

Nuevo Software para Nuevos Medios

Ambientes de Software Interactivos para Aprender (ASIA)

Por
Dr. Jaime Sánchez Ilabaca
Departamento de Ciencias de la Computación
Universidad de Chile

* Trabajo presentado en la Primera Jornada Regional de Informática Educativa y Aprendizaje Virtual,
San Juan, Costa Rica, Septiembre 29 y 30, Octubre 1, 1999

Introducción

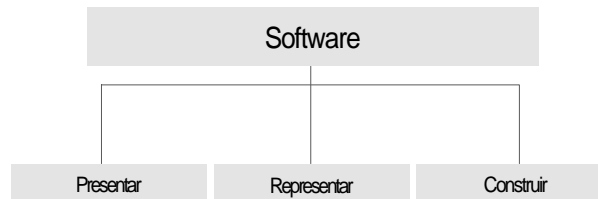
El desarrollo de ambientes de software interactivos para apoyar metodologías activas de aprendizaje ha sido descrito profusamente en la literatura. Es así como desde hace algún tiempo algunos autores piensan que tecnologías como *software*, *multimedios* y, recientemente, *Internet*, pueden constituirse en «buenas herramientas» para diseñar ambientes de software que faciliten y estimulen la construcción de aprendizajes en los aprendices.

Diversas experiencias en el mundo revelan que existe consenso que utilizar tecnologías más pertinentes al mundo en que vive el aprendiz, motiva su aprender y los procesos de construcción involucrados. Bajo esta perspectiva, desde hace unos años hemos visto un desarrollo creciente de una serie de productos computacionales destinados a estimular y apoyar el desarrollo de un aprendizaje activo y constructivo.

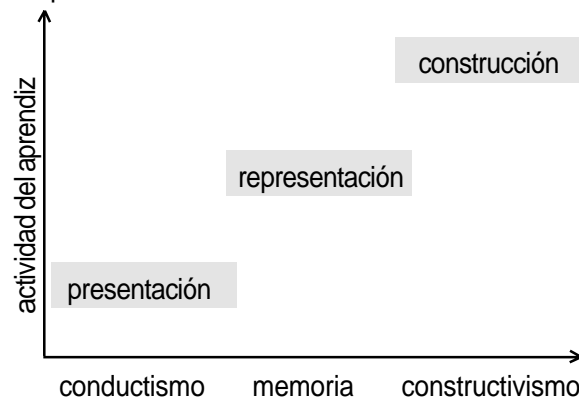
Del software centrado en el computador al software centrado en el aprendiz

Esta evolución tecnológica ha llevado a que surjan nuevas formas de clasificación del uso del software educativo más allá de las tradicionales formas tales como como *tutor*, *herramienta* y *aprendiz* (Taylor, 1980) y aprendizaje *acerca* del computador, aprendizaje *con* el computador, *acerca del pensamiento* con el computador, aprendizaje *a través* del computador, *administración* del aprendizaje con el computador (Sánchez, 1993).

Es así como en el último tiempo surge una nueva forma de clasificar el software basado en el rol del aprendiz, el rol de la tecnología y las demandas conductuales y cognitivas del software. De esta forma, clasificamos el software como software para *presentar información y conocimiento*, software para *representar información y conocimiento* y software para *construir información y conocimiento*.



Es así como, luego de observar un tipo de software educativo que utilizaban (y aún utilizan) conceptos e ideas conductistas del aprender como CAI, CAL, CBI y similares, surge un tipo de software que apunta al uso profuso de múltiples medios integrados, que otorgan cierto grado de interactividad con el usuario. Este software, aún de uso masivo, no ha crecido más allá del modelo *presentación de información y conocimiento*. El mayor cambio es en la tecnología medial (uso variado uso de medios como texto, sonido, imágenes), pero no en la calidad educativa del software. Si bien el formato de presentación comienza a ser variado, impactante y algo motivador, el modelo que subyace en este software sigue siendo el mismo. El eje de este software está en la adquisición, comparación/relación conceptual y recuperación de información y conocimiento. Así, podríamos decir que se ha cambiado el envase (nueva tecnología) pero el modo de lo que el usuario hace con el contenido sigue siendo el mismo. Es como el dicho, *el mismo vino con distinto envase*. Más aún, este tipo de software es usualmente distribuido en CDs de uso personal y aislado, haciendo la experiencia de utilización una actividad personal, sin mayor interacción social con otros aprendices.



Quizás el modelo más aportativo en esta línea sea algún software educativo multimedial que en su forma, metodología de secuencia y presentación de contenido, utiliza modelos clásicos de la literatura sobre representación y organización de información en memoria. La estructura del software, su navegación y la interacción con el usuario, intentan imitar la forma como se almacenaría información en memoria, según algunas teorías psicológicas. Este *software para representar información* cubre un amplio espectro del software actualmente disponible para uso educativo.

En los años recientes surge el denominado *software con el cual construir*, es decir software rico en elementos, materiales y herramientas que favorezcan y estimulen la construcción de conocimientos. Este software es posible gracias a una mayor disponibilidad de tecnología de software y hardware de mayor perfil y complejidad tecnológica, que permite una mayor flexibilidad. Este software presenta una mayor madurez en el tratamiento del contenido, la interactividad y su relación con el aprendizaje. El software responde a una demanda por software educativo que cumpla con requerimientos más activos y flexibles, donde el aprendiz pueda *hacer cosas con el software* y no que *el software haga cosas con él*. En esta categoría surge software tipo historietas, historias, comics, juegos de aprendizajes y ambientes interactivos como los que aquí se proponen. En general, este tipo de software se caracteriza por reutilizar conceptos e ideas de los juegos computacionales, pero con el valor agregado de la puesta de contenidos de aprendizaje explícitos y con la intencionalidad de desarrollar o estimular el uso de algún proceso cognitivo y la transferencia del aprender. Con ello, el aprendiz aprende, juega, se entretiene, resuelve complejidades, controla variables, se enfrenta a situaciones inciertas, etc., involucrándose en un ambiente desafiante, entretenido y motivador. Esto es, se aprende jugando, *se construye el aprender jugando*.

En Iberoamérica, el diseño, desarrollo y evaluación metodológica de este último tipo de productos es aún minimal. Si bien se ha observado un mayor interés por desarrollar software de calidad, el objetivo de muchas empresas de software sigue la línea de la apariencia, impacto y entretenimiento, a través del uso excesivo del tratamiento de medios, desvirtuándose muchas veces el objetivo de aprendizaje de estos productos y omitiéndose diseños centrados en el usuario, desarrollos considerando desde el inicio al usuario final y espacios para trabajo colaborativo multiusuario. La interacción social se realiza a través de muchas facetas (conversación, expresión artística, etc.) y una de ellas constituye compartir objetos. Es precisamente esta modalidad que aún está ausente en el esquema del aprender y conocer en el contexto de ambientes computacionalmente mediados. Inclusive la presentación de naturaleza inmersiva de estos ambientes es un área incipiente, que puede ser claramente explotada a través de mundos virtuales representados con Virtual Reality Modeling Language (VRML).

No sólo hay escasez de productos, sino que existe una seria necesidad por nuevos modelos, nuevos esquemas de interacción social tecnológicamente mediados, nuevas propuestas de interfaces para estimular el aprender, nuevos ambientes que estimulen activamente la construcción de conocimiento y del desarrollo y uso de destrezas de pensamiento de mediano y alto orden, como pensamiento lógico-aritmético, resolución de problemas, adaptación, juicio crítico, etc. Se requieren nuevas miradas educativas en la construcción de software y metodologías de uso de software. Miradas que se traducen en mayor interactividad, mayor interés por el aprendiz usuario, mayor interés por una metodología de uso del software.

Es en virtud a lo antes expuesto, que surge la idea de generar modelos armónicos de ambientes de software interactivo para aprender constructivamente, ASIA. Las características fundamentales de estos software son:

- Utilizar y reutilizar conceptos e ideas de ambientes interactivos tales como hiperhistorias y juegos de aprendizaje
- Proveer herramientas y materiales para apoyar trabajos que involucren colaboración local y remota, con la finalidad de estimular la construcción del aprender en aprendices
- Considerar en el diseño la interacción aprendiz/computador

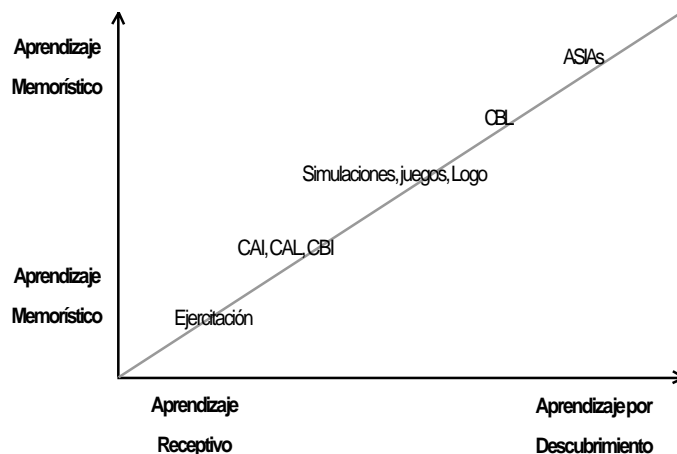
- Incluir interfaces simples, amistosas y diseñadas para el aprendiz
- Soportar la implementación de software abierto, interactivo, robusto, flexible, fácil de usar y controlado por el aprendiz
- Tener interfaces estéticamente placenteras
- Incluir modalidades y metodologías activas de uso y trabajo del software interactivo en el aula
- Dar opción a un uso individual y colaborativo
- Funcionar utilizando herramientas estándar de Internet, de tal modo que se pueda vincular aprendices que están físicamente distantes y que converjan en problemas, soluciones e inquietudes de un dominio dado

Más aún, el tipo tradicional de software refleja una falta de representación de la idea de estado. Todos los sistemas de la vida real lo poseen y evolucionan además de una manera dinámica. Por esto, los ASIA expresan a través de aplicaciones, sistemas que pueda ser navegados e interactuados a lo largo del tiempo, y no meramente explorados en sesiones aisladas.

Con los ASIA intentamos crecer a partir del nivel de software de presentación y representación de información y reutilizar los modelos y experiencias de la primera aproximación en el nivel construcción de conocimiento (software de hiperhistorias). La idea es evolucionar y dar un paso más adelante hacia propuestas de software más abiertas que incluyan características de juegos, historias, editores, que considere un testeo de su efectividad y una experimentación con metodologías de uso con aprendices y facilitadores.

La diferencia más fundamental entre ambientes interactivos y ambientes tradicionales es la inserción explícita de Internet en la conceptualización del software. Asimismo, los ASIA consideran la ejecución multiusuario en lugar de la opción monousuario de las hiperhistorias. Los ASIA permiten establecer y estimular mecanismos de colaboración e interacción entre usuarios distantes.

Así también, consideramos que los ASIA tienen un importante efecto en la construcción de los aprendizajes cuando son interactivos, flexibles, mediados, controlados por el aprendiz, así como cuando tienen características lúdicas y de alta interacción con el aprendiz, cuando proporcionan las herramientas y materiales para soportar metodologías de trabajo colaborativo y permiten construir nuevo conocimiento.



Atributos de los ASIA

Los siguientes son algunos de los atributos constitutivos: constructividad, navegabilidad, interactividad, contenido, interfaz.

Constructividad

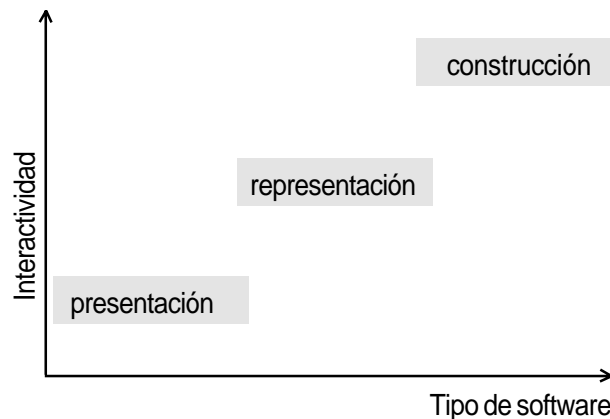
Se refiere a la posibilidad de construir nuevos escenarios a partir de la combinación de entidades y objetos en el espacio y tiempo. Cuando el aprendiz interactúa con el software, el construye y reconstruye, diseña y rediseña, estructura y reestructura, esto es, está en actividad constructiva. La acción en los ASIA ocurre en tanto el aprendiz haga cosas, tome decisiones, diseñe, es decir, no hay acción si el aprendiz está pasivo, la acción es el aprendiz.

Navegabilidad

Consiste en la posibilidad de explorar libremente los ambientes, realizar asociaciones, no siguiendo una secuencia lineal, sino que moviéndose libremente de un punto otro siguiendo el orden que determine su mente, su aprender.

Interactividad

Se refiere a la capacidad dinámica del ambiente, a la acción y reacción, a la comunicación de idea y vuelta entre el ambiente y el aprendiz. Esto implica proveer una retroalimentación al aprendiz en tiempo real, adaptar y modificar dinámicamente el comportamiento en función de los eventos recibidos y establecer algún tipo de conversación. La interactividad implica también un cierto grado de manipulación y control del aprendiz sobre las variables del ambiente, como por el tipo, nivel y calidad de respuesta percibida por el aprendiz. La interactividad implica que la actividad se gatilla en el receptor y recíprocamente se estimula en el receptor, fluyendo como una actividad dinámica.



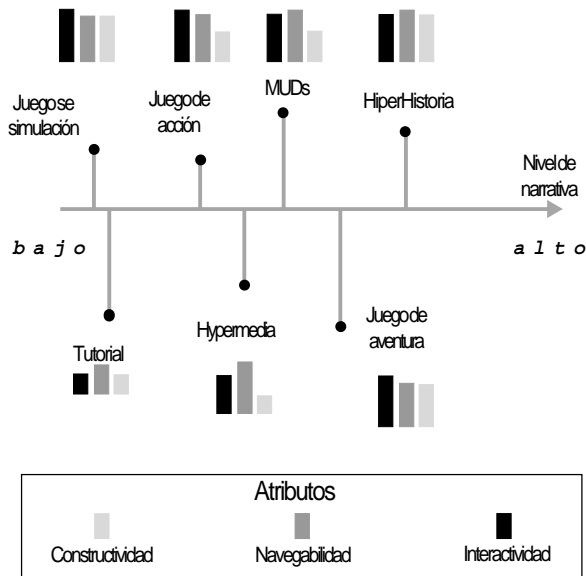
Contenido

Es el grado de calidad, fiabilidad, organización y relevancia de la información entregada. Es ortogonal a la presentación, debe ser adaptado y organizado en función del tipo y nivel del aprendiz.

Interfaz

Corresponde a la superficie de contacto entre el aprendiz y el computador. Es la forma como se captura la atención y acción del aprendiz y refleja el estado y contenido del ambiente. La interfaz tiene un fuerte impacto en la navegación, construcción e interactividad provista. Los ambientes interactivos proveen de interfaces diseñadas de forma tal que sean transparentes e invisibles para el aprendiz, dejando que la visibilidad recaiga en la tarea de aprendizaje, en la construcción del aprender.

Finalmente, los atributos de **constructividad**, **navegabilidad** e **interactividad** pueden ser identificados en los distintos tipos de software mono y multiusuarios, utilizados en el aprender, tales como juegos, tutoriales, hipermedios, MUDs, hiperhistorias y ASIAs. Así, podemos destacar a los juegos de aventura, hiperhistorias y ASIAs por presentar un significativo grado de constructividad, navegabilidad e interactividad.



Elementos constructivos de los ASIA

La literatura describe que además de los atributos de los ASIA es importante resaltar aquellos elementos constitutivos que distinguen a un software con características constructivas tales como el grado de desafío que genera en el aprendiz y la motivación que produce en los aprendices (Druin, 1999, Solomon & Druin, 1998). Algunos autores señalan que los ambientes de software deberían ser intrínsecamente motivadores como ocurre con algunos juegos (Malone, 1987). Asimismo, el componente lúdico es otro elemento distintivo de esta nueva generación de software, el juego y la entretención como apoyo al construir y aprender parecen ser condimentos fundamentales en software educativos del nivel construcción.

En la misma línea, la posibilidad de adaptación a los distintos requerimientos cognitivos del aprendiz, cautivarlo y envolverlo en la tarea, estimulando el desarrollo y uso de destrezas de descubrimiento, todos ellos son también elementos constructivos de los ASIA. El control de las acciones y de las variables principales del software tiene que estar del lado del aprendiz, quién puede tomar decisiones libremente sobre el curso de acción del software.

La posibilidad de proveer herramientas y materiales que permita a los aprendices diseñar situaciones o acciones, crear sus propios escenarios y construir experiencias sobre la base de sus acciones, son también elementos característicos de estos ambientes.

Finalmente, las estrategias y alternativas para que el aprendiz pueda desarrollar su imaginación, capacidad de maravillarse y construir fantasías basándose en su experiencia con estos ambientes, son también características de estos ambientes.

Tejido teórico de los ASIA

Los ASIA se enmarcan dentro de área de construcción del conocimiento a partir de software educativo interactivo. Para tal efecto, analizaremos el tema de los juegos computacionales educativos, y para ello conjugaremos diferentes tópicos provenientes de Internet, Colaboración, Realidad Virtual (RV) y Resolución de Problemas. La figura siguiente organiza jerárquicamente estos tópicos y permite establecer un marco argumentativo de la discusión bibliográfica de los ASIA.

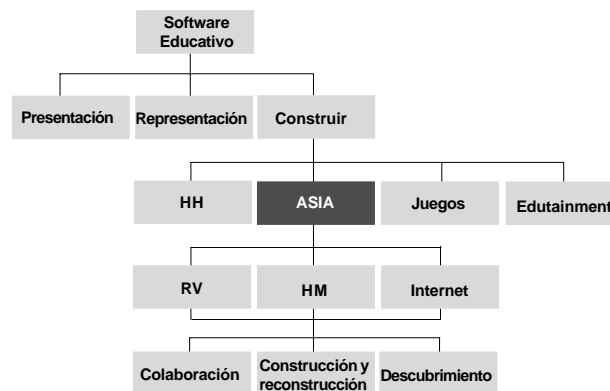
Evolución del software educativo

Diversos autores han propuesto recientemente diferentes tipos de software educativo, de acuerdo a la forma cómo los usuarios interactúan con el software, a la relación del software con el conocimiento y a las posibilidades que hoy nos otorga la tecnología para generar materiales para aprender. Así, el software educativo ha sido clasificado en tres amplias categorías: presentación de conocimiento, representación de conocimiento y construcción de conocimiento (Nelson 1994, Sánchez & Rossi 1994).

El uso de software para proveer acceso a cierta información al aprendiz corresponde a la categoría presentación de conocimiento. Esa información «idealmente» es incorporada como conocimiento por el aprendiz. Existen varias formas y niveles de uso de esta modalidad, generalmente determinadas por los objetivos educacionales, el software y el nivel de control requerido por el aprendiz. Independientemente de si el aprendiz tiene menos control en una experiencia (controlada por el profesor o por el programa), o bien que sea el aprendiz el que decida activamente su ruta de navegación a través de la información, escogiendo autónomamente la información para explorar, esta modalidad implica presentar información al aprendiz. Aún cuando este modelo es útil para ciertos propósitos, acción, control, ritmo e interactividad son dados por el software, demandando un usuario más bien pasivo. Usar este software ha sido descrito como «clickear flechas hacia adelante y atrás». No se incluye mayor valor agregado fuera de presentar información más dinámica y atractiva.

Una variación de este modelo es el software para representación de la información. La información puede ser representada mediante una comparación metafórica de la relación estructural entre conceptos del programa y posibles estructuras mentales formadas por el aprendiz. Aquí la idea es imitar la mente humana, presentando información que imita un modelo de la memoria humana. Esto podría ser beneficioso, ya que una premisa recurrente en el aprender es que mientras más se asemeje un cuerpo externo de información estructurada a las estructuras de conocimiento en memoria permanente, mayor sería la probabilidad que aprendices procesen esa información y la retengan. Algunos ejemplos de este modelo son los software que usan mapas/web conceptuales que permiten que la información sea relacionada e interconectada de forma no lineal (Novak 1984) o redes semánticas con información organizada en estructuras jerárquicas en forma top-down (Quillian 1968), generalmente usadas para fines de orientación, estructura, presentación de información, evaluación, etc.

Para muchos autores, las primeras dos categorías tienden a limitar al usuario, en términos de interacción, control, flexibilidad, no-determinismo, y acceso limitado a una metáfora del mundo real.



Diferentes conceptos y tecnologías se conjugan para dar lugar a la conceptualización e implementación de los ASIA. En particular los ASIA surgen de alguna manera como una extensión de las HiperHistorias (HH) y permiten incluir nuevas tecnologías y conceptos al software de construcción del conocimiento.

Un modelo más flexible y de mayor frontera en investigación sobre software educativo es el tipo de software para *construcción de conocimiento*. Aquí, el software provee los materiales y herramientas para que los aprendices hagan cosas, construyan, resuelvan, crean, corrijan, construyan a partir de errores, etc. (Kenneth et al. 1994). Es decir, se enmarcan en una propuesta constructivista (Jonassen 1996). Esto tiene el beneficio de permitir a los aprendices crear sus propias relaciones entre conceptos en la medida que ellos decidan cómo la información es conectada. Así, en la medida que el aprendiz decida qué información incluir y cuál eliminar, se involucra en generalizaciones y discriminaciones que luego desarrollan su propio entendimiento de los conceptos involucrados. Son ejemplos de este modelo, la construcción de software por parte de los aprendices, las hiperhistorias y los juegos educativos computacionales, que incorporan importantes estrategias cognitivas, motivando e involucrando completamente a los aprendices a través de otorgar control sobre la tarea de aprendizaje, desafío, interacción, adaptación al nivel y requerimientos del jugador, con un espectro de principiante a avanzado (Resnick 1993; Resnick, Martin & Bruckman 1996; Kafai 1995, 1996; Schank 1993; Alessi 1996; Alessi & Trollip 1991; Druin 1996; Druin & Solomon 1997).

La categoría menos explorada y con un desarrollo aún minimal es *construcción de conocimiento*, menos aún a través de software interactivo como juegos de aventuras, juegos de roles e hiperhistorias. Es así como, aún cuando juegos de acción, aventuras y roles tales como Doom, Myst, MUDs y MOOs consideran un escenario interactivo con los usuarios, sólo consideran modelos de diseño poco claros para aquellos que deseen recrear los mismos ambientes.

Las hiperhistorias como ambientes narrativos interactivos

Las hiperhistorias surgen de la necesidad de poner en acción historias en un contexto hipermedial interactivo.

Las historias son narrativas de eventos verdaderos o ficticios que intentan capturar e involucrar activamente a los aprendices. Las hiperhistorias son historias no secuenciales que ocurren en un ambiente virtual hipermedial (Cernuzzi et al. 1995; Sánchez et al., 1994, 1996; Sánchez & Rossi 1994; Sánchez & Cernuzzi 1996a, 1996b, 1997; Sánchez & Lumbreras 1995, 1996a, 1996b, 1996c, 1997). Las hiperhistorias son versiones electrónicas de las historias literarias.

Para comprender el concepto de Hiperhistoria primero debemos conceptualizar lo que es un Ambiente Virtual Hipermedial (AVH), este es la combinación de contextos hipermediales navegables, objetos dinámicos sobre los cuales el aprendiz puede realizar operaciones y un conjunto de caracteres que puede manipular. Los objetos y caracteres pueden tener su propio comportamiento y actuar en forma autónoma, eventualmente en forma concurrente con el comportamiento del protagonista. El protagonista puede interactuar con otros caracteres, a fin de resolver un problema o lograr una meta. Pero ello no es todo, las historias tienen un contenido, un argumento, los personajes tienen roles y esos roles generan sucesos. Es decir, una hiperhistoria tiene que tener una narrativa. Luego, una hiperhistoria es la sumatoria de AVH y narrativa, esto es, el valor agregado está en que existe una secuencia intencional de eventos basados en roles y caracteres (Lumbreras & Sánchez 1997).

El hiper de las hiperhistorias se refiere a un espacio bidimensional: el medio ambiente estático donde ocurre la historia y el espacio invisible de la historia donde la hipernarrativa es construida. Una hiperhistoria incluye un carácter manipulado libremente por el usuario que representa el enlace entre carácter y el sistema. Otros caracteres preprogramados interactúan con el *avatar* del usuario en el ambiente físico representado, extendiendo las ideas presentadas por Bernstein (1996).

Durante el desarrollo de una hiperhistoria el aprendiz puede moverse en ambientes estimulantes e interactuar flexiblemente con personajes y objetos.

La idea es explorar, navegar y construir estructuras mentales a través de una serie de componentes estructurales y funcionales de la metáfora, los que le proporcionan las herramientas y un medio rico en materiales con los cuales construir. Así, las Hiperhistorias fueron construidas para involucrar, desafiar, interactuar y motivar a los aprendices, debido a que se le otorga al aprendiz control sobre las historias, herramientas y materiales de construcción para construir cosas y desarrollar estrategias para testar hipótesis con la idea implícita de estimular el desarrollo y uso de relación espacio-temporales y lateralidad (Sánchez & Lumbreras 1996a, 1996b).

Así, interactuando con las hiperhistorias se pretende explotar las nociones de espacio, posición, secuencia y extensión en tiempo, así como el desarrollo de estructuras cognitivas de alto nivel. También se han desarrollado hiperhistorias para aprendices con discapacidades: ciegos (Lumbreras 1993, 1997; Lumbreras, Sánchez & Barcia 1996), mentales (Cernuzzi & Sánchez, 1996) y sordos (Campos, Valdeni & Sánchez 1995).

A los aprendices les encantan las historias y las recuerdan fácilmente, tienden a explotar su imaginación. Cuando se involucran en una historia, pueden identificar, recuperar y usar datos relevantes para resolver un desafío teniendo acceso rápido y flexible a la secuencia de la historia. Por ello, se estima que las hiperhistorias motivan a los aprendices, facilitan la navegación, y promueven aprendizaje de largo plazo, proveyendo un rico contexto de información y haciéndolo más recordable.

Una forma de interactividad que es muy similar a las hiperhistorias se refiere a los juegos tipo *branching*, primero popularizados en libros «elige tu propia historia». En este tipo de juegos, el aprendiz experimenta segmentos cortos y lineales de la historia. Al final de cada segmento hay un pequeño número de opciones (dos a cuatro), cada una lleva a un nuevo segmento lineal, lo cual conduce a otras opciones y así sucesivamente. Algunas veces los caminos convergen, otras divergen a fines diferentes. Joiner (1994) señala que la ventaja de un juego tipo *branching* es que permite escribir un pequeño número de vías alternativas de las historias. La desventaja es que se ve disminuida la interactividad. El jugador puede escoger caminos que han sido prediseñados por el diseñador.

Al respecto, para satisfacer las complejas demandas del concepto de hiperhistoria y enmarcándose en el concepto de software basado en un modelo (Davenport, 1996), se ha desarrollado un modelo de especificación y desarrollo de hiperhistorias (Sánchez & Lumbreras 1996b, Lumbreras & Sánchez 1997). El modelo extiende la idea de juegos tipo *branching* combinando objetos dinámicos y estáticos en contextos anidados para permitir navegación flexible en un mundo virtual. Las hiperhistorias crecen a partir de este concepto introduciendo la idea de oportunidad. Las historias de la vida real tienen una amplia variedad de oportunidades. Estas posibilidades ocurren a través del tiempo. Los caracteres pueden decidir si tomar o no una oportunidad. Este concepto fue previamente explorado por Sawney (1996), presentando la poderosa idea de enlace temporal disponible sólo en alguna ventana de tiempo o lapso. Las hiperhistorias van más allá aún, incluyendo en la conducta de las entidades algunas reglas que pueden ser gatilladas por un período dado de tiempo, sin saber el usuario a priori cuando comenzar esta ventana. Por ejemplo, un mapa del tesoro volando libremente en varios ambientes puede ser encontrado solamente si es percibido por el carácter. Si el carácter rechaza explorar el mapa, la oportunidad puede ser perdida para siempre. Como resultado, aún cuando el vuelo del mapa fue predefinido el encuentro mapa-carácter es un escenario no determinista. Esto otorga un comportamiento no predecible de la historia presentando una extensión poderosa del concepto de oportunidad.

Los juegos computacionales como ambientes interactivos

La literatura identifica distintos tipos de juegos computacionales (Withers 1997). En general, los juegos pueden clasificarse en las siguientes categorías amplias (un juego puede ubicarse en dos o tres categorías diferentes): acción, estrategia, guerra, simulaciones, deportes, aventuras, roles, tableros y puzzles. Cada tipo de juego plantea una organización temporal y navegacional propia, la cual impone ventajas y restricciones (Lumbreras & Sánchez 1997).

En este contexto se ha observado que los juegos de naturaleza lineal ofrecen menos alternativas y posibilidades de creación que los juegos con estructura de *hub* o concentrador, o que los juegos con una estructura arborescente. Es por ello que cada tipo de juego se perfila mejor para diferentes tipos de aplicaciones posibles: estructura lineal para aplicación educativa tipo tutorial, estructura de *hub* para aplicación guiada hacia selección de tópicos a modo de menú, estructura de camino ancho para el incentivo de soluciones en donde se precisen muchos componentes simultáneos para progresar, estructuras arborescentes para escenarios de aprendizaje guiados por la incertidumbre y exploración (Hayes-Roth 1995, Hayer-Roth, Gent & Huber 1996, Maes 1995. Este tipo de aplicaciones están siendo gradualmente portadas a ambientes de tipo inmersivos (Appino, Lewis & Koved 1992; Kelso, Weyhrauch & Bates 1993), tal como los que propone la plataforma VRML, la cual está diseñada para ser utilizada vía Internet (Hartman, Wernecke & Carey 1996; Lea, Matsuda & Miyashita 1996).

Las principales características educativas que poseen los juegos descritas por la literatura (Alessi & Trollip 1991; Cavazos 1996; Druin 1996; Ellis 1994; Kafai, 1995, 1996) son: metas (claramente establecidas o inferidas), reglas (qué acciones se permiten y qué restricciones existen), competición (en contra de un oponente, en contra de sí mismo, en contra del tiempo, en contra de la suerte), desafío (lo que el jugador tiene que realizar para lograr la meta), fantasía (fantasía para motivar, que van desde una representación cercana a la realidad, a más lejana, hasta llegar a una totalmente imaginaria), seguridad (forma segura de actuar en realidades peligrosas) y entretenimiento (para estimular motivación y aprendizaje). Todas estas características pueden tener una directa implicancia en el desarrollo y uso de diversos procesos cognitivos.

Diversos autores (Alessi & Trollip 1991; Man 1994; Klawe & Phillips 1995) han señalado tres tipos de factores a considerar en el diseño de un juego educativo, tales como: factores en la introducción de un juego, en el cuerpo del juego y factores en el término de un juego. Los factores en la introducción de un juego son metas, reglas, jugadores, equipamiento, procedimientos, restricciones, penalizaciones, instrucciones de uso y opciones. Los factores en el cuerpo del juego incluyen escenario, nivel de realidad, reparto, rol de los jugadores, presencia de incertidumbre y curiosidad, naturaleza de la competición, relación del aprendizaje y los objetivos, destrezas versus azar, opciones, flujo de información, movimientos, tipos de acciones y modos de interacción. Los factores en el término de un juego consideran reconocer al ganador, el premio, proveer información sobre el progreso y desempeño y mensaje final.

La literatura describe una amplia diversidad de estudios que destacan la importancia del uso de juegos para fines de aprendizaje, tales como el desarrollo de destrezas de competencia y concentración, motoras, verbales, matemáticas, visuales y de resolución de problemas (Cavazos 1996; Turkle 1996; Ellis 1990), desarrollo de destrezas de lecto-escritura (Rosas et al. 1996). Así también, se ha explorado el uso de software interactivo para estimular el desarrollo de destrezas de toma de decisiones basadas en conocimiento aprendido, aprendizaje intuitivo de probabilidad matemática, experiencia vicaria activa, estrategias y prudencia, coordinación ojo-mano, ritmo de trabajo (Zamora 1984; Sherwood 1991). De igual forma, Kafai & Soloway (1994) proponen el uso de software interactivo para estimular una rápida coordinación de acciones complejas, feedback inmediato, metas claras, despliegue visual de metas o resultados y niveles de desafío.

Otros autores comunican experiencias con el uso de juegos y software educativo interactivo para estimular el desarrollo y uso de destrezas de creatividad (Kahn 1996) y destrezas de integración curricular, interacción social y autocuestionamiento (Sherwood 1991). Klawe & Phillips (1995) informan del uso de software de tipo lúdico para desarrollar destrezas de investigación como obtener, manipular, organizar e interpretar información. Man (1994) sugiere utilizar ambientes interactivos para destrezas de mapeo y registro de información. Druin (1996, 1999) informa del uso de juegos para estimular habilidades y destrezas para tomar de notas y realizar listas y resúmenes de resultados. Todos estos estudios resaltan la importancia de combinar los aspectos de tipo lúdico con el desarrollo de alguna destreza o habilidad a través del trabajo con software educativo.

Realidad virtual y ambientes interactivos

El uso de Realidad Virtual (RV) en el área de educación es de novísima introducción. Sin lugar a dudas el objetivo de las aplicaciones de RV, ya sea por medio de un sistema aislado o por uno conectado a Internet, es proveer una experiencia de naturaleza inmersiva en donde por medios de estímulos acústicos, visuales y táctiles se intenta transferir una experiencia sintética de conceptos físicos y abstractos de un ambiente dado (Sánchez, Lumbreras & Silva 1997). Los resultados preliminares pueden ser controversiales acerca del uso de esta tecnología en educación, pero (Carpenter 1996) señala en sus pruebas preliminares una mejora en el rendimiento del aprendizaje; (Mikroupoulos & Nikolou 1996) indica que las aplicaciones de VR facilitan la usabilidad y mejoran la naturaleza de ambientes altamente interactivos. Otro de los valores agregados de VR es la posibilidad de revivir experiencias de aprendizaje muy difíciles de revivir en otros ambientes. Por ejemplo (Littman 1996) describe cómo sus alumnos pudieron recorrer en primera persona edificaciones egipcias, tal cual cómo si ellos estuvieran allí en la época del esplendor de los faraones. A pesar que las evidencias muestran a este tipo de tecnología como una herramienta sumamente interesante para transferir y construir conocimiento, diversos autores sostienen que puede ocurrir algún grado de socialización o distracción de la tarea principal para la que la aplicación fue formulada (Winslow 96, Grove, Williams & Hartley 1996). Sin embargo, la inclusión de estas tecnologías en un ambiente de colaboración asociado al aprendizaje basado en la metáfora de juegos con implementación basada en VRML, es un área incipiente que merece estudio y extensión.

WWW y ambientes interactivos

Asimismo, diversos estudios han comenzado a implementar en World Wide Web, software educativo de tipo lúdico, generalmente para uso colaborativo. Inicialmente las propuestas estaban muy orientadas a un formato más bien rígido de una página Web, luego applets ilustrativos de juegos en red, para en el último tiempo generarse software para el desarrollo de juegos por los niños, como es el caso de una versión Java para niños, *cocoa* (Resnick 1996) y software interactivo en Web (Soloway 1995, Kearsley 1996, Trentin 1996, Linn, Bell & Hsi 1998).

Un aspecto interesante de destacar en estas aplicaciones es que, a pesar que su idea es estimular la colaboración, en su acción el grado de colaboración logrado es bastante limitado.

Resolución de problemas y ambientes interactivos

La resolución de problemas puede ser vista como un continuo donde en un extremo existen actividades computacionales simples y, en el otro, tareas que involucran análisis, síntesis y evaluación. Generalmente, la resolución de problemas es vista como destreza de orden más alta (Barba & Rubba 1992). Un grupo importante de ambientes interactivos lúdicos incorpora el uso de estas destrezas para su logro o éxito final. Asimismo, de la misma forma como existe un amplio rango de problemas en el marco de la investigación en resolución de problemas, existe un continuo de estrategias para resolver problemas.

Sternberg (1996) describe un modelo de siete etapas en el ciclo de la resolución de un problema: identificarlo, definirlo, construir una estrategia, organizar información, asignar recursos, monitorear la resolución de problemas y evaluarlo. Mayer (1992) alude a secuencias tradicionales en la resolución de problemas dadas por Wallas (preparación, incubación, iluminación y verificación) y Polya (entender el problema, diseñar un plan, desarrollarlo y revisarlo).

Estas metodologías son reutilizadas en el diseño de los Ambientes Interactivos, de tal manera que para el aprendiz sean transparentes, centrando su acción en el desafío y descubrimiento.

Un aspecto final de destacar es que, si bien los modelos de Wallas, Polya y Sternberg parecen simples y fáciles de utilizar, en el diseño de software, cuando son implementados en el contexto de ambientes interactivos de software, la resolución de problemas se torna en un proceso mucho más complejo y menos generalizable que lo que reflejan estos modelos.