

Desarrollo dirigido por modelos para la creación de laboratorios virtuales

Model driven development for the creation of virtual laboratories

Yois Smith Pascuas Rengifo¹, José Joaquín Bocanegra García², Edson Johann Ortiz Lozada³, José Nelson Pérez Castillo⁴

Ingeniería de Sistemas, Universidad de Amazonia, Florencia, Colombia

yoispascuas@hotmail.com
jose.bocanegra@udla.edu.co
johannortiz@hotmail.com
nelsonp@udistrital.edu.co

Resumen- Los laboratorios virtuales apoyan y promueven el aprendizaje de los estudiantes y aumentan las opciones de experimentos disponibles en las instituciones educativas. A pesar de las ventajas pedagógicas que tienen los laboratorios virtuales, se ha identificado que estas herramientas no cumplen con una estandarización que permita su interoperabilidad, reutilización y accesibilidad. Esto ocasiona, entre otras cosas, la creación de aplicaciones no reutilizables y la inversión en tiempo y costo que esto supone. Como una posible solución a los problemas mencionados, se propone la utilización del desarrollo dirigido por modelos para generar laboratorios virtuales que puedan ser empaquetados en el estándar SCORM (Sharable Content Object Reference Model) y de esta forma, estar disponibles para su uso en sistemas de gestión de aprendizaje.

En el caso de estudio, se define por una parte, un conjunto de metamodelos que muestran la estructura y funcionalidades genéricas de los laboratorios virtuales, y por otra parte, una serie de transformaciones de esos modelos a código para que sean interoperables con el estándar SCORM. Esta aproximación inicial, permite elevar el nivel de abstracción y reutilización, ya que los laboratorios virtuales deben tener la capacidad de ser usados en diferentes contextos y propósitos educativos, reduciendo significativamente el tiempo y esfuerzo en su construcción.

Palabras Clave— Objeto de aprendizaje, estándar, metamodelo, modelo, transformación, código.

Abstract- Virtual labs support and promote students' learning and at the same time increase the options of available

¹ Ingeniera de Sistemas, Universidad de la Amazonia, Grupo de investigación en ingeniería de software y grupo de investigación GICOGE.

² Ingeniero de sistemas, Grupo de investigación en ingeniería de software Universidad de la Amazonia.

³ Ingeniero Químico

⁴ Ingeniero de Sistemas, Grupo de Investigación GICOGE, Universidad Distrital Francisco José de Caldas.

experiments in educational institutions. Despite pedagogical advantages of the virtual labs, it has been identified that these tools do not accomplish a standarization to allow its interoperability, reuse and accessibility. It causes, among other things, the creation of non-reusable applications and the time

and cost investment needed. As a possible solution to these problems, the use of model driven development has been

proposed to generate virtual labs that can be packaged in SCORM (Sharable Content Object Reference Model) standard and in this way, being available for its use in learning management systems.

In the case study, a set of metamodels that show the generic functions and structure of the virtual labs is defined and also, a series of transformations of those models in code to be interoperable with the SCORM standard. This initial approach allows to raise the abstraction and reuse level, due to virtual laboratories should have the capacity to be used in different contexts and educational purposes, reducing, in a high proportion, the time and effort on its construction.

Keywords- Learning object, standard, metamodel, model, transformation, code

I. INTRODUCCIÓN

LOS laboratorios virtuales (LV) son una representación informática de los laboratorios tradicionales y permiten realizar experimentos, investigaciones, prácticas académicas y científicas, dando la sensación de su existencia real; apoyan y promueven el aprendizaje de los estudiantes y aumentan las opciones de experimentos disponibles en las instituciones educativas. A pesar de las ventajas pedagógicas que tienen los LV, se ha identificado que algunas de estas herramientas no cumplen con una estandarización que permita su interoperabilidad,

reutilización y accesibilidad. Esto ocasiona, entre otras cosas, la creación de aplicaciones no reutilizables y la inversión en tiempo y costo que esto supone.

Como una posible solución a los problemas mencionados, se propone el uso del desarrollo dirigido por modelos (MDD) ya que los modelos permiten aumentar el nivel de abstracción y reutilización; como herramienta de especificación y diseño de software permiten simplificar tareas, análisis, detección de problemas en las etapas tempranas del desarrollo y la independencia de las plataformas, tecnologías o lenguajes de implementación. MDD y su arquitectura dirigida por modelos (MDA) son enfoques aceptados para el desarrollo de software, los cuales proponen el uso de modelos en todas sus fases. La transformación de modelos es la base de MDA, comenzando por un modelo independiente de la plataforma (PIM) y cuyo objetivo es lograr modelos específicos de plataforma (PSM).

El presente artículo muestra la utilización del paradigma MDD en la creación de LV que puedan ser empaquetados en el estándar SCORM y de esta forma, estar disponibles para su uso en sistemas de gestión de aprendizaje (LMS por sus siglas en inglés). En el caso de estudio, inicialmente se propone:

- Metamodelo origen denominado Experimento, que muestra la estructura y funcionalidades de los LV.
- Metamodelo destino llamado LaboratorioVirtual, que permite que la aplicación sea desarrollada en cualquier plataforma.
- Transformaciones modelo a modelo utilizando ATL.
- Transformaciones modelo a texto con MOFScript para generar código y empaquetarlo con SCORM.

Esta aproximación inicial ofrece a los diseñadores, desarrolladores e investigadores un elevado nivel de abstracción y reutilización que permite editar de forma flexible LV para que puedan ser usados en diferentes contextos y propósitos educativos. Adicional a esto, se reduce significativamente el tiempo y esfuerzo en su construcción, logrando aumentar la productividad en el desarrollo de este tipo de aplicaciones educativas.

II. CONTEXTUALIZACIÓN

A. Laboratorio Virtual

Un LV se define según [1], como un sistema computacional que pretende aproximar el ambiente de un laboratorio tradicional. Los experimentos se realizan paso a paso, siguiendo un procedimiento similar al de un laboratorio tradicional: se visualizan instrumentos y fenómenos mediante objetos dinámicos, imágenes o animaciones. Se obtienen resultados numéricos y gráficos, tratándose éstos

matemáticamente para la obtención de los objetivos perseguidos en la planificación docente de las asignaturas.

B. Desarrollo dirigido por modelos

El modelo de un sistema es la descripción o especificación de ese sistema y su adaptación a un propósito determinado. Frecuentemente se presenta a un modelo como una combinación de dibujos y texto. Para Pelayo [2], el texto puede estar en un lenguaje de modelado o en lenguaje natural. Según Vallecillo [3], los modelos pueden ser abstractos, comprensibles, precisos, predictivos y económicos. Un modelo se define de acuerdo a la semántica de un modelo de modelos llamado metamodelo. Un meta-metamodelo ecore (extensión de modelos) pretende introducir la semántica que se requiere para especificar metamodelos. Como modelo con su metamodelo, un metamodelo se ajusta al meta-metamodelo.

El primer metamodelo que utiliza MDA es el PIM, que contiene una vista del sistema centrada en la operación del mismo y esconde los detalles necesarios para una determinada plataforma, el cual muestra un grado de independencia tal que permite mapearlo a una o varias plataformas. Esto se logra al definir una serie de servicios al abstraer los detalles técnicos para que otros modelos especifiquen como será la implementación.

El segundo metamodelo es el PSM, que combina la especificación de un PIM con los detalles para indicar cómo el sistema usa una plataforma en particular. La diferencia entre un modelo PSM y uno PIM es el conocimiento que incorpora acerca de la plataforma final de implementación, el PIM no incorpora ninguna información acerca de la plataforma de destino; por su parte, un PSM contiene toda la información necesaria para generar el código de la aplicación. El enfoque basado en modelos supone proporcionar a los diseñadores y los desarrolladores un conjunto de operaciones dedicadas a la manipulación de los metamodelos. En este contexto, la transformación de metamodelos es una actividad central para el manejo de modelos: su objetivo es especificar la manera de producir una serie de modelos de destino en función de un conjunto de modelos de origen.

Las reglas de transformación de metamodelos [4], permiten convertir un modelo instancia de un metamodelo en otra instancia de un metamodelo del mismo sistema (Modelo a modelo: M2M), se pueden obtener diferentes PSM a partir del mismo PIM como resultado de la aplicación de diferentes transformaciones. Posteriormente, aplicando transformaciones modelo a texto (M2T) a los PSM se puede generar código.

Las transformaciones se basan en una serie de reglas denominadas mapeo o metainformación (mapping rules),

que definen los mecanismos para el paso de un metamodelo origen a un metamodelo destino. Se etiquetan o marcan los metamodelos para guiar los procesos de transformación. Esta metainformación se denomina genéricamente como marcas y se utilizan tanto en los PIM como en los PSM.

III. TRABAJO RELACIONADO

Se revisaron trabajos relacionados con modelos y su aplicabilidad en el desarrollo de aplicaciones y plataformas utilizadas para actividades educativas y de elearning. Se hallaron las principales y comunes características de estas propuestas. Entre estas características se define como primera la técnica de desarrollo, segunda el tipo de aplicación, tercera el estándar de interoperabilidad, cuarta la creación del PIM, quinta la obtención del PSM, sexta las transformaciones y séptima la generación automática de código.

Se inicia con la propuesta de Fátima [5], que considera que los objetos de aprendizaje (OA) no se adaptan fácilmente a perfiles de los estudiantes. Propone su personalización utilizando una estrategia dirigida por modelos, específicamente la metodología del Lenguaje Específico del Dominio (DSL) y describe una arquitectura de los OA personalizables.

En el trabajo de Wang [6], se propone una metodología basada en MDA para desarrollar sistemas elearning basados en la arquitectura J2EE, LTSA (Learning Technology Systems Architecture) y el estándar SCORM. El PIM es un artefacto que está modelado con UML y el PSM por UML profile para la interfaz en EJB.

En Bizonova [7] se muestra el enfoque MDA de una arquitectura marco para integrar plataformas educativas de diversas instituciones, reutilizar contenidos elearning y clasificar existentes o futuros LMS para simplificar la migración de datos. Propone una estrategia de integración de LMS en un modelo PIM generalizado tomando como referencia dos conjuntos de funcionalidades: permisos de acceso y gestión de OA, de dos sistemas LMS (Moodle y OLAT). Sin embargo el marco todavía necesita ser complementado por más LMS.

En Montenegro [8], se presenta la construcción de un Modelado Específico de Dominio (DSM) para la creación de módulos en los LMS. Se inicia con un metamodelo para la construcción de un Lenguaje Específico de Dominio (DSL) que con MDE y sus transformaciones genera un PIM; el despliegue de este modelo se realiza en varias plataformas LMS. Posteriormente utilizando redes de petri comprueban la hipótesis de que las herramientas para el diseño basado en modelos reducen el tiempo de desarrollo

de plataformas de educación virtual, pudiéndose extender a diversos campos de acción.

El enfoque de Dodero [9] está basado en la definición de un lenguaje abstracto para el diseño de aprendizaje (IMS LD), con un conjunto principal de elementos relacionados con la descripción de actividades y flujos de aprendizaje. Basado en el paradigma MDD, propone un PIM denominado FlexoLD que describe una secuencia de actividades; a partir de la definición de cursos expresados con este lenguaje (PSM), se pueden generar unidades de aprendizaje que cumplan con el estándar IMS LD. Presenta Malek [10] la utilización de MDD en el contexto de sistemas de aprendizaje adaptativos, propone ContAct-Me una herramienta basada en CAAML (Context-aware Adaptive Activities Modeling Language) y la utilización de dos metamodelos, el primero de actividades de aprendizaje y el segundo de adaptabilidad, con el fin de garantizar que el diseño de actividades sea interoperables en diferentes plataformas de aprendizaje.

La mayoría de estas propuestas consideran procesos para facilitar el desarrollo de objetos de aprendizaje, cursos e integración de plataformas LMS de manera general y específica en cada caso. Una propuesta aborda el tema MDD y DSL, en la creación de módulos para los LMS, sin embargo es un tema amplio que debe seguirse explorando.

Se evidencian problemas relacionados con la reusabilidad, específicamente la fácil adaptabilidad en diferentes contextos educativos. Un tema no tratado es la complejidad del modelamiento de posibles cálculos matemáticos que se deben ejecutar y mostrar dentro de estos contenidos educativos. Adicionalmente, poco se aborda la facilidad y ventajas de la interoperabilidad que ofrece el empaquetamiento de objetos de aprendizaje en el estándar SCORM para trabajar en diferentes LMS.

La Tabla 1, resume las características que se han tenido en cuenta en cada una de las propuestas. Se presentan como filas las características y como columnas las propuestas en donde el símbolo [✓] significa que lo contiene y el símbolo [✗] que no lo contiene.

TABLA 1
RESUMEN DE PROPUESTAS

Características	Propuestas					
	[5]	[6]	[7]	[8]	[9]	[10]
MDD	✗	✓	✓	✓	✓	✓
Aplicación Educativa	✓	✓	✓	✓	✓	✓
Estándar Interoperabilidad OA	✗	✓	✗	✗	✗	✗
PIM	✗	✓	✓	✓	✓	✓
PSM	✗	✓	✗	✗	✓	✗
Transformación	✗	✓	✓	✓	✓	✗
Código	✓	✗	✗	✓	✗	✗

IV. MODELO PROPUESTO

El modelo propuesto fue diseñado teniendo en cuenta la necesidad de implementar herramientas educativas que consideren el desarrollo de prácticas académicas en laboratorios virtuales. El modelo presenta ventajas en el ámbito educativo porque permite estructurar laboratorios virtuales con contenidos temáticos diferentes de manera rápida y sencilla.

El desarrollo dirigido por modelos puede ser aplicado en la creación de LV empaquetados en SCORM para su interoperabilidad en diferentes LMS. Como lo muestra el diagrama de actividades de la Figura 1, se parte de definir los metamodelos del proyecto: el metamodelo origen denominado Experimento y el metamodelo destino llamado LaboratorioVirtual tomando como referencia el meta-modelo ecore definido por MOF de la OMG (Object Management Group) [11], al realizarse la implementación de metamodelos se crean los ecore. Se toma como referencia el ecore del metamodelo origen Experimento y con GMF se crea un editor gráfico para la generación del modelo origen que corresponde a un XMI con datos para la transformación. Los insumos para realizar la transformación M2M son el metamodelo origen, el metamodelo destino y el modelo origen, esto implica la generación y validación del modelo destino.

Posteriormente se realiza la transformación M2T utilizando MOFScript y tomando como base el XMI del modelo destino y el metamodelo destino se generan archivos de código. El código de la aplicación de caso de estudio es un LV de química para el desarrollo de la práctica de neutralización ácido fuerte – base fuerte y el cual posteriormente es empaquetado en SCORM.

A. Metamodelo origen PIM: Experimento

La relación de los trabajos descritos en donde se analizan LV [12], permite detallar las características que contribuyen a su diseño, desarrollo e implementación; se destaca la definición de los LV como un tipo de aplicación educativa y científica, que debe cumplir preferiblemente con los lineamientos de un modelo pedagógico y la estandarización que permite la interoperabilidad entre plataformas de contenidos educativos SCORM. Los LV deben permitir realizar experimentos en donde se pueda manejar y controlar remotamente instrumentación real, efectuar simulaciones e interacciones con realidad virtual. Los datos e información que se manejan cumplen con el ciclo de un sistema: entrada de datos, proceso, salida.

Se propone el PIM denominado Experimento, que muestra la estructura y funcionalidades de las aplicaciones que ofrecen los LV en diferentes contextos académicos y científicos; es una propuesta de metamodelo libre de

detalles de implementación y aspectos tecnológicos que representa principalmente el modelo de negocio, se ha representado mediante un metamodelo ecore. El diagrama ecore de la Figura 3, contiene cuatro eclass (clases ecore): Experimento L, Laboratorio, Proceso y Dato.

Experimento L, representa el objetivo principal de los LV que es realizar experimentos y a partir del procesamiento de datos de entrada, se obtienen datos de salida como parte del resultado del procesamiento esperado. Un experimento tiene asociado un nombre, descripción, ayuda, investigador y un tipo de experimento que genera la opción de que este metamodelo PIM sea implementado posteriormente en cualquier plataforma que sea definida en el PSM. La EClass Laboratorio, hace referencia a su caracterización con su nombre y descripción. Proceso, contiene las operaciones a realizarse, los métodos que se emplean para obtener resultados a partir de diferentes datos que ingresan como colecciones o de manera individual. Un proceso estructura la interacción de datos que se necesitan (Cuerpo lógico). Dato, representa objetos identificados como datos que poseen un nombre, un tipo y un valor específico, es decir unas propiedades.

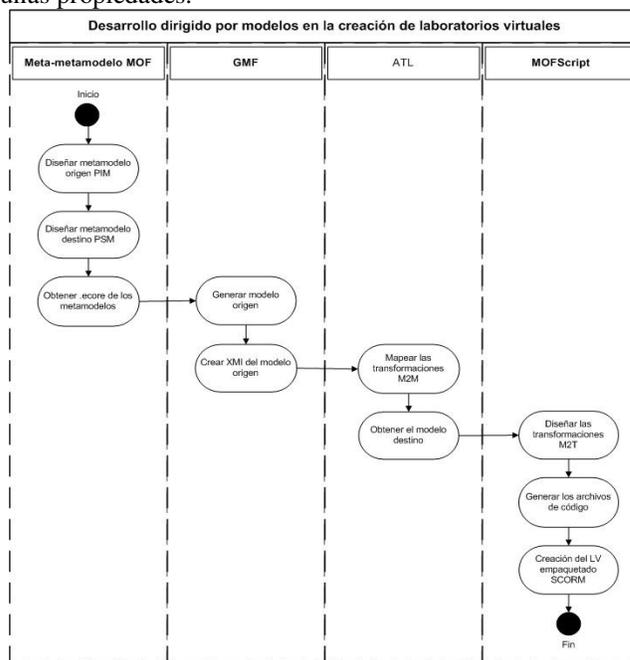


Fig 1. Diagrama de Actividades de MDD en la creación de LV

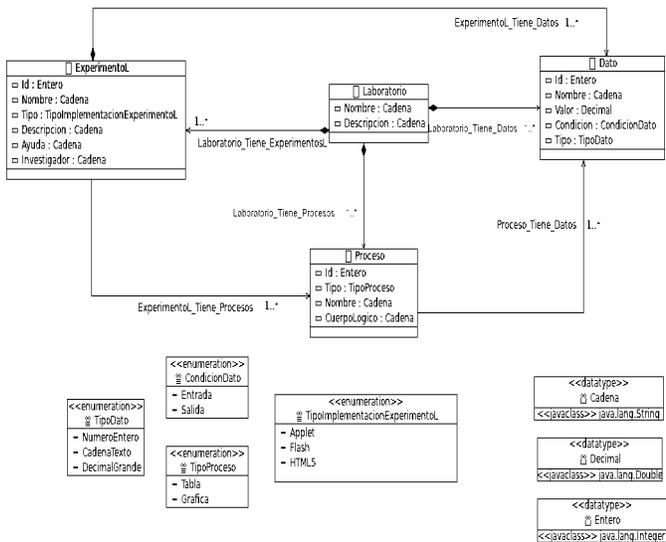


Fig 2. Metamodelo Origen PIM: Experimento

Se definen adicionalmente las enumeraciones <<enumeration>>, que son listas de valores contenidos en un esquema físico. Son variables estáticas en el modelo definidas previamente para ser relacionadas a una eclass mediante un tipo de dato o atributo. En el PIM existen cuatro listas <<enumeration>>, Tipo Implementacion Experimento L, Tipo Proceso, Condicion Dato y Tipo Dato que contienen diferentes valores que son usados en algunas propiedades de cada eclass representada en el PIM y el PSM. También se definen los datatype: Cadena, Entero y Decimal que son representados con el estereotipo <<datatype>> siendo herederos de tipos de datos de las clases nativas java.lang.*. Las EReference (relaciones ecore) corresponden a la comunicación que existe entre las eclass pertenecientes al metamodelo.

B. Metamodelo destino PSM: LaboratorioVirtual

El PSM Destino o metamodelo destino, es el metamodelo que permite que la aplicación sea desarrollada en cualquier plataforma. El PIM o metamodelo origen es una abstracción de los LV que se ha denominado Experimento.

El diagrama ecore de la Figura 4, contiene las clases, relaciones, tipos de datos y enumeraciones que se necesitan para la elaboración del LV bajo una plataforma específica: La EClass Aplicación, representa la información específica de la aplicación, como el nombre y el tipo, permite identificar la aplicación LV, es la encargada de contener la lógica del negocio y su respectiva interfaz gráfica, además representa el tipo de aplicación entre Applet, Flash o HTML, etc.

La EClass Contenedor, hace parte de los elementos contenedores de objetos en las capas de presentación de la

interfaz gráfica. Gráfica, que hace parte de las clases que representan las herramientas graficas que contendrán los datos del LV. La EClass Grafica es la encargada de dar visualización grafica a los resultados obtenidos en el proceso de variables. Tabla, representa las colecciones de datos que contienen los datos iniciales y los datos resultados de procesos cíclicos o lineales en la experimentación. Permite organizar los datos y mostrarlos en forma de tabla. Contiene una operación (EOperation) inicial básica llamada Generar Lista, la cual se encarga de cargar los datos de una colección en la Tabla.

Columna, representa una columna de datos dentro de las colecciones obtenidas como resultados en los experimentos, están contenidas por una tabla y se usan para mostrar grupos de datos de un mismo espacio de la Colección. Celda, está contenida dentro de una columna, en cada una de las posiciones de la columna se encuentra una celda, esta se marca en una posición (X,Y) un contenido y una operación (EOperation) que permite validar un tipo de dato (Testear un Numero). CajaDato, contiene el valor de una o más variables, se puede contener desde una celda en una tabla o desde un contenedor directamente. Método, contiene la lógica del negocio, es decir las operaciones o procesos que se realizan en el LV, el cual permite obtener resultados diferentes con operaciones diferentes. Estos son los que se asignan dentro de una clase java y que contienen el cuerpo lógico, vienen de la EClass en el PIM llamado Proceso. Variable, es aquel dato organizado con una representación clara dentro del negocio de Experimento (LV). Una variable contiene, un nombre, un tipo de datos y un valor, dos operaciones claramente identificadas para dejar en cero su valor y para inicializarlas.

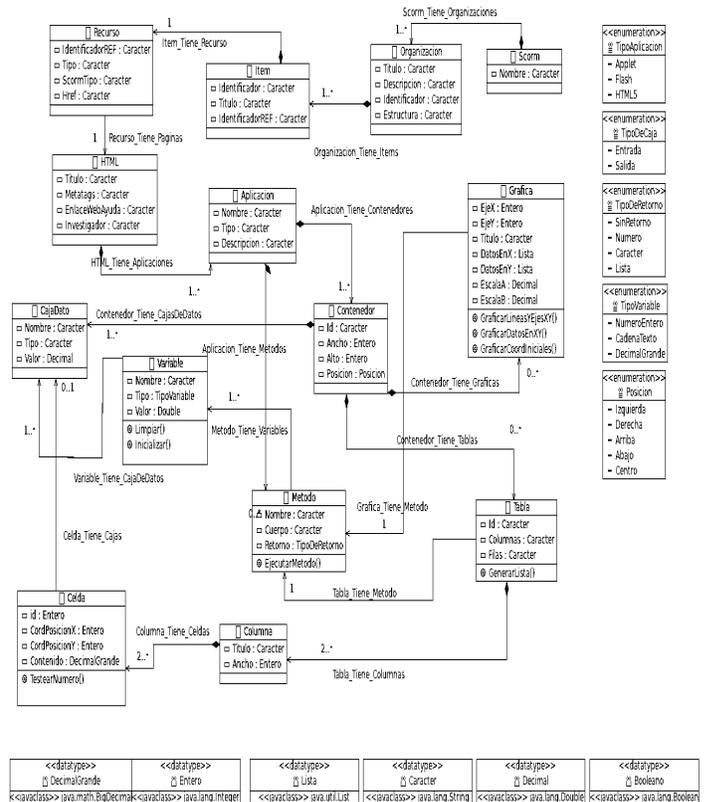


Fig 3. Metamodelo Destino PSM: Laboratorio Virtual

La EClass Scorm, que contiene los elementos a incluir en el LV como recurso físico permitiendo integración con plataformas LMS. Esta EClass determina el XMI que configura la organización como objeto de aprendizaje del LV. Organización, contiene información referente a los ítems, que son los contenedores virtuales de recursos de diferentes plataformas. Ítem, representa el nodo estructural del Scorm (LV) para organizar recursos y contenidos, es aquí donde se referencia la ubicación de los recursos. Recurso, contiene la información del objeto aplicación que se desarrolla en el proyecto e indica la ruta del HTML. La EClass HTML, hace parte de la estructura de la aplicación pero se referencia desde los objetos del Scorm, se representa de color amarillo también, porque hace parte de los archivos necesarios para la integración como SCORM a un LMS. Las relaciones en el metamodelo destino (PSM LV) al igual que el PIM se representan en el diagrama para señalar la multiplicidad que mantiene al ser usadas entre EClass. Las enumeraciones y los tipos de datos primitivos, cuentan con dos nuevos tipos de datos “Bigdecimal, boolean y List”, estos permiten almacenar resultados con mayor precisión y en cantidad. Adicionalmente aparecen dos enumeraciones para darle mayor nivel de configuración al Metamodelo: “TipodeRetorno” para los métodos y “PosiciónEnLayout” para la ubicación de los contenedores.

El PSM se orienta a la obtención de una aplicación tipo applet java empaquetado en SCORM que define como representar y vincular los elementos en un archivo XML. El objetivo es generar un objeto de aprendizaje que sea accesible, inter-operativo, reutilizable, adaptable y asequible en un LMS.

C. Transformaciones

La transformación de modelos constituye una fase clave para MDD, las reglas de mapeo, definen los mecanismos para el paso de un modelo origen a un modelo destino. Se realizan con el fin de validar si metamodelo destino (Laboratorio Virtual) cumple con lo propuesto en el metamodelo origen (Experimento).

Se etiqueta o marcan los modelos para guiar los procesos de transformación y lograr diferenciar por ejemplo datos persistentes de datos temporales, procesos remotos o locales. Existen dos tipos fundamentales de transformaciones: la transformación de un PIM en un PSM, denominada M2M y la transformación de un PSM en el código fuente de la aplicación (M2T). En la Tabla 2, se determinan las reglas que se mapean del metamodelo origen Experimento al metamodelo destino LaboratorioVirtual para posteriormente programar como reglas en ATL.

Transformación Modelo a Modelo

El proceso de transformación M2M, se realiza con la plataforma de desarrollo

Eclipse Indigo. Es llevada a cabo mediante el lenguaje ATL (Atlas Transformation Language) el cual permite la definición de reglas de mapeo entre metamodelos definidos en ecore [13].

TABLA 2
RESUMEN MAPEO DE TRANSFORMACIONES

Eclass Metamodelo	PIM Origen	Eclass Metamodelo PSM Destino
Experimento		Aplicación
		Contenedor
Dato		Columna
		Celda
		Variable
		CajaDato
Proceso		Metodo
		Gráfica
		Tabla
Laboratorio		Aplicación

El procedimiento se realiza para verificar si el metamodelo destino es el resultado esperado del metamodelo origen propuesto, se instancia un modelo del metamodelo origen y se transforma a un modelo destino con ATL. ATL al igual que un lenguaje de programación permite mediante una sintaxis definida introducir modelos y transformarlos. La transformación de modelos se realiza con el fin de validar los modelos, es decir verificar si una instancia del metamodelo origen puede generar una instancia del metamodelo destino.

Transformación Modelo a Texto

Después de realizar la validación del metamodelo destino, utilizando la transformación de modelos y se obtienen los resultados esperados, el metamodelo destino está listo para ser transformado a código. Esto se realiza mediante transformaciones M2T, utilizando MOFScript se elaboran los archivos físicos que corresponden al código fuente de la aplicación, además es un objeto de aprendizaje LV tipo Applet con la estructura SCORM.

MOFScript permite operar los componentes del metamodelo ecore que son proporcionados en EMF (Eclipse Modeling Framework) y transformar estas EClass, EAttributes, EOperations, ERelations en líneas de código y archivos respectivos que varían de acuerdo a la plataforma definida para el LV. Se han logrado generar los recursos código de la aplicación LV a través de archivos M2T cumpliendo con la estructura del estándar SCORM.

V. CONCLUSIONES

Los LV empaquetados con SCORM, como recurso pedagógico permiten aumentar las posibilidades académicas contribuyendo a la experimentación sin limitantes de espacio y tiempo, además pueden integrarse con otros recursos dentro de diferentes LMS.

MDD y su MDA, son aplicables en el contexto de diseño de LV reduciendo significativamente el tiempo y esfuerzo en la construcción de nuevos LV independientes del contexto académico. Actualmente se está dando mayor interés al modelado en el desarrollo de cualquier tipo de aplicación, debido a la facilidad que ofrece un buen diseño tanto a la hora de desarrollar como al hacer la integración y mantenimiento de sistemas de software.

La propuesta de MDD apoya el desarrollo de software, basándose en la separación entre la especificación de la funcionalidad del sistema y la implementación de dicha funcionalidad usando plataformas de implementación específicas. Algunos de estos aspectos están bien fundamentados y se están empezando a aplicar con éxito, otros sin embargo están todavía en proceso de definición.

La utilización de modelos en el ámbito educativo, proporciona una adecuada abstracción permitiendo a los diseñadores, desarrolladores, docentes e investigadores utilizar elementos comunes en el diseño de LV como actividades, flujos de aprendizaje y adaptaciones basadas en las evaluaciones; logrando optimizar recursos, talento humano, conocimiento y costos.

REFERENCIAS

- [1] Rosado L. y Herreros J.R., 2009, Nuevas aportaciones didácticas de los laboratorios virtuales y remotos en la enseñanza de la Física; En International Conference on Multimedia and ICT in Education.
- [2] Pelayo B., 2007, TALISMAN: Desarrollo ágil de Software con Arquitecturas Dirigidas por Modelos.
- [3] Vallecillo A., 2007, Desarrollo de Software Dirigido por Modelos. Visitado el 17 Enero 2011. Disponible en: <http://alarcos.inf-cr.uclm.es/per/fruiz/cur/santander/avallecillo-dsdm.pdf>
- [4] ATL User Manual -version 0.7, 2007, ATLAS group LINA & INRIA.
- [5] Fátima C de Souza M., de-Castro Filho J. y Andrade R., 2010, Model-Driven Development in the Production of Customizable Learning Objects. Advanced Learning Technologies (ICALT), IEEE International Conference.
- [6] Wang H., Zhang D. y Zhou J., 2003, MDA-based development of e-learning system" Computer Software and Applications Conference, Proceedings. 27th Annual International.
- [7] Bizonova Z., Ranc D. y Drozdova M., 2008, Model driven e-learning platform integration. Visitado 26 Noviembre 2011. Disponible en: <http://ceur-ws.org/Vol-288/p02.pdf>
- [8] Montenegro C., Gaona P., Cueva J. y Sanjuan O., 2011, Aplicación de ingeniería dirigida por modelos (MDA), para la construcción de una herramienta de modelado de dominio específico (DSM) y la creación de módulos en sistemas de gestión de aprendizaje (LMS) independientes de la plataforma; En: Revista DYNA Universidad Nacional de Colombia año 78, Nro. 169, pp. 43-52.
- [9] Doderó J., Martínez A., Ruiz I., Cornejo C. y León A., 2010, FlexoLD: Un Lenguaje Específico de Dominio para Diseños de

- Aprendizaje; En: Actas de los Talleres de las Jornadas de Ingeniería del Software y Bases de Datos, Vol. 4, No. 3.
- [10] Malek, J., Laroussi, M., Derycke, A. y Ben Ghezala, H., 2010, Model-Driven Development of Context-aware Adaptive Learning Systems; En: 10th IEEE International Conference on Advanced Learning Technologies.
 - [11] OMG, Object Management Group. Visitado el 12 Enero 2011, Disponible en: <http://www.omg.org/mof/>
 - [12] Pascuas Y., 2010, Laboratorios virtuales integrados con tecnología grid para la Universidad de la Amazonia; En: Revista Ingenierías & Amazonia 3(1), pp: 40-53.
 - [13] ATL, A model transformation technology. Visitado el 17 Enero 2011, Disponible en: <http://www.eclipse.org/atl/>