

Entrevista (!)

CONOCEMOS A UNO DE LOS RESPONSABLES DEL KERNEL DE WINDOWS

Desde adentro

Entrevistamos a Dave Probert, uno de los arquitectos del desarrollo del núcleo de Windows 2000, Windows XP y, recientemente, Windows Vista. También es el encargado del programa académico en el que la empresa comparte el código fuente de Windows con profesores universitarios de sistemas operativos.

¿Cuál es su trabajo en Microsoft?
 ← Ahora estoy trabajando en arquitectura de kernel dentro de Windows y, como antes había dedicado varios años a la universidad, aproveché esa experiencia para volver a trabajar con esas instituciones. No es mi responsabilidad fundamental, pero una de mis tareas es ayudar a que las universidades entiendan mejor la plataforma Windows. Como trabajé con UNIX durante mucho tiempo, creo entender el afecto que tienen los departamentos de computación, sus profesores y alumnos, por él. Yo tengo mi propio afecto en lo que respecta al interior de Windows; por lo tanto, estuve trabajando con mis colegas para que sea posible que las universidades comiencen a querer a Windows. Empecé a viajar por todo el mundo y a entregar materiales que describían la parte básica del sistema, y comenzamos a brindar el código fuente de partes principales del núcleo, junto a grandes cantidades de material de pre-

sentación que describen las operaciones internas de Windows. El objetivo es que los profesores se entusiasmen cuando enseñan sistemas operativos, que no sólo enseñen UNIX, sino que también enseñen Windows.

→ ¿Por qué cree que hasta ahora no se utilizaba Windows en la enseñanza universitaria?

← La razón principal es que, en ciencias de la computación, amamos UNIX, estamos familiarizados con él, y es un sistema que tiene muchas cualidades, que hace que sea interesante de enseñar.

→ ¿Ama UNIX?

← Sí.

→ ¿Ama Windows?

← Sí.

→ ¿Cuál cree que es la ventaja de entregar partes del código fuente de Windows en el mundo académico?

← Para poder entender las partes internas de Windows, es necesario poder

leer el código fuente, y se lo explico a los profesores de esta manera: si les diéramos sólo el material y los alumnos les hicieran una pregunta difícil, sin el código fuente, ¿cómo podrían darles una respuesta? Entonces, les damos el código fuente.

→ Hay alumnos que participaron del proyecto que cuestionan el solo hecho de mirar el código fuente por temor a, sin querer, copiar algún algoritmo y tener algún problema comercial. ¿Qué opina?

← (silencio)... Entiendo ese tipo de temor porque el clima que hace a la propiedad intelectual es muy difícil y complejo. Estoy muy contento de no ser abogado. Este programa está pensado para que los profesores se sientan cómodos utilizando el código fuente en el aula. Todos los días nos topamos con el tema de la propiedad intelectual y, en general, sabemos qué está bien y qué no. Si escucho una canción en la radio, no puedo escribirla y decir que es mía. Si escribo o leo el código fuente y decido escribirlo y decir que es mío, estaría mal; y la gente lo sabe. Pero si escucho una canción y me inspira a escribir poesía, entonces la poesía es totalmente mía, escrita con mis propias palabras, y entonces no hay problema. Simplemente, es inspiración; lo mismo es válido con el código fuente de Windows.

→ Por ahora, el programa es sólo académico. ¿Ve un futuro en el que el código fuente de Windows sea más accesible para desarrolladores comerciales?

Cuando uno de mis ingenieros tiene una idea le pregunto: "si hay un problema, ¿cuánto tiempo le costaría al usuario? Ahora multiplícalo por los mil millones de usuarios de Windows y dime cuántas vidas hombre acabas de gastar".

← Voy a dar mi opinión personal, que puede ser diferente de la de Microsoft. La gente que quiere el código fuente para fines comerciales lo quiere por dos motivos. El primero es para usarlo como un recurso sobre el cual construir otro producto, en cuyo caso necesita seguir los caminos normales para conseguir una licencia. Tal como si yo escuchara una canción en la radio y quisiera usarla para un comercial.

El segundo se refiere a los desarrolladores de aplicaciones comerciales, que quieren el código fuente para entender cómo funciona el sistema. Mi opinión personal es que no es bueno entregarlo, porque ellos van a aprender cómo funciona una versión específica del sistema operativo, y van a basar sus propios diseños sobre la manera en que funciona esa versión. Si cambiamos el sistema interno en la próxima versión, su programa no funcionará más. Y sus clientes también son nuestros clientes. Y entonces nuestros clientes van a estar disconformes con nosotros diciendo que ahora el programa no funciona. Entonces, es importante para nosotros, en las ciencias de la computación, lograr la interoperación entre programas y sistemas operativos a partir de contratos de software, y no a partir del código fuente. Quizá desde Microsoft brindemos el código fuente para motivos comerciales, pero mi opinión personal es que esto podría llegar a ser una mala idea si la gente basa su entendimiento en el código fuente y no en su comportamiento.

→ **¿Cuál es la importancia de estudiar un sistema operativo?**

← Hay muchos programadores que no aprecian la importancia de entender un sistema operativo y conocer el proceso de desarrollo en sí. La idea es estudiar los sistemas operativos para entender, por ejemplo, cómo funciona el multithreading, cómo afecta la memoria virtual a los programas y cómo podemos abordar los nuevos desafíos que llegan con el desarrollo de los multicore. Los sistemas operativos son importantes para quienes trabajan con datos, con servidores web, y lo que es más importante aún, con juegos.

→ **¿Qué diferencias hay entre el desarrollo de un kernel y el de aplicaciones comerciales?**

← Trabajar en un equipo de kernel en Microsoft es muy especial, porque hoy el kernel no es sólo la base del sistema operativo, sino que es la base de la plataforma Windows en sí misma. Es muy divertido porque mucha gente viene a vernos con preguntas interesantes y nuevos desafíos. El kernel es un lugar magní-

co y maravilloso. Cuando la gente está desarrollando, piensa en APIs, posiciones de memoria, instrucciones... nosotros creamos todo eso.

→ **¿Siente presión o responsabilidad teniendo en cuenta la masividad de Windows en el mundo?**

← Nunca sentí presión. Pero reconozco la responsabilidad y la oportunidad. Te doy un ejemplo: cuando uno de mis ingenieros tiene una idea le pregunto: "si hay un problema, ¿cuánto tiempo le costaría al usuario? Ahora multiplícalo por los mil millones de usuarios de Windows y dime cuántas vidas hombre acabas de gastar".

Ésa es la única escala razonable para medir el impacto de Windows.

Windows ha ayudado a cambiar la economía de todas las naciones del mundo, al aumentar la productividad humana, facultar a la gente para alcanzar sus sueños y darles una gran oportunidad para participar. Antes de Windows, las computadoras eran muy caras, pocas personas accedían a ellas, y la estrategia de Microsoft ayudó a bajar su costo. Ahora, la mayor parte de las personas en la mayoría de los países tienen acceso a una PC. Hemos achicado el mundo a través de la tecnología.

Para aquellos que no quieren a Windows: "El hecho de amar lo que nos resulta familiar es humano. Pero la sabiduría es ser abiertos a opciones diferentes, entenderlas y apreciar su belleza aunque, en un principio, nos parezca poco familiar".

→ **Habiendo trabajado en el núcleo de Windows, ¿qué podría decirle a aquellos usuarios que aman solamente UNIX y no quieren ni recomiendan a Windows?**

← Es una muy buena pregunta... Siempre tengo la esperanza de convencerlos. En general, las mayores críticas que tienen se refieren a la estabilidad, sustentadas en dos cosas: previas versiones de Windows que no están basadas en NT y problemas con drivers o de hardware. Por ejemplo, hemos visto muchos problemas de errores de memoria. En Windows Vista, hemos agregado un programa que se puede ejecutar para verificar la memoria. La base de Windows hoy es NT, es como VMS. Es extremadamente estable para lo que es el sistema operativo. Por lo tanto, si consideran que Windows no es estable, no conocen el Windows de hoy.

El otro punto que podría mencionar es que deberían abrir sus mentes y analizar muchas ideas diferentes. El hecho de amar lo que nos resulta familiar es humano. Pero la sabiduría es ser abiertos a opciones diferentes, entenderlas y apreciar su belleza aunque, en un principio, nos parezca poco familiar.

Maximiliano R. Firtman

¿Qué debe hacer un profesor universitario si quiere participar del programa?

Existe el sitio web www.microsoft.com/windowsacademic, donde todos los profesores pueden acceder a información sobre el programa. Allí hay instrucciones acerca de cómo contactar y calificar para el programa y, luego, habilitarles la descarga del código fuente de Windows y otros materiales referidos al tema. Si tienen preguntas, también pueden dirigirse a la dirección de e-mail compsci@microsoft.com.