

Una Revisión de los Aspectos Clave en el Diseño de los Sistemas de Enseñanza Basada en Casos

Guillermo Jiménez Díaz, Mercedes Gómez Albarrán

Departamento de Sistemas Informáticos y Programación
Universidad Complutense de Madrid
Profesor José García Santesmases, s/n. 28040, Madrid, España
gjimenez@fdi.ucm.es, albarran@sip.ucm.es

Resumen

La Enseñanza Basada en Casos sugiere que el alumno aprende a partir de experiencias pasadas. Este trabajo presenta los aspectos clave de los sistemas que usan este paradigma de enseñanza, así como un análisis sobre cómo son abordados dichos aspectos en algunos sistemas de enseñanza representativos.

Palabras clave: Enseñanza basada en casos; software educativo inteligente; tutores inteligentes basados en casos.

1. Introducción

Desde hace más de veinte años se han venido desarrollando sistemas de enseñanza que han empleado distintas técnicas de Inteligencia Artificial. Entre estos sistemas, los hay que usan técnicas de Razonamiento Basado en Casos (CBR, del inglés Case-Based Reasoning), debido a que el aprendizaje basado en casos es un método natural de enseñanza [Kolodner'97].

El CBR surgió como una metodología de resolución de problemas que posibilita obtener sistemas basados en experiencia de manera más rápida que elaborar sistemas expertos clásicos basados en reglas [Kolodner'93]. El CBR intenta aplicar el comportamiento que hay tras la memoria humana, así como la forma en que razonamos y aprendemos. La implementación de estas teorías de razonamiento y aprendizaje dio lugar a que investigadores como Roger Schank y Janet Kolodner se planteasen el uso de las técnicas de CBR en la enseñanza.

La Enseñanza Basada en Casos (EBC) [Schank'90, Schank'95b] tiene en cuenta algunos aspectos

fundamentales del aprendizaje natural humano [Schank'95c]:

- Dirigido por objetivos. Las personas aprendemos mientras perseguimos un objetivo que nos resulta interesante.
- Dirigido por errores. Los errores hacen que nos demos cuenta de que necesitamos aprender ya que hemos detectado una carencia en nuestro conocimiento.
- Basado en experiencias. Ante un problema solemos pensar en una situación similar, personal o ajena, que nos ayude a resolverlo.
- Aprendemos haciendo. El aprendizaje activo hace que nos aseguremos de cómo usar aquello que hemos aprendido.

La EBC enseña a los alumnos una determinada disciplina haciendo uso de experiencias pasadas (casos) y se apoya en que las personas aprendemos a partir de errores. De ahí que esta técnica sitúe al alumno ante una situación en la que pueda cometer fallos. En estos momentos es cuando el alumno está

preparado para aprender aquello que el profesor está dispuesto a contarle (experiencias pasadas).

Por tanto, la EBC se encuentra íntimamente relacionada con el aprendizaje activo. Se puede decir que la EBC y el aprendizaje activo se complementan:

- El aprendizaje activo proporciona la actividad interesante para perseguir el objetivo del alumno, durante el desarrollo de la cual puede cometer errores.
- La EBC proporciona los casos que serán usados para proporcionar ayuda al alumno durante el aprendizaje activo.

Existe un conjunto de aspectos clave que es necesario tener en cuenta cuando se diseñan sistemas EBC [Schank'90]:

- Los casos. Uno de los aspectos más relevantes de un sistema EBC consiste en definir la estructura de los casos del sistema y para qué van a ser usados.
- Interacción con el alumno. Se ha de definir cómo se lleva a cabo la interacción con el alumno, dando especial importancia a la tarea que éste realiza en el sistema EBC y al tipo de ayuda que se le va a proporcionar.
- Evaluación de las acciones del alumno. Es necesario definir los métodos usados para comprobar si las acciones que el alumno está realizando en el sistema son o no correctas.
- Recuperación e indexación de casos. Define cómo, cuándo y con qué objetivo son recuperados los casos del sistema, así como la manera en la que se encuentran indexados dentro de la base de casos.

A estos aspectos –interacción con el alumno, casos, recuperación de casos, indexación de casos y evaluación de las acciones del alumno– hemos de añadir otro aspecto relevante: las herramientas que ayudan en el proceso de autoría y mantenimiento del sistema EBC.

El presente trabajo examina las distintas aproximaciones que un conjunto representativo de sistemas EBC han empleado en cada uno de los aspectos anteriormente señalados. Los sistemas analizados conforman un grupo suficientemente variado dentro de la EBC y han sido seleccionados atendiendo a su relevancia en determinados aspectos. Estos sistemas son: Creanimate, CATO, Proyecto AMBRE, TAO ITS y J²M.

En la Sección 2 se describen brevemente el área de aplicación y la arquitectura de cada uno de estos sistemas. Posteriormente realizaremos un estudio más detallado de todos ellos desde el punto de vista de cada uno de los aspectos señalados en esta introducción. La última sección presenta las conclusiones obtenidas de este trabajo.

2. Presentación de los sistemas a estudio

Esta sección describe brevemente los sistemas EBC escogidos.

2.1. Creanimate

Los tutores socráticos son sistemas que conducen la enseñanza del alumno mediante un diálogo en el que el tutor hace preguntas que el alumno ha de responder. Creanimate es posiblemente uno de los primeros sistemas CBR usado en el campo de la enseñanza [Edelson'92, Edelson'93, Edelson'96]. Es un tutor socrático usado para la enseñanza de la adaptación animal a niños de educación primaria. Mediante un juego en el que el niño crea un nuevo animal a partir de otro real, se pretende enseñar la relación existente entre las características físicas de los animales y su comportamiento dentro de la naturaleza.

Creanimate se compone de dos módulos fundamentales: el entorno de trabajo y el contador de historias. En el entorno de trabajo, el alumno ha de crear nuevos animales a partir de animales reales a los que va añadiendo características físicas. En este entorno también interviene el módulo Gestor de Diálogo Socrático, encargado de introducir al alumno en un ciclo de preguntas y respuestas con el que se pretende que el alumno genere sus propias hipótesis y aprenda la relación entre las características físicas y el comportamiento animal. El contador de historias es el módulo encargado de buscar y recuperar los vídeos de animales que puedan aportar algo al aprendizaje del alumno en la situación actual. Como ya se verá más adelante, en el ciclo de diálogo se han introducido unas estrategias de recuperación que serán las encargadas de monitorizar las acciones del alumno e indicar al contador de historias qué vídeos ha de recuperar.

2.2. CATO

Una de las disciplinas clásicas para la EBC es el Derecho. CATO (Case Argument TutOrial) [Aleven'97a, Aleven'97b, Ashley'02] es un tutor inteligente para enseñar a generar argumentaciones legales a partir de casos sobre secreto industrial. Se

trata de una evolución de HYPO [Ashley'91], un sistema también capaz de generar argumentaciones citando otros casos para justificar conclusiones legales. CATO proporciona al alumno una serie de herramientas que le permiten buscar casos en la base de casos del sistema y ayudan al alumno en la generación de argumentaciones a partir de un conjunto de casos. Esto se realiza mediante un diálogo de preguntas y respuestas en el que el alumno es preguntado sobre las diferencias existentes entre dos casos. CATO también es capaz de generar automáticamente las argumentaciones a partir de casos seleccionados por el alumno.

La arquitectura de CATO diferencia tres elementos fundamentales:

- **Módulo Experto.** Está compuesto por la base de casos, el modelo para la generación de argumentaciones y la Jerarquía de Factores, usada para la indexación de casos y descrita más en profundidad en la Sección 3.3.
- **Módulo Pedagógico.** Se compone del Gestor de Mini-diálogo y del Verificador del Análisis de los casos, que analiza los puntos fuertes y las debilidades de cada caso.
- **Módulo de Interfaz.** Lo constituyen seis herramientas que sirven para recuperar y analizar casos, presentar argumentaciones de ejemplo y verificación y ayuda en las argumentaciones creadas por el alumno.

2.3. Proyecto AMBRE

El proyecto AMBRE (Apprentissage de Méthodes Basé sur le Raisonnement à partir de l'Expérience) propone la enseñanza de métodos para la resolución de problemas usando las fases del CBR. Bajo este proyecto se han desarrollado varios prototipos para la resolución de problemas numéricos: AMBRE-ILE [Guin-Duclosson'02], y AMBRE-AWP [Nogry'04].

La principal particularidad de los sistemas desarrollados bajo esta propuesta es que el procedimiento usado para la enseñanza de los métodos de resolución de problemas es similar al ciclo de ejecución CBR:

- **Reformulación del problema.** El alumno ha de buscar dentro de la base de casos del sistema un enunciado más general que le permita determinar la clase de problema que ha de resolver.
- **Selección.** Dentro de la clase seleccionada en la fase anterior, el alumno ha de elegir un problema concreto sobre el que realizará la

siguiente etapa de adaptación.

- **Adaptación.** En esta fase el alumno ha de detectar los parámetros similares entre el problema propuesto y el caso recuperado para que pueda adaptar la solución de éste al nuevo problema. Una vez decidido esto, se produce la adaptación y la generación de la solución para el problema propuesto.
- **Almacenamiento.** Una vez resuelto el problema, el alumno puede considerarlo como un caso útil para posteriores sesiones. Si es así, se propone al alumno que incluya el problema dentro de la base de casos. Durante esta tarea, el alumno ha de buscar dentro del árbol sobre el que se indexan los casos el lugar donde debería situarse el problema para su recuperación en posteriores sesiones. Incluso se permite que el alumno modifique este árbol, si el nuevo problema no encaja correctamente en ninguno de los índices ya existentes.

2.4. TAO ITS

TAO ITS (Tactical Action Officer Intelligent Tutoring System) [Stottler'00, Stottler'01b] es un tutor inteligente basado en simulación y en el uso de escenarios para la enseñanza de habilidades a los futuros TAOs (TAO –del inglés, Tactical Action Officers) de la Marina de los EEUU, encargados de controlar los sensores y armas de los barcos, así como de monitorizar los movimientos de otros aviones, naves y misiles, tanto de tropas aliadas como enemigas. Esta herramienta comercial permite a los alumnos practicar en escenarios tácticos simulados, evalúa automáticamente sus acciones – comprobando así el conocimiento que tienen de los principios o conceptos tácticos y su capacidad para aplicar dicho conocimiento en situaciones concretas– e intenta remediar las deficiencias detectadas presentando información descriptiva sobre los conceptos tácticos fallados y haciendo que practiquen en nuevos escenarios que requieren la aplicación del conocimiento cuya carencia se ha detectado. La relevancia de TAO ITS radica en que combina un sistema EBC con simuladores comerciales reales.

El sistema TAO ITS se divide en tres partes:

- El generador de escenarios, usado por el instructor para crear los escenarios que se van a presentar al alumno para que practique sus habilidades.
- La interfaz del estudiante, donde se presentan al alumno los escenarios en los que ha de

practicar y donde se muestra la evaluación final que ha hecho sobre las acciones del alumno, proporcionándole la información de ayuda necesaria para corregir sus errores.

- La interfaz del instructor, donde éste puede revisar las actividades que ha realizado cada alumno, añadir nueva información de ayuda y asociar los principios tácticos que el alumno ha de practicar con los escenarios creados.

Cabe destacar que TAO ITS no está integrado con el simulador sino que es un tutor inteligente con un módulo EBC que se encuentra a la escucha de las acciones del alumno y de los eventos producidos por el simulador.

2.5. JV²M

Comprender la estructura de la Máquina Virtual de Java (JVM –del inglés Java Virtual Machine [Lindholm'99]) así como el lenguaje de compilación de Java es una tarea nada sencilla. Para hacer más atractivo el aprendizaje de la JVM se ha diseñado el sistema JV²M [Gómez-Martín'02, Gómez-Martín'03], un entorno virtual 3D que simula de manera metafórica la JVM. En este entorno, el alumno es representado por un avatar usado para interactuar con los objetos del mundo virtual que representan la estructura de la JVM. El alumno, a través de este entorno, deberá simular el comportamiento de la JVM durante la ejecución de código compilado (*Java byte codes*) resultado de la compilación del código fuente de un ejercicio. JV²M tiene la cualidad de incluir un nuevo elemento en estos sistemas: el uso de agentes pedagógicos animados. Un agente pedagógico es un agente especializado que reside en un entorno interactivo de aprendizaje. Los agentes pedagógicos animados son representados mediante una presencia física en el entorno que crea la ilusión de tener vida, lo que repercute positivamente en la motivación del alumno [Lester'97]. JAVY (Java taught Virtually), el agente pedagógico de JV²M, es el encargado de monitorizar las acciones del alumno y proporcionarle ayuda durante la realización de los ejercicios.

La arquitectura del sistema JV²M se compone de tres elementos:

- El mundo virtual. Contiene objetos estáticos y objetos dinámicos que representan estructuras de la JVM o artefactos con los que el avatar del alumno y el agente pueden interactuar.
- La interfaz del alumno. El alumno observa el mundo virtual en una simulación 3D y puede interactuar con el mismo haciendo uso de las

primitivas básicas usadas en muchas aventuras gráficas: “Mirar a”, “Coger”, “Usar” y “Usar con”.

- JAVY. Es el agente pedagógico. Dispone de toda la información relativa al escenario que el alumno está actualmente resolviendo y la información específica relativa al dominio de la JVM. Esta última se encuentra representada mediante la jerarquía conceptual, una estructura que contiene información sobre los conceptos del dominio dividida en cinco niveles de abstracción: conceptos de compilación, instrucciones de la JVM, estructura de la JVM, microinstrucciones y operaciones de la interfaz de usuario.

3. Análisis de los sistemas EBC según aspectos de diseño

En esta sección se describe el análisis realizado sobre los sistemas EBC propuestos atendiendo a sus aspectos clave de diseño.

3.1. Interacción con el alumno

En esta sección pasamos a describir el tipo de entorno en el que el sistema EBC va a interactuar con el alumno. Además, en este estudio hemos considerado que la ayuda proporcionada al alumno también forma parte de la interacción, por lo que será tratada más adelante en esta misma sección.

Los sistemas EBC incluyen algún entorno en el que el alumno realiza o resuelve una tarea. Existen distintos tipos de entorno:

- Entornos de diálogo socrático. En estos sistemas el alumno realiza una tarea que no persigue encontrar una solución correcta o incorrecta sino que busca las lagunas cognitivas del alumno y su motivación. Este es el caso de Creanimate, que propone al alumno la creación de un nuevo animal a partir de otro ya existente en la naturaleza. Una vez que el alumno selecciona un animal y una característica física que desea añadirle, Creanimate entra en un ciclo de diálogo en el que hace preguntas al alumno sobre los motivos que le han llevado a incluir esta nueva característica al animal. Con estas preguntas, se pretende que el alumno reflexione sobre sus decisiones y crear en él inquietudes sobre el comportamiento animal. Cuando Creanimate considera que se presenta una buena oportunidad para instruir al

alumno propone enseñarle un vídeo que pueda resolver algunas de sus dudas. El alumno puede ver el vídeo propuesto o, por el contrario, desechar la oferta y seguir dialogando con Creanimate para continuar con su sesión de aprendizaje.

- Entornos de resolución de ejercicios. En estos entornos el alumno tiene acceso a los medios necesarios para la resolución de un ejercicio concreto. Este es el caso de los sistemas AMBRE, en los que se propone al alumno la resolución de un problema numérico de manera guiada.
- Entornos de simulación de escenarios. En estos sistemas existe un simulador de escenarios en el que va a trabajar el alumno. Se distinguen dos tipos de simuladores, dependiendo de su integración con el sistema de enseñanza:
 - Simuladores externos. Son simuladores que son independientes del sistema de enseñanza. Este es el caso de TAO ITS. El sistema de enseñanza se encuentra a la escucha de los eventos ocurridos en el simulador y de las acciones que el alumno realiza. El principal inconveniente es que el sistema EBC no puede interactuar con el simulador.
 - Simuladores propios. Estos simuladores están integrados en el sistema de enseñanza, de modo que éste puede intervenir durante la simulación del escenario. Este es el caso de JV²M. El principal inconveniente de este tipo de sistemas es que, como el simulador no es independiente del sistema de enseñanza, cualquier modificación sobre el simulador supone hacer modificaciones sobre el propio sistema de enseñanza y sobre los casos que el mismo contiene.
- Sistemas híbridos: Son aquéllos que combinan distintos enfoques en el entorno. CATO y JV²M son dos ejemplos. CATO enfrenta al alumno con la construcción de una argumentación legal, lo que correspondería con la resolución de un ejercicio. Pero, por otro lado, CATO también puede realizar comparaciones sobre dos casos, de modo que permite guiar al alumno en la tarea de construcción de una argumentación mediante un proceso de preguntas y respuestas, al igual que los entornos socráticos. JV²M, por su parte, propone al alumno la resolución de un ejercicio relativo a la JVM. Pero, a diferencia de AMBRE, la resolución de este problema se hace introduciendo al alumno en un mundo

virtual donde se encuentran representados metafóricamente los elementos fundamentales de la JVM. Así, el alumno termina trabajando en una simulación, en lugar de resolver el ejercicio directamente en términos de la JVM.

Para analizar la ayuda de estos sistemas nos hemos centrado en dos aspectos: cuándo proporciona ayuda el sistema y qué tipo de ayuda es proporcionada.

La mayoría de los sistemas estudiados proporcionan ayuda al alumno durante la ejecución de la tarea, ya sea a petición del alumno o porque el sistema de enseñanza lo considere oportuno. TAO ITS, sin embargo, proporciona ayuda al comienzo y al final de la simulación, ya que no puede intervenir durante la misma. Cuando el alumno finaliza el escenario, TAO ITS genera un informe que resume la actuación del alumno durante la ejecución del escenario. Dicho informe señala las situaciones en las que el alumno ha mostrado comprensión de los principios por medio de su correcta aplicación, así como las situaciones en las que el alumno no ha actuado correctamente.

En lo que respecta al tipo de información de ayuda proporcionada por el sistema de enseñanza, se pueden destacar tres medios fundamentales de ayuda:

- Razonamiento dialéctico. Este tipo de ayuda supone introducir al alumno en un diálogo de preguntas y respuestas que permiten al sistema EBC comprobar que el razonamiento llevado a cabo por el alumno es correcto. Este tipo de ayuda es proporcionada por el agente pedagógico JAVY en el sistema JV²M y por el sistema CATO.
- Ejemplos. El sistema de enseñanza puede proporcionar ejemplos similares a aquél en el que se encuentra involucrado el alumno. CATO también puede proporcionar este tipo de ayuda, ya que el sistema posee una herramienta que es capaz de generar de manera automática argumentaciones anotadas con referencias a las conclusiones legales que está utilizando y que sirvan al alumno como ejemplo de comparación con las argumentaciones que él mismo está creando. De un modo similar, AMBRE propone guiar al alumno en la búsqueda de ejemplos que le sirvan para resolver el ejercicio propuesto.
- Material de referencia. La mayoría de los sistemas incluyen este tipo de información como ayuda, ya que es la más sencilla de generar. Creanimate, por ejemplo, presenta

los vídeos sobre animales para que el alumno comprenda la relación entre el comportamiento de un determinado animal y sus características físicas. TAO ITS usa el informe final de las simulaciones para que el alumno pueda acceder a los principios tácticos y revisar documentos multimedia asociados a cada uno de ellos.

3.2. Los casos del sistema

Los casos son la piedra angular de los sistemas EBC. En esta sección se analiza el uso encontrado para los casos, así como la estructura que suelen tener en los sistemas EBC.

Los casos son utilizados fundamentalmente para cubrir dos objetivos:

- Servir de información de ayuda. Las historias o anécdotas contadas por el tutor durante las clases tienen unas propiedades que las hacen particularmente útiles para la enseñanza [Schank'95a]. Los casos usados como información de ayuda suelen representar estas anécdotas que el tutor cuenta y pueden ser de muy distinta naturaleza. En Creanimate, los casos son vídeos que se presentan al alumno y que tienen un cierto valor pedagógico. Por otro lado, en CATO, cada caso guarda un breve resumen de la sentencia de un caso judicial de secreto industrial, junto con una breve explicación sobre cómo se ha llegado a tomar dicha decisión. Estos resúmenes se guardan en forma textual.
- Representan la tarea que el alumno ha de realizar. Como ya se mencionó con anterioridad, la EBC suele estar relacionada con la resolución de una tarea. En TAO ITS, JV²M y los sistemas AMBRE, un caso es una representación estructurada de una tarea que el alumno ha de realizar. En sistemas como los del proyecto AMBRE o JV²M, los casos definen ejercicios a resolver por el alumno, mientras que en TAO ITS los casos representan escenarios reales, es decir, una tarea situada en un determinado contexto de ejecución en la que el alumno va a trabajar en el simulador.

Aunque con ciertos matices, los casos que sirven de ejercicio o escenario tienen una estructura general similar en los distintos sistemas estudiados. Estos ejercicios o escenarios contienen la siguiente información:

- Una descripción del escenario o un enunciado

del ejercicio. Esta descripción puede adoptar diversas representaciones (por ejemplo, textual, como en los sistemas AMBRE o JV²M, o material multimedia, como es el caso de TAO ITS).

- Una representación de la solución al problema. Esta representación es dependiente del dominio de enseñanza:
 - AMBRE. Ya que aquí se pretende enseñar técnicas generales de resolución de problemas, la solución es representada mediante información sobre la técnica de resolución y los parámetros usados para obtener la solución al problema concreto que el caso representa.
 - TAO ITS. Este sistema contiene información procedural para la obtención de la solución correcta e información para la evaluación de las acciones que el alumno realiza. Para la primera, es necesaria una representación de las acciones correctas a realizar para resolver el problema. En algunas situaciones las acciones pueden tener un orden prefijado, en otras el orden de las acciones no es relevante, pudiendo aparecer acciones opcionales o alternativas. Para la segunda, es necesario definir los métodos que determinan la corrección de la solución del alumno. Aunque en [Stotler'99] se analizan distintos métodos, en TAO ITS se han empleado máquinas de estado finitas para la verificación de las acciones del alumno.
 - JV²M. Cada escenario de este sistema contiene una jerarquía de bloques que representa un subconjunto del árbol sintáctico de un programa en código objeto de Java. Relacionado con cada instrucción de la JVM (código objeto) existe un grafo de ejecución. En estos grafos un nodo representa un estado en el proceso de resolución del escenario, mientras que las aristas representan microinstrucciones, esto es, las acciones primitivas de la JVM que hay que usar para resolver el escenario.

Además de esta información general, los casos pueden contener información adicional para otros fines específicos. Por ejemplo, ya que en AMBRE existe un proceso de adaptación de los casos recuperados, éstos contienen características estructurales que son usadas durante dicha fase de adaptación.

Para finalizar, queremos destacar que los objetivos que pretenden cubrir los casos –tarea a resolver e información de ayuda– no tiene por qué ser excluyentes. Un ejemplo claro es la propuesta AMBRE. Los casos representan los ejercicios y, además, durante la resolución del ejercicio, el alumno consulta otros casos para razonar sobre el método que ha de usar para resolver el ejercicio propuesto. Esto muestra la posibilidad de usar los mismos casos como material de ayuda en los sistemas EBC.

3.3. Indexación de casos

Las estructuras de índices que son usadas para almacenar los casos están íntimamente relacionadas con el dominio que pretende enseñar el sistema, con los métodos de recuperación usados y con los fines de dicha recuperación. Por esto estas estructuras suelen ser propias de cada sistema, siendo demasiado heterogéneas como para poder agruparlas en este estudio.

La indexación es uno de los puntos más relevantes de Creanimate. Para cada vídeo de la base de casos de Creanimate se ha construido un índice usando una estructura de índices compuesta de dos elementos fundamentales:

- **Objetos:** pueden ser animales, características físicas, acciones y comportamientos de los animales y objetos físicos.
- **Relaciones:** se encargan de conectar los objetos entre sí para generar la estructura. Algunas de las principales relaciones existentes son: FeaFun (Feature for a Function), que relaciona una característica física con la acción para la que es usada –por ejemplo, patas largas para correr rápido; Plans, que relaciona una acción con el comportamiento que proporciona –por ejemplo, correr rápido para perseguir a su presa; o Bplans, que sirve para crear jerarquías de comportamientos – por ejemplo, perseguir una presa es una forma de cazar.

Para la indexación en AMBRE se usa un grafo de clasificación de problemas. Este grafo define un conjunto de clases de problemas y relaciones entre ellas. Para cada clase se define un caso prototípico, un problema que representa de manera prototípica a la clase. Además, cada clase contiene también un conjunto de casos que el alumno va a usar para poder resolver el problema que le ha sido propuesto.

Aunque todos los casos que almacena CATO corresponden a casos judiciales sobre secreto industrial, los términos empleados en el texto de las sentencias pueden ser muy diferentes. El uso de técnicas de Recuperación de Información en este contexto ha resultado ser insuficiente, por lo que es necesario realizar una indexación conceptual a través de lo que se denominan *factores*. Un factor es un hecho relevante dentro de un caso judicial que va a permitir guiar las comparaciones entre casos. En un ejemplo extraído de [Brünninghaus'99], el factor F15 (que representa el hecho “Producto único”), se usa para indexar tres casos diferentes y, en cada caso, se corresponde con un texto bien diferente, como se puede ver a continuación:

“Varias características del proceso eran totalmente únicas en la fabricación de transistores” (*de Sperry Rand v. Rothlein*)
 “La información del diagrama no era conocida para el público ni para ninguno de los competidores de Tri-Tron” (*de Tri-Tron v. Velto*)
 “Es obvio que no se podía pedir un Lynchburg Lemonade en ningún establecimiento que no fuera del demandante” (*de Mason v. Jack Daniel Distillery*)

Asociado a cada caso existe una lista de factores aplicables. Éstos se encuentran relacionados con otros factores de conocimiento legal más abstracto formando una estructura llamada Jerarquía de Factores (Figura 1). CATO usa dicha jerarquía para interpretar y razonar sobre los casos, de modo que pueda realizar comparaciones entre casos, identificar el asunto principal de un caso y organizar las argumentaciones con múltiples casos de acuerdo a un asunto.

En TAO ITS, los principios o conceptos tácticos que los futuros TAOs deben conocer constituyen los objetivos formacionales del proceso de entrenamiento. Estos principios se organizan en una jerarquía de principios tácticos, usada para indexar los casos del sistema. La organización jerárquica de los principios es resultado de representar las relaciones de subsunción y de dependencia (por ejemplo, principios que son prerequisites de otros) entre los principios tácticos. Estas últimas, además, se emplean para definir el orden en el que se practican los principios en el proceso de entrenamiento. Los principios pueden llevar asociadas descripciones multimedia que resultan útiles en el proceso de interacción con el usuario. Cada caso incluye una lista con los principios tácticos que pretenden ser cubiertos con la realización de dicho escenario.

Al igual que TAO ITS, JV²M usa los conceptos de la JVM que el alumno ha de aprender para construir una jerarquía que sirve como índice de los casos. Aunque la jerarquía conceptual de JV²M contiene cinco tipos distintos de conceptos sólo dos, las instrucciones de la JVM y los conceptos de compilación, son los que se emplean en la indexación de los ejercicios en la base de casos. Al igual que TAO ITS, la información de ayuda para el usuario está contenida en la jerarquía de conceptos. Asociados a éstos existen descripciones que son usadas por el sistema para proporcionar ayuda al alumno durante la resolución del ejercicio.

3.4. Recuperación de casos

Los métodos de recuperación son los encargados de obtener los casos más interesantes para la situación actual. En esta sección se analizan los fines con los que los casos son recuperados, así como las estrategias de recuperación utilizadas.

Se ha observado que los métodos de recuperación son también dependientes de los objetivos que pretenden cubrir los casos: servir de información de ayuda o representar la tarea a realizar.

- Recuperación cuando se trata de buscar ayuda relevante al alumno de acuerdo con el contexto. Durante la ejecución de la tarea propuesta al alumno, el sistema EBC evalúa la situación actual y revisa la base de casos en busca de algún caso que pueda ser interesante para el alumno. Esta es la estrategia seguida por Creanimate y CATO, aunque cada uno de

ellos implementa sus propios métodos de recuperación, que detallamos a continuación.

Creanimate puede proponer la visualización de un vídeo relacionado con el estado actual del animal que el alumno está creando. Estos vídeos pretenden hacer reflexionar al alumno sobre las características físicas que tienen animales reales semejantes al que está creando y la relación con su comportamiento en la naturaleza. Para realizar esto, durante el ciclo de diálogo de Creanimate se han introducido estrategias de recuperación, reglas que permiten detectar momentos oportunos para la recuperación de un caso. El sistema define tres estrategias fundamentales:

- Recuerdo de ejemplos. Recupera vídeos que ejemplifican los conceptos discutidos actualmente.
- Recuerdo basado en similitud. Recupera vídeos que pueden ayudar al alumno a hacer generalizaciones.
- Recuerdo por violación de expectativas. Recupera vídeos que violan los conceptos generales que están siendo discutidos.

Por su parte, CATO sólo dispone de un método de recuperación basado en la información del caso que el alumno está realizando actualmente. Dado que CATO realiza argumentaciones legales y comparaciones entre casos de manera automática,

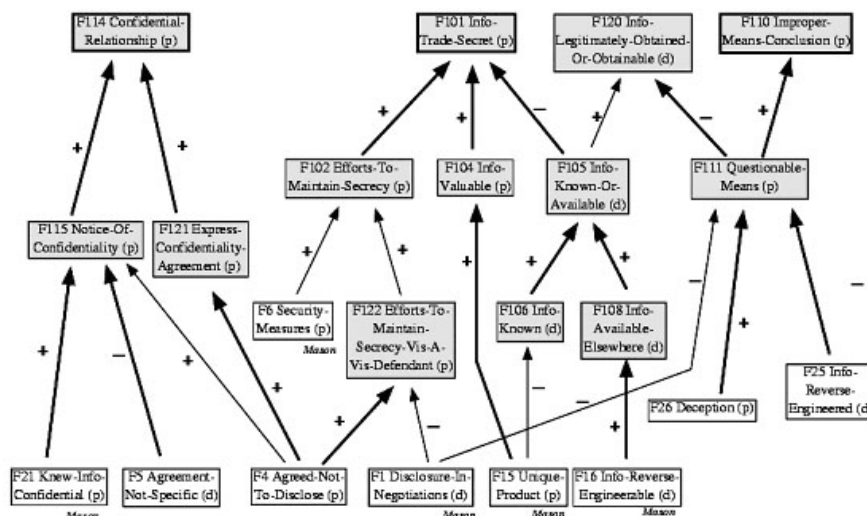


Figura 1. Fragmento de la jerarquía de factores en CATO [Alevén'97a].

es capaz de recuperar todos aquellos casos que puedan serle útiles para la realización de su tarea. Para ello hace uso de un complejo algoritmo de recorrido sobre la base de casos, descrito en [Aleven'97a].

- Recuperación cuando se trata de seleccionar el caso que el alumno ha de resolver. Algunos de los sistemas estudiados sugieren inicialmente al alumno el ejercicio que han de resolver. Esta selección se hace en función de los conocimientos que el alumno tiene actualmente sobre el dominio concreto que el sistema pretende enseñar. Esta información es almacenada en un modelo del alumno. Por tanto, los métodos de recuperación deberán usar esta información para buscar en la base de casos los ejercicios que sean lo suficientemente complejos para el alumno pero que no superen excesivamente sus conocimientos actuales. Tanto TAO ITS como JV²M siguen una estrategia de recuperación de casos muy similar. Los casos son indexados por los conceptos pedagógicos que cubren, por lo que los métodos de recuperación harán uso de la información contenida en el modelo de usuario del alumno para encontrar aquellos casos que más se asemejen a los conceptos que el alumno actual conoce.

Otros sistemas actúan bajo demanda. Por ejemplo, en TAO ITS el alumno puede solicitar al sistema escenarios en los que practique principios en los que aún no ha sido evaluado o que ha fallado recientemente.

Además del desarrollo de estrategias de recuperación automática de casos, hay sistemas que incluyen herramientas que ayudan al alumno a buscar en la base de casos para recuperar aquellos casos que él mismo considere relevantes. CATO dispone de una herramienta que permite al alumno navegar por la base de casos y que le permite recuperar casos. AMBRE proporciona al alumno una serie de mecanismos que le permiten buscar en la base de casos otros problemas similares. Una ventaja del proceso de AMBRE frente a CATO es que el primero va evaluando el proceso de recuperación, por lo que puede informar al alumno sobre la inconveniencia de la recuperación de algunos casos o sugerir otros.

3.5. Evaluación de las acciones del alumno

La evaluación del alumno supone comprobar la validez de las acciones que el mismo está realizando

durante la interacción con el sistema de enseñanza. Es obvio que para aquellos sistemas en los que la tarea no consista en la resolución de un problema, como es el caso de Creanimate, no existirá tarea alguna de evaluación. Para el resto, esta sección analiza cuándo se realiza la evaluación de las acciones, qué medios se emplean para dicha evaluación, y cuándo se informa y se corrige al alumno sobre los errores cometidos.

Lo habitual es que los sistemas realicen la evaluación del alumno mientras éste interactúa con el sistema. Ahora bien, existen diferencias en cuanto a la permisividad del sistema ante los errores del alumno. Algunos de los sistemas son totalmente restrictivos, de modo que reconducen la actividad del alumno en el momento en el que éste realiza acciones no contempladas en la solución correcta. Eso ocurre en AMBRE, que monitoriza todos los pasos por los que va a ir pasando el alumno. AMBRE guía en todo momento la ejecución de la sesión con el alumno. Durante cada una de las fases por las que ha de pasar el alumno para la resolución del problema –reformulación, recuperación, adaptación y almacenamiento– el sistema corrige las decisiones erróneas del alumno, da una explicación sobre el error en que ha incurrido y propone alternativas correctas para que el alumno tome nuevas decisiones.

Los grafos de ejecución de microinstrucciones del sistema JV²M son empleados en la evaluación de las acciones del alumno para alcanzar una solución correcta. En el momento en el que el alumno no realiza una acción contemplada en el grafo de ejecución, se le informa del error y se le plantea cuál es la acción correcta en la situación actual. Pero además, los grafos de ejecución tienen la ventaja de poder incluir la existencia de soluciones alternativas y de caminos erróneos. Estos caminos suelen representar errores que el instructor considera que suelen ser comunes entre los alumnos, de modo que el sistema, aún conociendo que las acciones realizadas son erróneas, permite que el alumno continúe con la resolución del ejercicio. Siguiendo el camino erróneo el alumno llegará a un punto de retorno en el que se le informará precisamente del error que se le ha dejado que cometiera, devolviéndole a la siguiente acción correcta que debiera haber realizado.

CATO es otro de los sistemas que no es permisivo con el alumno cuando éste está haciendo la comparación de dos casos jurídicos. Dados dos casos, el sistema solicita al alumno que seleccione aquellos factores que puedan ser relevantes a favor del demandante o del defensor. Una vez seleccionados, CATO evalúa las respuestas que el

alumno está dando. Si éste omite algún factor o incluye factores erróneos, el sistema le informa sobre su error de manera argumentada.

En contraposición con los anteriores, TAO ITS se comporta de manera totalmente permisiva ante los errores del alumno, aunque no por ello deja la evaluación de lado. Una decisión de diseño tomada en TAO ITS fue permitir al alumno interactuar libremente con el simulador y realizar la evaluación al final de la ejecución del escenario. Esta forma flexible de ejecutar escenarios es posible en TAO ITS gracias al uso de Máquinas de Estado Finitas (MEF) que reconocen si las acciones del alumno ponen en práctica correcta o incorrectamente un principio ligado a un escenario. Cada estado de la MEF representa un estado de conocimiento del alumno en base a su actuación en la ejecución del escenario: principios correctamente aplicados y principios incorrectamente aplicados. Cada transición entre estados tiene asociado un grupo (simple o complejo) de condiciones que deben cumplirse para que dicha transición se efectúe y se pase al estado destino; también se pueden asociar eventos a las transiciones, de modo que sólo se efectúa la transición si el evento ocurre y las condiciones se cumplen. Cuando una transición se efectúa, se supone que el alumno es conocedor de todos los “principios correctamente aplicados” asociados al estado actual y falla en todos los “principios incorrectamente aplicados” ligados a dicho estado. Cada transición puede anotarse con comentarios textuales para ser mostrados al alumno tras la evaluación del escenario. Cuando se finaliza el escenario, TAO ITS genera un informe que resume la actuación del alumno a partir de la información recogida por las MEFs que han evaluado el escenario. Dicho informe señala las situaciones en las que el alumno ha mostrado comprensión de los principios por medio de su correcta aplicación, así como las situaciones en las que el alumno no ha actuado correctamente. Además, estos informes están a disposición del instructor humano, de modo que pueda revisar las actividades y los resultados de la evaluación de los alumnos.

3.6. Herramientas de autoría y mantenimiento

La fiabilidad de la EBC depende del número de casos almacenados, de manera que será más eficaz cuanto mayor sea dicho número [Stottler'01b]. Por esto sería conveniente que los sistemas EBC dispusieran de herramientas que facilitaran la generación y posterior ampliación de la base de casos del sistema. Además, sería interesante que estas herramientas fueran de fácil manejo, ya que

suelen ser los instructores, sin ayuda de los desarrolladores del sistema, los encargados de mantener el sistema EBC. En esta sección se analiza la existencia de estas herramientas para los sistemas estudiados, así como el alcance que tienen en el mantenimiento del sistema.

El mantenimiento de las bases de casos consiste fundamentalmente en la realización de dos tareas: creación e indexación de los casos, y actualización de los índices. No todos los sistemas proporcionan herramientas para las dos tareas. Además, no todas las herramientas facilitan en igual medida ambas tareas.

Creanimate y CATO sólo disponen de herramientas de indexación de casos, ya que la creación de los casos (vídeos y textos sobre sentencias) no corre a cargo del desarrollador y la estructura de indexación permanece fija. En cambio, las herramientas de cada uno de ellos ayudan en distinta medida al encargado de añadir nuevos casos. Mientras que en Creanimate la indexación se realiza manualmente, en CATO se ha automatizado dicha tarea aplicando técnicas de CBR textual [Brünninghaus'01].

Tanto AMBRE como JV²M y TAO ITS presentan herramientas para las dos tareas de mantenimiento. Durante el diseño de los sistemas AMBRE un experto es el encargado de definir el grafo de clasificación de los problemas, así como el conjunto de casos prototípicos que representan cada clase. Pero más adelante es el alumno el encargado, durante la fase de almacenamiento, de añadir nuevos problemas al sistema e incluso de modificar el grafo de clasificación guiado por el sistema.

En JV²M se han desarrollado dos herramientas usadas para el mantenimiento del sistema:

- JaCoMon (JAVy COnccepts and Micro-Instruction). Esta herramienta se usa para la construcción y mantenimiento de la jerarquía conceptual. Permite añadir nuevos conceptos, establecer sus propiedades y relacionarlo con otros conceptos de la jerarquía. También es usada para la creación de los grafos de ejecución ya que éstos contienen relaciones con los conceptos de la jerarquía y es necesario verificar la integridad de los grafos –por ejemplo, un grafo de ejecución no puede incluir microinstrucciones que no estén incluidas en la jerarquía de conceptos.
- JADE (JAVy: Developing tool for Exercises). Esta herramienta se usa para transformar el código Java de los ejercicios propuestos por el instructor en escenarios ejecutables en JV²M. JADE compila el código fuente y

asocia los bloques de código objeto generados con las líneas del código fuente original. Posteriormente es el instructor el encargado de modificar estas relaciones o añadir explicaciones haciendo uso de la interfaz gráfica de esta herramienta.

TAO ITS dispone de la herramienta IITSAT (Internet ITS Authoring Tool) [Stottler'01a, Richards'02] que permite la creación de jerarquías conceptuales y la generación de las MEFs usadas para la evaluación de nuevos escenarios. TAO ITS también cuenta con una herramienta que permite al instructor revisar las actividades de los alumnos, ya que facilita el acceso a los resultados de la evaluación de las ejecuciones de escenarios. También permite modificar los escenarios para corregir las deficiencias que puedan existir en lo que se refiere a su comprensión o incluir nueva información multimedia.

4. Conclusiones

La EBC es una metodología que se basa en:

- Aprendizaje activo. La EBC propone que el alumno aprenda mientras está desempeñando una tarea.
- Aprendizaje basado en ejemplos. La EBC se basa en el uso de casos, experiencias que serán adaptadas para la resolución de nuevos problemas.

En este artículo se ha realizado un análisis de un conjunto representativo de sistemas EBC usados en distintos dominios. Este estudio se ha realizado de acuerdo a los aspectos que han de tenerse en cuenta en el desarrollo de dichos sistemas:

- Interacciones con el alumno. Aquí se incluye tanto el entorno de trabajo como la ayuda proporcionada al alumno. En los sistemas revisados se han detectado hasta tres entornos distintos de trabajo –entornos de diálogo socrático, de resolución de ejercicios y simuladores– e incluso algunos de los sistemas combinan varios de ellos. Consideramos que la bondad del entorno de trabajo es dependiente del dominio del sistema. En cuanto a la ayuda, lo más habitual es proporcionarla durante la ejecución de la tarea. La ayuda proporcionada es muy variada, siendo principalmente razonamiento dialéctico, ejemplos relacionados con la situación actual o material de referencia. Los dos primeros tipos de ayuda tienen una mayor capacidad

pedagógica, ya que obligan al alumno a razonar sobre los errores cometidos.

- Casos. Los casos cubren dos objetivos que no tiene por qué ser disjuntos: representar la información de ayuda proporcionada al alumno, o representar la tarea que el alumno ha de realizar. La ventaja de utilizar los casos para ambos objetivos es que los propios casos que el alumno está resolviendo pueden servir de ayuda para futuras tareas, por lo que se consigue una realimentación del sistema a partir de las experiencias del alumno. En cuanto a su estructura, los casos suelen componerse de una descripción, una solución y alguna otra información adicional. Aquí se ha visto que la información contenida en cada uno de estos apartados puede ser de muy distinta naturaleza.
- Recuperación de casos. Por un lado, la recuperación de casos se realiza para proporcionar ayuda al alumno usando distintas estrategias y reglas de recuperación que se van comprobando durante la realización de la tarea por parte del alumno. Por otro lado, la recuperación se realiza para seleccionar la tarea siguiente que el alumno ha de ejecutar. Para este objetivo, los métodos de recuperación han de relacionar la información sobre los conocimientos del alumno con la información del conocimiento que cubre cada caso.
- Indexación de casos. Este tal vez sea el aspecto más heterogéneo de todos los estudiados. Se ha llegado a la conclusión de que la indexación está estrechamente relacionada con el dominio del sistema EBC y con el fin que persigue la recuperación de los casos. Especialmente interesante resulta la indexación por conceptos pedagógicos, ya que permite la recuperación de los casos basándose en los conocimientos que el alumno tiene sobre la disciplina que enseña el sistema EBC.
- Evaluación de las acciones. Los sistemas EBC suelen realizar la evaluación de las acciones del alumno durante la ejecución de la tarea. Una vez que se ha detectado un error del alumno existe un problema que se ha de tener en cuenta a la hora de tomar decisiones de diseño en un sistema EBC: la permisividad del sistema ante los errores. Es necesario llegar a un término medio en cuanto a la corrección al alumno durante la interacción con el sistema ya que, aunque un sistema muy permisivo fomenta que el alumno sea dueño de su propio proceso de aprendizaje,

con las ventajas que ello conlleva, demasiada libertad hace que el sistema EBC pueda llevar a la frustración del alumno si éste no recibe la ayuda necesaria cuando no sabe cómo continuar en la tarea propuesta.

- Herramientas de autoría y mantenimiento. Ya que la EBC basa su fiabilidad en el número de casos del sistema, consideramos necesarias las herramientas que faciliten la tarea de creación de casos y de mantenimiento de los ya existentes. La ayuda proporcionada por estas herramientas es muy variada. Las más beneficiosas para la continuidad y uso del sistema no se centran exclusivamente en los casos sino que van más allá, permitiendo el mantenimiento del sistema completo.

Por último, destacar que en esta revisión no se ha incluido un elemento importante en los actuales sistemas de enseñanza basada en computador: el modelo de usuario. A pesar de la importancia de este elemento para proporcionar al alumno una enseñanza individualizada, la mayoría de los sistemas estudiados no contemplan el uso de modelos de usuario o, si lo poseen, no le conceden especial relevancia, como es el caso de los sistemas AMBRE. Cabe destacar el uso que se le da al modelo de usuario en los sistemas JV²M y TAO ITS, en los que éste guarda los conceptos de dominio que son conocidos por el usuario y que son usados para recuperar el caso que mejor corresponde con el conocimiento actual de cada alumno.

Agradecimientos

Este trabajo ha sido posible gracias a la financiación del Ministerio de Ciencia y Tecnología (TIC2002-01961).

Referencias

- [Aleven'97a] V. Aleven, "Teaching case-based argumentation through a model and examples", Tesis Doctoral Intelligent Systems Program and Learning Research and Development Center University of Pittsburgh, 1997.
- [Aleven'97b] V. Aleven and K. D. Ashley, "Teaching case-based argumentation through a model and examples. Empirical evaluation of an intelligent learning environment." Procs. 8th World Conference on Artificial Intelligence in Education, 1997, pp. 87-94.
- [Ashley'91] K. D. Ashley, "Reasoning with Cases and Hypotheticals in HYPO", International Journal of Man-Machines Studies, vol. 34, pp. 753-796, 1991.
- [Ashley'02] K. D. Ashley, R. Desai, and J. M. Levine, "Teaching Case-Based Argumentation Concepts Using Dialectic Arguments vs. Didactic Explanations", Procs. 6th International Conference on Intelligent Tutoring Systems, 2002, pp. 585-595.
- [Brünninghaus'99] S. Brünninghaus and K. D. Ashley, "Bootstrapping Case Base Development with Annotated Case Summaries", Procs. Third International Conference on Case-Based Reasoning, 1999, pp. 59-73.
- [Brünninghaus'01] S. Brünninghaus and K. D. Ashley, "The Role of Information Extraction for Textual CBR", Procs. 4th International Conference on Case-Based Reasoning, 2001, pp. 74-89.
- [Edelson'92] D. C. Edelson, "When Should a Cheetah Remind You of a Bat? Reminding in Case-Based Teaching." Procs. 10th National Conference on Artificial Intelligence, 1992, pp. 667-672.
- [Edelson'93] D. C. Edelson, "Learning from stories: indexing and reminding in a Socratic case-based teaching system for elementary school biology", Northwestern University, Informe Técnico 43, 1993.
- [Edelson'96] D. C. Edelson, "Learning from questions and cases: The Socratic case-based teaching architecture", The Journal of the Learning Sciences, vol. 5, pp. 357-410, 1996.
- [Gómez-Martín'02] M. A. Gómez-Martín, P. P. Gómez-Martín, and P. A. González Calero, "JAVY: Agente Pedagógico para enseñar la estructura de la JVM", Departamento de Sistemas Informáticos y Programación, Universidad Complutense de Madrid, Informe Técnico 129-02, 2002.
- [Gómez-Martín'03] P. P. Gómez-Martín, M. A. Gómez-Martín, and P. A. González Calero, "Javy: Virtual Environment for Case-Based Teaching of Java Virtual Machine", Procs. 7th International Conference on Knowledge-Based Intelligent Information & Engineering Systems, 2003, pp. 906-913.
- [Guin-Duclosson'02] N. Guin-Duclosson, S. Jean-Daubias, and S. Nogry, "The AMBRE ILE: How to Use Case-Based Reasoning to Teach Methods", Procs. 6th International Conference on Intelligent Tutoring Systems, 2002, pp. 782-791.

- [Kolodner'93] J. L. Kolodner, Case-based reasoning. San Mateo, CA: Morgan Kaufmann Publishers, 1993.
- [Kolodner'97] J. L. Kolodner, "Educational Implications of Analogy. A View from Case-Based Reasoning", American Psychologist, vol. 52, pp. 57-66, 1997.
- [Lester'97] J. C. Lester, S. A. Converse, S. E. Kahler, S. T. Barlow, B. A. Stone, and R. S. Bhoga, "The Persona Effect: Affective Impact of Animated Pedagogical Agents", Procs. Conference on Human Factors in Computing Systems, 1997, pp. 359-366.
- [Lindholm'99] T. Lindholm and F. Yellin, The Java(TM) Virtual Machine Specification (2nd Edition). Oxford: Addison-Wesley Professional, 1999.
- [Nogry'04] S. Nogry, S. Jean-Daubias, and N. Duclosson, "ITS Evaluation in Classroom: The Case of Ambre-AWP", Procs. 7th International Conference on Intelligent Tutoring Systems, 2004, pp. 511-520.
- [Richards'02] R. A. Richards, "Principle Hierarchy Based Intelligent Tutoring System for Common Cockpit Helicopter Training", Procs. 6th International Conference on Intelligent Tutoring Systems, 2002, pp. 473-483.
- [Schank'90] R. C. Schank, "Case-Based Teaching: Four experiences in education software design" in Interactive Learning Environments, vol. 1, 1990, pp. 231-253.
- [Schank'95a] R. C. Schank, Tell Me a Story. Narrative and Intelligence: Northwestern University Press, 1995.
- [Schank'95b] R. C. Schank and C. Cleary, Engines for education. Hillsdale, NJ: L. Erlbaum Associates Publishers, 1995.
- [Schank'95c] R. C. Schank, M. Korcuska, and M. Jona, "Multimedia applications for education and training: revolution or red herring?" ACM Computing Surveys, vol. 27, pp. 633-635, 1995.
- [Stottler'99] R. Stottler and S. Ramachandran, "A Case-Based Reasoning Approach to Intelligent Tutoring Systems (ITS) and ITS Authoring", Procs. The 12th International Florida Artificial Intelligence Research Symposium, 1999, pp. 181-186.
- [Stottler'00] R. Stottler and M. Vinkavich, "Tactical Action Officer Intelligent Tutoring System (TAO ITS)", Procs. Industry/Interservice, Training, Simulation & Education Conference, 2000.
- [Stottler'01a] R. Stottler, D. Fu, S. Ramachandran, and T. Jackson, "Applying a Generic Intelligent Tutoring System Authoring Tool to Specific Military Domains", Procs. Industry/Interservice, Training, Simulation & Education Conference, 2001.
- [Stottler'01b] R. Stottler and N. Harmon, "Transitioning an ITS Developed for Schoolhouse Use to the Fleet: TAO ITS, a Case Study", Procs. Industry/Interservice, Training, Simulation & Education Conference, 2001.