

PLEIAD: Explorando Nuevos Lenguajes para Mejores Programas

PLEIAD (Programming Languages and Environments for Intelligent, Adaptable and Distributed Systems) está dedicado a explorar en qué forma los lenguajes de programación y sus ambientes de desarrollo pueden permitir la construcción de software evolucionable y adaptable. Considerando fundamentalmente contextos desafiantes como la computación distribuida y ubicua.



Los lenguajes de programación a medida que son utilizados van adquiriendo cierta popularidad. Hoy Java y C# son muy populares. Su antecesor en popularidad fue C++ y los antecesores a este fueron C y FORTRAN sucesivamente. Los lenguajes de programación son diseñados para satisfacer determinados ambientes tecnológicos de ejecución, con el propósito de facilitar el desarrollo de ciertos tipos de soluciones. A medida que la tecnología va cambiando, los lenguajes de programación van quedando obsoletos. Es por esto que un lenguaje de programación puede verse afectado por cambios y evoluciones a lo largo de su vida. La comunidad de Ingeniería en Software y Lenguajes está constantemente en búsqueda de lenguajes de programación que satisfagan las nuevas tecnologías. Por ejemplo, ahora que Java ya tiene más de 10 años, se está explorando lo que a veces se denomina el "mundo post-Java". Otro ejemplo claro de esta búsqueda es la utilización de forma más masiva de los lenguajes de programación dinámicos como Python y Ruby.

Es importante señalar que los lenguajes de programación tienen un rol fundamental en la forma en la cual los desarrolladores manejan la complejidad de los sistemas computacionales. Por ejemplo, la programación por objetos y por componentes es la base sobre la cual muchos sistemas de gran envergadura están contruidos. La ubicuidad incesante de la computación a todos los niveles de la sociedad implica más complejidad para el software. Esto llama a desarrollar mejores formas lingüísticas que manejen esa complejidad. Un ejemplo de este fenómeno es la emergencia del paradigma de Programación Orientada a Aspectos (AOP), el cual permite la definición modular de preocupaciones que son transversales a los objetos de un sistema, como la seguridad o la coordinación de actividades. Con AOP, se logra mejor desacoplamiento y reusabilidad de los componentes y objetos. Esto también permite mejor adaptabilidad del sistema en forma dinámica.

Principales Áreas de Investigación

En el laboratorio PLEIAD se investigan al día de hoy tres áreas principales:

- Programación por aspectos (AOP)
- Depuración y comprensión de programas

- Programación de sistemas de computación pervasiva.

A continuación detallamos brevemente estas áreas.

Programación por Aspectos (AOP). AOP introduce nuevos mecanismos para definir software en forma más modular y más adaptable. AOP tiene una fuerte herencia de los trabajos sobre reflexión computacional, uno de los temas sobre los cuales integrantes de PLEIAD han trabajado previamente. En el contexto de AOP, PLEIAD participa de la investigación sobre definición de lenguajes AOP. En efecto, aunque la comunidad ya cuenta con un lenguaje AOP de alcance industrial llamado AspectJ, muchos temas quedan por explorar para revelar toda la potencialidad de los aspectos en el desarrollo de software. Actualmente dos temáticas principales se desarrollan en PLEIAD: la construcción y definición de lenguajes AOP, incluyendo lenguajes específicos a cierto dominio (por ejemplo un lenguaje dedicado a la definición del manejo de transacciones), y el mejoramiento de los mecanismos provistos para el control del impacto de un aspecto sobre un sistema dado. Esta investigación se hace con una mirada particular a las problemáticas asociadas a la introducción de aspectos en sistemas complejos, es decir, concurrentes y distribuidos.

Depuración y comprensión de programas. La depuración consiste en ayudar al desarrollador a encontrar errores en un programa. De forma más general, esto pertenece al área de "entendimiento de programas" (program understanding). Es decir, cómo ayudar a un ser humano a entender con suficientes detalles lo que pasa en un programa como para ser capaz de modificarlo o corregirlo, manteniendo un nivel de abstracción que le permita aprender el programa sin ser sobrepasado por su complejidad. En particular, en PLEIAD se está trabajando sobre un sistema de Depuración Omnisciente llamado TOD (Trace-Oriented Debugger). Es decir un depurador donde se puede navegar en la historia de ejecución de un programa, tanto hacia adelante como atrás en el tiempo. Esto permite recorrer vínculos causales que son muy difíciles de reconstituir en los depuradores actuales. Mediante la utilización de un depurador omnisciente podemos saber en qué momento y en qué contexto, determinada variable fue asignada un valor inválido, lo que con un depurador normal no se puede saber.

Computación Pervasiva. La computación pervasiva o "inteligencia ambiental", se refiere al desarrollo de sistemas computacionales para usuarios móviles con aparatos móviles, en los cuales la integración de los sistemas en la vida diaria es lo más transparente posible. Esto requiere de sistemas que sepan captar su ambiente de ejecución y adaptarse a ello en forma dinámica. Programar dichos sistemas con lenguajes tradicionales implica un nivel de complejidad enorme para manejar todos los detalles relacionados con la naturaleza volátil del ambiente y de las conexiones. PLEIAD trabaja en lenguajes dedicados, que proveen abstracciones adecuadas para que el programador pueda especificar tanto la percepción del ambiente cómo la adaptación del sistema. En estos momentos se experimenta con el lenguaje AmbientTalk, desarrollado en el laboratorio PROG de la Vrije Universiteit Brussel (Bélgica) con el cual PLEIAD está colaborando. También se exploran técnicas de inteligencia artificial aplicadas a computación pervasiva.

Historia, Logros y Desafíos

El laboratorio PLEIAD fue inaugurado recién en noviembre de 2007, con la presencia de Ron Goldman, Ingeniero de Investigación de Sun Labs en EE.UU., a cargo del proyecto de Sun SPOTs; unos aparatos para inteligencia ambiental dotados de sensores y que corren una máquina virtual Java.

PLEIAD está inicialmente formado por dos profesores full-time, Johan Fabry y Éric Tanter, cuatro estudiantes de doctorado y un estudiante de magister. Y se están integraron nuevos miembros, principalmente provenientes de posgrado.

A pesar de ser un laboratorio joven, PLEIAD ya está logrando exponerse a nivel internacional en las distintas áreas en que se desempeña. En la conferencia ACM sobre desarrollo de software orientado a aspectos, organizada en abril (AOSD 2008), se presentó un artículo técnico sobre "scoping" de aspectos dinámicos, se organizó un workshop sobre lenguajes de aspectos específicos a dominios, y se hizo una demostración del Depurador TOD. Un artículo técnico sobre el depurador omnisciente ha sido publicado en OOPSLA 2007, la conferencia de referencia del ACM en el área, y otro, sobre uso de TOD para programas orientados a aspectos, se presentó en el simposio de computación aplicada del ACM en marzo (SAC 2008). También se organizó un workshop sobre computación inspirada en la biología en diciembre en Valparaíso (BIC 2007), Chile, que contó con la presencia de varios expertos internacionales, tanto de Europa como de EE.UU. PLEIAD está involucrado como miembro fundador de la red de colaboración sudamericana Latin AOSD, y participa en otros proyectos de cooperación internacional, en particular con el INRIA de Francia.

Información y contacto en <http://pleiad.dcc.uchile.cl/>