personas interactúan con computadores (Caroll, 2001).

Surge una visión del desarrollo de interfaces hombre-computador como una disciplina derivada de teorías y métodos de la ciencia cognitiva aplicada al desarrollo de software y centrada principalmente en la creación de conceptos y técnicas de interfaces de usuario gráficas que permitan la creación de sistemas más poderosos, más útiles y más utilizables. Un primer enfoque estuvo centrado en la evaluación de la usabilidad de las interfaces, otro en los métodos de diseño de las mismas, y un tercero en el desarrollo de arquitecturas específicas para las interfaces de usuario, que separaran la funcionalidad del sistema de su manipulación. Por otro lado, la parte más visible de la interacción hombre-computador son las interfaces de usuario, cuyas herramientas y software se relacionan con conceptos y metáforas de interfaz con técnicas de interacción y despliegue, y con métodos de desarrollo de software. Una meta crucial para el área ha sido siempre el desarrollo de sistemas interactivos, meta que ha traído consigo grandes desafíos y que, gracias al desarrollo tecnológico, se ha ido cumpliendo con niveles de interactividad cada vez mayores.

A principios de los 80, el principal medio de interacción era textual, pero hacia el fin de la década Apple introdujo HyperCard en sus computadores personales, generando un cambio importante en la situación. Al mismo tiempo, se estandarizaron formatos de imágenes y surgieron herramientas que hicieron más fácil crear y visualizar estas imágenes; por otro lado surgieron también los primeros sintetizadores de voz. Nickerson (1976) fue uno de los primeros en modelar interfaces de computadora sobre la base del modelo de conversación humana, proponiendo una lista de características de la conversación cara-a-cara que podrían ser aplicadas exitosamente a la interacción hombre-máquina, incluyendo iniciativa mixta, comunicación noverbal y reglas para transferencia de control. Brennan (1995) elaboró guías

para el desarrollo de interfaces conversacionales basándose en observaciones de conversaciones humanas.

Las técnicas de visualización y gráficas en general fueron centrales en el desarrollo del actual paradigma de interfaces de usuario gráficas. Actualmente, los continuos avances tecnológicos permiten animaciones a gran escala, en que los usuarios pueden navegar y participar activamente. Con el uso de dispositivos adecuados, incluso es posible una “inmersión” en el mundo virtual que se presenta. La inteligencia artificial ha jugado un importante rol en el desarrollo de estas interfaces, especialmente en la integración de las interfaces gráficas a los sistemas tutoriales inteligentes, primero haciéndolos inteligentes al separar el conocimiento del dominio de las estrategias pedagógicas y de la planificación e interacción con el alumno; y luego, a través de la utilización de agentes, que actúan como planificadores, coordinadores de actividades y como “rostro” de la aplicación para la interacción con el estudiante.

En su origen, fue Bolt (1980) quien presentó uno de los primeros sistemas con interfaz-usuario gráfica, basado en el estudio de modalidades no-verbales en conversaciones; éste fue “Put-thathere”, sistema que usaba reconocedores de lenguaje y dispositivos con seis grados de libertad para capturar la entrada de gestos del usuario y permitirle manipular objetos desplegados en un telón. Utilizaba una arquitectura simple que combinaba entradas de texto y gestos en comandos simples que eran resueltos por el sistema. Primero se intentaba solucionar las instrucciones textuales y luego, si se requería, se interpretaban los gestos (generalmente apuntando a algo).

Luego, hubo variados intentos por personificar agentes con humanos o animales, sin embargo, Cassell et al (1999) señala haber sido la primera en construir un agente conversacional personificado en el Centro para Modelación y Simulación humana, de la Universidad de Pennsylvania, en conjunto con su equipo de trabajo y luego de 10 años de estudio de aspectos verbales y no-verbales de comunicación humana. Ella señala específicamente que su trabajo introdujo las primeras reglas para la generación autónoma de comportamiento en la comunicación verbal y no-verbal de los personajes animados. Por otro lado, ya en el año 1998, Johnson et al (1998) publicaban sus primeros resultados de desarrollo de un agente pedagógico personificado, llamado “STEVE”, en un ambiente virtual de simulación, en el que podía interactuar con sus estudiantes. En paralelo, otros investigadores de diversos lugares han ido publicando resultados de sus experiencias con agentes, por lo que se torna difícil establecer cronologías y autorías específicas tan recientes.

Johnson et al (2000) señalan el surgimiento de los agentes pedagógicos animados (AP) como un nuevo paradigma, señalando que estos personajes autónomos habitan entornos de aprendizaje en que los tutores pueden desarrollar interacciones cara-a-cara con los estudiantes más ricas y complejas, lo que a su vez genera nuevas posibilidades: pueden efectuar demostraciones prácticas de problemas complejos, movilizarse y gesticular para capturar la atención del estudiante sobre algún aspecto específico de la tarea, y enfocar las respuestas emocionales hacia la situación de tutoría: prometen ampliar los canales de comunicación tutoriales y mejorar la habilidad de los entornos virtuales para motivar y comprometer a los estudiantes en su actividad de aprendizaje.

Los autores sostienen que este nuevo paradigma de agentes pedagógicos animados surge producto de dos áreas de investigación, ambas con un fuerte respaldo de trabajo de investigación:

- agentes de interfaz animados: que provee una nueva metáfora para el diálogo hombre-máquina basada en el diálogo cara a cara.
- ambientes de aprendizaje basados en el conocimiento: que pretenden que el software instruccional sea capaz de adaptarse a los estudiantes individualmente a través del uso de IA.

Se destaca que las raíces de estos agentes están en el estudio previo de los ambientes de aprendizaje basados en el conocimiento, en que los estudiantes pueden aprender y practicar destrezas en un “mundo virtual”, en que el computador generalmente puede interactuar con los estudiantes a través de iniciativa mixta en diálogos tutoriales.

Un agente que convive con los estudiantes en un entorno de aprendizaje puede comunicarse con éste por medios verbales (el tradicional medio de interacción), y usando lenguaje no verbal (miradas, movimientos, gestos) para capturar y enfocar la atención del estudiante, o bien regular los turnos de conversación de iniciativa mixta. Movimientos de cabeza y otros gestos o expresiones faciales pueden realimentar al estudiante sin interrumpir necesariamente su línea de pensamiento.

Todos estos componentes no verbales son habituales en un diálogo cara a cara entre humanos. Johnson plantea que esto motiva el interés de los estudiantes en el sistema, y se fundamenta en estudios desarrollados por Lester (1997) y Walker et al (1994) que sostienen que la sola presencia de un AP es suficiente para motivar a los estudiantes a buen desempeño.

En resumen, los APs tendrían dos ventajas claras sobre el trabajo anterior:

- aumentar los medios de comunicación/interacción computador-estudiante
- motivar y comprometer mejor y más rápido al estudiante.

Al igual que otros agentes, un AP necesita ser veraz en su comportamiento, el cual debe ser natural y apropiado, emulando al comportamiento humano. Johnson justifica esta necesidad señalando que:

- 1º- los agentes más similares a los humanos comprometen más a los estudiantes, haciendo más agradable la experiencia del aprendizaje.
- 2º- los comportamientos “antinaturales” (poco humanos) habitualmente llaman la atención sobre el agente y distraen al usuario del problema en cuestión.

Señala también que un AP requiere de una estrecha relación entre su apariencia física y sus estados internos; y de la capacidad de ejecutar tareas y lograr metas en ambientes complejos y dinámicos. Arquitecturas como RAP (Firby, 1994) y SOAR (Laird et al, 1987) se han usado para crear agentes con capacidad de adaptación a cambios en su ambiente, que pueden planear y ejecutar acciones. También pueden colaborar con otros agentes en el logro de metas comunes mostrando un comportamiento robusto en ambientes no predecibles, como el de aprendizaje en que deben tratar con el entorno en sí y con el comportamiento del estudiante sobre el mismo, que por naturaleza es impredecible. Johnson destaca que la necesidad de dar soporte a la instrucción y sus interacciones genera requerimientos específicos sobre un AP, especialmente de comprensión profunda de acciones que de otra forma serían solo parte de la tarea a desempeñar y de sus consecuencias (para ser capaz de responder y explicarse)

### **Ejemplos de APs en la actualidad**

CARTE (Centre for Advanced Research in Technology for Education), del instituto de ciencias de la información de la USC ha desarrollado dos agentes:

- Steve (Soar Training Expert for Virtual Environments), un agente pedagógico que facilita el proceso de aprendizaje en el entrenamiento de dispositivos navales (operación de un submarino nuclear) (Johnson et al (1998); Johnson & Rickel (2000)).
- Adele, desarrollada para ejecutarse sobre equipos de escritorio con interfaz convencional. Los cursos a los que da soporte actualmente están orientados a la formación médica continua. (<http://www.isi.edu/isd/ADE/ade-body.html> , Shaw et al (1999), Johnson et al (1998) ).

La North Carolina State University's IntelliMedia Initiative ha desarrollado tres agentes:

- Herman the bug, que habita un entorno de aprendizaje para niños que trata de la anatomía botánica y la fisiología. "Design-A-Plant" simula condiciones ambientales para que los niños puedan cuidar plantas y verlas crecer en esas condiciones. Herman es un insecto hablador que da consejos para la resolución de problemas a los estudiantes mientras observa sus acciones. Para explicar conceptos, Herman camina, se recoge, se expande, nada, salta, hace acrobacias entre otras cosas. (Moreno et al (2001), <http://condor.depaul.edu/~elliott/papers/ijcai97/node16.html>, <http://www.csc.ncsu.edu/eos/users/l/lester/www/imedia/DAP.html> )
- Cosmo da consejos para la resolución de problemas en Internet Protocol Advisor. Los estudiantes que interactúan con él pueden aprender de rutas de redes y navegación por subredes. Para cada subred, pueden enviar su paquete al ruteador especificado y observar los ruteadores adyacentes. (<http://www.csc.ncsu.edu/eos/users/l/lester/www/imedia/IPA.html>, Towns et al (1998 a/b). Al tomar decisiones sobre factores como resolución de direcciones y congestión de tránsito, aprenden de los fundamentos de las topologías de redes y los mecanismos de ruteo. Cosmo explica como están conectados los computadores, cómo se ejecuta el ruteo y el rol que juegan las consideraciones de tráfico. Cosmo fue diseñado para estudiar "spatial deixis" en agentes pedagógicos (AP), es decir, la habilidad de los agentes de combinar gestos, movimientos y palabras para referirse a los objetos del entorno mientras se entregan consejos para la resolución de un problema.
- WhizLow habita el entorno "CPU city 3D" que representa una tarjeta madre con tres componentes principales: RAM, CPU y disco duro, una unidad de control simple y una UAL, más algoritmos de sistema como el ciclo fetch, paginación y memoria virtual y lo básico de compilación y ensamblaje. (Lester et al, 1999).

El DFKI, centro de investigación Alemán para IA ha desarrollado PPP Persona, un agente animado que da instrucciones de ayuda en línea en un entorno 2D, guiando al estudiante a través del material web, usando gestos de apuntamiento para llamar la atención hacia determinados elementos y haciendo comentarios en voz alta. El sistema PPP genera planes multimedia para las presentaciones del agente, plan que se ejecuta adaptivamente, modificándose de acuerdo a las acciones del usuario (en tiempo real) como reposicionamiento del agente en pantalla o preguntar algo nuevo. (André, 1998).

La VR-Valley Twente Foundation de Netherlands posee dos programas pilotos para el desarrollo de agentes personalizados para la instrucción en entornos virtuales 3D: Jacob y AneuRx. ([www.vr-valley.com](http://www.vr-valley.com), Evers & Nijholt (2000)).

Vincent es un agente pedagógico que ayuda al entrenamiento en procesos de aprendizaje basado en web (Paiva & Machado, 2002). Tiene representación antropomórfica, puede percibir

su entorno y actuar sobre él con la capacidad de inferir acerca de estas percepciones, resolviendo problemas y determinando qué acciones deben ser ejecutadas para lograr las metas.

### **Mejoramiento de entornos de aprendizaje a través de Agentes Pedagógicos (AP)**

Johnson et al (2000) plantea ocho aspectos beneficiados por la aplicación de AP en entornos de aprendizaje:

**1.- Demostraciones interactivas :** un entorno virtual que simula las condiciones reales de trabajo y habitado por un agente pedagógico (AP) ofrece nuevas y buenas oportunidades para enseñar al estudiante como ejecutar las tareas en ese ambiente a través de la demostración directa de los pasos a seguir, siendo posible además reforzar estas demostraciones con comentarios orales u escritos.

Se destaca que ver una demostración en un entorno virtual interactivo es mucho más útil que su descripción o incluso, que su presentación en vídeo; esto porque en un entorno virtual el estudiante puede observar la tarea desde distintos ángulos, repetir pasos, formular preguntas e incluso interrumpir para finalizar por sí mismo la tarea. Por otro lado, aún cuando el desempeño real de la actividad no sea operativo o manipulativo de maquinaria, un AP presentando gráficamente los conceptos que se desean enseñar, mostrando ciclos y etapas, siempre con la posibilidad de interactuar en cualquier momento, resulta ser muy efectivo, dada la motivación que genera en el estudiante.

**2.- Guía para la navegación:** en entornos virtuales inmersos o en ambientes complejos y grandes, los estudiantes pueden llegar a sentirse desorientados y perdidos; en estos casos, un AP que actúe como guía mientras se familiarizan con el ambiente es de gran utilidad.

Permitir una “inmersión” en un entorno virtual 3D con guías de navegación ayuda a los estudiantes a desarrollar o ampliar modelos espaciales del entorno aún desconocido hasta que sean capaces de desarrollar la navegación por sí mismo. Sin embargo, estas intervenciones no están restringidas a entornos virtuales de inmersión, y en un entorno “de escritorio” o 2D, pueden utilizarse exitosamente un AP como lo muestran algunas experiencias de agentes en la web. (<http://www.isi.edu/isd/ADE/ade-body.html>).

**3.- Miradas y gestos para orientar la atención:** con el desarrollo de la computación gráfica, las aplicaciones son cada vez más complejas y detalladas y poseen gran variedad de medios para destacar algún objeto; parpadeo y flechas entre otros. Existen agentes con una capacidad de comunicación no-verbal bien desarrollada y útil: un agente personificado habitando un entorno virtual puede destacar algún objeto por el medio más simple y más comprensible: mirándolo e indicándolo con una mano. Un AP puede emplear estos comportamientos para generar referencias específicas a objetos físicos moviéndose en el entorno virtual, apuntándolos ya sea con el dedo índice, un puntero o algún otro medio (Cassell, 2000).

**4.- Realimentación no-verbal:** una actividad básica de un tutor es la de dar realimentación a las acciones de un estudiante; de manera similar a lo que ocurre con un tutor humano, se espera que el agente pueda dar esta realimentación por medios no-verbales complementarios o sustitutos de la comunicación verbal. El lenguaje del cuerpo puede causar una fuerte impresión a los estudiantes evitando que cometan un error. Se sostiene que al ser menos intrusivo se interfiere menos en el proceso mental del estudiante, permitiéndole continuar con su trabajo. Por otro lado, puede ser usado también para felicitar más efusivamente los logros alcanzados.

**5.- Señales de conversación:** un AP permite implementar características de conversaciones cara a cara sostenidas entre humanos en un diálogo humano-computador, tornando más “natural” y por ende más verídica la interacción. Estos elementos incorporados incluyen aspectos de comunicación no-verbal (Cassell (2000), Johnson et al (2000, 1998)).

Aunque ciertamente la comunicación por vía verbal excluyente de aspectos no-verbales es posible (correo electrónico o postal, teléfono), la inclusión de ésta la enriquece, tornándola más atractiva y efectiva. En este aspecto, existen interesantes trabajos con agentes: REA (Real Embodied Agent), el agente conversacional personificado más completo desarrollado, que incluye comportamientos verbales y no verbales, generados por funciones conversacionales subyacentes, y Gandalf, que al interactuar con un usuario “vestido” con un traje especial es capaz de detectar movimientos corporales y direcciones de miradas, “oir” y analizar no solo las palabras, sino también la entonación de las mismas. Como respuesta a todos esos estímulos, Gandalf combina el texto de su discurso con la entonación, expresión facial y corporal y mirada adecuada. Cassell (1999)

**6.- Conducción de la emoción:** la motivación por aprender es un factor clave para el éxito del aprendizaje, y el rol que juegan las emociones en el logro de la motivación es crucial. Existen investigaciones en torno a la incorporación de la emoción en los agentes; un AP que “cuida” más a un estudiante lo compromete más en el resultado de sus estudios y su progreso, y puede transmitir entusiasmo por la tarea al estudiante, haciendo la experiencia del aprendizaje más interesante y entretenida para el alumno. Un alumno que disfruta y se entretiene en la interacción ocupará más tiempo en trabajar con el sistema, invirtiendo mayor tiempo en su aprendizaje.

En torno al tema, un aspecto especial está en los diversos estudios que analizan los efectos de la antropomorfización del software y sus efectos en la interacción hombre-máquina. Aún cuando el análisis hasta ahora se ha manifestado más bien teórico con un número de experiencias empíricas aún insuficientes, toda la evidencia indica que personajes similares a los humanos, o a quienes se pueda asociar características del comportamiento humano, tienden a ser pedagógicamente más eficientes motivando fuertemente a los estudiantes (Lester, 1997) A través de una buena coordinación de expresión facial, ubicación corporal, movimientos de brazos y gestos manuales un AP puede potenciar significativamente un consejo verbal ofrecido para la solución de un problema, dando ánimo, transmitiendo empatía e incrementando (quizás) su motivación.

**7.-compañeros virtuales:** las tareas complejas muchas veces requieren de acciones coordinadas de miembros de un equipo (ej. emergencias en distintos contextos). Para desempeñar una labor conjunta correctamente es necesario que cada miembro del equipo sepa claramente cuáles son sus funciones, aprenda a efectuarlas correctamente y a coordinarse adecuadamente con los restantes integrantes del equipo.

Las aplicaciones virtuales distribuidas, que representan un escenario de trabajo son una buena opción para el entrenamiento de equipos de trabajo que pueden practicar juntos en situaciones simuladas. Los AP's aquí pueden jugar dos roles: como entrenadores o como “reemplazo” de otros compañeros ausentes.

Johnson destaca que los estudios relacionados con las interacciones entre estudiantes y agentes en distintos roles (no solo como tutor o entrenador) comparten la visión del aprendizaje como un proceso social, lo que hace de esta perspectiva una “dirección” natural para futuras

investigaciones. En este punto es necesario destacar la estrecha relación que se observa entre el enfoque Vigotskyano de instrucción y el enfoque didáctico del AP.

**8.- Interacciones Pedagógicas Adaptativas:** además de las características propias de un AP, éste debe también ser capaz de efectuar correctamente las actividades propias de un tutor inteligente no personificado: contestar preguntas y dar explicaciones, plantear pruebas y evaluar las destrezas del estudiante, entre otras cosas, todo al mismo tiempo que interactúa con éste en un contexto cara-a-cara. Esto es lo que obliga a que el enfoque pedagógico incorporado a los STI con AP sea dinámico y adaptativo, en oposición a estrategias secuenciales o preplaneadas estáticas. Esta habilidad de dar instrucción oportuna basada en la situación del alumno (actualizada constantemente) es una característica del AP, implementada de diversos modos en cada aplicación.

### **Algunos resultados empíricos**

Un estudio empírico del impacto de un AP en el proceso de enseñanza aprendizaje ha sido conducido por Lester(1997) sobre la aplicación Design-A-Plant (agente Herman the bug) en 100 estudiantes. Cada estudiante interactuó con una de las varias versiones de Herman: uno que entregaba sólo consejos visuales, otro solo consejos orales y otro una combinación oral/visual. Por otro lado, cada versión presentaba diferentes niveles de consejos, algunos sólo consejos a alto nivel (basados en principios), otros a bajo nivel (orientados a la tarea) y otros ninguno. A los estudiantes se les aplicaron pre y post test, y a todos se les presentó el mismo conjunto de actividades-problemas a resolver. Los resultados del experimento presentaron una mejora significativa en aquellos estudiantes que interactuaron con el agente. Estos resultados indican, según los autores, que un agente bien diseñado en un entorno bien diseñado siempre será un aporte al aprendizaje y no un elemento distractor, como señalan algunos. Los resultados indican también que existen importantes beneficios en la introducción de agentes que empleen mensajes orales y visuales a alto y bajo nivel, así como se ve que mientras mayor es la utilidad que le encuentran al agente, mayor es la motivación por utilizar la aplicación, lo que lleva a los autores a buscar la definición de áreas en que serían particularmente efectivos.

El estudio efectuado por Lester(1997) entregó otro resultado importante: sólo la presencia de un AP en el entorno interactivo de aprendizaje fue un elemento suficiente para generar un gran efecto positivo (“efecto persona”) en la percepción de los alumnos de su propia experiencia de aprendizaje. Adicionalmente los estudiantes encontraron que los agentes más expresivos (comunicación y niveles de consejos) fueron percibidos de gran utilidad y que se comunicaban con gran claridad. Beun et al (2003) han establecido empíricamente resultados similares, al obtener resultados en que los alumnos encontraban de mayor utilidad y aceptaban mejor a aquel agente que gesticulaba más, aún cuando el contenido de las advertencias era siempre el mismo. Otro estudio sobre el agente PPP\_Persona fue efectuado para evaluar el grado en que el agente contribuye al aprendizaje; con este fin, el equipo de trabajo confeccionó dos versiones de PPPPersona: con y sin agente, ambas con el mismo contenido, los mismos mensajes e idénticas narraciones. Los usuarios fueron adultos que observaron información técnica y no-técnica, sin resolución de problemas. Los resultados mostraron que no había diferencias en la comprensión y retención de temas por la presencia (o ausencia) del agente. Esto no contradice, dice Johnson, los resultados anteriores dado el contexto absolutamente diferente de esta evaluación. Sin embargo, 29 de 30 estudiantes prefirieron la aplicación con el agente, los temas técnicos se encontraron menos difíciles y más entretenidos con el agente (lo que concuerda con el “efecto

persona” mencionado anteriormente). Johnson destaca que estas evaluaciones presentadas han sido sobre agentes de 1ª generación, pero que se espera que en pocos años las limitantes técnicas se vean superadas y se genere una mayor y mejor interacción. Destaca que en este sentido, como los AP serán más sofisticados, se torna crítico el repetir y efectuar más experimentos que den la base empírica de una teoría de AP y aprendizaje.

### **Aspectos técnicos de la implementación de un AP.**

Las nuevas tecnologías de la información abren nuevos horizontes a las aplicaciones en diferentes áreas de la vida moderna, y al expandirse y generalizarse en nuestra sociedad, se va produciendo también un proceso de integración entre diversas tecnologías que a su vez da lugar a nuevas y ricas aplicaciones. Con los AP ocurre algo así: aún siendo una nueva (y potencialmente rica) aplicación, surge producto de avances tecnológicos y reúne en sí características propias de STI y entornos interactivos de aprendizaje (e incluyen aspectos de computación gráfica, reconocimiento de lenguaje, técnicas de representación y razonamiento sobre conocimiento, modelamiento del estudiante, diseño de estrategias pedagógicas, entre otras).

Un AP ofrece nuevas ventajas y oportunidades, pero también nuevos desafíos para su desarrollo tecnológico, relacionados principalmente con temas de: interfaz, construcción, control de comportamiento, credibilidad del agente, emoción y aspectos de conectividad y plataforma.

**1.- interfaz:** permite ampliar el canal de comunicación con el estudiante por diferentes vías verbales y no-verbales. Considera el ambiente de aprendizaje, a los estudiantes y a los agentes que colaboran en el entorno. La interfaz entre el agente y su ambiente puede ser dividida en dos partes: su percepción o consciencia del entorno y sus acciones motoras, es decir, su capacidad de afectar a este ambiente. La interfaz con el usuario considera aspectos de funciones de comunicación, ubicación del estudiante en el entorno y de acciones permitidas; también considera acciones para motivar o estimular al estudiante, o para verificar que presta atención a una demostración, o para ver el turno de conversación, o para intentar orientar su atención hacia algún objeto específico de su entorno, entre otras. Johnson destaca que algún agente es capaz ya de distinguir gestos y seguir las miradas de sus estudiantes, y también que se espera potenciar esta capacidad de detección de expresiones faciales y de emociones en un futuro próximo.

La interacción entre el “cuerpo” de un AP y su entorno exige conocimiento espacial del mismo que permita la “visualización” de objetos para su posterior manipulación o señalamiento, o simplemente para desplazarse de una forma “natural” (sin atravesar paredes ni objetos) por el entorno.

**2.- Construcción:** los enfoques más conocidos son por “espacios de comportamiento” o bien por “generación dinámica de comportamiento”. El primer enfoque es el método más común para la construcción del comportamiento de un AP y consiste básicamente en una biblioteca de fragmentos de comportamiento. Para generar el comportamiento del agente, un motor secuenciador enlaza dinámicamente estos fragmentos uniéndolos en tiempo de ejecución. Cuando esto está bien hecho, el comportamiento del agente parece un todo al estudiante y le provee un consejo visualmente contextualizado para la resolución del problema.

La generación dinámica de comportamiento, en cambio, es un enfoque alternativo al anterior; en éste se genera completamente cada animación que es requerida sin reutilizar segmentos de animación, ni siquiera un frame. Este enfoque ha sido usado por Jack

([www.agentsoftware.com](http://www.agentsoftware.com)), Steve y WhizLaw, entre otros, y considera modelos 3D de cada agente segmentado en sus partes movibles. Además, incluyen algoritmos que pueden tomar especificaciones de la postura deseada y generar los movimientos “corporales” adecuados para llevar al agente desde la posición actual hasta la deseada. Por ejemplo, si Steve quiere mirar un objeto, un algoritmo generará un camino de animación para dirigir el movimiento de su cabeza. La dificultad está en el número de partes del cuerpo que se deben mover coordinadamente. Este enfoque generativo trabaja bien tanto para texto como para animaciones. Mientras el enfoque anterior unía piezas preexistentes, éste enfoque genera el discurso desde fonemas individuales, incluso variando la entonación, si así se le instruye, desde un tono duro hasta uno amigable. Es difícil lograr el mismo nivel de calidad que es posible con otros enfoques. Por ahora se debe intentar equilibrar la flexibilidad y la calidad, aunque se espera que en corto plazo, producto del avance tecnológico y la investigación en animación gráfica, sea más atractivo este enfoque generativo.

**3.- Control de Comportamiento:** controlar el comportamiento de un AP requiere atención a muchos aspectos. Al igual que los demás agentes, estos deben ser capaces de responder autónomamente a su entorno dinámico. Adicionalmente, como un STI, el AP debe ser capaz de establecer un diálogo coherente con el estudiante y tomar decisiones pedagógicas apropiadas. La personificación del AP impone un énfasis en el cuidado del control de movimientos corporales, aspectos de comunicación no verbal y niveles de presencia recomendables. Las capacidades y potencialidades del AP de establecer una comunicación “más natural” son su gran ventaja frente a tutores simples, de diálogo específico, pero al mismo tiempo su desarrollo y control exige mayor trabajo y esfuerzo.

La clave para mantener un comportamiento coherente en un ambiente dinámico es mantener una rica representación del contexto. La habilidad para reaccionar ante eventos inesperados y manejar interrupciones intermedias es vital para los AP's, y se relaciona con la coherencia global del comportamiento del agente. Una buena representación del contexto permite al agente ser receptivo mientras se enfoca en el contexto general. Los AP deben mantener al menos los siguientes tres tipos de contexto:

- Contexto Pedagógico: que incluye las metas educacionales y un modelo del conocimiento del estudiante. La contribución de los AP a esta área no es relevante.
- Contexto de la tarea: representa el estado de la solución al problema, tanto del estudiante como del agente. Incluye las metas de la tarea, el estado actual del ambiente de aprendizaje y las acciones requeridas para completar una tarea. Por ejemplo, Steve y Adele usan un plan de representación jerárquico que se genera automáticamente usando planificación de tareas por descomposición. El monitoreo sobre el mundo virtual es continuo y utiliza el modelo de tareas para mantener un plan para completar la tarea, usando una variante de las técnicas de planificación. Muchas veces se incorporan también modelos de representación para la bitácora de estudiantes.
- Contexto del diálogo: representa el estado de la interacción colaborativa entre el estudiante y el agente y puede expresarse de varias maneras: pila de enfoque (focus stack) que representa toda la jerarquía de tareas, subtareas y acciones en las cuales el estudiante y el agente están involucrados; el estado de su interacción en la etapa actual de desarrollo de la tarea; un registro de cuando el estudiante o el agente es responsable de completar una tarea (puede cambiar en interacción mixta); la última respuesta a dudas del alumno y las acciones que han llevado a cabo tanto el estudiante como el agente.

Un área importante de investigación futura es la sincronización adecuada de acciones no verbales al discurso en una conversación cara-a-cara. Aunque Cassell y sus colegas han avanzado mucho en el tema queda aún la incorporación de estos resultados en una conversación AP-estudiante en tiempo real (Cassell, 2000).

**4.- Credibilidad:** Dada la aceptación casi inmediata que un AP produce en los usuarios, estos están dispuestos a ocupar más tiempo en el uso de la aplicación, por lo que sus beneficios motivacionales sean tal vez mayores que sus beneficios pedagógicos. La credibilidad que posee un AP es producto de dos fuerzas: las cualidades visuales del agente y las propiedades computacionales del sistema de control de comportamiento que crea sus comportamientos en respuesta a las interacciones con el usuario.

El logro de la credibilidad en un AP presenta tres desafíos principales:

- 1º. La meta primaria de un AP es promover el aprendizaje, y cualquier comportamiento del agente que distraiga o desconcentre al estudiante de la resolución del problema (aún cuando mejore su credibilidad) debe ser evitado.
- 2º los comportamientos de motivación y credibilidad deben complementar y algunas veces intercalarse con los comportamientos de consejos o explicaciones que los AP llevan a cabo.
- 3º Si un observador ve que un agente se está comportando como un simple autómatas le quitará gran parte de la credibilidad otorgada (o posible de obtener).

Para lograr credibilidad, los agentes típicamente exhiben comportamientos *ah-doc* que se suman a los consejos, explicaciones o atención. Las implementaciones de APs emplean algoritmos que permiten seleccionar los comportamientos más adecuados a cada circunstancia, considerando las distintas variables involucradas. Cuando el agente no está interactuando con el alumno, es activado por una máquina secuenciadora que le permite ejecutar uno de varios comportamientos apropiados, motivadores y creíbles como: enfoque visual, reubicación física (cambio de postura) desplazamiento, movimientos corporales, relajamiento y otros (limpiar gafas, por ejemplo). Herman, por ejemplo, cuando “ve” a un estudiante concentrado en un problema se dedica a algo más interesante para él, como saltar en pantalla o limpiar sus gafas.

**5.- Emoción:** Los agentes pedagógicos son visualmente expresivos y al tiempo que ayudan a los estudiantes a resolver un problema, son capaces de motivarlos, darles consejos y empatizar con ellos a través de expresiones faciales y corporales, entonaciones de voz y otros medios.

Lograr también que a través de estos medios sean capaces de transmitir emoción, o de intentar generarla con fines pedagógicos determinados en un estudiante es un desafío. El comportamiento emotivo de los agentes debe dar soporte a movimientos expresivos y complementar visualmente la solución del problema. Más aún, estos comportamientos deben ser planeados y coordinados en tiempo real en respuesta a los progresos de los estudiantes y para dar la impresión de ser verídicos deben ser comunicados por canales visuales y auditivos a la vez.

Un AP que sea visualmente expresivo puede al mismo tiempo dar consejos para la resolución de un problema y ser una fuerte motivación para los alumnos. Si resulta posible proveer a los agentes de un conjunto de expresiones faciales y gestos expresivos ellos podrían explotar el canal visual para aconsejar, motivar y empatizar con los estudiantes. Sin embargo, lograr la comunicación de contenido afectivo es un serio desafío, la expresión emotiva no es solo facial, sino que involucra al cuerpo entero, el cuerpo del agente debe ser capaz de generar

movimientos expresivos que complementen las expresiones y gestos faciales, dando así un soporte acorde al problema que se está resolviendo. Más aún estos comportamientos deben ser planificados y coordinados en tiempo real en respuesta a los progresos del estudiante. Para crear la ilusión de vida los AP deben ser capaces de comunicarse por ambos canales: auditivo y visual, también deben ser capaces de expresar distintas emociones: alegría, euforia, tristeza, pena, envidia, vergüenza y deleite entre otras. Como manejan distintos actos orales/escritos se les debe dar soporte adecuado con un amplio rango de comportamientos emotivos. Sin embargo, de todo el conjunto de posibles comportamientos emocionales, solo algunos pocos serán utilizados a la vez, dependiendo de la experiencia de aprendizaje que esté experimentando el alumno. Así, la reacción del agente será de alegría cuando el alumno logre la meta, de desaprobación si la resolución es incorrecta, inquisitiva para situaciones inciertas, etc.

Tanto el comportamiento emotivo del agente como el modelamiento afectivo del estudiante jugarán importantes roles en la investigación futura del tema. Existe una importante actividad de investigación en modelos computacionales de la emoción y están disponibles varios marcos de trabajo, sin embargo, la aplicación de estos marcos en ambientes de aprendizaje interactivos solo está comenzando.

**6.- Plataforma y aspectos de conectividad:** Hasta la fecha, los agentes que generan comportamientos interactivos trabajan sobre procesadores de alta velocidad con poderosas aceleraciones gráficas. Para reducir los requerimientos técnicos, es mejor usar el enfoque de “espacio de comportamiento”.

El problema de incorporar agentes pedagógicos en materiales de aprendizaje basados en web es interesante. La web ha surgido como un mecanismo para los cursos on line; al mismo tiempo, ésta instrucción puede ser muy impersonal, con limitantes para adaptarse y responder al usuario.

Un agente que pueda integrarse con materiales basados en web es deseable porque puede aplicarse a un amplio número de materiales de cursos distintos, y porque puede mejorar la interactividad y responsabilidad sobre el material.

El principal problema técnico asociado con los agentes web es intentar hacer coincidir la rapidez propia de la interacción cara a cara con la lentitud de respuesta de la red. Por otro lado, necesitan responder a las acciones de los alumnos, mirar sus acciones, aprobar, interrumpirlo, responderle si procede. La interactividad puede ser difícil de lograr.

Dos agentes, PPP Persona y Adele, solucionan este problema a través de un comportamiento reactivo desde el servidor. PPP baja una máquina de estados al cliente para la ejecución del comportamiento general del agente, mientras la planificación está en el servidor. (André(1998), <http://www.isi.edu/isd/ADE/ade-body.html>).

### **Entornos Virtuales y Agentes Pedagógicos**

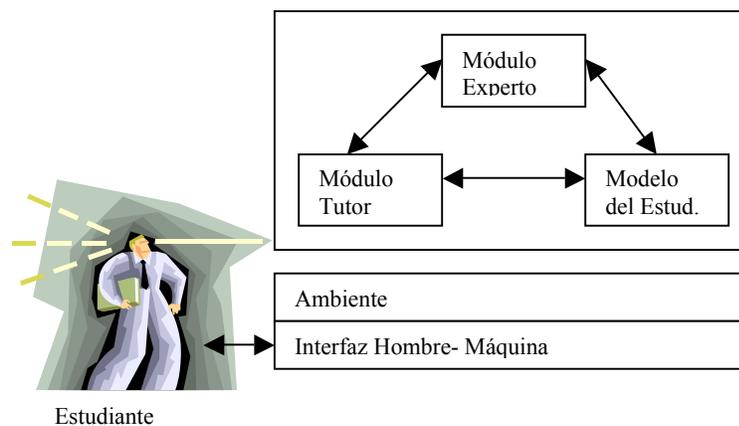
Un entorno virtual (EV) es una simulación informática en la que se representa el funcionamiento de un ambiente real específico en un entorno gráfico 2D o 3D (de escritorio o inmersivo) en el que se desea que los estudiantes adquieran alguna destreza y/o conocimiento a través de la ejercitación práctica simulada de algún comportamiento. Estos EV son especialmente útiles en el entrenamiento/aprendizaje de la operación de maquinaria de alto costo y/o riesgo: navegación espacial, plantas nucleares, maquinaria militar sofisticada, otras.

En los entornos más interactivos, el estudiante es representado físicamente al interior del

mismo a través de un “avatar” con el que puede desplazarse y ejecutar acciones libremente. Las decisiones pedagógicas están a su vez centradas en un agente personificado de software inteligente, el “agente pedagógico”, cuyas características ya se han descrito y que, según las características particulares de cada aplicación, puede jugar uno o más roles: el tradicional de tutor, un entrenador, un compañero de equipo o un contrincante.

El desarrollo de entornos virtuales es muy reciente. Las primeras aplicaciones datan de los años 90, y dado esto y la complejidad que poseen debido al gran número de áreas involucradas en su desarrollo, no se puede hablar aún de que existan arquitecturas o procesos de desarrollo estandarizados para la construcción de ellos. Muchas de las arquitecturas propuestas se enfocan sólo al problema que se está resolviendo, sin ser generalistas; por otro lado, algunos procesos de desarrollo olvidan normas básicas de la ingeniería de software: flexibilidad, modularidad, cohesión y otros. En particular, De Antonio et al (2005) presentan una arquitectura basada en agentes para el desarrollo de entornos virtuales para entrenamiento, que tiene su fundamento en la arquitectura original de los Sistemas Tutoriales Inteligentes.

Distintos autores en la materia plantean diferentes modelos de arquitecturas para STI's. La arquitectura general más conocida para la implementación de estos sistemas presenta cuatro módulos interrelacionados entre sí:



Estudiante

Figura 5. Arquitectura de un Sistema Tutorial Inteligente (STI)

donde:

- **Módulo Experto:** contiene el conocimiento del dominio; tiene entre sus funciones la identificación de fallos y la resolución de problemas. Se puede implementar a través del uso de simulaciones, reglas de producción, redes semánticas, etc.
- **Modelo del Estudiante :** diagnostica lo que el estudiante conoce; es la representación abstracta del estudiante en el STI, permite adaptar el funcionamiento del tutor a las características cognitivas del alumno, se construye como un modelo diferencial, por enumeración de errores o por generación de errores.
- **Módulo de Tutoría o instructor:** selecciona cuáles deficiencias de conocimiento enfocar y selecciona estrategias para presentar ese conocimiento; construye el modelo del estudiante a través de la evaluación de su nivel de conocimientos, determina qué temas presentar y las estrategias de tutoría a utilizar, cuándo interrumpir y en general los aspectos relacionados con la enseñanza.

- **Módulo de Comunicación (Ambiente Instruccional e Interfaz Hombre máquina):** establece la comunicación con el usuario-estudiante a través de la interfaz de la aplicación.

De esta arquitectura se desprenden algunas características:

- Los STI son capaces de representar separadamente, a través del uso de técnicas IA, los distintos dominios de conocimiento que manejan.
- Se les califica como "inteligentes" en contraste con los sistemas de instrucción por computadora (CIA), siendo la diferencia el uso de técnicas IA para la representación del conocimiento y explicación de su razonamiento.
- Otra razón para atribuir "inteligencia" a estos sistemas está en su capacidad de resolver los problemas que presenta a los estudiantes, y explicar cómo lo hizo.

*Aplicaciones de Agentes Pedagógicos en Entornos Virtuales para la Enseñanza .*

- En un STI, se usan técnicas IA tales como la planeación, optimización y búsqueda, dejando que el sistema decida "inteligentemente" el orden de presentación del contenido al alumno.
- En torno a la interacción tutor-alumno en un STI; la actitud del tutor puede ser **reactiva** (reacciona ante acciones del alumno) o **preactiva** (interviniendo cada vez que las acciones del alumno lo permitan). Un buen tutor presta **asesoría**, conjugando una actitud proactiva para intervenir en momentos importantes ofreciendo ayuda al estudiante, e interviniendo reactivamente cuando el alumno lo requiera.

Los STI son difíciles de diseñar e implementar, se requiere la integración de los distintos módulos. Son aplicaciones complejas, basadas en el conocimiento, resuelven problemas y prestan tutoría asistida por computador. Luego, no es de sorprender que la suma sea mayor que sus partes.

La primera suposición básica para la adaptación de esta arquitectura es que un agente pedagógico en un EV de entrenamiento es la personificación del módulo tutor; adicionalmente otras consideraciones importantes son: que en un EV de este tipo se entrenan generalmente equipos de trabajo, por lo que el agente pedagógico deberá atender a las necesidades individuales de cada estudiante y a las del grupo en general; que el modelo del estudiante deberá reflejar no sólo el conocimiento individual de cada estudiante, sino también el del equipo; que cada estudiante está realmente interactuando desde el interior del entorno a través de su avatar; y, que en un sistema de estas características, la interacción se produce a través de manipulación de elementos gráficos, pudiéndose utilizar además comunicación verbal y no-verbal en el desempeño de la tarea (De Antonio et al, 2005).

La arquitectura basada en agentes presentada por De Antonio et al (2005), presenta cinco agentes principales asociados a los cinco módulos claves de la arquitectura de un sistema tutorial inteligente extendida con la incorporación de un módulo "world" que permite el control sobre el entorno en sí mismo. Los agentes considerados son: un agente de comunicación (Communication Agent), un agente de modelamiento del estudiante (Student Modeling Agent), un agente mundo (World Agent), un agente experto (Expert Agent) y un agente pedagógico (Tutoring Agent). Cada uno de estos agentes puede interactuar con los otros y delegar tareas a agentes subordinados de la sociedad multi-agente resultante. De esta forma, cada estudiante podrá ser atendido individualmente, al tiempo que las necesidades del equipo están también cubiertas.

Los autores sostienen que esta proposición, y su implementación en una plataforma genérica de

agentes configurables, facilitarán el diseño y la implementación de nuevos entornos virtuales de entrenamiento, permitiendo la reutilización de componentes de software y el crecimiento de un sistema por la incorporación de nuevas funcionalidades. La arquitectura basada en agentes presentada es una buena solución para un problema existente, permite visualizar la organización interna del sistema y de los agentes, al tiempo que da las bases para un desarrollo modular. De una u otra manera, los investigadores del área están tendiendo al planteamiento de soluciones como ésta, algunas basadas en objetos u otros enfoques, pero todos coincidiendo en la necesidad de presentar una arquitectura de software generalista y un enfoque de desarrollo claro para la construcción de nuevas y robustas aplicaciones.

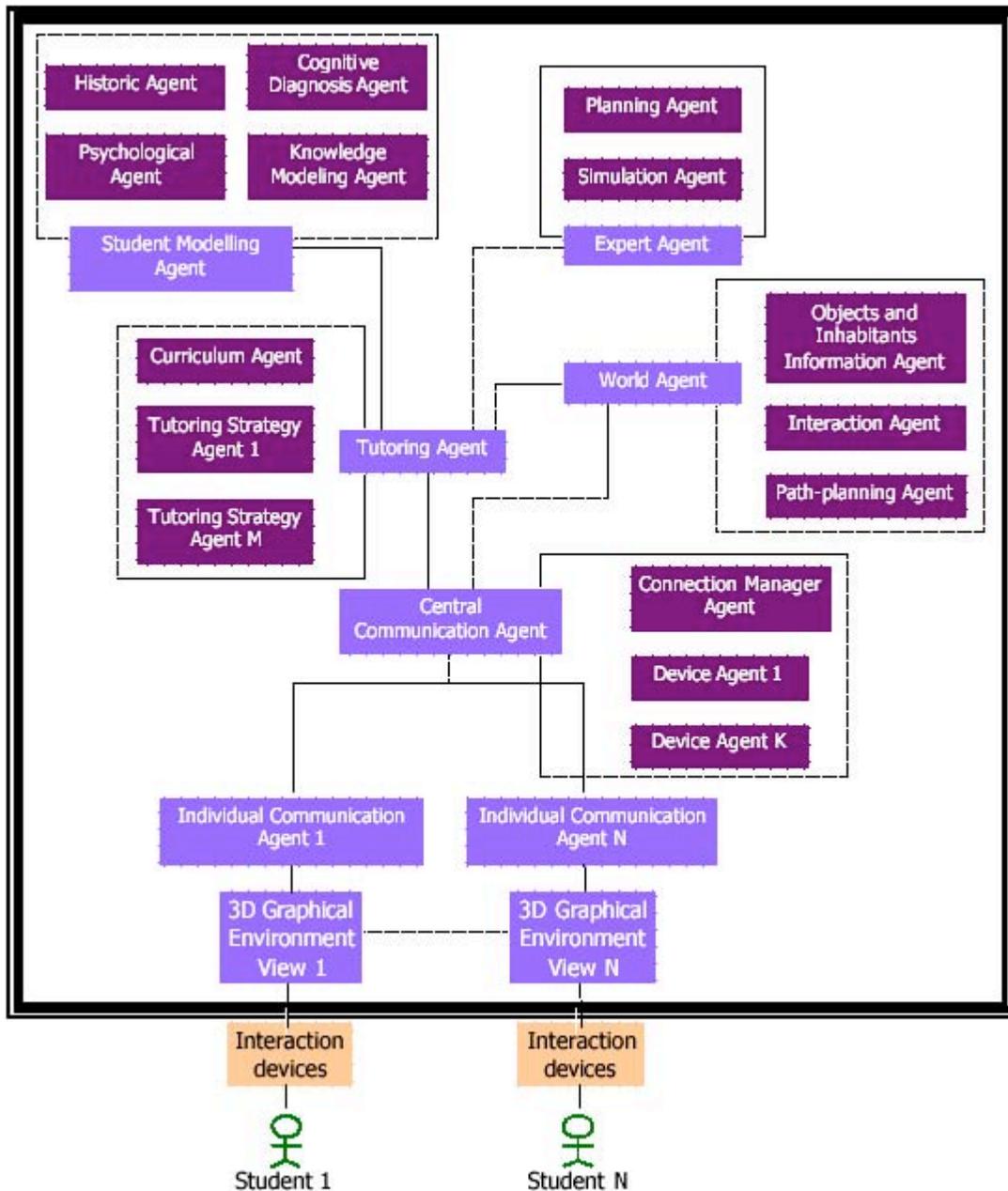


Figura 6. Arquitectura basada en Agentes para un Entorno Virtual de Entrenamiento.

## Conclusiones

El desarrollo continuo de nueva tecnología y aplicaciones de software impulsa a su vez al crecimiento y la mejora de las aplicaciones informáticas educacionales, perfeccionándolas y mejorándolas al incorporar en ellas los nuevos avances del área.

Así, la incorporación de agentes de software a los sistemas tutoriales inteligentes, enmarcados entornos de aprendizaje virtuales ha dado origen a una nueva generación de aplicaciones educacionales, de grandes potencialidades y que demanda aún una gran cantidad de trabajo en investigación y desarrollo.

El desarrollo de estas aplicaciones es reciente, en los años 90 se comenzaron a desarrollar las primeras experiencias. Dado el interés que despiertan y sus potencialidades pedagógicas, se espera un potencial incremento en el desarrollo de éstas, y en un futuro cercano su salida del ámbito académico al mundo industrial y comercial. Sin duda el continuo crecimiento tecnológico favorecerá esto.

## Referencias Bibliográficas

1. De Antonio, Angélica; Ramírez, J. Méndez, G. chapter 8 “*An agent-based architecture for virtual environments for training*” In: “*Developing Future Interactive Systems*”, Sánchez S., María Isabel, Carlos III Technical University of Madrid - Spain, 2005.
2. Dowling, Carolyn “*The socially Interactive Pedagogical Agent within Online Learning Communities*”, Proceedings of the international Conference on Computers in Education (ICCE’02), 2002.
3. Fernández et al[a] Fernández Chamizo, Carmen; Gómez Sanz, Jorge ; Pavón Mestras, Juan *Desarrollo de Agentes software - Introducción a la tecnología de agentes*, Dep. de Sistemas Informáticos y Programación <http://www.fdi.ucm.es/profesor/jpavon/doctorado/sma.pdf> accesado en abril de 2004.
4. Fernández et al[b] Fernández Chamizo, Carmen; Gómez Sanz, Jorge ; Pavón Mestras, Juan *Desarrollo de Agentes software – Metodologías de desarrollo de sistemas multiagente*, Dep. de Sistemas Informáticos y Programación
5. Franklin & Graesser[1996] Franklin, Stan; Graesser, Art *Is it an Agent, or just aProgram?: A Taxonomy for Autonomous Agents* Proceedings of the [Third International Workshop on Agent Theories, Architectures, and Languages](http://www.msci.memphis.edu/%7Efranklin/AgentProg.html), Springer-Verlag, 1996. En <http://www.msci.memphis.edu/%7Efranklin/AgentProg.html> 04/2004
6. Giraffa, LMM & Viccari, RM “*The use of Agents Techniques on Intelligent Tutoring Systems*”, Computer Science, 1998. SCCC’98 International Conference of the Chilean Society of, 9-14 nov. 1998, p.76-83.
7. González, C. *Sistemas Inteligentes en la Educación: Una revisión de las líneas de investigación actuales*. *Revista ELección de Investigación y EValuación Educativa*, v.10, n.1. [http://www.uv.es/RELIEVE/v10n1/RELIEVEv10n1\\_1.htm](http://www.uv.es/RELIEVE/v10n1/RELIEVEv10n1_1.htm). Consultado en 05/2004.
8. Guardia Robles, Bruno *Asesores inteligentes para apoyar el proceso de enseñanza de*

- lenguajes de programación** tesis de maestro en cs. de la computación, Instituto Tecnológico y de Estudios superiores de Monterrey, Ciudad de México, 1997.
9. Johnson, WL; Rickel, JW ***Animated Pedagogical Agents: Face to Face Interaction in Interactive Learning Environments***, International Journal of Artificial Intelligence in Education, 2000.
  10. Lester, James C.; Converse, Sharolyn A.; Kahler, Susan E.; Barlow, S. Todd; Stone, Brian A.; Bhoga, Ravinder S. ***The Persona Effect: Affective Impact of Animated Pedagogical Agents*** CHI 97 Electronic Publications: Papers en <http://www.acm.org/sigchi/chi97/proceedings/paper/jl.htm> accesado en abril de 2004.
  11. Pfeifer, Rolf & Sheier, Christian ***Understanding Intelligence***, 1999 MIT Press
  12. Polson, Martha C. & Richardson, J. Jeffrey ***Foundations of intelligent tutoring systems*** Hillsdale: Lawrence Erlbaum Associates, 1988.

©CiberEduca.com 2005

La reproducción total o parcial de este documento está prohibida sin el consentimiento expreso de/los autor/autores.  
CiberEduca.com tiene el derecho de publicar en CD-ROM y en la WEB de CiberEduca el contenido de esta ponencia.

**® CiberEduca.com es una marca registrada.**

**©™ CiberEduca.com es un nombre comercial registrado**