

# EDITORIAL

La Computación, como todas las ciencias, desarrolla sus ideas creativas, avances e innovaciones mediante la interacción de dos efectos o *fuerzas*:

- el *empuje*, que viene dado por los nuevos desarrollos, invenciones o descubrimientos (por ejemplo, los *fractales* de Mandelbröt), y
- el *tirón*, dado por las necesidades actuales del mercado o de un sector o servicio, por ejemplo, la oportunidad de usar la red www para comercio electrónico.

Daré un ejemplo de avance mediante el empuje. En 1963, Lotfi Zadeh inventó los conjuntos difusos («fuzzy sets»), que ampliaron en mucho la percepción de un conjunto o colección de objetos. Pronto siguieron funciones difusas, relaciones difusas, algoritmos difusos, etc. Proliferaron los artículos sobre temas difusos, se volvió asunto de moda. Todo esto tardó mucho más en *aplicarse*, y fueron los japoneses los que primero lograron la introducción de la lógica difusa a productos comerciales, por ejemplo, circuitos para eliminar la inestabilidad de grabadoras de video portátiles de 8 mm. Dijéramos que estos descubrimientos nacen “por inspiración”, sin idea de dónde y cómo pueden usarse o aplicarse. Otro ejemplo: el abaratamiento de la memoria de acceso directo, que hizo posible, por ejemplo, la popularización de interfaces gráficas con abundantes colores y buena resolución.

La “ventaja” de los avances clasificados como empujes es que no necesitamos definir para qué sirven, no hay un juez externo que nos diga: muy bonito lo que acabas de inventar, pero no resuelve mi problema, y ni idea tengo de cómo usarlo.

La necesidad de las empresas geográficamente dispersas de tener su información ordenada y a la mano, es un ejemplo de tirón. La necesidad, el problema a resolver, está ahí. Es útil resolver este problema, la solución puede venderse bien. Este tirón ha impulsado avances en computación distribuida, en groupware, productos como Lotus Notes, conceptos como bodega de datos, minería de datos.

Otro ejemplo de tirón: la necesidad de entender textos escritos en lenguaje natural (en español, digamos): averiguar cuáles son los temas principales de un artículo; hacer un resumen (por computadora, se entiende) del mismo; analizar tendencias entre varios artículos; descubrir analogías o concordancias (¿cuáles artículos tienen el mismo punto de vista con respecto a “los derechos humanos?”); hacer operaciones de promedio o consenso (“¿cuál es la opinión promedio de estos artículos, con respecto a la guerra del Golfo Pérsico?”).

La “ventaja” de los avances clasificados como tirones es que la aplicación (el cliente) está ahí, esperando. El problema está definido, por lo que no hay que hacer suposiciones, basta con familiarizarse (con el problema).

Ahora bien, ¿qué conviene recomendar a un científico joven que se aventura en una ciencia como la Computación, nacida apenas hace 50 años?

- Dedícate a “empujes”. Inventa algo nuevo, sin preocuparte de cómo puede utilizarse. La ventaja de este enfoque es que casi todo está por inventarse, por descubrirse (en Computación). Un colega que siga este consejo llegará cada mañana a su escritorio y se preguntará: ¿qué deseo inventar hoy? De lo único que tiene que preocuparse es que *no haya sido ya previamente inventado o descubierto*.
- Dedícate a “tirones”. Observa qué problemas importantes de tu organización, de tu empresa, de tu región, de tu país, no han sido resueltos. Escoge uno. De las herramientas que tú dominas, escoge las más afiladas y trata de vencer el problema. La ventaja de este enfoque es que *es fácil ver si ya se resolvió o no*. La desventaja: quizá tu solución (aunque útil y probablemente bien pagada) no sea novedosa, y no puedas publicarla como una verdadera aportación original.

Qué escogerá nuestro joven amigo depende de sus inclinaciones, contacto con problemas reales, velocidad de reacción, etc. Un científico dedicado exclusivamente a “empujes” pronto perderá contacto con la realidad, y correrá el riesgo de trabajar mucho en aportaciones estériles, aunque originales. Un colega dedicado exclusivamente a “tirones” corre el riesgo de perder el hilo

de las nuevas contribuciones, de dejar de aportar trabajos originales, aunque supuestamente sus desarrollos son útiles (y probablemente bien remunerados).

En países con escasos recursos (como muchos de los iberoamericanos), quizá el investigador de tirones tenga mejor fortuna y sea más aceptado.

Empero, en Computación podemos usar una estrategia híbrida: tratar de resolver un problema real (tirón) utilizando cualesquiera métodos que funcionen. Una vez resuelto el problema, conviene retomarlo, buscando cómo generalizarlo, cómo plantearlo de una manera más sistemática, más global. Después de todo, estamos bien familiarizados con él. De aquí puede surgir un nuevo método, una innovación “de empuje” (inspirada por nuestro problema original). Esta nueva herramienta podría tal vez usarse en otros problemas no resueltos, y volver así al tirón. Es como “estar en el frente de batalla” o en la línea, preocupado por un problema real (abatir al enemigo). Después de cierto éxito, conviene ahora ir a la retaguardia, a analizar qué fue lo bueno y lo malo que hicimos. En la retaguardia trataremos de inventar un arma mejor, dada nuestra experiencia previa en la línea.

Esta oscilación entre línea y retaguardia no puede darse en todas las ciencias, sino sólo en las que (¡oh nosotros, afortunados!) por su poco desarrollo, hacen posible que lo último que se acaba de inventar sea pronto aplicable a problemas reales.

Una observación final: tenga claro si lo que usted está haciendo es “aplicación” o “investigación”. ¿Qué es aplicación, o transferencia de tecnología? Algo útil. Por consiguiente, se caracteriza por la existencia de un cliente, de alguien que desea usarlo. Y está dispuesto a pagar, a patrocinar, a colaborar en la solución a su problema. ¿Qué es investigación? Algo que persigue inventar o descubrir o definir cosas nuevas, originales. Usted tiene que estar seguro de que lo que pretende hacer no se ha hecho. Dos errores comunes:

- Construir algo (que no es tema de investigación) que *pudiera ser útil*, para que, una vez construido, busquemos un cliente para ver si le interesa. Construir para un cliente inexistente es construir para el anaquel de exhibiciones.
- Inventar o proponer conceptos que nos parecen *muy bellos* o muy interesantes, pero que, por no haber leído la literatura suficientemente, resulta que fueron inventados hace tiempo, de manera más general, y presentados más elegantemente. Es decir, inventar el agua tibia.

Adolfo Guzmán Arenas  
Editor en Jefe

---

“Computación y Sistemas” es una publicación trimestral, de circulación internacional, que fomenta y patrocina el *Centro de Investigación en Computación (CIC)* del *Instituto Politécnico Nacional (IPN)* para la comunidad científica y académica iberoamericana. **Volumen I, Número 4**, abril-junio de 1998. Tiraje: 2500 ejemplares. —Registro ante la Dirección General de Derechos de Autor de la Secretaría de Educación Pública número 000998/97. —Autorización de Licitud de Título y Contenido por la Comisión Calificadora de Publicaciones y Revistas Ilustradas de la Secretaría de Gobernación en trámite. —El contenido de los artículos es responsabilidad exclusiva de sus respectivos autores. —Queda prohibida la reproducción total o parcial, por cualquier medio, sin el permiso expreso del editor.

Informes y Suscripciones: Revista Computación y Sistemas. Centro de Investigación en Computación del IPN. Apartado Postal 75-476. 07738 México DF, MÉXICO. Teléfono: +52-5 729-6000 ext. 56571 Fax: +53-5 729-6000 ext. 56608  
Página Web: <http://www.cic.ipn.mx/~CyS> Correo electrónico: [revista@pollux.cic.ipn.mx](mailto:revista@pollux.cic.ipn.mx)

---

The journal *Computación y Sistemas* (Computer Science and Systems) is published by the Centro de Investigación en Computación (Computer Science Research Center) of the Instituto Politécnico Nacional (National Polytechnic Institute), and is published quarterly.  
Volume I, Number 4, April-June, 1998. Printing 2500.  
The authors are responsible for the contents of the articles.

Subscriptions. Revista Computación y Sistemas. Centro de Investigación en Computación del IPN, Apartado Postal 75-476 México D.F. 07738.  
Telephone +52-5 729.60.00 ext. 56571, fax +52-5 729.60.00 ext 56607  
Web Page: <http://www.cic.ipn.mx/~CyS> E-mail: [revista@pollux.cic.ipn.mx](mailto:revista@pollux.cic.ipn.mx)