

“INGENIERÍA DE SOFTWARE EDUCATIVO, TEORÍAS Y METODOLOGÍAS QUE LA SUSTENTAN”

Prof. Pedro Salcedo Lagos

INTRODUCCIÓN

El aprendizaje es una actividad consustancial al ser humano. Se aprende a lo largo de toda la vida, aunque no siempre en forma sistemática; a veces es fruto de las circunstancias del momento, otras, de actividades planeadas por alguien (la persona o un agente externo) y que el aprendiz lleva a cabo en aras de dominar aquello que le interesa aprender.

Sin embargo, según Alvaro Galvis (Galvis, 94), aprender por uno mismo o ayudar a otros a que aprendan no es algo innato, ni se adquiere por el simple hecho de asistir durante una buena parte de la vida a ambientes de enseñanza - aprendizaje. Hace falta entender y aplicar teorías de aprendizaje humano que den sustento al diseño de ambientes de aprendizaje efectivos.

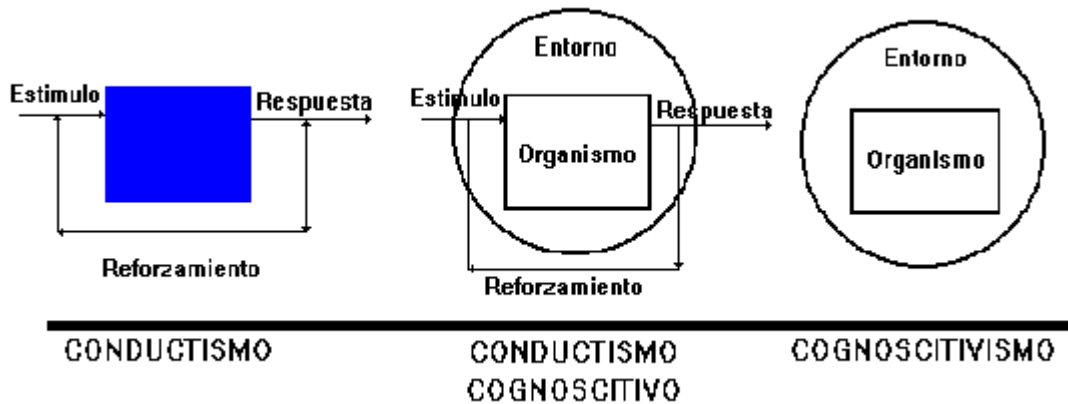
¿Por qué estudiar teorías psicológicas del aprendizaje humano, como uno de los fundamentos para un proceso de selección o desarrollo de un sistema computacional?. La respuesta es obvia: quienes intentan desarrollar ambientes de enseñanza-aprendizaje basados en el computador sin tener un buen sustento teórico respecto al aprendizaje humano y a las características del computador como medio de enseñanza, *pueden entrar a replicar, indiscriminadamente, las estrategias de enseñanza-aprendizaje que se conocen y además pueden desaprovechar algunas características útiles del computador.*

PRIMERA PARTE

TEORÍAS DEL APRENDIZAJE

Todas las aproximaciones psicológicas al fenómeno del aprendizaje humano tienen algo que decir como fundamento para el diseño de ambientes de enseñanza - aprendizaje. Sin embargo, los aportes no necesariamente son convergentes, como no lo es la perspectiva desde la cual se analiza el fenómeno en cada caso, ni los métodos usados para obtener el conocimiento. Si hubiera una teoría que atendiera todos los aspectos del fenómeno, que abarcara las demás teorías, no habría que estudiar las otras; pero la realidad es diferente, de ahí la necesidad de por lo menos conocer los puntos más importantes de los diferentes aportes en relación al tema.

Las aproximaciones al fenómeno del aprendizaje oscilan entre dos polos: conductismo y cognoscitivismo.



Focos de atención en las diferentes teorías del aprendizaje

En el primer polo no se toma en cuenta el organismo (el sujeto que aprende), sólo las condiciones externas que favorecen su aprendizaje; por esto se habla de un modelo de "caja negra" en el que lo fundamental es la programación en pequeños pasos, de eventos que conduzcan a lograr el resultado esperado (respuesta) y el reforzamiento de las respuestas, que confluyen hacia el logro de lo que se desea.

En el otro polo lo que cuenta es el individuo, el aprendiz, con todo su campo vital, su estructura cognoscitiva, las expectativas que tiene. Por contraposición se habla de un modelo de "caja traslúcida" en el que lo que cuenta es el aprendiz dentro de su entorno psicológico y social. La motivación interna, la significancia, el procesamiento de la información, las aptitudes de las personas, entre otros, son tomados en cuenta como factores que promueven el aprendizaje.

En medio de los enfoques anteriores, es posible encontrar una combinación de ambos. En el cual se comparten algunas características. Ya no se habla de caja negra como en el primer enfoque, pero tampoco se considera al aprendiz, como único elemento de relevancia (enfoque de "caja traslúcida").

A pesar de las anteriores diferencias, las teorías de aprendizaje tienen en común su objeto de estudio: el aprendizaje. No es de extrañar, por consiguiente, que se logre un efecto de "triangulación" (ver de varios ángulos un mismo asunto) cuando se analizan los distintos aportes. Desde cada teoría existe una perspectiva que complementa a otras.

El evolucionismo genético de Piaget, propone que el aprendizaje está subordinado al desarrollo orgánico y de estructuras cognoscitivas (a la madurez del estudiante), además se basa en la experiencia, actividad inquisitiva sobre el objeto de conocimiento. Por otra parte para Piaget el aprendizaje resulta de alcanzar nuevos estados de equilibrio a partir de desequilibrios cognitivos que se solucionan mediante asimilación de nuevos conocimientos y acomodación de las estructuras cognoscitivas a partir de experiencias.

"El uso instruccional del computador reproduce los viejos esquemas de la instrucción programada en cualquiera de sus modalidades. Por el tipo de postulados epistemológicos de su ciencia base, le es imposible a esta tecnología trazarse como objetivo desarrollar la inteligencia. El uso interactivo del computador hace parte de una tecnología educativa derivada del estructuralismo genético de Piaget. Esta ciencia base ha desarrollado los instrumentos conceptuales y epistemológicos necesarios para darle significado empírico a enunciados acerca del desarrollo de la inteligencia." (Hernan,87)

Una teoría pedagógica de gran auge actualmente y que parte de las teorías de Piaget es el "construccionismo", cuyos principios han sido elaborados por Papert desde la implementación del lenguaje Logo, al final de la década de los 60's.

Esta teoría constructivista plantea un cambio a la educación tradicional, de modo que todo desarrollo de herramientas para asistir el aprendizaje, debieran tener sus bases en esta teoría. Papert se refiere al construccionismo como una teoría que maximiza lo aprendido y minimiza lo enseñado.

SEGUNDA PARTE

Usos del Computador en Educación

El computador puede emplearse en varios sectores de la educación como:

- Administrativo: procesamiento de calificaciones, sistemas de información (hojas de vida de empleados y estudiantes), proceso de matrícula y pagos, etc..
- Docentes: herramientas para aumentar la productividad (procesador de texto, hoja de cálculo, presentaciones, etc.), herramienta de cómputo (cálculos difíciles y monótonos), herramienta de consulta, investigación y actualización, asistencia en tareas específicas (planeación de cursos, evaluación y seguimiento de cada estudiante).
- Estudiantes: herramientas para aumentar la productividad (procesadores de texto, hoja de cálculo, presentaciones, bases de datos, etc.), herramienta de consulta, investigación y actualización, computador como objeto de estudio (aprender a programar), asistencia en el proceso de evaluación (exámenes por computador), **Asistencia en la construcción del conocimiento.**

La única que puede representar un cambio real en la educación tradicional es la última: asistencia a los estudiantes en la construcción del conocimiento. Esta se presenta como una alternativa a la **fracasada educación actual.**

Al fracaso de la educación tradicional se han atribuido muchas razones: el curriculum no tiene relación con la vida real de los alumnos, los docentes no están bien preparados, los estudiantes carecen de conocimientos básicos, falta individualización y muchas otras. El cambio que requiere con urgencia la educación no se dará apoyando las labores usuales de la educación (tareas administrativas, empleándolo para hacer trabajos, como enciclopedia, como calculadora, o simplemente asistiendo la evaluación tradicional), sino cambiando sus cimientos y volviendo a

pensar la pedagogía. Hoy, el reto de la educación es aprovechar los nuevos medios (vídeo, audio, computadores, inteligencia artificial, realidad virtual, etc.) para atraer al estudiante y permitirle la construcción de su conocimiento.

MATERIALES EDUCATIVOS COMPUTARIZADOS

Bajo este nombre (abreviado MEC) se agrupan diversos tipos de aplicaciones encaminados a apoyar el aprendizaje. Una referencia bastante apropiada es "Ingeniería de Software Educativo" de Alvaro Galvis (Galvis, 94), de donde se ha tomado la clasificación que se presenta.

Una primera clasificación de herramientas y materiales para asistir el aprendizaje los divide en **algorítmicos** y **heurísticos**. En los materiales algorítmicos predomina el aprendizaje vía transmisión de conocimiento desde quien sabe hacia quien lo desea aprender; quien diseña la herramienta planea secuencias de actividades para conducir al estudiante; el rol de alumno es asimilar el máximo de lo que se le transmite. Por otra parte en los materiales heurísticos predomina el aprendizaje por experimentación y descubrimiento; el diseñador crea ambientes ricos en situaciones que el alumno debe explorar; el alumno debe llegar al conocimiento a partir de la experiencia, creando sus propios modelos de pensamiento, sus propias interpretaciones del mundo, las cuales puede someter a prueba con la herramienta.

Tal clasificación puede refinarse aún más:

Algorítmicos	Sistemas tutoriales
	Sistemas de ejercitación y práctica
Heurísticos	Simuladores y Juegos educativos
	Micromundos exploratorios
	Sistemas expertos
Algorítmicos y Heurísticos	Sistema tutorial inteligente

Los sistemas tutoriales pueden presentar las cuatro fases del aprendizaje, y resultan particularmente útiles cuando se requiere alta motivación, retroalimentación inmediata, ritmo propio y secuencia controlable por el usuario parcial o totalmente.

Los sistemas de ejercitación y práctica permiten reforzar las dos fases finales del proceso de instrucción: aplicación y retroalimentación por medio de ejercicios tradicionales.

Los simuladores pretenden apoyar el aprendizaje por medio de experimentos, de forma que el estudiante descubra conceptos en un micromundo semejante a una situación real. En este

tipo de MEC, que puede emplearse en cualquier de las cuatro fases, el alumno es agente activo.

Los juegos educativos, al igual que los simuladores apoyan el aprendizaje semejando situaciones, sin embargo, en la simulación se trata de situaciones reales mientras que esto no se da necesariamente en los juegos, además en éstos se dan situaciones excitantes o entretenidas.

Los micromundos exploratorios, emplean un lenguaje de programación sintónico, es decir no hay que aprenderlo, simplemente se está sintonizado con sus instrucciones y se emplea para interactuar en un micromundo. (e.g. Logo y Karel). La diferencia básica con los simuladores es que además de exigir la solución de problemas, la exige de forma estructurada (es decir, una que conlleve división de problemas en subproblemas). Por esta razón, los lenguajes sintónicos permiten el desarrollo de estrategias para solución de problemas.

Los sistemas expertos son sistemas capaces de representar y razonar acerca de algún dominio rico en conocimientos, con el ánimo de resolver problemas y dar consejo a quienes no son expertos en la materia. Estos sistemas además de demostrar gran capacidad de desempeño en términos de velocidad, precisión y exactitud, cuentan con una base de conocimientos construida a partir de experiencia humana. Con la base de conocimientos y con reglas de alto nivel es capaz de hallar o juzgar la solución a algo, explicando o justificando lo que halla o lo que juzga, de modo que es capaz de convencer al usuario de que su razonamiento es correcto.

Un sistema Tutorial Inteligente presenta un comportamiento "inteligentemente" adaptativo, es decir, adapta el tratamiento educativo en función de aquello que se desea aprender y de las características y desempeño del aprendiz. Además de tener los componentes típicos de un sistema experto (base de conocimiento, motor de inferencia, hechos e interfaz con usuario) hay un "modelo de estudiantes" donde se plasman sus conocimientos, habilidades y destrezas y un "modulo de interfaz" capaz de ofrecer distintos tipos de ambiente de aprendizaje a partir de las cuales se puede llegar al conocimiento buscado.

INGENIERÍA DE SOFTWARE EDUCATIVO

La enseñanza asistida por computador se ha convertido en una rama de investigación importante de la inteligencia artificial. Francisco Ruedas (Ruedas,92) menciona algunas técnicas, empleadas para desarrollar software educativo, tales como representación del conocimiento, sistemas expertos, redes neuronales y procesamiento de lenguaje natural.

Metodología para el desarrollo de software educativo

En cuanto a metodología de desarrollo, varios autores han tratado el tema, por ejemplo Jaime Prelusky (Prolusky,95) o Alvaro Galvís (Galvis,94). De éste último: "Ingeniería de Software Educativo", es una referencia bastante completa y es una buena guía para el desarrollo del software. En esencia se conservan los grandes pasos o etapas de un proceso sistemático para desarrollo de materiales (análisis, diseño, desarrollo, prueba y ajuste, implementación). Sin embargo, en este caso se da particular énfasis a los siguientes aspectos: la solidez del análisis, como punto de partida; el dominio de teorías sustantivas sobre el aprendizaje y la comunicación humanas, como fundamento para el diseño de los ambientes educativos computarizados; la

evaluación permanente y bajo criterios predefinidos, a lo largo de todas las etapas del proceso, como medio de perfeccionamiento continuo del material; la documentación adecuada y suficiente de lo que se realiza en cada etapa, como base para el mantenimiento que requerirá el material a lo largo de su vida útil.

ANÁLISIS DE NECESIDADES EDUCATIVAS

Todo MEC debe cumplir un papel relevante en el contexto donde se utilice. Su incorporación a un proceso de Enseñanza/Aprendizaje no se puede deber simplemente a que el MEC *“es chévere”*, o a que *“está disponible”*. Estas y otras razones probablemente lleven a dedicar recursos a labores que no producen los mejores resultados.

A diferencia de las metodologías asistemáticas, donde se parte de ver de qué soluciones disponemos para luego establecer para qué sirven, de lo que se trata acá es de favorecer en primera instancia el análisis de qué problemas o situaciones problemáticas existen, sus causas y posibles soluciones, para entonces si determinar cuales de éstas últimas son aplicables y pueden generar los mejores resultados.

Esta metodología entonces pretende responder a las preguntas: ¿cómo identificar las necesidades o los problemas existentes?, ¿qué criterios usar para llegar a decidir si amerita una solución computarizada? ¿con base en qué, decidir si se necesita un MEC y qué tipo de MEC conviene que sea, para satisfacer una necesidad dada?.

Es por lo anterior que esta metodología se centra en:

- Consulta a fuentes de información apropiadas e identificación de problemas
- Análisis de posibles causas de los problemas detectados
- Análisis de alternativas de solución
- Establecimiento del papel del computador en la solución al problema

SELECCIÓN O PLANEACIÓN DEL DESARROLLO DE MEC

El proceso de análisis de necesidades educativas que ameritan ser atendidas con MECs no termina aún. Falta establecer si existe o no una solución computarizada que satisfaga la necesidad que se detecta, en cuyo caso podría estar resuelta, o si es necesario desarrollar un MEC para esto.

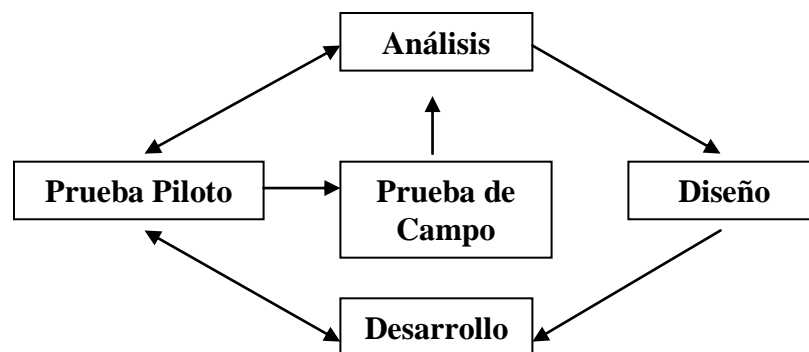
Cuando se identifican uno o más paquetes que parecen satisfacer las necesidades, es imprescindible someterlos al ciclo de revisión y prueba de MECs que asegure que al menos uno de ellos satisface la necesidad. Para esto es indispensable tener acceso a una copia documentada de cada MEC, como etapa final de la fase de análisis, y hacerlo revisar por expertos en contenido, metodología e informática. Los primeros, para garantizar que efectivamente corresponde al contenido y objetivos de interés. Los expertos en metodología para verificar que

el tratamiento didáctico es consistente con las estrategias de enseñanza/aprendizaje que son aplicables a la población objeto y al logro de tales objetivos. Los expertos en informática para verificar que dicho MEC se puede ejecutar en la clase de equipos de que dispondrán los alumnos y que hace uso eficiente de los recursos computacionales disponibles. Si todo esto se cumple habrá terminado el análisis con al menos un MEC seleccionado para atender la necesidad.

Cuando no se identifica un MEC con el cual satisfacer la necesidad, la fase de análisis culmina con la formulación de un plan para llevar a cabo el desarrollo del MEC requerido. Esto implica consultar los recursos disponibles y las alternativas de usarlos para cada una de las etapas siguientes. Se debe prever tanto lo referente a personal y tiempo que se dedicará a cada fase, así como los recursos computacionales que se requieren para cada fase en particular las de desarrollo y pruebas piloto y de campo.

CICLOS PARA LA SELECCIÓN O EL DESARROLLO DE MECS

La anterior explicación permite entender la razón de ser del doble ciclo, para selección o desarrollo de MECs, que ha propuesto Galvis como metodología básica de trabajo y que se ilustra en la siguiente figura.



Modelo sistemático para selección
o desarrollo de MECs propuesto por Alvaro Galvis

El punto de partida de ambos ciclos es la identificación de necesidades educativas reales que conviene atender con material educativo computarizado. Dependiendo del resultado final de esta etapa, se procede en el sentido contrario al avance de las manecillas del reloj, cuando se trata de seleccionar un MEC; pero en el mismo sentido del avance del avance de las manecillas, si conviene efectuar su desarrollo.

En cualquiera de los dos ciclos, una vez que se dispone de un MEC, se requiere evaluarlo con un grupo piloto de alumnos que pertenezca a la población objeto, bajo las condiciones para las cuales está diseñado. Esta es la base para decidir si el MEC debe llevarse a la práctica en gran escala, o para echar pie atrás, rediseñarlo, ajustarlo o desecharlo. Durante su implementación

también es importante que se evalúe el MEC, de modo que se pueda establecer la efectividad real del material, éste es el sentido de la prueba de campo.

DISEÑO DE MECs

El diseño de un MEC está en función directa de los resultados de la etapa de análisis. La orientación y contenido del MEC se deriva de la necesidad educativa o problema que justifica el MEC, del contenido y habilidades que subyacen en esto, así como de lo que se supone que un usuario del MEC ya sabe sobre el tema; el tipo de *software* establece, en buena medida, una guía para el tratamiento y funciones educativas que es deseable que el MEC cumpla para satisfacer la necesidad.

Entorno para el diseño del MEC

A partir de los resultados del análisis, es conveniente hacer explícitos los datos que caracterizan el entorno del MEC que se va a diseñar: destinatarios, área del contenido, necesidad educativa, limitaciones y recursos para los usuarios del MEC, equipo y soporte lógico que se van a utilizar.

Entorno del diseño

- ¿A quiénes se dirige el MEC ?, ¿ qué características tienen sus destinatarios ?
- ¿ Qué área de contenido y unidad de instrucción se beneficia con el estudio del MEC ?
- ¿ Qué problemas se pretende resolver con el MEC ?
- ¿ Bajo que condiciones se espera que los destinatarios usen el MEC?
- ¿Para un equipo con las características físicas y lógicas conviene desarrollar el MEC?

A lo anterior habrá que agregar un:

Diseño educativo del MEC

El diseño educativo debe resolver los interrogantes que se refieren al alcance, contenido y tratamiento que debe ser capaz de apoyar el MEC.

Diseño de comunicación

La zona de comunicación en la que se maneja la interacción entre usuario y programa se denomina *interfaz*. Para especificarla, es importante determinar cómo se comunicará el usuario con el programa, estableciendo mediante qué dispositivos y usando qué códigos o mensajes (interfaz de entrada); también se hace necesario establecer cómo el programa se comunicará con el usuario, mediante qué dispositivos y valiéndose de que códigos o mensajes (interfaz de salida).

Diseño computacional

Con base en las necesidades se establece qué funciones es deseable que cumpla el MEC en apoyo de sus usuarios, el profesor y los estudiantes. Entre otras cosas, un MEC puede brindarle al alumno la posibilidad de controlar la secuencia, el ritmo, la cantidad de ejercicios, de abandonar y de reiniciar. Por otra parte, un MEC puede ofrecerle al profesor la posibilidad de editar los ejercicios o las explicaciones, de llevar registro de los estudiantes que utilizan el material y del rendimiento que demuestran, de hacer análisis estadísticos sobre variables de interés, etc.

La estructura lógica que comandará la interacción entre usuario y programa deberá permitir el cumplimiento de cada una de las funciones de apoyo definidas para el MEC por tipo de usuario. Su especificación conviene hacerla modular, por tipo de usuario, y mediante refinamiento a pasos, de manera que haya niveles sucesivos de especificidad hasta que se llegue finalmente al detalle que hace operacional cada uno de los módulos que incluye el MEC. La estructura lógica deberá ser la base para formular el programa principal y cada uno de los procedimientos que requiere el MEC.

Finalmente, es necesario determinar de cuáles estructuras de datos es necesario disponer en memoria principal y cuáles en memoria secundaria (archivos de disco), de modo que el programa principal y los procedimientos de que se compone el MEC puedan cumplir con las funciones definidas.

DESARROLLO DE MECs

Desde la fase de análisis, cuando se formuló el plan para efectuar el desarrollo, debieron haberse asignado los recursos humanos temporales y computacionales necesarios para todas las demás fases. Tomando en cuenta esto, una vez que se dispone de un diseño debidamente documentado es posible llevar a cabo su implementación (desarrollarlo) en el tipo de computador seleccionado, usando herramientas de trabajo que permitan, a los recursos humanos asignados, cumplir con las metas en términos de tiempo y de calidad de MEC.

PRUEBA PILOTO DE MEC's

Con la prueba piloto se pretende ayudar a la depuración del MEC a partir de su utilización por una muestra representativa de los tipos destinatarios para los que se hizo y la consiguiente evaluación formativa. Para llevarla a cabo apropiadamente se requiere preparación, administración y análisis de resultados en función de buscar evidencia para saber si el MEC está o no cumpliendo con la misión para la cual fue seleccionado o desarrollado.

PRUEBA DE CAMPO DE MECs

La prueba de campo de un MEC es mucho más que usarlo con toda la población objeto. Si exige hacerlo, pero no se limita a esto. En efecto, dentro del ciclo de desarrollo de un MEC hay que buscar la oportunidad de comprobar, en la vida real, que aquello que a nivel experimental parecía tener sentido, lo sigue teniendo.

EL MODELAJE Orientado por objetos: UN MEDIO PARA DESARROLLAR MEC's

La ingeniería de software como disciplina ha evolucionado significativamente en lo que se refiere a modelos conceptuales y herramientas de trabajo, que hacen del proceso de desarrollo y mantenimiento de software una actividad cada vez menos dependiente del arte de quienes llevan a la práctica un diseño elaborado. Dentro de estos aportes se destacan los de la orientación por objetos, que cubre todo el ciclo de vida del software.

Alvaro Galvis en su trabajo "Ingeniería de Software Educativo con Modelaje Orientado por Objetos: un medio para desarrollar micromundos interactivos" (<http://www.minerva.uevora.pt/simposio/comunicacoes/rigomezmarino.html>) integra el modelaje O.O. con la metodología de ISE propuesta por él, para enriquecer el proceso de desarrollo de MEC. Como punto de partida identifica las características que debería poseer un MEC, particularmente un Micromundo Interactivo, fruto de evaluar varias aplicaciones existentes en el mercado con este tipo de Micro mundos. A partir de allí se realiza una adaptación y/o redefinición de los pasos que debe seguir una metodología de ISE en su componente computacional.

CONCLUSIONES

En el momento de estar escribiendo estas conclusiones, me he topado con la propuesta de un enfoque para el desarrollo de software educativo, que no solo ha pasado por alto las teorías del aprendizaje como sustento al ciclo de desarrollo de vida de un software educativo, sino mas aún ha demostrado un desconocimiento total de las metodologías existentes en este campo.

Lo anterior nos lleva a concluir que el estudio de las teorías psicológicas del aprendizaje humano, de los tipos de software educativos existentes y de las metodologías para desarrollarlos, pueden entrar a ayudarnos a no *replicar indiscriminadamente, las estrategias de enseñanza-aprendizaje que se conocen y además a no desaprovechar algunas características utiles del computador y de las Ciencias de la Computación*. Su estudio y conocimiento, es prioritario para aquellos profesionales que se involucren en el proceso de desarrollo de este tipo de software.

Por último, como proyección debemos indicar que la utilización cada día más efectiva, de técnicas de IA en Educación, deberá obligar a la Ingeniería de Software Educativo a complementar sus propuestas con las que nos presentan la Ingeniería del Conocimiento, permitiendo así al Ingeniero de Software y a los expertos (docente, comunicador, psicólogo, etc.) compartir más eficazmente el conocimiento que tanto le cuesta entregar a estos últimos, y que tanto le cuesta formalizar a los primeros.

BIBLIOGRAFÍA

(Galvis, 94) "Ingeniería de Software Educativo". Alvaro Galvis Panqueva. 1994. Ediciones Uniandes.

(Hernan,87) "El uso instruccional vs. el uso interactivo del computador respecto al desarrollo de la inteligencia", Hernan Escobedo David. Aparece en Memorias del Primer Congreso Colombiano de Informática, Educación y Capacitación. 1987

(Holt,32) "Purposive behavior in Animals and Men" Holt, M. y Tolman (1932), E. New York: Appleton-Century-Crofts.

(Lysaught, 75) "Introducción a la enseñanza programada". Lysaught, J.P. y Williams, C.M. (1975). México, editorial Limusa.

(Papert,81) "Desafío a la Mente. Computadoras y Educación". Seymour Papert (1981). Ediciones Galápagos.

(Prolusky,95) "A Practical Guide for Developing Educational Software", Jaime Prilusky. Aparece en Proceedings of a Meeting on the Use of Microcomputer for Developing Countries.

(Ruedas,92) "La inteligencia artificial, sus principios básicos y sus aplicaciones educativas", Francisco Rueda. Aparece en Memorias del Congreso Colombiano de Informatica Educativa. 1992

(Skinner, 53) "Science and Human Behavior". Skinner, B.F. (1953) New York: free press (en Español: Ciencia de la Conducta Humana. Barcelona: Fontanela, 1970).