

Del software al fatware al nanosoftware



El que se considera primer ordenador electrónico programable es el ENIAC, construido como sistema de propósito general (es decir, capaz de llevar a cabo diferentes tareas en función del programa que se le suministrara). Los operarios del sistema tenían que tener muchísimo cuidado con los programas que elaboraban, ya que un error suponía un montón de tiempo de revisión de las tarjetas perforadas que le suministraban información. Y los programas debían ser muy pequeños, ya que su memoria era pequeña, disponía de 17.468 tubos de vacío capaces de almacenar números: cada 36 tubos almacenaban un número.

¿Cómo debían sentirse las personas que manejaban los datos de ese sistema? Supongo que como un biólogo pegado al microscopio, mirando cada dato, cada número, cada instrucción, para que cupiera en la exigua memoria del sistema.

Rápidamente empezaron a crecer en capacidad y velocidad los ordenadores, y el primer PC (el de IBM, el modelo 5150) ya era capaz de almacenar 16.384 números en su memoria basada en transistores. La pequeña (por aquel entonces) empresa Microsoft había desarrollado un intérprete del lenguaje Basic que únicamente ocupaba cuatro kilobytes, y estaba incluido en la memoria ROM de ese micro ordenador. Visto ahora parece casi increíble. Que un intérprete de un lenguaje de programación sea capaz de “caber” en tan poco sitio suena a leyenda urbana.

El progreso en la informática ha facilitado que cada vez se puedan producir microprocesadores más rápidos y dispositivos de almacenamiento más fiables y todo ello más barato. ¿Quién podría imaginar que un sistema informático como el actual en 1981, cuando se presentó el IBM PC? Un microprocesador con una frecuencia de reloj que se mide en Gigahertzios, almacenamiento en disco y en memoria medido en Gigabytes, y todo ello por menos de la cuarta parte de lo que valía en su momento.

Pero sin embargo, los usuarios ¿qué han ganado con todo esto? ¿Tarda menos en hacerse una hoja de cálculo con cualquier sistema actual que hace unos años con los sistemas preparados para trabajar en pantallas de texto de fósforo verde? No mucho menos. Se hacen más bonitas, con más tipos de letra, con más efectos... pero no son mucho más efectivas. La inmensa mayoría de los usuarios desconocen el 80% de las funciones de su procesador de textos, lo que le sirve a los “expertos” de oficina a aparecer como héroes cuando enseñan cómo poner una palabra en negrita sin necesidad de separarse del teclado y usar el ratón.

El software ha ido creciendo y volviéndose cada vez más complejo ya que los sistemas lo permitían. Y ese crecimiento supone consumo de recursos: memoria, disco, procesador, tarjeta gráfica, etc. Recuerdo el paso del famoso dBaseIII, que dejó de ocupar un par de disquetes de 5 pulgadas y cuarto al dBase IV, que se distribuía en ¡once disquetes! Muchos se llevaron las manos a la cabeza por ese dispendio en disquetes y en espacio. Cuando al final, el uso que se le daba a ese programa era, en muchos casos, únicamente dar de alta y consultar una base de

Del software al fatware al nanosoftware



datos sencillita. ¿Cuántos usuarios llegaron a emplear el sistema de consultas SQL embebido en dBase IV?

Por no hablar de lo sistemas operativos. MS-DOS 3.3, por ejemplo, ocupaba dos disquetes de 360 Kb, MS-DOS 6, 4 disquetes de 1,44 Mb (16 veces más), Windows 95 en 13 disquetes con un formato especial que conseguía algo más de espacio, y ya dio el salto al CD ROM para instalarse. Windows Vista se distribuye en DVD, un soporte que es capaz de almacenar 4,7 Gb, es decir, más de 13.000 discos como los de la distribución de MS-DOS 3.3.

Este crecimiento ¿en qué ha repercutido? En muchos gadgets, en mucho interfaz gráfico tridimensional, en imágenes fotorrealistas, pero a costa de consumir recursos a mansalva.

¿Pensamos en el malware? ¿Cuánto ocupaba el Viernes 13? Utilizaba únicamente 2 Kb de memoria, y los ficheros infectados crecían en 1.813 bytes. ¿Y el gusano Brontok.FT? Más que un gusano parece una anaconda, o una serpiente pitón. ¡Ocupa 12 megabytes!

Todo crece: los discos, las memorias, las funciones del sistema operativo... ¿No es posible hacer que las aplicaciones en lugar de crecer vayan a menos? ¡Si en un reloj digital hay más capacidad de procesamiento de información que en el Apollo XI y llegó a la luna!

Pues sí, es posible. En la mecánica se está empezando a investigar en el campo de las nanotecnologías, de manera que se están construyendo máquinas a escala atómica. Por ahora son experimentos, engranajes en los que los piñones no son más que átomos o tubos por los que únicamente puede pasar una molécula. En la informática, están empezando a despuntar sistemas nanotecnológicos. Programas que, a pesar de la tendencia actual a utilizar más recursos, más memoria y más funciones, son extremadamente ligeros y rápidos.

Los nanoprogramas pueden estar diseñados para funciones muy concretas, como puede ser mostrar un pequeño reloj en pantalla, o un juego sencillo pero adictivo. O incluso funciones tremendamente complejas, como la aplicación “nanoscan” de Panda Software.

Nanoscan es un sistema de búsqueda de malware activo en un sistema que es capaz de encontrar cientos de miles de programas malignos sin necesidad de ocupar megas o gigas en el sistema. Gracias a un sistema de desarrollo tremendamente cuidado y teniendo como objetivo el tamaño mínimo con las máximas funciones, se ha conseguido, por fin, hacer olvidar al mercado la tendencia a engordar el software.

¿Y cómo es posible? Sencillamente, olvidándonos de que un sistema va a poder ofrecernos más recursos de manera ilimitada. Las aplicaciones clásicas han sido

Del software al fatware al nanosoftware



desarrolladas pensando en que van a ser instaladas en un sistema lleno de API distintos, muy útiles y llenos de funciones, pero que deben ser cargados en memoria para usarlos. Y cada aplicación hace lo mismo, por lo que el consumo de recursos crece de manera desorbitada.

Si se desarrolla una aplicación prácticamente “autocontenida”, sin optar por el fatware; y con un I+D realmente investigador y desarrollador, se pueden conseguir resultados impresionantes en la industria del software.

Estamos a las puertas de una nueva era, la del nanosoftware. Quizá en poco tiempo volvamos a tener que recuperar los disquetes de 3,5 pulgadas para instalar un procesador de texto. ¿Por qué no? Sólo es cuestión de tomarse el desarrollo de software como una ciencia, y no solo como una colección de archivos entrelazados comerecursos.

Fernando de la Cuadra
Editor Técnico Internacional
Panda Software (<http://www.pandasoftware.com>)
E-mail: Fdelacuadra@pandasoftware.com