

Resumir una panorámica de la Informática desde una perspectiva de diez años puede hacerse desde diversos puntos de vista. Todos ellos, sin embargo, pueden compartir la idea global de comprender como ésta ha ido formando parte de nuestro entorno habitual de trabajo.

En diez años ha descendido del mundo privilegiado de algunos especialistas al mundo de lo cotidiano, a lo accesible a una gran cantidad de personas; ha salido de los centros de proceso de datos a instalarse en nuestra mesa de trabajo y a constituirse en centro de conversación y preocupación de gran parte de la sociedad.

Y todo esto se ha producido ampliando fuertemente el espectro de aplicaciones y con ellas alcanzando a nue-



Gonzalo León. Doctor Ingeniero de Telecomunicación es profesor Titular del Departamento de Ingeniería Telemática de la U.P.M. Su actividad de investigación se centra en la ingeniería software de sistemas de comunicación orientada al diseño de entornos de programación para sistemas concurrentes.

¿Evolución o revolución Informática?

Texto: Gonzalo León Serrano

vos usuarios. No es que el proceso sea nuevo, simplemente se ha acelerado en esta década pasada.

Entender este proceso desde la perspectiva de las Telecomunicaciones es aún más significativo porque es este sector uno en el que las necesidades de diseño hardware y software son más exigentes. No es pretencioso afirmar que gran parte de la evolución Informática se ha realizado en los últimos años a instancias de lo que esta industria necesitaba (cierto que no solamente la industria del sector de las Telecomunicaciones pero en mi opinión ésta posee un peso específico muy importante).

En definitiva, comprender esta evolución desde esta perspectiva no limita en absoluto la percepción del fenómeno sino que, por el contrario, nos sitúa en una excelente atalaya para entenderlo en su verdadera magnitud.

Desde una perspectiva personal también he vivido en estos diez años como esta evolución se reflejaba en la importancia que ha tomado en la formación de nuestros ingenieros; no sólo desde el punto de vista del número de materias informáticas impartidas sino por lo que, a mi juicio, es más importante: en su masiva utilización como herramienta de trabajo en muchas otras disciplinas. Y esto no era más que un reflejo de lo que estaba pasando en la industria.

Mucho más complicado es predecir o que puede ocurrir en próximo futuro. Y esta previsión se hace cada vez menos segura cuando este período sea superior al tiempo de vida útil de las actuales tecnologías de diseño hardware o software. La aparición de nuevos avances no previsibles hoy día o la comercialización masiva de desarrollos aún en fase experimental de laboratorio pueden distorsionar totalmente las previsiones.

Recordar lo sucedido,

por ejemplo, con las previsiones de refrigeración de los ordenadores a válvulas en los años cuarenta con el fin de incrementar la potencia de cálculo o las previsiones de la necesidad de operadoras telefónicas para poder atender el incremento de llamadas telefónicas son casos muy aleccionadores.

En sentido contrario, las expectativas una y otra vez defraudadas del incremento de productividad de los programadores o los progresos tan lentos en programación automática o procesamiento de lenguaje natural o verificación formal de programas por citar algunos ejemplos conocidos son claros indicadores de dificultades teóricas aún no resueltas.

En este trabajo pasaremos revista brevemente a los hechos más significativos ocurridos en torno al diseño hardware/software, comentaremos la situación actual centrándonos en la fuerte relación entre ambos tipos de evolución y plantearemos unas previsibles líneas de evolución.

EVOLUCION DE LA ARQUITECTURA DE LOS ORDENADORES

Aunque por razones de estructuración del artículo (y también por razones de evolución histórica) trataremos la evolución hardware y software por separado, los puntos de contacto entre ambos son tan fuertes que no es posible entender una sin la otra.

Utilizaremos, sin embargo, dos puntos de vista complementarios: el del usuario y el del diseñador. Tanto uno como otro han visto esta evolución desde diferentes coordenadas no siempre totalmente sincronizadas. La aparición del PC no representó realmente ningún "avance" hardware o software en sí mismo aunque constituye el hecho socio-informático de mayor envergadura de toda la década que comentamos.

En 1977 la arquitectura básica de los ordenadores de la tercera generación estaba totalmente estabilizada. El microprocesador como componente electrónico estaba ya en el mercado.

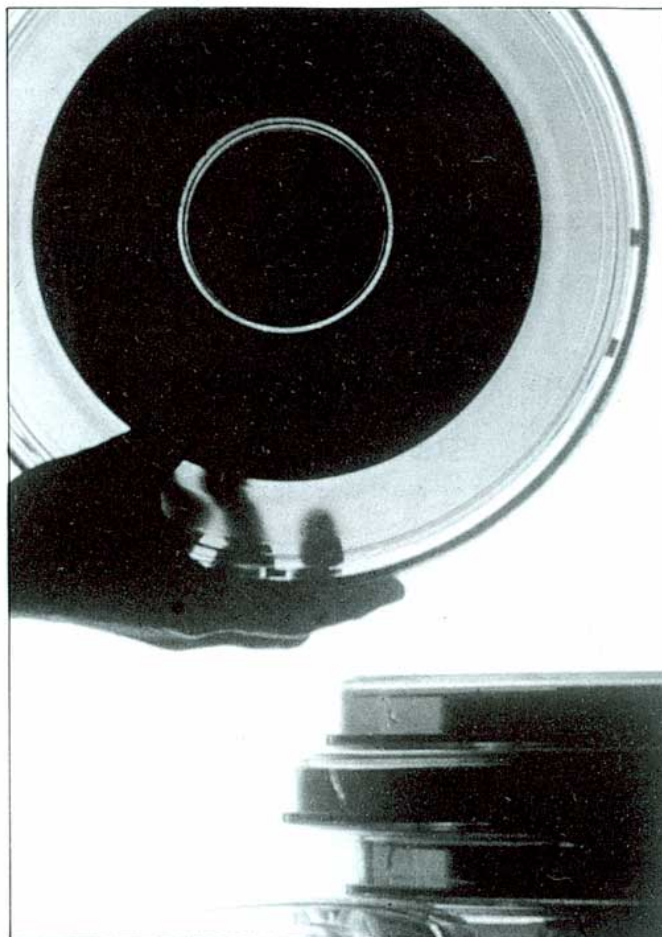
Los arquitectos de computadores se encontraban entonces trabajando en dos grandes líneas: la de reproducir en el diseño de nuevas generaciones de microprocesadores los conceptos que ya existían en grandes máquinas y la de tomar una visión radicalmente distinta. Un microprocesador podía ser considerado como un bloque elemental sobre el que construir sistemas de enorme potencia de procesamiento mediante la interconexión de los mismos.

Ambas líneas de trabajo han progresado decisivamente en esta década. Seguidamente comentaremos lo más significativo de cada una de ellas.

Por la primera de ellas se dispone hoy día de microprocesadores con una sofisticada arquitectura interna. Estos están a su vez rodeados de complejos circuitos integrados con los que diseñar computadores es hoy una técnica totalmente madura. Los problemas han venido ligados al proceso de fabricación de estos circuitos. Es el propio desarrollo de la Microelectrónica quien ha marcado realmente el ritmo de incorporación de los mismos al mercado.

La segunda de las líneas de trabajo ha sido mucho más difícil, quizás menos espectacular y todavía no introducida comercialmente de forma masiva. Su evolución ha llevado al estudio y conocimiento de los problemas de interconexión de cientos de procesadores por un lado y al diseño de circuitos adaptados a facilitar esta interconexión por otro. Este es el caso del "transputer" como un bloque elemental para construir sistemas complejos.

Pero no ha sido en la CPU donde la evolución ha



“
Desde la
perspectiva del
usuario la
evolución ha sido
más visible en
lo que se
relaciona
con dispositivos
periféricos.”

sido más rápida. Desde la perspectiva del usuario la evolución ha sido más visible en lo que se relaciona con dispositivos periféricos.

Tanto en terminales gráficos (hace diez años un lujo reservado a aplicaciones muy específicas y hoy casi una condición fundamental para poder vender un simple PC) como en las funciones gráficas implantadas en los procesadores gráficos la evolución ha sido espectacular. Su disponibilidad no sólo ha facilitado su introducción en muchos otros campos de aplicación sino que ha alterado profundamente las interfaces de usuario.

Con respecto a la tecnología de memoria secundaria, en diez años se ha multiplicado la capacidad de almacenamiento por 100 manteniendo constante el coste tanto en memoria principal

como secundaria. La importancia de este fenómeno no reside sólo en la mejora de fiabilidad, consumo, peso, tamaño, etc., sino, y mucho más importante, en su incidencia en la forma de diseñar sistemas software, en su impacto en la organización de equipos de trabajo, en el incremento de productividad asociado, etc.

Finalmente, la necesidad de comunicarse con otras máquinas (con independencia de la tecnología implicada) también ha pasado de ser una opción costosa empleada sólo en grandes máquinas a constituir algo estándar en todas las estaciones de trabajo como parte del equipo básico que se adquiere.

EL DISEÑO SOFTWARE Y EL SOFTWARE PARA EL DISEÑO

Los problemas más importantes desde el punto de vista del diseñador software no han venido ligados a los aspectos hardware. En realidad, los parámetros más importantes hace unos años han comenzado a relativizarse.

La memoria principal disponible, los registros de desplazamiento, es decir, la arquitectura interna de la máquina, ya casi no interesan; no es que no sean importantes sino que desde el punto de vista de la elección de configuración, adquirir 4 ó 8 MB es barato. Sin embargo, el tipo de S.O. que posee, o las herramientas de diseño disponibles o los paquetes de aplicación instalados constituyen los aspectos fundamentales a la hora de tomar una decisión.

Hace diez años también estos aspectos se cuidaban por supuesto, pero existía una diferencia importante. El software iba ligado básicamente a lo que el fabricante ofrecía con la máquina. Durante esta década pasada la disociación entre el fabricante de la máquina y los proveedores de software

se ha hecho mucho más evidente.

En estos momentos, es posible obtener un paquete para una docena de máquinas diferentes soportando un mismo tipo de sistema operativo (imaginemos MS-DOS o UNIX) sin más que pedir la versión adecuada.

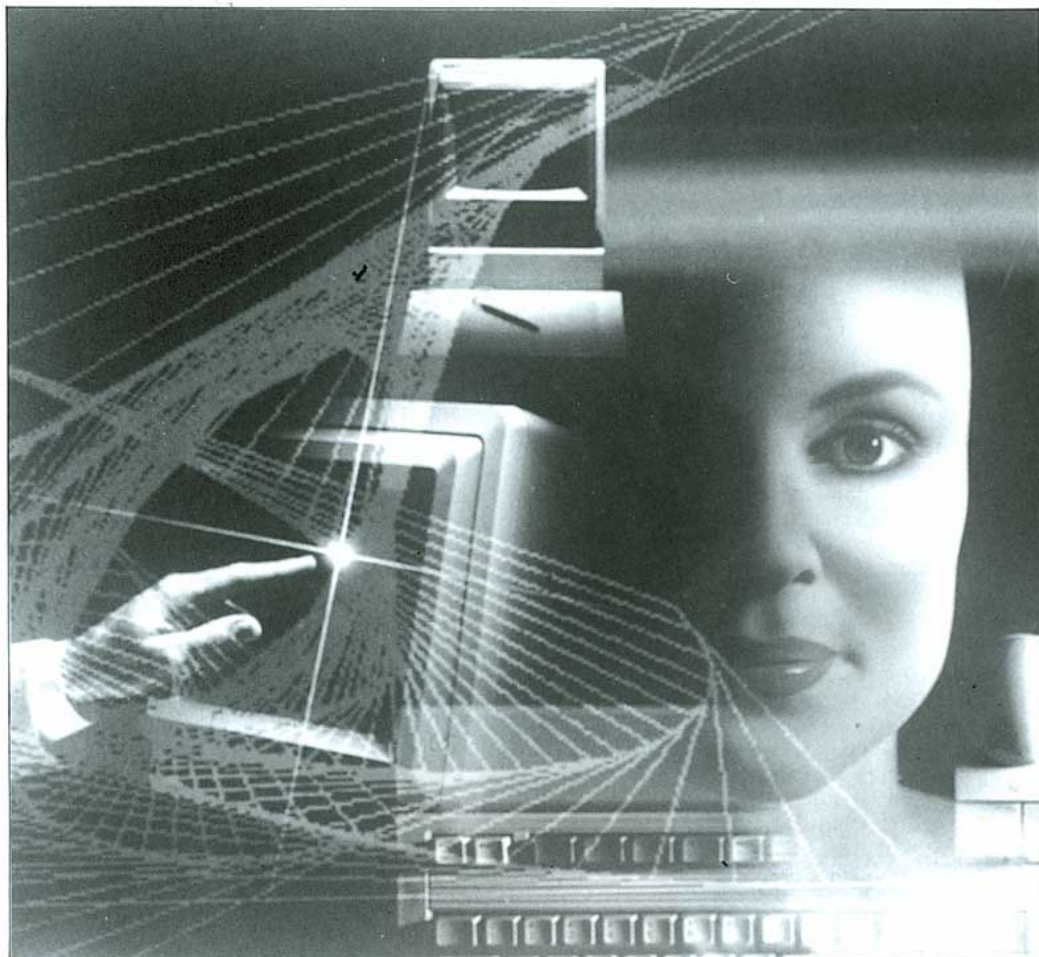
Si hace diez años se compraba una máquina y luego se pensaba en paquetes disponibles, ahora empieza a pensarse en comprar un paquete software y después buscar la máquina que mejor lo soporte.

El caso de los PC's es un caso extremo ya que es el número elevadísimo de paquetes disponibles lo que justamente se valora. Esta misma situación también empieza a aparecer en estaciones de trabajo basadas en UNIX.

Visto de una forma global y con la perspectiva que ofrecen estos diez años transcurridos, todo este proceso no es más que el reconocimiento que para los usuarios tienen los estándares (ya sean reconocidos internacionalmente o de facto). Estos estándares constituyen una garantía de portabilidad cada vez más apreciada. Nos estamos acostumbrando a reconocer equipos funcionalmente equivalentes por debajo de carcasas muy distintas.

Esta relativización de la importancia del hardware nos obliga a centrar la atención en como la nueva herramienta es cómoda, sencilla, ... amigable. En definitiva, centrar de nuevo la atención en el usuario. Desde este punto de vista el tipo de lenguaje de control de trabajo adquiere gran importancia.

Una definición ya clásica de un sistema operativo le consideraba como una máquina virtual para la que fuese sencillo trabajar. La disponibilidad de hardware y software gráfico, un mayor interés en potenciar las interfaces de usuario así como un mejor conocimiento de



“
*Es previsible una
rápida extensión
en el uso de
equipos con
capacidad de
procesamiento
distribuido.*
”

aspectos ergonómicos han creado una rama de la informática aplicada de enorme interés.

Volviendo la atención al propio proceso de diseño de sistemas software. En mi opinión, el eje fundamental de la evolución software se ha centrado en el problema del diseño de sistemas software complejos, grandes, con problemas de concurrencia y restricciones en tiempo real, implicando la necesidad de diseño asociado de hardware especializado (sistemas embebidos).

En estos la predicción del coste y tiempo necesario para acabar un determinado sistema constituía una tarea casi imposible hace diez años. Razonar sobre la corrección del producto generado imposible. Todo ello no es más que un índice de la

madurez de la tecnología software implicada.

Para que la actividad de diseño fuese realmente una ingeniería era necesario formalizar el proceso de desarrollo, formalizar los productos generados en el proceso, convertir los conjuntos ordenados de recetas de descomposición (pomposamente denominadas metodologías de desarrollo) en refinamientos sobre los que se pudiera razonar la satisfacción de propiedades, etc.

Las herramientas disponibles tanto notacionales (lenguajes) como paquetes software específicos bebieron de este proceso de formalización reflejo de avances teóricos importantes y dando lugar a una generación de herramientas que constituyen el mayor logro de la década.

Tres puntos me parecen significativos para entender lo que ha ocurrido: el desarrollo de lenguajes de especificación, la creciente disponibilidad de entornos de programación y el incremento de proyectos buscando reutilizar diseños anteriores en busca de una mayor productividad.

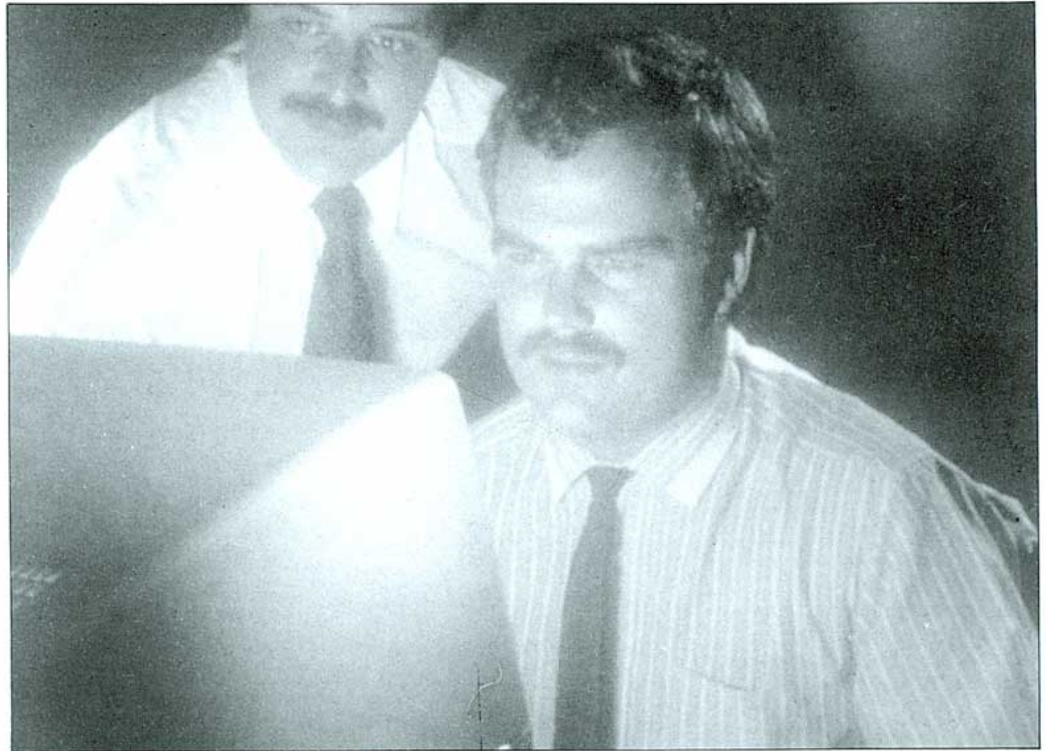
Sobre el primero de los aspectos habría que decir que no sólo representa la punta de lanza de la tecnología software abriendo un camino muy válido de aprovechamiento de nuevos desarrollos teóricos sino que va a alterar profundamente la capacitación técnica necesaria de nuestros ingenieros. En pocas palabras, necesitaremos especificadores/diseñadores en vez de programadores.

Los lenguajes basados en tipos abstractos de datos y cálculos algebraicos básicos para poder razonar sobre sistemas concurrentes empiezan a tener interés incluso para diseñadores hardware (fundamentalmente en la especificación funcional de circuitos integrados). Es el anticipo de las técnicas de diseño asociadas a una emergente crisis del hardware.

El segundo aspecto que deseo comentar se refiere a las herramientas que soportan el desarrollo de un producto software. Aquí el esfuerzo ha sido importante obligando a pensar en la industria de diseño software como una industria intensiva en capital.

La elección del sistema operativo y demás herramientas de edición, documentación, prueba, generación de programas, etc. (más o menos ligadas a un lenguaje concreto) es una condición sine qua non para lograr un nivel de calidad y productividad aceptable.

Esta productividad sólo se puede lograr automatizando partes del diseño, controlando la calidad del producto, reutilizando trabajo anterior, etc.



“
*Se empieza
a pensar
en comprar
un paquete
software y
después se busca
la máquina que
mejor lo soporte.*
”

Podríamos resumir todos estos esfuerzos en la búsqueda de la tecnología suficiente para crear factorías software. Algo todavía lejos pero con un camino previsible. Tanto en los programas ESPRIT como RACER, EUREKA, etc. de la CEE se aborda el diseño de esta cadena de producción de software como el objetivo básico de los años noventa. Nos adentramos en el futuro.

¿UN FUTURO CONOCIDO?

Sería fácil dar cifras de capacidades de almacenamiento, velocidades de procesamiento medidas en MIPS, MFLOPS o KLIPS (miles de inferencias lógicas por segundo) extrapolando sobre los valores actuales el proceso seguido durante la última década.

Imaginar, por otro lado, una estación gráfica de diseño personal dotada de decenas de MB's de memoria, discos ópticos del orden de GB's, conectada a través de una fibra óptica con otras

muchas a través de una red local y mediante pasarelas con todo el mundo (la verdadera aldea global) no es realmente un avance científico espectacular sino una clara expansión de una tecnología ya conocida en la actualidad. Y no por esto es menos importante el proceso.

Dicho de otra manera, considero previsible una rápida extensión en el uso de equipos con capacidad de procesamiento distribuido haciéndose común su empleo por una gran cantidad de usuarios potenciales.

La creciente disponibilidad de estaciones de trabajo para diseño cuyo coste (hardware, software y transferencia tecnológica) se está acercando al coste medio de un ingeniero senior obligará a dotar a éstos de este tipo de herramientas.

Al final de la próxima década este coste será muy inferior (¿la cuarta o quinta parte?). Creo que esto tendrá enormes consecuencias sobre la forma de trabajar. Si en los años sesenta la regla de cálculo era la herramien-

ta de trabajo personal. la calculadora electrónica lo fue en los setenta y el ordenador personal se ha ido poco a poco convirtiendo en la de los ochenta, la estación de trabajo gráfico como una evolución de los ordenadores personales lo será en los noventa.

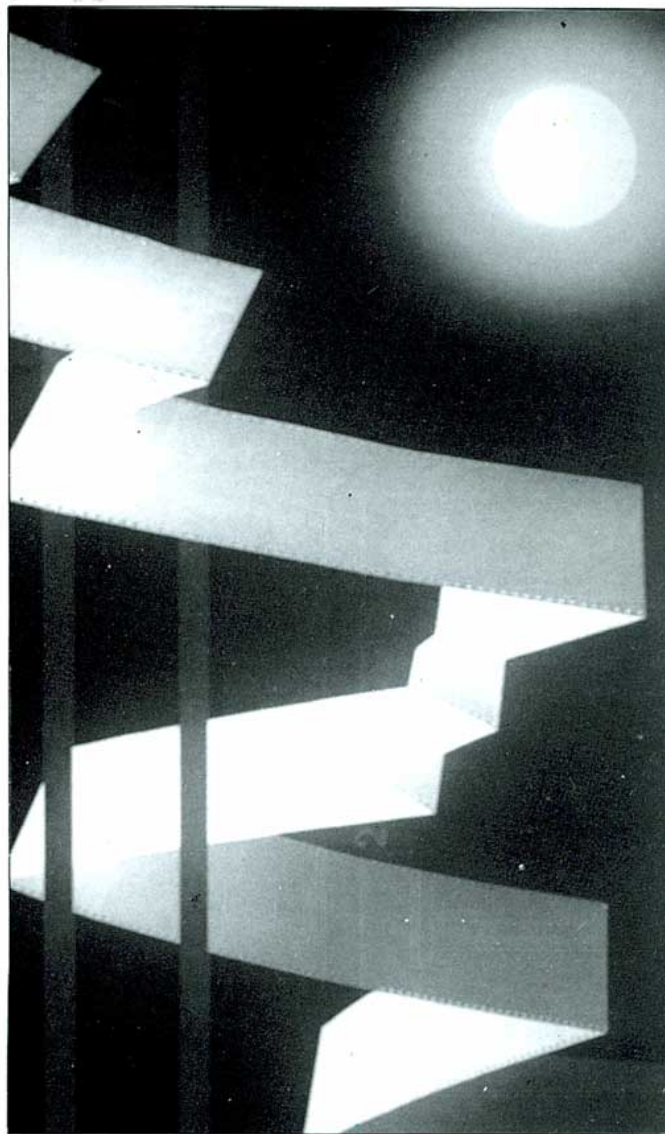
Si asociamos este proceso de disponer herramientas de trabajo más potentes con la posibilidad a través de servicios telemáticos de elegir donde realizar la computación, de donde tomar los datos de entrada o a donde enviar los resultados, también dónde trabajar será irrelevante. Es cierto que este tipo de previsiones no se ha cumplido fácilmente y no son sólo problemas tecnológicos los que lo han impedido.

Desde el punto de vista técnico el esfuerzo se seguirá orientando a lograr una mayor transparencia geográfica del proceso distribuido. Para ello, será necesario poner a punto una nueva generación de sistemas operativos distribuidos y aplicaciones que provean servicios al usuario final.

El desarrollo teórico seguirá buscando la aplicación práctica de modelos, cálculos o teorías ideadas en un plano abstracto. Temas como el de la verificación formal de programas que por ahora, no permiten pasar de pequeños ejercicios elementales sufrirán un fuerte avance llegando a formar parte de las técnicas habituales de diseño (reservándolos quizás a aspectos críticos de sistemas concurrentes o en tiempo real).

La fiabilidad, mantenibilidad, portabilidad de los sistemas software, actuales caballos de batalla seguirán siendo desarrollados. Aunque no espero resultados espectaculares si que las técnicas actuales de prototipado se emplearán profusamente.

Debemos ser optimistas: el empleo de entornos de programación implicando el uso de más herramientas, el



manejo de mejores lenguajes y, sobre todo, una mayor capacitación técnica de programadores e ingenieros software estoy seguro que producirán un efecto multiplicador beneficioso.

Existen tres áreas prometedoras actualmente y cuyo desarrollo depende en gran medida de la disponibilidad de software más evolucionado y nuevos desarrollos hardware. Nos referimos a la inteligencia artificial, la robótica y los nuevos servicios telemáticos integrados.

En todos ellos hay depositadas grandes esperanzas

“vendidas” a la sociedad en conjunto. Es más, incluso creer en ellas como un objetivo dinamizador es bueno. Los ejemplos del proyecto japonés de la quinta generación o los servicios integrados de banda ancha son ejemplos elocuentes. No tanto por el logro de los objetivos finales como por lo que implica el propio avance (salpicado de pequeños logros de aplicación práctica inmediata).

COMENTARIO FINAL

Una última reflexión: titulamos el artículo con una

pregunta: ¿evolución o renovación? Depende.

Vivido día a día, esperando el último paquete, la última versión de un sistema operativo o el último número de una revista que nos resolviese el problema, la espera era muy lenta.

En muchos casos se sigue diseñando prácticamente igual que hace diez años. Hay sectores donde la evolución sólo se ha visto desde el hardware (eso sí, de una manera fortísima).

Para otros, posiblemente a los que más les iba en el envite, aquellos para los que mejorar sus técnicas de diseño constituía un asunto de supervivencia, la evolución se produjo en todos los sentidos. Y no fue fácil.

Implicó, entre otras cosas un proceso de reconversión técnica del personal y, sobre todo, de un proceso de mentalización de sus dirigentes a la necesidad de incorporar una tecnología más avanzada y a entenderla. Para ellos la evolución fue rápida, a un ritmo si no superior al que su personal era capaz de asimilar si al que la propia empresa podía hacerlo.

Finalmente, otro gran grupo de usuarios, procedentes en gran parte de otras disciplinas ha vivido estos años como una revolución gigantesca en su forma de trabajar. Ingenieros mecánicos, civiles, arquitectos, etc. han encontrado en sofisticados sistemas hardware/software de CAD/CAM un nuevo mundo para su habitual trabajo.

Evolución, revolución... o parálisis son diferentes perspectivas de una misma historia, de una década de trabajo en la que, al menos, conocemos mejor los productos que manejamos y sabemos hacer las cosas un poco mejor.

Ojala que en otro aniversario, en 1997 podamos decir que estos objetivos se han cumplido y podamos levantar la vista hacia otros más ambiciosos.