

Workshop en Técnicas de Programación Científica: Enseñando a desarrollar software colaborativo para la ciencia

Pablo Nicolás Alcain¹, Cecilia Jarne², Rodrigo Lugones¹ y María Graciela Molina³

¹Departamento de Física, Facultad de Ciencias Exactas y Naturales, Universidad de Buenos Aires, Buenos Aires, Argentina e Instituto de Física de Buenos Aires (IFIBA) - CONICET, Buenos Aires, Argentina;

²Universidad Nacional de Quilmes e Instituto de Física de Buenos Aires (IFIBA) - CONICET, Buenos Aires, Argentina;

³ Departamento de Ciencias de la Computación, Facultad de Ciencias Exactas y Tecnología, Universidad Nacional de Tucumán, Tucumán, Argentina.

Resumen

A partir del año 2016 iniciamos el dictado del Workshop en Técnicas de Programación Científica que se reedita anualmente en diferentes localidades del país. Durante el mismo enseñamos herramientas formales y actualizadas para el desarrollo de software, con un enfoque específico en el software científico. El concepto de eficiencia en este tema actualmente se ciñe solo a la velocidad de ejecución del programa, dejando de lado otros factores claves como la colaboración, versatilidad, reusabilidad y facilidad de mantenimiento. Mediante charlas teóricas seguidas de ejercicios prácticos después, permitimos a los alumnos adentrarse en nuevas técnicas con supervisión de los docentes. El curso finaliza con el desarrollo colaborativo de un proyecto grupal, en el que se articulan e integran los conceptos aprendidos. El workshop en sus dos ediciones tuvo muy buena repercusión en los ámbitos en los que se desarrolló, mostrando así la necesidad del aprendizaje de estas herramientas para muchos grupos de investigación. Actualmente está en desarrollo la segunda edición del Workshop, en la Facultad de Ciencias Exactas y Tecnología (FACET) de la Universidad Nacional de Tucumán (UNT).

Palabras clave: computación científica, programación, trabajo colaborativo.

Introducción

La principal motivación para la realización del Workshop en Técnicas de Programación Científica (WTPC), es que detectamos la necesidad, tanto en estudiantes de posgrado como en grupos de investigación ya consolidados, de contar con instrumentos más formales y herramientas más actuales para el desarrollo de software científico y también para su uso eficiente. Algunas de estas habilidades blandas o *soft skills* incluyen la elección apropiada de librerías, desarrollo de código de manera colaborativa mediante herramientas adecuadas a tal fin, mejorar la interfaz del usuario con el programa, además de realizar optimización en general y aprovechar eficientemente el hardware existente.

Dado el impacto que tienen este tipo de capacidades o *soft skills* durante el proceso de investigación científica, muchas disciplinas se ven beneficiadas. Esto se ve reflejado, no solo por el aumento en la calidad de los trabajos sino también por la capacidad de acelerar los diferentes procesos. Muchos grupos de investigación cuentan con profesionales del área de las Ciencias de la Computación, sin embargo la comunicación con el científico se hace dificultosa si ambos no hablan el mismo "idioma". Saber cómo optimizar esta comunicación es esencial. En consecuencia, este Workshop está orientado a científicos jóvenes en formación, en particular estudiantes de doctorado y estudiantes avanzados de carreras de grado científicas.

El objetivo principal del Workshop es presentar estrategias y buenas prácticas al momento de diseñar

e implementar software científico de modo eficiente. Consideramos la eficiencia del software desde un enfoque integral, no sólo basado en el tiempo de ejecución de un programa (idea de *eficiencia* proveniente de muchos científicos que realizan simulaciones computacionalmente intensivas), sino también en el tiempo de desarrollo del científico, la versatilidad de la interfaz entre la aplicación y el usuario, la reusabilidad, la colaboración, el uso de librerías y el mantenimiento.

Para cubrir los conceptos que consideramos dentro de la eficiencia, los contenidos del Workshop forman un abanico de temas que va desde el uso de sistemas de control hasta programación en entornos paralelos pasando, entre otros, por el uso de lenguajes interpretados.

Historia

El origen de este Workshop data de abril de 2015, cuando por primera vez, los autores de este trabajo (excepto Rodrigo Lugones) se reunieron junto a Pablo Echevarría, en un Workshop que realiza periódicamente el Centro Internacional de Física Teórica (ICTP, por sus siglas en inglés) [1]. El hecho de haber podido implementar rápidamente en nuestro trabajo cotidiano las herramientas que habíamos adquirido en dicho Workshop, nos motivó a compartir con la comunidad científica local nuestra experiencia. Inmediatamente después nos reunimos para diagramar un pequeño curso de capacitación de una semana de duración (40 hs) a pedido de los integrantes del Servicio Meteorológico Nacional, del cual formaba parte Pablo Echevarría, y que se realizó en noviembre de 2015.

Luego de esa experiencia tan positiva, decidimos impartir un curso de una escala mayor: nuestro primer Workshop en Técnicas de Programación Científica WTPC2016 [2], realizado en la Universidad de Buenos Aires en marzo de 2016. Para nuestra grata sorpresa, durante el periodo de selección se inscribieron más de 100 personas. Esta cantidad de gente superaba ampliamente nuestra capacidad física, de modo que seleccionamos 25 participantes. El balance de las dos semanas que duró el evento fue altamente positivo, por lo que decidimos continuar dictando el Workshop. A mediados de ese año, se incorporó al equipo Rodrigo Lugones (coautor de este trabajo), que en primera instancia fue asistente al WTPC2016.

Posteriormente, la Universidad Nacional de Quilmes seleccionó nuestro proyecto para dictar una versión intensiva, pero esta vez en la modalidad Curso de Posgrado [3]. Otra vez resultó muy gratificante la experiencia y la buena recepción por parte de la Universidad y de los alumnos, confirmó la necesidad concreta por parte de la comunidad de adquirir habilidades y herramientas de desarrollo de software que muchas carreras de grado no proporcionan actualmente a los científicos. Este fue el momento en que decidimos hacer de nuestro Workshop un evento periódico y comenzamos a planificar la edición de marzo de 2017, con sede en la Facultad de Ciencias Exactas y Tecnología de la Universidad Nacional de Tucumán.

Objetivos

El principal objetivo de este Workshop es brindar a la comunidad científica un conjunto de *soft skills* que facilitan el desarrollo diario de software. Esto se enmarca en una serie de objetivos más generales como organización. Entre estos objetivos se encuentran la federalización en la formación de posgrado en técnicas de computación científica, y el fomento de la participación de los grupos sub representados en el área, como es el caso de las mujeres.

En ambos sentidos creemos que estamos en el camino adecuado. Mediante la revisión continua y actualización de los contenidos, la incorporación de colaboradores, y manteniendo nuestra filosofía de desarrollo colaborativo, pensamos continuar con el dictado de estos cursos.

Metodología y Contenidos

El Workshop, de una duración de dos semanas (70 hs) se divide en dos partes. En la primera parte, de una semana de duración, combinamos clases teóricas de carácter formativo por la mañana (3 hs por día), con ejercicios prácticos individuales por la tarde. En éstos, los alumnos, con asistencia permanente de los docentes, ponen en práctica los conocimientos teóricos desarrollados.

En la segunda parte, docentes e investigadores invitados brindan charlas de carácter informativo, en las que comparten cómo implementan distintos as-

pectos de la programación científica en su trabajo cotidiano.

A su vez, por las tardes, los asistentes del Workshop trabajan en grupos de cuatro o cinco personas desarrollando colaborativamente un proyecto. Estos proyectos, que articulan e integran los contenidos vistos en el Workshop, son propuestos por los docentes al finalizar la primera semana. Al finalizar el evento, cada grupo expone la que muchas veces es su primera experiencia en el desarrollo colaborativo de software. Esta metodología de trabajo permite incluir todos los problemas típicos que aparecen en la dinámica del trabajo colaborativo.

En el marco del desarrollo colaborativo, elegimos que todos los ejercicios prácticos y las clases teóricas utilicen exclusivamente herramientas de software libre. Asimismo, las clases y charlas se encuentran disponibles en nuestra página web para consulta [2]. Además, en esta edición en la FACET-UNT, gracias al aporte del Centro de Educación a Distancia e Investigación en Tecnologías Educativas (CEDITE) de la FACET, hemos grabado como prueba piloto algunas de las clases [4].

Los contenidos de este Workshop abarcan: introducción al desarrollo de software, introducción a lenguaje interpretado (Python), sistemas de control de versiones (git), introducción a la programación orientada a objetos, compilación y linkeo comunicando distintos lenguajes, arquitectura del computador, optimización, debugging y profiling, documentación e introducción a la programación en entornos paralelos. El programa más detallado se encuentra en la página web del curso [2].

Una de las actividades didácticas más efectivas en este caso tiene que ver con la resolución de un problema de aplicación de manera grupal. Siguiendo la filosofía de software colaborativo, cada grupo debe, durante la segunda semana del curso diseñar e implementar una solución computacional al problema planteado. Al finalizar pudimos comprobar que con este ejercicio los alumnos logran incorporar todas las estrategias, herramientas y metodologías presentadas en la primera semana. Además el resultado en la mayoría de los casos supera las expectativas iniciales de los propios alumnos.

Conclusiones

Al cierre del presente artículo nos encontramos en plena actividad del WTPC2017 con gran asistencia y entusiasmo por parte de los alumnos de diferentes zonas geográficas del país.

Queremos dejar como mensaje la importancia de capacitarse en estas nuevas herramientas de cara a mejorar la calidad del trabajo científico. Cada vez se vuelve más ineludible el aprender estas herramientas de desarrollo del software y cada vez más los referatos realizan requerimientos para poner el código a disposición de los revisores o incluso ser publicados para los lectores. De este modo, el proceso de programar debe ser más claro y su escritura, simple y clara para poder ser contrastada, revisada, integrada a otros proyectos y, de esta manera, poder volverse parte y nuevo pilar del conocimiento científico que vendrá.

Los invitamos a seguir nuestras actualizaciones en redes sociales y página para poder aplicar a nuestra próxima edición, el WTPC2018.

Agradecimientos

En primer lugar, más que agradecidos estamos orgullosos de haber comenzado este camino con Pablo Echevarría quien actualmente pertenece al BSC (Barcelona Supercomputing Centre). Agradecemos también muchísimo el invaluable aporte de Gabriel Lio en el diseño de muchos de los posters y logos relacionados con el WTPC. También les agradecemos a Ivan Giroto, David Grellscheid y Axel Kohlmeyer por habernos dictado las clases del workshop original y por motivarnos a realizar este curso en Argentina.

Como así también a todos los profesores invitados y a las personas que nos ayudaron mucho con la logística en los eventos: Patricio Jarne Ferrarotti, Andrés Litwiller, Ticiano Torres Peralta y Leonardo Albarracín. Finalmente, agradecemos también a las autoridades de la FACET-UNT y UNQ por haber confiado en nosotros.



Lecturas sugeridas

Página oficial del Workshop en Técnicas de Programación Científica (2017) <https://wtpc.github.io/clases/>

Haslwanter, Thomas (2016) An Introduction to Statistics With Applications in the Life Sciences, Springer.

Hans Petter Langtangen (2014) A Primer on Scientific Programming with Python.

Referencias bibliográficas

[1] Workshop on Advanced Techniques for Scientific Programming and Management of Open source Software Packages (2017), <http://indico.ictp.it/event/7657/>

[2] <https://wtpc.github.io/> (2017)

[3] Técnicas de Programación Científica (2017), <http://www.unq.edu.ar/cursos-posgrado/107-t%C3%A9cnicas-de-programaci%C3%B3n-cient%C3%ADfica.php>

[4] Centro de Educación a Distancia e Investigación en Tecnologías Educativas (2017), <http://www.cedite.unt.edu.ar/index.php/es/>

Este artículo se realizó en el mes de Marzo de 2017 en el Laboratorio de Computación Científica del Departamento de Ciencias de la Computación de la Facultad de Ciencias Exactas y Tecnología de la Universidad Nacional de Tucumán.

Pablo Alcain

Es Licenciado en Ciencias Físicas de la Universidad de Buenos Aires, actualmente cursando el doctorado en física en la misma universidad, con Claudio Dorso como director. En su doctorado, trabaja en el estudio computacional de sistemas de mecánica estadística, específicamente en las estructuras de pasta nuclear en las estrellas de neutrones. Es también ayudante del Departamento de Física en Mecánica Estadística e Introducción a la Química Cuántica.

Cecilia Jarne

Es Doctora en Ciencias Físicas de la Universidad Nacional de la Plata, bajo la dirección de H. Walhberg y M. T. Dova en estudios experimentales de rayos cósmicos. Actualmente realiza una estancia posdoctoral en el Laboratorio de Sistemas Dinámicos del IFIBA-UBA bajo la dirección de G. Mindlin donde estudia las unidades del canto en aves oscinas. Además, es docente de la Universidad Nacional de Quilmes y de la Escuela Técnica N°2 de Quilmes, donde enseña técnicas digitales y sistemas de comunicaciones.

Rodrigo Lugones

Es Licenciado en Física de la Universidad de Buenos Aires, actualmente cursando el doctorado en física en la misma universidad, con Pablo Dmitruk como director. En su doctorado, trabaja en el estudio computacional de turbulencia en plasmas. Es también ayudante del Departamento de Física de la Facultad de Ciencias Exactas y Naturales de la UBA, en el área de Fluidos y Electromagnetismo.

Graciela Molina

Es Licenciada en Informática, actualmente finalizando el doctorado en Ciencias Exactas e Ingeniería de la Facultad de Ciencias Exactas y Tecnología (FACET), Universidad Nacional de Tucumán (UNT) bajo la dirección del Dr. Miguel A. Cabrera. Es profesora adjunta e investigadora en la misma Universidad. Es docente en las asignaturas de grado "Métodos Numéricos I" y "Métodos Numéricos II". Integrante del Laboratorio de Computación Científica (LabCC), del Laboratorio de Telecomunicaciones, ambos de la FACET-UNT. Estudia algoritmos numéricos para detección de ecos de radares para aplicaciones geofísicas y Space Weather. Mediante el estudio de técnicas de High Performance Computing desarrolla software para el procesamiento de señales de radares.

**cet**

REVISTA DE CIENCIAS EXACTAS E INGENIERIA
FACULTAD DE CIENCIAS EXACTAS Y TECNOLOGÍA

Visite nuestra página:
www.facet.unt.edu.ar/revistacet