

Utilización de estándares en el aprendizaje virtual

Citation for published version (APA):

Burgos, D., Tattersall, C., & Koper, R. (2005). *Utilización de estándares en el aprendizaje virtual*.

Document status and date:

Published: 01/05/2005

Document Version:

Peer reviewed version

Document license:

CC BY-NC

Please check the document version of this publication:

- A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

<https://www.ou.nl/taverne-agreement>

Take down policy

If you believe that this document breaches copyright please contact us at:

pure-support@ou.nl

providing details and we will investigate your claim.

Downloaded from <https://research.ou.nl/> on date: 08 Sep. 2024

Open Universiteit
www.ou.nl



UTILIZACIÓN DE ESTÁNDARES EN EL APRENDIZAJE VIRTUAL

Funcionalidades didácticas de la especificación IMS Learning Design

Daniel Burgos, Colin Tattersall y Rob Koper
Educational Technology Expertise Center (OTEC)
Open University of the Netherlands
Valkenburgerweg 177; PO Box 2960
6401 DL Heerlen; The Netherlands
email: daniel.burgos@ou.nl

Resumen

El aprendizaje virtual está caracterizado entre otras cosas por una adaptación de la metodología y de los contenidos a la capacidad de la herramienta o entorno donde se implementan. Esto conlleva una dependencia unilateral del estudiante y del profesor hacia la plataforma que obliga a suscribirse a una cierta opción tecnológica, con sus limitaciones, actualizaciones y circunstancias de supervivencia. La incorporación de estándares y especificaciones al mercado del aprendizaje-enseñanza en línea facilita la independencia del recurso frente a la metodología didáctica, así como de las unidades de aprendizaje frente a la aplicación que las edita o las ejecuta.

Mostramos en estas líneas la fundamentación y razonamiento de este hecho y utilizamos ejemplos concretos de funcionalidades didácticas desarrolladas con la especificación europea IMS Learning Design que muestran sus posibilidades para abordar la definición de escenarios de aprendizaje independientemente de la pedagogía utilizada para confeccionarlos. Todos ellos han sido creados en la Open University of The Netherlands, dentro del proyecto europeo UNFOLD, Understanding New Frameworks of Learning Design.

Palabras Clave: IMS Learning Design, aprendizaje virtual, especificación, estándar, código abierto, UNFOLD

1. INTRODUCCIÓN

¿Realmente necesitamos especificaciones y estándares en el aprendizaje virtual? Si realizamos un estudio rápido del panorama actual (Mayo, 2005) podemos comprobar que existen decenas de LMS (sistemas de gestión de aprendizaje) y de VLE (entornos de aprendizaje virtual). Fácilmente encontraremos uno que se adapte a nuestras necesidades, dentro de un amplio rango económico. Ahora bien, ¿cuántos entre ellos son aplicaciones de código abierto o generan ficheros en código abierto

con intención de interoperabilidad? Solamente algunos, empezando por el bien conocido Moodle (Dougiamas, 2004), un CMS (sistema gestor de cursos) basado en diseño instructivo con facilidades y servicios para aprendizaje colaborativo, pasando por Reload (University of Bolton, 2005), editor de Scorm y ciertas especificaciones IMS, y siguiendo con Edubox (OUNL y Perot Systems, 2004), basado en código EML y utilizado actualmente como principal plataforma de enseñanza en esta universidad.

Pero ¿con cuántos podemos intercambiar contenidos y estructuras de aprendizaje? La respuesta es sencilla. Ninguno.

2. POR QUÉ NECESITAMOS ESTÁNDARES EN EL APRENDIZAJE VIRTUAL

Pongamos un ejemplo práctico. Imaginemos que somos profesores que queremos incorporar facilidades de aprendizaje virtual a nuestras clases, o que somos diseñadores de aprendizaje que buscamos definir metodologías utilizando internet como plataforma o, finalmente, que somos proveedores de contenidos que construimos cursos y unidades de aprendizaje para ser ejecutadas en línea. Para resumir: e-profesores, e-didactas y e-escritores. O, al menos, de una manera potencial. Quizá realizamos las mismas tareas pero todavía no online. Si la institución donde trabajamos, ya sea una universidad, una empresa privada o pública, un colegio, un instituto o cualquier otra de las muchas posibilidades donde desempeñamos una de estas funciones (o varias al tiempo, porque la línea entre ellas es muchas veces muy difusa) adquiere una aplicación comercial o una de código abierto con el objeto de introducir una plataforma virtual en el proceso de enseñanza, tenemos dos problemas principalmente. Primero, debemos aprender la aplicación. Cómo instalarla, utilizarla y rentabilizarla lo máximo posible. Segundo, debemos crear el material didáctico y construir la estructura pedagógica adaptada a la nueva aplicación. Quizás horas, días o meses de trabajo, según nuestra capacidad y el tamaño del proyecto.

¿Y qué ocurre si pasado un tiempo, después de haber realizado una buena adaptación de nuestro material para ser ejecutado en línea, la aplicación informática cambia? Puede que por una drástica actualización del programa, puede que sea sustituida por otra aplicación, puede que sea porque cambiamos de empleo o de institución y en nuestra nueva plaza hay otro sistema. ¿Qué ocurre entonces con todo el material ya transferido del papel al ordenador? Que debemos rehacerlo, en el mejor de los casos, y volver a migrarlo.

En realidad, no deberíamos empezar desde cero. Es de suponer que cierto material sigue siendo válido y que podemos reutilizarlo. Pero si no podemos ejecutar las lecciones ya construidas para una aplicación informática determinada en otra distinta es seguro que tenemos que comenzar de nuevo y volver a realizar una nueva adaptación del material en bruto originario o del material electrónico generado para la versión anterior. Por lo tanto, más trabajo, más esfuerzo y más tiempo.

2.1. PARA QUÉ SIRVE UNA ESPECIFICACIÓN

Una especificación sobre aprendizaje virtual permite escribir y modelar algo, trabajar con ello y mantener el nuevo material funcionando exactamente igual, independientemente de la plataforma que se utilice (Burgos et al, 2005). Es decir, puede ser migrado automáticamente y el contenido y la estructura del curso son independientes de la plataforma de ejecución. Además, se escribe en código abierto (interpretable y modificable, por tanto) y es gratis. Así de sencillo. Es cierto que el primer paso que anunciábamos en el apartado anterior sobre migración del papel al ordenador y adquisición de la habilidad para hacerlo satisfactoriamente no se puede obviar. Pero también es cierto que todo lo creado permanece y puede ser utilizado en cualquier plataforma que cumpla con el estándar. Esta plataforma podrá ser open-source, comercial, share-ware o free-ware, pero nuestro trabajo podrá ser ejecutado y reutilizado en cualquiera de ellas.

3. UN CASO PRÁCTICO GENÉRICO

Expliquémoslo con un ejemplo práctico sobre generación de una Unidad de Aprendizaje (Figura 1). Partamos del supuesto que un profesor quiere migrar sus planificaciones de aula y contenidos a un sistema online. Para ello, crea contenidos en TXT, PDF, PPT, DOC, AVI, XLS, HTML, RTF, SXW y otros formatos de fichero, además de añadir enlaces web. El profesor puede incorporar tantos recursos como desee y puede posteriormente crear documentos de trabajo que enlacen estos recursos. Por último, también puede añadir evaluaciones e información

adicional sobre objetivos y pre-requisitos, por ejemplo.

Para realizar todo esto y hacer que funcione adecuadamente el profesor necesita algo de tiempo modelándolo y montándolo. Puede decidir crear la estructura basándose en páginas HTML que puedan ser visualizadas en un navegador web (como Internet Explorer, Opera, Mozilla o Navigator). Esta solución es útil si se quieren mostrar documentos y si únicamente se pretende que los usuarios consulten el “curso” de manera no guiada. Si, por el contrario, se decide incorporar cierto grado de control sobre la metodología, la evaluación, la retroalimentación, el flujo de aprendizaje, las propiedades o el proceso de matriculación, por citar unas cuantas posibilidades, es necesario insertar los contenidos en un sistema que permita hacer todo esto. Un LMS (sistema de gestión de aprendizaje) o un VLE (entorno de aprendizaje virtual) son las mejores opciones. De esta manera, el profesor dispondrá de contenidos y de método, es decir, el recurso de los materiales pasados de papel a ordenador y las funcionalidades adicionales ya comentadas. Además, puede añadir foros, charlas en línea, servicios de comunicación; puede mostrar u ocultar información dependiendo del perfil del usuario o del grado de destreza adquirido, etcétera.

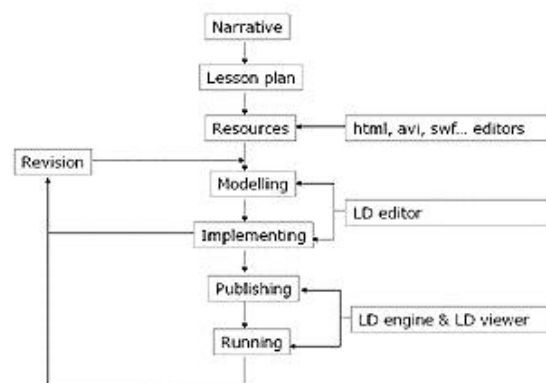


Figura 1: Ciclo de creación de una Unidad de Aprendizaje

De esta manera, contamos tres cosas: contenidos/recursos, actividades/funcionalidades complementando los contenidos/recursos y método/pedagogía, que engloba, justifica y sustenta todo lo demás. Lo primero es el producto no manufacturado, que puede ser creado con cientos de diferentes aplicaciones y que puede ser utilizado donde se necesite. Lo segundo es un conjunto de servicios que apoyan o complementan los contenidos y la metodología y depende del sistema informático. Lo último, el método, la pedagogía, consiste en el enfoque didáctico y está completamente ligado a la herramienta que hayamos usado para modelar la planificación de aula o lección. Si mantenemos estas dos últimas partes tan independientes de la plataforma como la primera, no necesitaremos reconstruirla cuando modifiquemos los contenidos, o

los cambios de plataforma o se actualice la versión de la aplicación. La pedagogía estará enlazada a los recursos pero será independiente de la plataforma. Se puede cambiar el contenido del recurso pero su enlace permanecerá y la estructura entera funcionará igual.

Y aquí es donde entran en juego las especificaciones y los estándares.

4. ESPECIFICACIÓN Y ESTÁNDAR

Un estándar es una tecnología, formato o método, reconocido, nacional o internacionalmente, documentado en detalle y ratificado por una autoridad respectada de su campo, como ISO (International Standards Organisation), BSI (British Standards Institute) o IEEE. Por el contrario, una especificación es el paso previo, creado por alguna compañía u organismo, que no ha sido ratificado todavía por ninguna autoridad, y que suele utilizarse de manera provisional pero suficientemente respaldada.

Mientras que la especificación existe como tal es plenamente operativa y se produce mucha actividad corporativa e investigadora para identificar necesidades y carencias, para realizar re-revisiones y para depurarla lo más posible hasta que se obtiene el estándar (Cetis, 2005).

Así pues, el proceso completo consiste en detectar una necesidad en un entorno profesional, crear una especificación para abordarla y, finalmente y si es necesario, estandarizarla.

5. LA ESPECIFICACIÓN IMS LEARNING DESIGN

En el año 2003, la especificación IMS Learning Design (IMS LD, de ahora en adelante) fue publicada por IMS Global Learning Consortium Inc. Con un amplio bagaje en especificaciones (Content Packaging, Enterprise, Question and Test Interoperability, etc.) IMS apostó por una especificación centrada en el proceso de aprendizaje y no tanto en los contenidos finales, intentando asegurar la interoperabilidad de los módulos o Unidades de Aprendizaje (UoL) generados con ella. Mediante IMS LD el profesor, pedagogo o escritor puede representar un escenario de aprendizaje sustentando en cualquier pedagogía, ya que es completamente neutro en este aspecto (Tattersall et al, 2003). Es, por tanto, pedagógicamente flexible, lo que implica además una ejecución multiplataforma, independientemente del visualizador LD utilizado.

En IMS LD un usuario ejecuta un rol (profesor, tutor, alumno, supervisor, oyente...) dentro del proceso de aprendizaje-enseñanza y sigue una serie de actividades de aprendizaje para completar los

distintos momentos o actos de dicho proceso (Burgos y Berbegal, 2005). Un escenario de aprendizaje completo viene definido por los recursos, la metodología, las actividades de aprendizaje y soporte, los servicios complementarios, la relación entre los roles, la agrupación de usuarios y un largo etcétera. Todo ello se define en un fichero llamado manifiesto, escrito en XML, que junto con los recursos es empaquetado en un fichero en formato ZIP, consiguiendo la Unidad de Aprendizaje (UoL). Este fichero comprimido podrá ser ejecutado, abierto o modificado en cualquier sistema compatible LD.

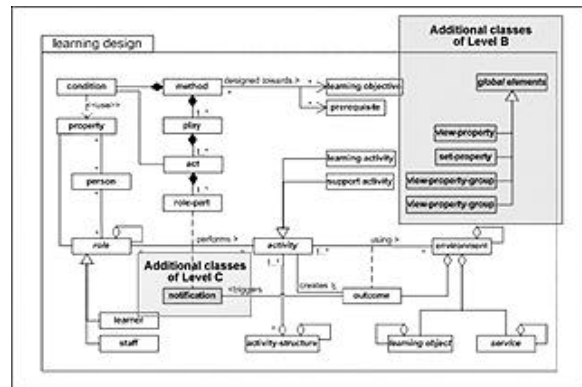


Figura 2: Niveles de implementación en IMS LD

Existen tres niveles de implementación (Figura 2) que definen la estructura conceptual de IMS LD (Koper y Tattersall, 2005^a):

- Nivel A, que constituye la base y comprende la definición de usuarios, actividades de aprendizaje, actividades de soporte, entornos, recursos, método, ejecuciones (plays), actos, roles y la coordinación entre todos ellos, es decir, la expresividad pedagógica. Además los usuarios podrán utilizar recursos externos, enlaces web y diversos servicios (foros, chats...).
- Nivel B. Añade al Nivel A la utilización de propiedades, condiciones, servicios de monitorización y elementos globales para gestión de la especificación desde ficheros externos a la misma. Constituye el nivel que aporta más flexibilidad a la hora de la representación didáctica ya que permite esconder y mostrar elementos, condicionar el flujo de aprendizaje, almacenar datos del usuario y la instancia, bien a nivel local y personal, bien a nivel global y compartido. De esta manera podemos incorporar cuestionarios, evaluaciones, cálculos numéricos, seguimiento de usuarios y un largo etcétera en el que profundizamos en el siguiente apartado.
- Nivel C. Añade al Nivel B la utilización de notificaciones, es decir, de mecanismos de lanzamiento y ejecución automática de procesos según el cumplimiento de ciertas

condiciones o la ejecución de ciertas acciones por parte de alguno de los usuarios implicados.

En su conjunto, IMS LD aporta una flexibilidad pedagógica y una serie de recursos que posibilitan la implementación de la casi totalidad de escenarios de aprendizaje, incluyendo modelos mixtos distancia-presencia (b-learning). En la actualidad, existen diversas iniciativas para diseminar y apoyar la especificación (UNFOLD Project, Learning Networks for Learning Design, Learning Networks) así como motores (CopperCore), Editores (CopperAuthor, Reload LD Editor, ASK LDT, Cosmos) y visualizadores (CopperCore Player, Reload LD Viewer, Alphanet Project). (Recursos LD, 2005).

6. APLICACIONES CONCRETAS DE IMS LD EN EL APRENDIZAJE VIRTUAL

Todas las facilidades que proporciona el Nivel B en cuanto a propiedades, condiciones, elementos globales y servicio de monitorización, pueden ser utilizadas en una amplia gama de aplicaciones (Koper y Burgos, 2005). A continuación describimos las más frecuentes desde un punto de vista didáctico: aprendizaje activo, aprendizaje colaborativo, aprendizaje adaptativo, personalización, retroalimentación dinámica, seguimiento en tiempo de ejecución, nuevas alternativas de evaluación y e-Portfolios. Aunque el código aportado refleja el flujo y la estructura correctamente hemos optado por escribir nuestro propio pseudo-código fuente con vistas a una mayor claridad. Por otro lado, para el desarrollo y prueba de los ejemplos, hemos utilizado el motor CopperCore (Vogten y Martens, 2005), único en la actualidad capaz de ejecutar satisfactoriamente todos los niveles de la especificación.

6.1. APRENDIZAJE COLABORATIVO Y APRENDIZAJE ACTIVO

El aprendizaje colaborativo consiste (Cole y Engestrom, 1993) en la compartición de información alumno-alumno y alumno-profesor con el fin de afianzar conocimientos, criticar puntos de vista, aportar nuevas ideas a la luz del trabajo de otros o introducir conceptos para discusión y elaboración grupal, por ejemplo. Esto implica que la misma información puede ser vista por distintas personas y que se produce un flujo constante de intercambio de datos que cada cual utilizará según sus objetivos personales dentro del grupo y del curso.

Una segunda posibilidad, el aprendizaje activo, es la monitorización del progreso del estudiante por parte del tutor, analizando dinámicamente las aportaciones

que este vaya realizando durante el curso y proporcionando un feedback al mismo. De esta manera, se establece también un flujo bilateral de información entre alumno y tutor orientado al aprovechamiento académico y personalizado.

El Nivel B proporciona el objeto 'monitor' que permite visualizar propiedades propias o ajenas de una manera estructurada. Estas propiedades deben ser definidas con anterioridad, e inicializadas para tipos numéricos, y pueden ser operadas mediante, como veremos más adelante. Fijémonos en dos ejemplos sobre definición e inicialización:

```
<loc-property identifier="LP-LA-1-completed">
  <title>Respuesta al cuestionario inicial</title>
  <datatype="boolean"/>
  <initial-value>true</initial-value>
</loc-property>
```

La propiedad 'LP-LA-1-completed', cuyo título o etiqueta es 'Respuesta al cuestionario inicial', es de tipo lógico y su valor inicial es TRUE. El uso de esta variable será almacenar el estado terminado o no de una actividad de aprendizaje .

```
<locpers-property identifier="LP-personalgoals">
  <title>Qué objetivos tienes para el curso?</title>
  <datatype="text"/>
</locpers-property>
```

La propiedad 'LP-personalgoals', cuyo título o etiqueta es 'Qué objetivos tienes para este curso?', es de tipo texto y no se inicializa con ningún valor. El uso de esta variable será que el usuario introduzca comentarios y, por tanto, se parte en blanco. Es una propiedad local y depende de cada usuario.

Tomando esta última variable 'LP-personalgoals' puede ser leída y grabada dentro del flujo normal del curso:

```
<Id:set-property ref="LP-personalgoals"/>
```

nos permite escribir ('set') el contenido de la variable.

```
<Id:view-property ref="LP-personalgoals" property-of="self"/>
```

nos permite leer ('view') el valor ('value') de nuestra propia variable ('self').

Pero también puede ser monitorizada mediante el objeto 'monitor' que puede recoger esta variable para realizar su seguimiento. Por ejemplo, la siguiente línea nos permite leer ('view') la variable de otro estudiante ('supported-person'):

```
<Id:view-property property-of="supported-person" ref="LP-personalgoals"/>
```

Para realizar esta acción, primero deberemos configurar el objeto ‘monitor’ dentro del entorno de trabajo (en este caso concreto):

```
<environment identifier="E-personalgoals">
  <title>Qué objetivos tienen los demás?</title>
  <service identifier="S-personalgoals">
    <monitor>
      <role-ref ref="Student"/>
      <title>Metas de los demás alumnos</title>
      <item identifierref="R-personalgoals"/>
    </monitor>
  </service>
</environment>
```

Como podemos ver, se define el servicio de monitor para el estudiante (‘Student’). Esto permite que cada estudiante pueda visualizar el contenido de las variables de sus compañeros. Para la visualización por parte del tutor se construiría una estructura similar, facilitando un seguimiento de los participantes en el curso.

6.2. APRENDIZAJE ADAPTATIVO

Para ilustrar esta sección nos basaremos en el ejemplo ‘Learning to listen to Jazz’ (Tattersall y Burgos, 2005) desarrollado originalmente en EML y adaptado a IMS LD. En él un alumno/a puede seguir un curso y elegir dos itinerarios distintos, temático e histórico en función de sus preferencias. En el ejemplo también se realizan funciones de monitorización como las descritas en el apartado anterior.

El aprendizaje adaptativo busca (Shuell, 1999), partiendo de una base de contenidos comunes, elegirlos y presentarlos al estudiante, en función de su perfil inicial y de los resultados progresivos que vaya proporcionando durante el desarrollo del curso. La personalización complementa el aprendizaje adaptativo proveyendo a cada estudiante la capacidad de elegir ciertas características del contenido, de la presentación, de la evaluación y del itinerario, dentro de un marco prefijado de aprendizaje.

En referencia al aprendizaje adaptativo, en el ejemplo referenciado, Jazz, se define la propiedad ‘LP-choose-itinerary’ para saber si el usuario ha elegido o no el itinerario. Cada itinerario se encuentra descrito en una estructura de actividad, ‘AS-historic’ y ‘AS-thematic’, definidas previamente y que no detallamos aquí. Todo el proceso se detalla como un conjunto de condiciones. En caso de que todavía no se haya elegido nada, y la variable no tenga ningún valor aún, se esconden las dos Activity Structures:

```
<conditions>
  <if>
    <no-value>
      <property-ref ref="LP-choose-itinerary"/>
    </no-value>
  </if>
```

```
<then>
  <hide>
    <activity-structure-ref ref="AS-thematic"/>
    <activity-structure-ref ref="AS-historic"/>
  </hide>
</then>
```

El estudiante puede elegir las dos opciones ‘historic’ y ‘thematic’ presentes en una combobox. Si el usuario elige la opción ‘thematic’, la estructura ‘AS-thematic’ se muestra y la ‘AS-historic’ se oculta. Lo mismo en el caso contrario:

```
<if>
  <is>
    <property-ref ref="LP-choose-itinerary"/>
    <property-value>thematic</property-value>
  </is>
</if>
<then>
  <show>
    <activity-structure-ref ref="AS-thematic"/>
  </show>
  <hide>
    <activity-structure-ref ref="AS-historic"/>
  </hide>
</then>
```

Las diferentes estructuras pueden contener distintos contenidos o los mismos contenidos reorganizados de diferentes maneras para dos enfoques distintos y todo estar gestionado dentro del propio manifiesto de la Unidad de Aprendizaje.

6.3. PERSONALIZACION

Por otro lado, un caso sencillo de personalización es la incorporación de los datos personales del alumno/a en el desarrollo del curso. Ya hemos visto en el apartado anterior cómo definir y rellenar una propiedad. Otra posibilidad consiste en utilizar un grupo de propiedades para tener una operativa más sencilla. Por ejemplo:

a) definiendo una propiedad de tipo ‘string’ y una propiedad de tipo ‘integer’, e inicializando esta última a cero

```
<locpers-property identifier="LP-name">
  <title>your name</title>
  <datatype datatype="string"/>
</locpers-property>
<locpers-property identifier="LP-age">
  <title>age</title>
  <datatype datatype="integer"/>
  <initial-value>0</initial-value>
</locpers-property>
```

b) agrupando las dos propiedades

```
<property-group identifier="LP-group-personalinfo">
  <title>personal details</title>
  <property-ref ref="LP-name"/>
  <property-ref ref="LP-age"/>
</property-group>
```

```
</property-group>
```

c) solicitando al usuario los datos ('set') y visualizándolos ('view'), según necesidad

```
<ld:set-property-group ref="LP-group-personalinfo"
property-of="self"/>
```

```
<ld:view-property-group ref="LP-personalgoals"
property-of="self"/>
```

6.4. RETROALIMENTACIÓN DINÁMICA

Para ilustrar esta sección utilizaremos el ejemplo 'GeoQuiz' (Burgos, 2005) en el que el alumno/a contesta un cuestionario de cinco preguntas que devuelve información adicional según las respuestas elegidas y que proporciona una valoración numérica final del ejercicio de evaluación del estudiante.

Como aplicación concreta del aprendizaje adaptativo, IMS LD también permite trabajar con retroalimentación dinámica, proporcionando contenidos adaptados a la evolución del estudiante dentro de cada Learning Activity y lectura de resultados en función de la actividad del mismo.

En referencia a la aportación de contenidos en función de la evolución ya hemos visto en el ejemplo anterior que una posibilidad dentro del propio manifiesto es la utilización de diferentes estructuras con contenidos iguales o distintos y distinta reorganización dentro del manifiesto. En definitiva, es ocultar o mostrar una actividad o una estructura, que al final estarán enlazadas con recursos apuntando a ficheros externos. Otra posibilidad consiste en la utilización de clases para modificar la visualización del propio contenido dentro de un fichero apuntado por un recurso, es decir, fuera del manifiesto. Estas clases hacen referencia a capas XHTML definidas en ficheros XML aparte de ims manifest.xml.

Aunque el mecanismo que se puede utilizar para visualizar y ocultar las capas es similar al visto en el apartado anterior para Activity Structures, la acción apuntada se realiza en otro fichero y se ejecuta, por tanto, de otra manera:

```
<if>
<is>
  <property-ref ref="Answer1"/>
  <property-value>C</property-value>
</is>
</if>
<then>
<hide>
  < class="Feedback_Wrong"/>
</hide>
<show>
  <class="Feedback _Right"/>
</show>
</then>
```

En el ejemplo anterior se muestra la clase 'Feedback_Right' si el contenido de la respuesta 'Answer1' es 'C' (que se elige de una lista enumerada), ocultando la clase 'Feedback_Wrong', que contiene otro contenido distinto. En el fichero externo, se establece el valor de la variable de respuesta 'Answer1' y se definen tanto las clases como el contenido de las mismas:

a) se elige la respuesta ('set')

```
<p>Your answer is:
<set-property ref="Answer1" of="self"/></p>
```

b) se definen las clases

```
<div class="Answer1_Wrong">
  <p><view-property ref="Answer1"/> is not
  right</p>
</div>
<div class="Answer1_Right">
  <p>Congratulations!</p>
  
</div>
```

6.5. SEGUIMIENTO EN TIEMPO DE EJECUCIÓN

En referencia a la lectura de resultados en función de la actividad del usuario, podemos realizar operaciones matemáticas con los valores almacenados en el transcurso del estudio y proveer un feedback contextual de los mismos, lo que constituye un seguimiento de resultados en tiempo de ejecución. Si imaginamos dos preguntas con dos respuestas y dos valores posibles para cada respuesta, 0 y 100, podríamos:

a) definir cada propiedad

```
<locpers-property identifier="QuestionTrue1">
  <datatype datatype="integer"/>
  <initial-value>0</initial-value>
</locpers-property>
```

b) asignarle un valor en función de la respuesta

```
<if>
  <is>
    <property-ref ref="Answer1"/>
    <property-value>C</property-value>
  </is>
</if>
<then>
  <change-property -value>
    <property-ref ref="QuestionTrue1"/>
    <property-value>100</property-value>
  </change-property -value>
</then>
<else>
  <change-property -value>
    <property-ref ref="QuestionTrue1"/>
    <property-value>0</property-value>
  </change-property -value>
```

```
</else>
```

c) y, finalmente, realizar la media aritmética de las dos respuestas

```
<change-property-value>
  <property-ref ref="sum"/>
  <property-value>
    <calculate>
      <divide>
        <sum>
          <property-ref ref="QuestionTrue1"/>
          <property-ref ref="QuestionTrue2"/>
        </sum>
        <property-value>2</property-value>
      </divide>
    </calculate>
  </property-value>
</change-property-value>
```

6.6. FORMAS ALTERNATIVAS DE EVALUACIÓN

La evaluación tradicional se basa en la confrontación de los conocimientos de un usuario frente a la máquina o frente a los conocimientos del profesor. Pero no es la única. Las nuevas formas de aprender y de enseñar deben ir de la mano de nuevas formas de evaluar. La evaluación debe concebirse dentro del flujo de aprendizaje y no como un recurso aislado a) comprobando el nivel de conocimiento de un estudiante antes de decidir el mejor itinerario, b) comprobando si un concepto ha quedado fijado antes de avanzar hacia el siguiente y c) proporcionando feedback de alta calidad para mantener una motivación alta. También deben contemplarse d) los escenarios compuestos por varios estudiantes compartiendo, por ejemplo, respuestas individuales con el grupo para una evaluación y un debate colectivos (Koper y Tattersall, 2005). Formalizar cualquiera de estas propuestas en LD no es difícil, aunque sí laborioso, si utilizamos los apuntes descritos hasta ahora. Las cuatro posibilidades pueden ser implementadas como combinación de condiciones y propiedades.

Del mismo modo, como ya vimos en el apartado anterior, la formalización de cuestionarios de evaluación es posible dentro de IMS LD, como también lo será la integración de esta especificación con Question and Test Interoperability (IMS, 2004). Si bien la codificación de QTI no es objeto de este artículo, sí podemos apuntar cómo es la actual integración de QTI en IMS LD. Básicamente, la secuencia es la misma que la comentada anteriormente para describir la relación entre un entorno y un recurso asociado. La diferencia está en la referencia del recurso tipo QTI ('imsqti_item_xmlv2p0') que apuntaría a un fichero escrito en esta especificación ('question_1.xml'):

```
<environment identifier="Env-1">
```

```
<title>A test linking QTI and IMS LD</title>
<learning-object identifier="LO-QTI-question-1">
  <title>First question</title>
  <item identifier="I-1" identifierref="R-que-1"/>
</learning-object>
</environment>
```

```
<resource identifier="R-que-1" type="imsqti">
  <file href="question_1.xml"/>
</resource>
```

6.7. E-PORTFOLIOS

Por último, la utilización de ePortfolios es posible si mantenemos el espíritu propio de IMS ePortfolio (IMS, 2002), es decir, que la grabación y compartición de datos externamente a cada aplicación se realice de manera estandarizada dentro del paquete de información, posibilitando que la información pueda ser utilizada en sitios diferentes por diferentes sistemas. Las propiedades globales que tiene IMS LD permiten definir información de un usuario (globpers-property), de un grupo (property-group) o, a modo de constantes, para todo el mundo (glob-property). La primera es llamada también portfolio-property porque permite la incorporación de ficheros a la ejecución de la unidad de aprendizaje. Como siempre, la utilización requiere dos momentos distintos:

- a) la creación e inicialización de la variable global

```
<globpers-property identifier="GP-suggestions">
  <global-definition uri="GP-suggestions">
    <title>Suggestions about the course</title>
    <datatype datatype="file"/>
  </global-definition>
</globpers-property>
```

- b) la utilización de la propiedad en un fichero XML

```
<div class="upload-file">
  <p>Choose the file with your suggestions:</p>
  <id:set-property ref="GP-suggestions"/>
</div>
```

6.8. CONCLUSIÓN

Los estándares en aprendizaje virtual son necesarios porque ayudan a independizar el contenido y la metodología de la herramienta que los usa. De esta manera cualquier creador (profesor, pedagogo o escritor) puede asegurar la interoperabilidad de sus programas lectivos o materiales didácticos sin temor a sufrir una caducidad prematura de los mismos obligada por elementos técnicos, en lugar de por requerimientos educativos. La especificación IMS Learning Design aborda el problema de juntar recursos planos con metodología haciendo posible el modelado de escenarios pedagógicos donde controlar el flujo de información y adaptar el proceso de

aprendizaje en función de la actividad de los participantes, de sus perfiles y de sus necesidades concretas. Hemos podido comprobar mediante funcionalidades concretas que ciertas necesidades básicas en aprendizaje virtual son abordadas y solucionadas satisfactoriamente por la especificación, lo que facilita su utilización en entornos reales de trabajo, impulsando a su vez la construcción de herramientas concretas de edición y ejecución que permitan una implantación satisfactoria de la misma.

Bibliografía

- Burgos, D., Berbegal, N., (2005) “IMS LD Level 0”, The Open University of The Netherlands (OUNL) [en <http://hdl.handle.net/1820/307,200405>]
- Burgos, D., (2005) “Geo-Quiz1” y “Geo-Quiz2”, OUNL, [en <http://moodle.learningnetworks.org/course/view.php?id=20,200405>]
- Burgos, D., Berbegal, N., Griffiths, D., Tattersall, C., Koper, R., (2005) “Do we need specifications in e-learning”, *International Magazine of the University of Moscow*, Russia
- Cetis, (2005) CETIS, Bolton, Reino Unido [en www.cetis.ac.uk,280405]
- Cole, M., Engestrom, Y., (1993) “A cultural-historical approach to distributed cognition”, en G. Salomon (Ed.), *Distributed cognitions*, Cambridge University Press, Reino Unido.
- Dougiamas, M., (2004) “Moodle”, [en www.moodle.org,310305]
- IMS (2003) “IMS Learning Design. Information Model, Best Practice and Implementation Guide, XML Binding, Schemas. Version 1.0 Final Specification”, Boston, USA, [en <http://www.imsglobal.org/content/learningdesign/,290405>]
- IMS, (2002) ePortfolio [en www.imsglobal.org,290405]
- IMS, (2004) Question and Test Interoperability [en www.imsglobal.org,290405]
- Koper, R. y Burgos, D., (2005) “Designing Learning Activities: From Content-based to Context-based Learning Services”, *International Journal on Advanced Technology for Learning, Special Session*. Volumen: 2, número 3, Octubre, 2005 [en <http://hdl.handle.net/1820/333>]
- Koper, R., Tattersall, C., (2005) “Learning Design: a handbook on modelling and delivering networked education and training”, chapter 10, Springer, Germany.
- OUNL y Perot Systems (2004) Edubox [en <http://www.perotsystems.com/netherlands/,310305>]
- Recursos LD (2005)
UNFOLD Project [www.unfold-project.net]
Learning Networks for Learning Design [<http://moodle.learningnetworks.org>]
- Learning Networks [www.learningnetworks.org],
IMS LD Nivel 0 en castellano
[<http://moodle.learningnetworks.org/mod/resource/view.php?id=196>]
CopperCore [www.coppercore.org]
CopperAuthor [www.copperauthor.org]
ASK LDT [www.iti.gr]
- Shuell, Th.J, (1999) “The role of the student in learning from instruction”, *Contemporary Educational Psychology*. Volumen: 13, páginas 276-295.
- Tattersall, C. y Burgos, D., (2005) “Learning to listen to Jazz”, OUNL, [en <http://moodle.learningnetworks.org/course/view.php?id=20,200405>]
- Tattersall, C., Vogten, H., Brouns, F., Koper, R., Van Rosmalen, P., Sloep, P. Van Bruggen, J., (2003) “Delivering courses modelled using IMS Learning Design”, OUNL, [en <http://hdl.handle.net/1820/35,100105>]
- University of Bolton, (2005) “Reload”, JISC Project [en www.reload.ac.uk,220405]
- Vogten, H., Martens, H., (2005) “CopperCore 2.2.2”, OUNL [en www.coppercore.org,290405].