

# Un entorno de desarrollo visual para reforzar la comprensión de algoritmos y estructuras de datos

Alejandra Schiavoni, Laura Fava, Jorge Rosso

LINTI - Laboratorio de Investigación en Nuevas Tecnologías Informáticas.

Facultad de Informática. Universidad Nacional de La Plata

Calle 50 esq. 120, 2do Piso. Tel: +54 221 4223528

{ales, lfava, jrosso}@info.unlp.edu.ar

## Resumen

La enseñanza de Algoritmos y Estructuras de Datos, es fundamental en las carreras de Ciencias de la Computación ya que éstas representan las posibles maneras de organizar los datos y el acceso a los mismos de manera eficiente. Estos temas se incluyen en las asignaturas del área de programación en los primeros años de las carreras. En general, se observa que a los estudiantes de estos cursos, les resulta complejo comprender los conceptos relacionados con estructuras de datos avanzadas y sus algoritmos de acceso.

En este contexto, es de mucha utilidad para los estudiantes, contar con herramientas que acompañen el aprendizaje de esta temática. Este artículo presenta una línea de investigación cuyo objetivo central es diseñar una herramienta educativa innovadora destinada a mejorar la comprensión de algoritmos y estructuras de datos a través de la visualización de las estructuras de datos y de la exploración de la ejecución de sus propios algoritmos.

**Palabras clave:** algoritmos y estructuras de datos, visualización de estructuras de datos, recursión, programación, Programación Orientada a Aspectos, JAVA.

## Contexto

La Facultad de Informática cuenta actualmente con más de 4000 alumnos y 300 docentes [1]. Se dictan tres carreras de grado: Licenciatura en Informática, Licenciatura en Sistemas y Analista Programador Universitario. La asignatura *Algoritmos y Estructuras de Datos* corresponde a una materia obligatoria de segundo año de las tres carreras que cuenta con más de 400 alumnos inscriptos. Además, en conjunto con la Facultad de Ingeniería se dicta la carrera de Ingeniería en Computación desde el año 2011. En el plan de la carrera se encuentra la asignatura *Programación 3*, que es de carácter obligatorio, corresponde a segundo año y cuenta con más de 100 alumnos inscriptos.

Dentro de los contenidos de ambas asignaturas se encuentra el estudio de estructuras de datos básicas y avanzadas tales como listas, pilas colas, árboles generales y binarios y grafos. Los alumnos aprenden los conceptos básicos de las representaciones y los algoritmos de acceso. Durante la cursada los alumnos deben implementar las estructuras de datos vistas usando el lenguaje de programación Java.

La herramienta propuesta podría ser usada como un complemento novedoso de las clases teóricas y prácticas en las que el alumno incorpora los conceptos y los aplica en el desarrollo de sus propios

algoritmos.

El proyecto descrito en este artículo se desarrolla en el Laboratorio de Investigación en Nuevas Tecnologías Informáticas, LINTI de la Facultad de Informática de la UNLP y está enmarcado en el proyecto 11-F014 “*Innovación en TICs para el desarrollo de aplicaciones en educación, inclusión, gobierno y salud*”, acreditado en el marco del Programa de Incentivos, bajo la dirección del Lic. Javier Díaz.

## Introducción

En Ciencias de la Computación el estudio de las Estructuras de Datos, es fundamental dado que las mismas son utilizadas en el desarrollo de casi todo tipo de software. Estudiar estructuras de datos implica estudiar las diferentes maneras de almacenar y organizar datos para facilitar el acceso y modificación [2]. El uso de una herramienta que visualice las diferentes estructuras de datos y el comportamiento de los algoritmos sobre ellas resulta de suma importancia, especialmente para los estudiantes de los primeros años de las carreras de informática. Actualmente existen varias herramientas dedicadas a reforzar la enseñanza de las estructuras de datos a través de una representación visual que ayuda a disminuir la abstracción inherente al estudio de las mismas [3,4,5]. Sin embargo algunas sólo consisten en animaciones de las operaciones sobre las estructuras a partir de un código inaccesible, otras si bien permiten visualizar la ejecución de un código propio, fuerzan a usar alguna técnica *intrusiva* -decorando o adaptando el código fuente original de la estructura- o *conductista* -usando un lenguaje no estándar/propietario-.

Este artículo presenta una línea de investigación cuyo objetivo central es

diseñar una herramienta educativa innovadora destinada a la enseñanza de las estructuras de datos, atendiendo a las dificultades observadas en los alumnos en cuanto a la comprensión y asimilación del comportamiento de las diferentes estructuras de datos.

La herramienta a implementar está orientada a ofrecer una funcionalidad amplia y diversa basada en la posibilidad de contar con una representación visual de las estructuras de datos y de inspeccionar el estado de ejecución de algoritmos definidos por los alumnos. Esto último resulta novedoso y a la vez muy útil, en especial en algoritmos recursivos, en los que va a ser posible realizar un seguimiento detallado del estado de cada una de las llamadas realizadas.

En la actualidad, después de un análisis profundo de las tecnologías disponibles, se está implementando un primer prototipo con las herramientas seleccionadas. En principio, como el lenguaje utilizado para la implementación de las estructuras de datos en las asignaturas Algoritmos y Estructuras de Datos y Programación 3 es JAVA, se tuvieron en cuenta soluciones basadas en este lenguaje. Por un lado, en las prácticas, los estudiantes utilizan Eclipse como entorno de desarrollo, motivo por el cual se decidió implementar un plug-in<sup>1</sup> para integrarlo con este entorno, en vez de desarrollar una herramienta desde cero.

De esta manera, los estudiantes seguirán usando Eclipse, un entorno de amplia aceptación en la comunidad de desarrolladores JAVA, pero con la

---

<sup>1</sup> Un plug-in es una componente de software, es la unidad funcional más pequeña de la plataforma Eclipse que puede ser desarrollada de manera separada e integrada al entorno Eclipse para potenciar su funcionalidad.

potencialidad del plug-in. Podrán ejecutar sus programas de la manera tradicional o elegir la opción ejecutar bajo las funcionalidades del plug-in.

Por otro lado, para lograr la visualización de los algoritmos, el seguimiento de los mismos, el paso a paso en determinados métodos, y el estado de las variables en los sucesivos llamados recursivos, se decidió utilizar Programación Orientada Aspectos con el objetivo de no ser intrusivo en el código fuente de las estructuras de los estudiantes. Los aspectos que definen qué código sería interceptado, son independientes del código fuente implementado por los estudiantes [6].

Las funciones ofrecidas por la herramienta serán:

- Visualización del estado de ejecución de los códigos fuentes Java implementados por los alumnos
- Acceso al estado actual de las variables y al estado de las mismas en cada uno de los llamados recursivos anteriores.
- Visualización de las estructuras de datos seleccionadas por el alumno en un panel específico.
- Representación animada de la ejecución de los algoritmos implementados por los alumnos de las operaciones comunes que se realizan sobre las estructuras de datos: inserción, eliminación, recorridos.

## **Líneas de Investigación, Desarrollo e Innovación**

Las líneas de investigación, desarrollo e innovación que se llevan a cabo en este proyecto se basan en dos ejes principales: uno vinculado a la enseñanza de las estructuras de datos y el otro relacionado con tecnologías avanzadas de JAVA.

El primer eje incluye aspectos relacionados con el estudio, uso y

comprensión de las estructuras de datos y de técnicas de programación, teniendo en cuenta las dificultades que ellas presentan en los alumnos de los primeros años de las carreras. Respecto a las estructuras de datos, es importante brindar una herramienta que les permita a los alumnos experimentar de manera de reducir la abstracción que su estudio conlleva. Respecto a las técnicas de programación, se analizan diferentes metodologías de enseñanza que apuntan a mejorar la comprensión y aplicación en casos concretos. La herramienta fue pensada para atacar los puntos críticos que se presentan al momento de aprender una nueva técnica de resolución de problemas. Tal es el caso de la recursión que involucra una forma distinta de pensar y encarar la solución a problemas [7, 8].

El segundo eje incluye el análisis y aplicación de tecnologías avanzadas basadas en el lenguaje Java para la implementación de un entorno de desarrollo visual, fácil de usar y que ayude a reforzar la comprensión de algoritmos y estructuras de datos.

Como ya se mencionó, la herramienta será implementada usando tecnologías JAVA y para se establecieron las siguientes tres líneas:

### ***1. Desarrollo de un plug-in y análisis para su integración con Eclipse.***

Eclipse es un framework de código abierto para la creación de entornos de desarrollo integrado utilizando un conjunto de herramientas y componentes GUI pre-construidas. Desde su lanzamiento, ha generado gran interés en muchos desarrolladores. El Workbench de Eclipse se compone de:

- Editores: los diferentes editores son paneles asociados con tipos de documentos específicos.
- Vistas: son paneles que apoyan a los

editores.

- Perspectivas: son combinaciones de vistas específicas y editores.

La idea de este plug-in es que incorpore una nueva Perspectiva a Eclipse, donde una vista muestre el código fuente java, otra el estado de las variables en los diferentes llamados recursivos (de métodos interceptados por el aspecto), otra la visualización de las estructuras de datos y algoritmos asociados [9].

## 2. Análisis de las características de la programación orientada a aspectos y su aplicación en la herramienta

Desde un principio se busco poder visualizar estructuras de manera independiente del código desarrollado por los alumnos. Es decir, se buscaba alguna manera de indicarle a la herramienta las clases y métodos que permitirían describir la composición de la estructura y los datos contenidos en ella, sin imponer algún requerimiento de convención de nombres o alguna inclusión de anotaciones JAVA que indique de qué tipo de estructura se trata y cuáles son sus métodos de acceso a la información.

Por otro lado utilizar archivos descriptores (con algún formato estándar como XML o JSON) asociados con las estructuras utilizadas, resultaría en una solución un tanto estática.

De esta manera llegamos a la conclusión de que la Programación Orientada a Aspectos nos permitiría resolver este tema de una manera más elegante y dinámica.

Con AspectJ es posible indicar puntos de un programa en los cuales uno desea examinar los objetos y realizar alguna determinada acción sin modificar el código original. La especificación de estos puntos de intervención (Pointcuts) y

las acciones que se llevarán a cabo se escriben en un aspecto (Aspect).

El momento en que se entrelaza (weaving) el código origen con los aspectos se puede dar en tres momentos: en compilación (compile-time weaving), durante el empaquetado de un archivo JAR, o en ejecución (load-time weaving). La Fig. 1 muestra una esquematización del funcionamiento de AOP para entremezclar los aspectos con los programas.

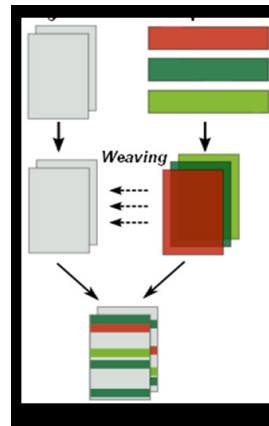


Fig. 1: Weaving de aspectos con el código fuente

## 3. Estudio de las componentes de interfaz de usuario provistas por JAVA para la implementación de la visualización de las estructuras de datos y la animación de algoritmos.

La plataforma estándar de JAVA, provee una librería de componentes gráficos para crear GUIs de alta calidad. Se profundizarán tópicos avanzados de Java 2D, Swing y técnicas de animación para lograr una GUI amigable, expresiva que capture la atención de los estudiantes [10].

## Resultados y Objetivos

Como se mencionó anteriormente, la posibilidad de contar con una herramienta de apoyo al proceso de enseñanza de conceptos relacionados a la programación

como son las estructuras de datos y la técnica de recursión, resulta novedoso y más que útil en los años iniciales.

El proyecto presentado en este artículo está en una etapa inicial, razón por la cual los resultados representan las primeras pruebas de la tecnología a utilizar y de la integración entre ellas. Los resultados alcanzados en esta etapa son:

- Análisis y diseño visual de la herramienta propuesta en este artículo.
- Implementación e integración de un plug-in para Eclipse con funcionalidad mínima para testear la incorporación de nuevas funcionalidades en el ambiente.
- Estudio y aplicación de la Programación Orientada a Aspectos a través de la definición de un *Aspecto* básico y general que inspecciona algoritmos.

Los objetivos inmediatos se relacionan con la implementación de la visualización del seguimiento de la ejecución de los algoritmos:

- Analizar las distintas componentes de interfaz de usuario de SWING, y seleccionar la más apropiada para visualizar en forma clara las sucesivas llamadas de algoritmos recursivos.
- Implementar las funcionalidades del plug-in que representa el seguimiento de la ejecución de algoritmos recursivos, y su visualización en la componente de interfaz de usuario seleccionada.
- Implementar la lógica del *Aspecto* para que cumpla con la funcionalidad de inspeccionar métodos/algoritmos seleccionados por el alumno.

Los objetivos posteriores están relacionados con la representación

animada de las estructuras y algoritmos implementados por los estudiantes. Esto implica un trabajo intenso de desarrollo de Interfaz de Usuario Gráfica, teniendo en cuenta que tanto la visualización de las estructuras como la animación de los algoritmos depende en forma directa del código propio que está ejecutando el alumno y se resuelve en tiempo de ejecución.

Como objetivo final del proyecto se prevé el testeado de la herramienta desarrollada con un grupo reducido y voluntario de estudiantes para evaluar el impacto en su aprendizaje. Se espera poder usarla como complemento innovador a las clases teóricas, explicaciones prácticas y textos recomendados por la Cátedra.

## Formación de Recursos Humanos

El equipo de trabajo de la línea de I+D+i presentada en este artículo se encuentra formado por docentes investigadores categorizados del LINTI y alumnos avanzados de las carreras de Lic. en Informática y Lic. en Sistemas de la Facultad de Informática.

Basada en las líneas de investigación de este proyecto se está comenzando a desarrollar una tesina de grado.

## Referencias

- [1] Anuario Estadística 2013/2014: <http://www.unlp.edu.ar/indicadores>
- [2] T. Cormen, C. Leiserson, R. Rivest, C. Stein; *Introduction to Algorithms*, Second Edition, ISBN 0-07-013151-1 (McGraw-Hill), 2001.
- [3] T. Chen and T. Tarek Sobh; *A Tool for Data Structure Visualization and User-defined Algorithm Animation*, Department of Computer Science and engineering, University of Bridgeport, USA. Accesible en: <http://www1bpt.bridgeport.edu/~risc/pdf/jp29.pdf>

- [4] J. Cross, D. Hendrix; *Workshop jGRASP: An Integrated Development Environment with Visualizations for Teaching Java in CS1, CS2, and Beyond*, 36<sup>th</sup> ASEE/IEEE Frontiers in Education Conference, ISBN 1-4244-0256-5, 27 -31 Oct. 2006.
- [5] G. Cardim, I. Marcal, C. de Sousa, D. de Campos, C. Marin; A. do Carmo, D. Toledo, A. Saito, R. Correia, R. Garcia, *Teaching and Learning of Data Structures Supported by Computer: An Experience with the CADILAG tool*. Information Systems and Technologies (CISTI), 2012 7th Iberian Conference, Madrid, p:1-5. ISBN 978-1-4673-2843-2
- [6] J. Gradecki, N. Lesiecki, *Mastering AspectJ: Aspect-Oriented Programming in Java*. ISDN 0-471-43104-4, Wiley, 2003.
- [7] J. Zhang, M. Atay, E. Smith, E. Caldwell, E. Jones, *Using a Game-Like Module to Reinforce Student Understanding of Recursion*, Frontiers in Education Conference (FIE), IEEE, ISBN: 978-1-4799-3921-3, Madrid, Spain, 22 – 25 Oct, 2014.
- [8] A. Chaffin, K. Doran, D. Hicks, T Barnes, *Experimental Evaluation of Teaching Recursion in a Video Game*, Proceedings of the 2009 ACM SIGGRAPH, Shimposium of Video Games, 79-86.
- [9] A. Blewitt, *Eclipse 4 Plug-in Development by Example: Beginner's Guide*. ISBN 978-1-78216-032-8, Packt Publishing, 2013.
- [10] C. Haase and R. Guy, *Filthy Rich Clients: Developing Animated and Graphical Effects for Desktop Java Applications*, ISBN-10: 0132413930, Addison Wesley, 2007.