

DESARROLLO DE SOFTWARE EDUCATIVO MEDIANTE LENGUAJES O SISTEMAS DE AUTORIA

Fernando Ferrer Olivares

RESUMEN

Este artículo presenta una visión del concepto de autoría y de las herramientas computacionales disponibles al respecto. Se detallan las características principales de los lenguajes y de los sistemas autores, analizando algunos ejemplos. Finalmente se discuten aspectos que se deben considerar al seleccionar una herramienta de autoría. Con esto se espera orientar la acción de los docentes que desean producir por sí mismos sus propios materiales educativos computarizados.

"Existe la sensación de que la educación por computador puede ser un logro importantísimo; la forma de alcanzarlo, sin embargo, no parece del todo clara".

CONCEPTO Y HERRAMIENTAS DE AUTORIA

Uno de los enfoques de producción de materiales educativos computarizados está relacionado con el concepto de autoría. Este término se refiere a todas las actividades tendientes a desarrollar materiales de enseñanza-aprendizaje por computador (contenidos y metodología), similar a las que realiza un autor para escribir un libro de texto.

Las herramientas provistas por la informática para llevar a la práctica este enfoque se clasifican en dos grupos: los lenguajes de autoría y los sistemas de autoría; ambas tratan de simplificar la interacción entre el computador y el hombre (particularmente educadores), cuando esta se orienta a la producción de materiales educativos computarizados.

En este artículo se tratará de explorar las características y el potencial de dichas herramientas, de modo que quienes se interesan por hacer uso de ellas tengan una base para seleccionarlas y sacar el mayor provecho.

LENGUAJES AUTORES

Un lenguaje de autoría es un lenguaje de programación altamente especializado, útil para desarrollar aplicaciones del computador relacionadas con el proceso de enseñanza-aprendizaje; con él se intenta poner a disposición del instructor-autor todas las capacidades del computador en una forma simple y adecuada a sus necesidades [1].

Lo anterior se puede obtener proveyendo al usuario-autor los siguientes elementos:

- conjuntos de funciones y subrutinas diseñadas especialmente para realizar las operaciones más frecuentemente desarrolladas en la programación de diálogo instruccional. Por ejemplo, desplegar textos, gráficos o animaciones, producir sonido, recibir y analizar mensajes.
- facilidades para dar control al usuario sobre medios instruccionales externos tales como proyectores, televisión y videodisco.
- un conjunto fácil de comandos orientados hacia los requerimientos del autor. En particular, comandos para edición de cada uno de los elementos que el autor tiene a su disposición.
- facilidades de manipulación de archivos para:
 - . Posibilitar el almacenamiento de material instruccional para presentación a estudiantes.
 - . Registrar y recuperar la información acerca de las respuestas de alumnos, recolectadas durante un diálogo instruccional

La mayoría de los lenguajes de autoría existentes permiten:

- Presentación de pantallazos usando texto, gráficos, animaciones y sonido.
- Hacer preguntas a los estudiantes, tanto cerradas como relativamente abiertas.
- Recepción y análisis de las respuestas de los estudiantes.
- Ofrecer material remedial y de refuerzo, pre-programado con el lenguaje u ofrecido con ayuda de otro dispositivo (p.ej., video-cinta) y de la interfaz apropiada.
- Flexibilidad en la secuencia, dependiendo de los eventos que el autor predefina asociados a las decisiones y acciones que tome el estudiante (individualización).

La mayoría de los lenguajes autores operan en la misma forma como los compiladores e interpretadores de los lenguajes convencionales. Esto es, tienen un archivo previamente editado que contiene instrucciones (código fuente) y traslada éstas a un programa en código de máquina ejecutable. Este código controla la actividad del computador mientras dura la sesión de estudio del estudiante.

Algunos Lenguajes autores

Los lenguajes autores han evolucionado grandemente desde su aparición en los sesentas, cuando apareció TIP (Traslato for Interactive Programs), el cual corría en un IBM 650 y que luego evolucionó en el lenguaje COURSEWRITER. Por la época en que se puso a disposición también lo hizo CATO (Compiler for Automatic Teaching Operations), ambos orientados por "marcos" (cada uno para un pantallazo), siendo la principal diferencia entre los dos que el segundo incorporó una serie de funciones ligadas a ciertas teclas. CATO fue reemplazado por TUTOR, también orientado por marcos. Los anteriores son ejemplos de lenguajes autores hechos para sistemas de computación específicos [2].

También existen lenguajes autores que son independientes del sistema de computación. Por

este motivo hay versiones para diferentes tipos de computadores (main-frame, minis, micros) y marcas. Entre los más destacados están : SAL (Simple Authoring Language); es esencialmente un lenguaje autor multimedial. Está compuesto por recursos para aprendizaje audio-visual (grabadora y proyector) e instrumentos controlados por el computador. STAF (Science Teachers Authoring Facility). Desarrollado por el Programa Nacional en Aprendizaje Asistido por Computador de la Gran Bretaña [3]. Es altamente portable (main-frame, mini y microcomputadores). Aunque el lenguaje fué desarrollado inicialmente para propósitos tutoriales, también puede ser empleado en técnicas de simulación y modelaje con diálogo tutorial. CICERO. Desarrollado por la Universidad Abierta de Gran Bretaña, sirve fundamentalmente instrucción de tipo tutorial [4]. ACL (Author Command Language). Orientado a ayudar al estudiante en dos áreas: velocidad y seguridad. Permite evaluar a los estudiantes dentro de ciertos límites de tiempo y rangos de error. PILOT (Programmed Inquiry, Learning Or Teaching) : el lenguaje está disponible para un número grande de microcomputadores y algunos minicomputadores. Permite preparar material computarizado y multimedial de tipo tutorial; algunas versiones permiten encadenar módulos escritos en otros lenguajes [5]. NATAL (NATional Author Language), sistema desarrollado en Canadá y cuyo principal atributo es el de permitir autoría utilizando un ambiente multimedial (microcomputador, videodisco).

SISTEMAS AUTORES

El primer paso para la definición de sistemas autores fue la aparición de lenguajes autores, los cuales facilitaron la escritura de programas de instrucción. Los sistemas de autoría van más allá y tratan de anticipar las necesidades del instructor para armar el material que desee.

Un sistema autor es un programa o conjunto de programas que permiten a un instructor crear materiales de enseñanza usando el computador, sin necesidad de programar. Esto se logra debido a que la comunicación entre el sistema de autoría y el usuario se efectúa mediante la presentación de gráficos, listas o menús; se intenta hacer la programación tan transparente como sea posible.

Los sistemas autores eliminan la necesidad de que el autor del material trate con la sintaxis o secuenciamiento de comandos de un lenguaje autor. De esta forma, puede decirse que un lenguaje autor hace fácil la autoría a quien sabe programar y desea enseñar algo, mientras que un sistema autor hace posible a no-programadores desarrollar por sí mismos sus propios materiales.

Los sistemas autores responden fundamentalmente a tres razones :

- Ofrecer facilidad de uso de, o de acceso a, computadores con propósitos instruccionales. La necesidad de saber programar limita significativamente el número de profesores que intenta preparar sus propios materiales y aún quienes algo saben tienen dificultades para usar eficientemente las facilidades interactivas del computador.

- Reducir los de costos y tiempos requeridos para crear materiales de enseñanza, particularmente en la etapa de desarrollo y prueba de los materiales.
- Favorecer transportabilidad del material, cuando a nivel regional, sectorial o interinstitucional se ha adoptado un sistema para autoría de cursos o un estándar básico para adquisición de equipos.

Características funcionales de los sistemas de autoría

Para favorecer el logro de sus propósitos, en los sistemas autores suelen materializarse de uno a cuatro niveles de capacidades, las cuales están organizadas en orden creciente de complejidad. La primera es lo mínimo que puede ofrecer un sistema autor y las cuatro serían lo que ofrecería un sistema autor bien completo [6]:

- Creación de contenidos

Se refiere a la entrada, formateo y modificación de texto, gráficas, audio o cualquier otra información, la cual será mostrada y/o almacenada por el sistema. Este nivel representa las capacidades mínimas de un sistema autor. Puede incluir un simple editor de texto, pero puede ofrecer también edición de gráficos, animaciones, música y efectos de sonido. Este nivel es el que más depende de los dispositivos de entrada y salida, ya que la forma en que el contenido es entrado, la forma en que puede ser modificado y los posibles tipos de presentación son funciones relacionadas con el hardware.

- Definición de lección.

Incluye funciones que permiten especificar la estructura de cada una de las lecciones. Incluye tres categorías principales de funciones de apoyo: (1) las relativas a la presentación de la información (para especificar los mensajes a ser presentados, sea digitándolos o haciendo referencia a una gráfica ya creada o a una secuencia audiovisual); (2) las que permiten procesar la respuesta del estudiante (qué se considera correcto, qué hacer si la respuesta no coincide con la que prevista, qué mensaje o acción de retroalimentación presentar); (3) las relativas al control de la secuencia de ejecución (indican que acción se debe realizar al completarse la presentación de una información, el análisis y tratamiento de una respuesta o la lección misma).

- Administración del curso

Esta capacidad incluye la posibilidad de definir o seleccionar una estrategia instruccional particular (p.ej., ejercitación y práctica, quiz, tutorial...), de especificar qué datos de las respuestas conviene recolectar, la posibilidad de revisar una lección en modo estudiante desde el modo autor, así como las opciones de control que se ponen a disposición del estudiante por lección o a nivel global. A diferencia de las opciones a nivel de lección, estas

funciones tienen efectos globales sobre todas las lecciones que comprenden un curso.

- **Medio ambiente del sistema de autoría.**

En primera instancia esta función adecúa un sistema autor a las características del autor que lo usa. Se trata de tener diferentes máscaras (interfaces) para interacción entre el sistema y el autor, dependiendo de si es un novato o tiene alguna o mucha experiencia como autor. Un segundo aspecto ligado a esto es el de la configuración del hardware con que se desarrolla y utiliza el material generado con el sistema. Cuando son las mismas, no hay problema. Pero si el sistema autor corre en máquinas más elaboradas o más poderosas que aquellas en que se correrán los programas generados, es importante que el sistema autor pueda emular el funcionamiento de la máquina objeto cuando el autor decide observar lo hecho en modo estudiante. Esto le permitirá balancear los tiempos y el ritmo de trabajo.

Algunos sistemas autores disponibles.

Al final de los sesenta aparecieron los primeros sistemas autores. VAULT fué un sistema autor escrito en PL/I para el computador IBM 1500. Este sistema introduce el concepto importante de separar la lógica instruccional del contenido durante el proceso de autoría. El sistema TICCIT [7] (Time-shared, Interactive, Computer-Controlled Information Television) aprovechó el concepto anterior y se caracterizó por incluir una lógica particular de instrucción (i.e., control del aprendizaje), de modo que el autor simplemente provee el contenido. Varios sistemas autores se desarrollaron para el sistema PLATO (p.ej., MONIFORM [8]).

Con la aparición de mini y microcomputadores, nuevos sistemas autores han aparecido especialmente diseñados para máquinas pequeñas. WISE es uno bastante completo, pero dependiente de la máquina, al menos en modo autor; la autoría requiere un equipo WICAT [9], y las lecciones corren en micro-computadores bajo MS-DOS con tarjeta gráfica apropiada. Otros sistemas que corren en equipos pequeños y demuestran los tres primeros niveles de un sistema autor son SCENARIO [10] e IDEA, el primero de los cuales corre en equipos PC compatibles con tarjeta CGA, y el segundo sobre equipos EPSON.

CONSIDERACIONES SOBRE USO DE LOS LENGUAJES Y SISTEMAS DE AUTORIA

Es importante evaluar en cada caso qué limitaciones impone un lenguaje o un sistema de autoría sobre el autor del material y sobre los usuarios del mismo. Para hacer esto es conveniente determinar qué facilidades ofrece y qué restricciones impone la herramienta para la autoría y para el uso del material generado. Una herramienta puede tener variados grados de potencia, dependiendo de las capacidades que tenga. Pero al mismo tiempo, es importante dilucidar a qué costo y con qué esfuerzo es viable hacer uso de tal potencia. Por ejemplo, usted puede tener una herramienta muy poderosa, pero si los requerimientos de máquina, sus costos, o el esfuerzo que implica usarla exceden sus capacidades, de poco le

sirve. Lo contrario también es cierto. Por más que la herramienta esté al alcance de sus posibilidades, que corra en su equipo y sea amigable, si la clase de material que permite hacer es poco relevante a sus necesidades, tampoco vale la pena.

La siguiente lista de factores pueden servir de base para efectuar la selección de herramientas de autoría (lenguajes o sistemas):

- Facilidad de uso. Dependiendo de la preparación del usuario-autor, la herramienta no debe demandar demasiado esfuerzo del mismo para hacer uso eficiente y eficaz de esta. Tal cualidad asegura aumento en la productividad del autor.
- El mejoramiento en la productividad del autor no es suficiente. También es necesario que la clase de software que se genere logre un efecto semejante en el aprendizaje del estudiante. ¿Permite el software generado cumplir funciones de apoyo que enriquezcan los ambientes de aprendizaje existentes? ¿Operan eficientemente los programas creados en el tipo de máquinas de que dispondrán los usuarios-estudiantes ?
- Costo. Considerando el beneficio que puede traer el uso de la herramienta de autoría ¿vale la pena invertir el costo asociado a obtener copia legal y con soporte técnico?
- Disponibilidad. ¿Qué tan accesible es la herramienta para quienes deseen usarla? ¿Qué cantidad y calidad de materiales se han generado con ella, para pensar en compartir productos?
- Portabilidad. Hasta qué punto la herramienta y los productos que se generan con ella se pueden usar o convertir para uso en otras máquinas? ¿Qué tanto son transportables los materiales generados, de manera que se puedan compartir esfuerzos de producción?
- Extensibilidad. ¿En qué medida y con qué dificultad los materiales producidos pueden crecer o adecuarse, de manera tal que puedan acomodarse a nuevas necesidades?
- Tecnología. ¿Requiere alguna instalación y configuración especial el uso de la herramienta? ¿Qué dispositivos gráficos, de sonido, de almacenamiento, de tele-comunicación y medios de comunicación audio-visual se pueden soportar con ella? ¿Con qué esfuerzo y a qué costo?
- Documentación y soporte técnico para la herramienta. ¿Qué tan completos y claros son los manuales? ¿En qué medida los representantes locales de la herramienta están preparados para dar soporte técnico cuando el problema excede a lo que dicen los manuales? ¿Qué entrenamiento ofrecen como soporte al uso de la herramienta?

Las consideraciones anteriores pueden ayudar a dilucidar la conveniencia o no de la herramienta, cuando de lo que se trata calza dentro de la filosofía con que están hechas la mayoría de ellas, es decir, orientadas a apoyar ambientes de aprendizaje que se basan en "marcos" y secuencias de ellos, con las limitaciones particulares que cada herramienta

pueda tener para manejar gráficos, materiales multimediales y simulaciones. Como dice Kearsley [6], el problema con la "orientación hacia marcos" es que estos no permiten individualizar la instrucción de una manera profunda, ni tampoco son capaces de un diagnóstico y remediación sofisticados. La orientación alternativa proviene de los sistemas "basados en conocimientos", aproximación en la que las estrategias de enseñanza toman la forma de conceptos relacionados y reglas de inferencia. Estas aplicaciones, derivadas de los sistemas expertos, están aún a nivel de prototipos y están llamadas a enriquecer el potencial de las herramientas de autoría.

REFERENCIAS

- 1 Taylor, SS (1979). Create: a Computer-based authoring Curriculum. En H. O'Neil (Ed). Issues in Instructional Systems Development. Nueva York: Academic Press.
- 2 Dean, P.M. (1978) Computer-Assisted Instruction Authoring Languages. Educational Technology, 18 (4), 20-23.
- 3 Murphy, P.J. (1982). Interactive Computer Tutorials in Open University Biology Courses. Milton Keynes, England: OU-CAL Research Group, technical report N° 31.
- 4 Jones, Ann & O'Shea, Tim (1981). An evaluation of Tutorial CAL at the Oen University : The use of CICERO in SDT286. Milton Keynes, England: OU-CAL Research Group, technical report N° 5.
- 5 Yob, G. (1977). PILOT. Creative Computing, 3 (Mayo/Junio 1977), 57-63.
- 6 Kearsley, Greg (1982). Authoring Systems in Computer-based Education. Communications of the ACM. 25 (7), 429-437.
- 7 Kazeltine Corporation (1979). APT (Authoring Procedure for TICCIT); TAS (TICCIT Authoring System). MacLean, Va.
- 8 Schulz, R.E (1975). Lesson MONIFORM : An Authoring aid for the PLATO IV CAI System. Alexandria, VA: Human Resources Research Organization.
- 9 WICAT, Inc. (1984). CDS Products Brief. Orem, Utah.
- 10 Techbyte Inc. (1987) SCENARIO - A Modular Integrator. Quebec : autor.