

Akademo: un Sistema de Gestión de Contenidos semiestructurados

Eduardo Grosclaude¹

¹ Universidad Nacional del Comahue, Neuquén, Argentina

Resumen

Las experiencias de extensión en transferencia de Tecnologías de Información nos han aportado una visión diferente de la enseñanza. Para adecuarnos a una nueva iniciativa de dictados con contenidos propios, basados en tecnologías libres, desarrollamos un modelo de transferencia y un sistema de gestión de contenidos (SGC) que lo soporta. Se describen el modelo de transferencia de la iniciativa CTL y el SGC para contenidos semiestructurados Akademo. Se consigna una experiencia de uso de ambos desarrollos.

Palabras clave: Gestión de contenidos, CMS, SGC, Software Libre, Transferencia.

1. Introducción

El Departamento de Ciencias de la Computación de nuestra Facultad ha ofrecido, en el pasado reciente, cursos en Tecnologías de Información abiertos a la comunidad, con programas de dictado elaborados por proveedores de tecnologías propietarias. Este es el caso del programa CCNA de Cisco Systems, dictado en la UNC desde 1999 y hasta el momento actual; y, más recientemente, el programa DBA de Oracle. Estas experiencias nos han brindado, en cierta manera, una visión estereoscópica de la enseñanza, al mostrarnos la existencia de objetivos, audiencias, modalidades, formatos, etc., diferentes de lo acostumbrado en el acontecer académico de las carreras regulares de la Universidad.

Con posterioridad a estas experiencias surge dentro del Departamento la iniciativa, que hemos llamado *CTL (Capacitación en Tecnologías Libres)*, de desarrollar una oferta de extensión basada en tecnologías libres y con contenidos propios. Esta iniciativa parte del reconocimiento del Software Libre como un elemento tecnológico beneficioso para la sociedad en general, y en particular para la región donde se inserta nuestra

Universidad. Representa para nosotros el nuevo desafío de organizar un cuerpo de conocimientos adquiridos durante la práctica docente y profesional, para transferirlo a una audiencia heterogénea.

Este desafío tiene múltiples aristas:

- Como docentes de Ciencias de la Computación, hemos debido revisar nuestra propia tendencia a reproducir en este nuevo contexto, de disciplinas tecnológicas, los esquemas de aprendizaje y enseñanza de las disciplinas científicas.
- El dominio de conocimiento, relacionado en este caso con la temática de las aplicaciones del Software Libre, es multiforme y cambiante, quizás como ningún otro frente de desarrollo de la tecnología informática actual.
- Este conocimiento, además de ser un blanco móvil, presenta una compleja trama de competencias interdependientes y, en general, no establecidas formalmente en los contenidos académicos tradicionales, por lo cual debemos elaborar nuestro propio mapa o modelo del conocimiento.
- La experiencia anterior en dictados de extensión muestra diferencias notables entre las audiencias de estos cursos y las compuestas por alumnos regulares de las carreras de la Universidad, y plantea exigencias didácticas para anticiparnos a las brechas en competencias, visiones y expectativas que traen a la misma aula diferentes personas con diferente formación.
- La iniciativa de Capacitación en Tecnologías Libres se realiza mediante un equipo de varios instructores, que en su rol académico usual funcionan como docentes de nuestro Departamento, por lo cual es natural el requerimiento de un espacio colaborativo donde desarrollar y combinar los componentes curriculares a ser diseñados.

Reunidas, estas observaciones nos enfrentaron a la necesidad de definir una cantidad de estrategias para aproximarnos al problema. En este trabajo mostramos el camino recorrido en la definición de estos elementos, y describimos el soporte tecnológico que, para ayudarnos, elaboramos durante este camino.

2. Modelo de transferencia CTL

La forma de transferencia elegida para los cursos CTL se inspira en las experiencias de extensión anteriormente mencionadas. Como se dijo, la audiencia de estos cursos es heterogénea en todo sentido. Se trata con frecuencia de adultos que han interrumpido su formación hace algún tiempo, y su interacción con el docente y con el material de estudio presentado tropieza a veces con la falta de “gimnasia” estudiantil. Las destrezas concretas que se proponen impartir estos cursos de tecnologías están al alcance de la mayoría; sin embargo, leer grandes cantidades de material teórico, mantener la atención, extraer lo importante, administrar su esfuerzo, son dificultades insoslayables. Para el docente, preparar material que cubra todos los aspectos formativos que necesita este alumnado diverso es a veces frustrante, tan pronto como se advierte que este material, con frecuencia, no será utilizado.

De estas experiencias aislamos por oposición, y vimos conveniente para nuestra iniciativa, un elemento clave que es la definición de objetivos específicos basados en destrezas, y la estructuración de las actividades en función de estos objetivos. A diferencia de las clases de nuestras materias regulares, que se organizan alrededor de los contenidos teóricos y cuentan con la práctica como medio de fijación, nuestra visión parte de la meta de lograr, para cada unidad didáctica, una habilidad técnica determinada, enunciable y verificable. Para los estudiantes observados en nuestra experiencia de extensión, expresar esta habilidad técnica como centro de la clase funciona como consigna visible, motivadora y orientadora. Opuestamente, el material textual expositivo que sirve de

introducción a las clases centradas en “temas”, presenta para ellos interrogantes que la práctica no siempre alcanza a poner de manifiesto ni a resolver.

De esa habilidad, destreza o competencia propuesta, una vez puesta en foco, resultan inmediatamente las actividades a desarrollar; y se extraen, por un proceso menos directo pero acotado, los contenidos teóricos estrictamente necesarios para su realización comprensiva. Para favorecer este proceso, el docente refleja este orden de prioridades al momento de diseño, en la construcción del material de aula dividido en *unidades*. Los contenidos teóricos necesarios se adjuntan como *recursos* a las unidades de prácticos desarrolladas, y su función es integrar esas destrezas con su explicación en un contexto dado. Estas entidades de bajo nivel, unidades y recursos, son el núcleo del modelo CTL.

Las demás entidades en el modelo CTL definen las secuencias y supersecuencias de unidades. En el tope de la jerarquía, los objetivos generales de un *programa* se corresponden con el rol técnico del estudiante (aquí visualizado en el trabajo con las Tecnologías Libres como un usuario, administrador de sistemas, administrador de redes, etc.). Este es el análogo, en nuestro contexto, de una carrera académica. En un nivel intermedio, los objetivos de un *módulo* definen los conjuntos cohesivos de conocimientos o saberes que, integrados, posibilitan este rol. Estos son los análogos de los cursos o materias que componen cada carrera.

Estos cuatro componentes: *recursos*, *unidades*, *módulos* y *programas*, definidos por objetivos a varios niveles, constituyen el modelo de transferencia desarrollado para la iniciativa CTL. Esta organización de objetos puede resumirse como en la Tabla 1.

Componente CTL	Nivel de objetivos	Análogo	Representa
Programa	General	Carrera, Especialidad	Rol profesional
Módulo	Intermedio	Materia, Curso	Conjunto cohesivo de saberes
Unidad	Específico	Clase, Práctico	Destrezas
Recurso	Contextual	Libro, Apunte	Fundamentos

Tabla 1: Elementos del modelo de transferencia CTL

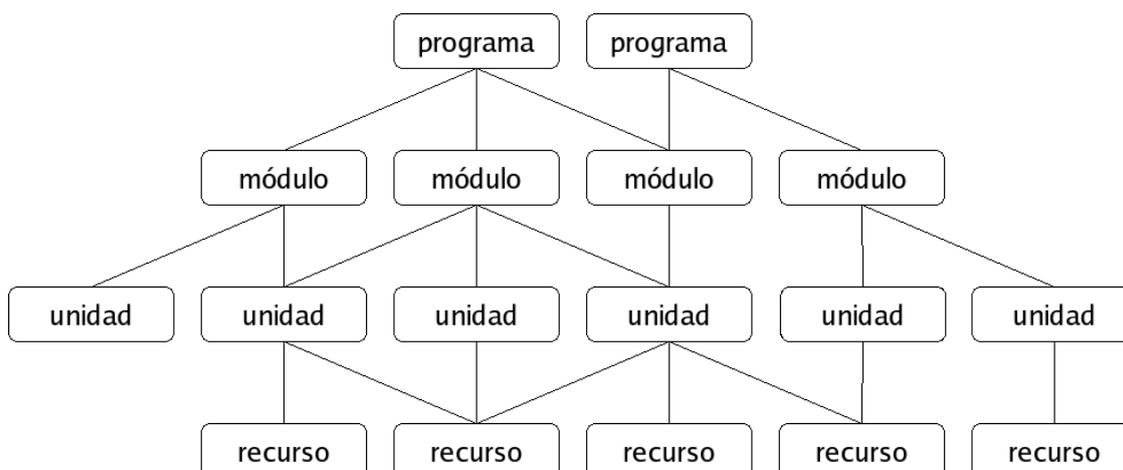


Figura 1: Modelo de información CTL implementado por Akademo

3. Herramientas de gestión de contenidos

Como se dijo, CTL pretende transferir conocimiento de un dominio vasto y móvil, a un alumnado significativamente plural. Los requerimientos más importantes para las herramientas con las cuales desarrollar esta iniciativa incluyen entonces:

- Mapear adecuadamente el modelo de transferencia CTL, permitiendo editar y administrar elementos del modelo
- Facilitar el proceso de creación de contenidos en forma modular y colaborativa.
- Permitir reutilizar y reestructurar las unidades didácticas desarrolladas en forma especialmente flexible, para adaptarse a las diferentes audiencias.
- Permitir modificar, reutilizar y recombinar el material desarrollado para adaptarse a los cambios en el dominio de conocimiento.
- Permitir extraer o publicar diferentes vistas de los contenidos mantenidos, para diferentes usos.

Las características propuestas para las herramientas de generación y administración de contenidos buscadas tienen intersección a primera vista con dos líneas tecnológicas actuales: por un lado, el procesamiento de documentos XML, y por otro, la edición y publicación simultánea de páginas, conocida como Wiki.

- La tecnología XML resulta inmediatamente atractiva. Al expresar una base de conocimiento en XML, obtenemos acceso a los contenidos en función de su semántica, y libertad para recuperar estos contenidos en los formatos finales que sean

necesarios. Lamentablemente, es difícil la interacción con cualquier herramienta de edición de XML, porque su interfaz es necesariamente complicada por la lógica del *tagging* o marcado semántico de elementos.

- Al contar con un sistema Wiki, la edición de contenidos es directamente colaborativa, soportada por web, y su lenguaje de *markup* permite concentrarse en lo esencial. Sin embargo, el lenguaje de marcado del Wiki no es estándar, a veces es no intuitivo, y no resulta aún una práctica común entre los docentes, probablemente por la separación entre edición y presentación contraria a la característica *wysiwyg* de los procesadores de texto tradicionales.

Con estas características en mente, relevamos una cantidad de proyectos diversos dentro del Software Libre, los que aproximadamente pueden ubicarse dentro de las tres categorías de *Editores XML*, *Wikis* y *Sistemas de preparación de documentos*. Entre los editores XML libres revisados se encuentran *MView* [10] y *Conglomerate* [11]. Ambos presentan los conocidos problemas de navegabilidad no trivial del texto, más la restricción de no hallarse disponibles en versión web. Los wikis examinados fueron *Tavi*, *Twiki*, *DokuWiki* y *MediaWiki* [1][2][3][4].

Twiki resultó el más aproximado a lo pretendido, al ser capaz de generar documentos XML con una determinada estructura a partir de páginas de texto wiki ingresadas. Sin embargo, problemas de ejecución resultaron en pérdidas de datos durante las pruebas, y el costo de obtención de contenidos en los formatos requeridos pareció alto, al requerirse programación XSLT intensiva. Los sistemas de preparación de documentos evaluados (como *SiSU* [5] y *Borges* [6]) están basados en comandos de línea. Representan una base sobre la cual construir un sistema, aunque apelan a lenguajes no estándar (*SiSU*) o no abordan el problema de la interfaz XML (*Borges*). En una

categoría intermedia se sitúa *LyX*, que es un editor que también produce XML DocBook.

Como conclusión de esta búsqueda nos encontramos con un abanico de tecnologías que, aunque sumamente poderoso e interesante como software de propósito general, dejaba vacante la mayoría de nuestros requerimientos específicos. Así fue como concebimos el diseño preliminar de un SGC (*sistema de gestión de contenidos*, en inglés *CMS*) para contenidos semiestructurados, *ad hoc* del modelo CTL.

4. Un SGC para material semiestructurado

El SGC desarrollado, que ha sido bautizado Akademo, combina la estructuración de contenidos que puede brindar XML, con el paradigma Wiki de facilitar la edición colaborativa; pero resuelve las dificultades con las interfaces habituales de ambos sistemas al funcionar mediante un editor XHTML *wysiwyg* convenientemente restringido.

Akademo es una aplicación web CGI convencional donde la edición de contenidos se hace por medio de formularios. Opera sobre una base de datos cuyo esquema, sin redundancia, soporta las entidades y relaciones del modelo CTL. Al reflejar este modelo utilizando relaciones muchos-a-muchos, Akademo posibilita la recombinación de contenidos. La Figura 1 idealiza una instancia de esta base, donde un recurso puede ser referenciado por varias unidades, cada unidad puede aparecer en varios módulos, y un módulo puede constituir más de un programa.

Los contenidos volcados al sistema Akademo pueden extraerse en XML, en HTML y en PDF, con vistas para uso del equipo docente y para el aula, en cualquiera de los cuatro niveles. Por ejemplo:

- Un docente puede imprimir una guía de aula para una unidad, donde aparezca un listado de recursos físicos necesarios y un conjunto de notas para el instructor, mientras que un alumno puede acceder a una vista de la misma unidad donde solamente figuren las consignas y el material complementario.
- Esta vista para el alumno queda disponible cuando el administrador del sistema decide publicar un subárbol cualquiera de la base. Mientras tanto, el cuerpo docente puede continuar editando la copia de uso interno sin afectar la vista del alumno.
- Un docente puede solicitar un apunte en PDF correspondiente a cualquier subárbol (programa, módulo o unidad) de la base.

La salida de texto XML a partir de los contenidos gestionados por Akademo subsume en un solo mecanismo de *tagging* semántico el texto plano ingresado por el

usuario, material en XHTML y contenidos generados por el sistema. Esta salida XML, al venir estructurada por el esquema de la base, puede ser convertida (por ejemplo, por transformaciones XSLT) a otros formatos estructurados. En particular, el formato DocBook puede manipularse con herramientas externas convencionales, postprocesarse con hojas de estilo, transformarse a otros formatos finales o importarse a otros sistemas de tratamiento de documentos; al mapear los documentos generados por Akademo al formato DocBook, un programa puede transformarse en un *Book*, o bien, los módulos en *Articles*, y las unidades subordinadas, en sus *Sections*.

Para adecuarlo completamente a los fines de la oferta de Extensión, el sistema ha sido complementado con subsistemas de gestión de contactos, gestión de dictados e inscripciones, obtención de certificados y emisión de circulares. A esto se agrega un mecanismo de *blogging* elemental para compartir observaciones o novedades entre los docentes. Las diferentes zonas del sistema son el espacio para los diferentes actores, a saber, público, alumnos, docentes y administrativos.

- La zona Pública permite ofrecer información general sobre la iniciativa y realizar registraciones o preinscripciones voluntarias de público interesado.
- La zona de Aula permite publicar el material de clase en estado elaborado.
- La zona Docentes es el lugar donde elaboran los contenidos, e interactúan, los miembros del equipo docente.
- La zona de Administración es utilizada por el personal administrativo de la Secretaría de Extensión de la Facultad para recibir inscripciones e interactuar con los contactos, emitir circulares y certificados.

La Figura 2 muestra estas zonas y los flujos de datos más importantes.

4.1. Implementación

Akademo ha sido desarrollado en lenguaje Perl, usando extensivamente componentes preexistentes, como los módulos *Class::DBI*, *CGI::Application*, *HTML::Template*, *HTML::HTMLOutput*, *Class::DBI::ToSax* y *MIME::Lite* [7]. El editor XHTML utilizado es *TinyMCE* [9], que se mostró comparativamente superior a otros similares. El gestor de base de datos utilizado para esta primera implementación es *SQLite 3* [8], aunque la separación aportada por *Class::DBI* hace trivial su migración a otros motores.

Conclusiones

- El modelo CTL resultó una aproximación útil y practicable a la nueva actividad de extensión.
- El modelo de transferencia CTL es directamente aplicable a otros dominios de la enseñanza de tecnologías, y así lo refleja la construcción de su herramienta básica Akademo, que puede ser reutilizada para otros fines dentro del ámbito de la extensión.
- La actividad de transferencia CTL nos enfrentó con una cantidad de oposiciones dialécticas a lo acostumbrado. Una de ellas fue el tránsito de la didáctica de “verificación experimental de lo que digo” a la de “explicación de lo que hago”. El cuerpo docente de CTL encontró dificultad en adaptarse a esta inversión metodológica. La estructuración de contenidos del sistema Akademo ayudó a mantener la atención sobre

los componentes básicos del proceso, las unidades, desplazando los recursos a su lugar correcto en el modelo.

Al momento presente se ha hecho uso de Akademo en el marco de la iniciativa CTL, dictando cursos de *Introducción a GNU/Linux* y *Administración de GNU/Linux*, niveles inicial e intermedio, desde Marzo de 2006 [12].

Desarrollo futuro

Se piensa que Akademo tiene la posibilidad de servir como útil para el codiseño curricular, ya sea admitiendo feedback de los estudiantes que completan transferencias (el flujo faltante en el diagrama de Figura 2) como exponiendo de forma controlada la edición de unidades y módulos, para que los potenciales interesados puedan sugerir los contenidos de esas transferencias.

En el aspecto técnico, si bien su uso ha sido satisfactorio, quedan pendientes varias mejoras que será deseable incorporar:

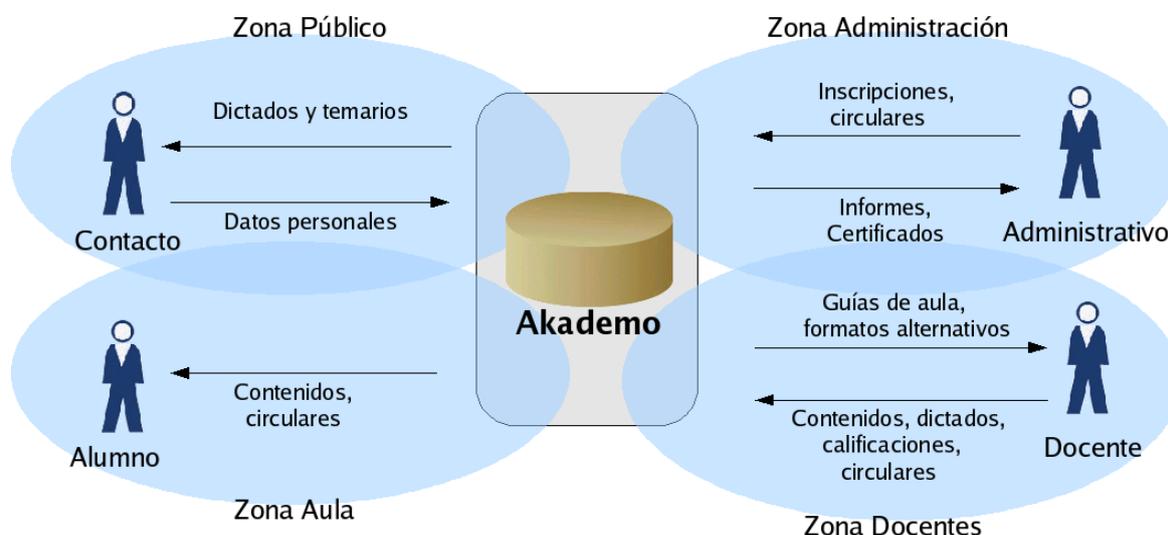


Figura 2: Zonas del sistema Akademo

- Producción de documentación DocBook mediante transformaciones XSLT.
- Categorización de los elementos del *pool* de recursos, actualmente organizado en forma plana.
- Obtener mayor control sobre la generación de PDF, lo cual depende de mejoras al componente HTMLDoc.
- Publicación y seguimiento de exámenes online.

Referencias

- [1] Tavi: <http://tavi.sourceforge.net/>
[2] Twiki: <http://twiki.org/>
[3] MediaWiki: <http://www.mediawiki.org/wiki/MediaWiki>
[4] DokuWiki: <http://wiki.splitbrain.org/wiki:dokuwiki>
[5] SiSU: <http://www.jus.uio.no/sisu/>
[6] Borges: <http://www1.mandrivalinux.com/documentation/project/borges>
[7] Módulos ubicados en el repositorio CPAN, <http://www.cpan.org>
[8] SQLite: <http://www.sqlite.org>
[9] TinyMCE: <http://tinymce.moxiecode.com/>
[10] MlView: <http://www.freepiders.org/projects/gmlview/>
[11] Conglomerate: <http://www.conglomerate.org>
[12] Capacitación en Tecnologías Libres: <http://linux.uncoma.edu.ar>

Dirección de Contacto del Autor:

Eduardo Grosclaude
Buenos Aires 1400
Neuquén
Argentina

email:
oso@uncoma.edu.ar

sitio web: <http://pedco.uncoma.edu.ar>

Eduardo Grosclaude Licenciado en Cs. de la
Computación, Universidad Nacional del Comahue.
Docente del Área de Sistemas del Departamento de Cs. de
la Computación de la Facultad de Economía y
Administración, UNC.
