

# Contenido

## **¿Qué son los Agentes Inteligentes?**

Lic. Rosa Eunice Echeverría Portillo, Lic. María Teresa Lozano Hernández (CIDETEC-IPN)

1

## **Algoritmo para Aproximar el Área Bajo la Curva de la Función Normal Estándar**

6

M. en C. Víctor Manuel Silva García, M. en C. Eduardo Vega Alvarado (CIDETEC-IPN);  
Ing. Salvador Pérez Cárdenas, Ing. Miguel Ángel Jiménez Cruz. (ESIQUE-IPN)

## **El Problema de Adopción de las Tecnologías de Información en las PyME's**

Lic. Elizabeth Acosta Gonzaga (CIDETEC-IPN), Abraham Gordillo Mejía (UPIICSA-IPN)

11

## **Una Perspectiva de las Bases de Datos Orientadas a Objetos Respecto a las Relacionales**

16

M. en C. Ivonne Rivera Santos, LAI. Maria del Carmen Celis Ortega (CIC-IPN);  
LAI. Eunice Echeverría Portillo (CIDETEC-IPN)

## **Interpretación y Maquinado de Archivos Gerber**

Ing. Agustín Cruz Contreras, M. en C. J. Carlos Herrera Lozada,  
M. en C. J. Carlos González Robles (CIDETEC-IPN)

27

## **Técnicas, Estrategias y Herramientas para el Desarrollo de Software Educativo**

31

Ing. Patricia Pérez Romero (CIDETEC-IPN)

## **Morfología Matemática en Tiempo Real para Imágenes Binarias (B&N)**

Ing. Agustín Cruz Contreras (CIDETEC-IPN)

35

# ¿Qué son los Agentes Inteligentes?

Lic. Rosa Eunice Echeverría Portillo  
Lic. María Teresa Lozano Hernández  
Profesoras del CIDETEC-IPN

**E**l mundo en el que vivimos, con el que interactuamos y donde desarrollamos nuestras actividades, es una sociedad dominada por la información, la cual se ha convertido en una materia prima muy valiosa. El disponer de la información de mayor calidad y más reciente es vital en la mayoría de los sectores de nuestra sociedad.

La red Internet ha superado fronteras geográficas, físicas y morales. La tecnología interactiva ha generado cambios destacados en nuestros modos de vida. Estas fronteras están siendo enriquecidas por la posibilidad de aprender y explorar con un nivel de profundidad que antes no era posible.

Pero no todo es dulce, la red mundial (world wide web) también proporciona una importante lección negativa. Por emocionante que sea, todavía es muy caótica y dista mucho de ser una verdadera infraestructura de información. La mejor manera de encontrar algo es buscar, y esto puede llevar horas; es como caminar por un gigantesco mercado con miles de calles y callejones rebosantes de material. La mayoría es aburrida o irrelevante para nuestras necesidades.

Merece especial atención el advenimiento de una nueva generación de

sistemas inteligentes, más humanos y accesibles, que vendrán de la mano del software inteligente.

Lo cierto es que aún no existen máquinas que superen la inteligencia y habilidad humana, pero mientras tanto los agentes están diseñados para ahorrarnos tiempo y trabajo, al tiempo que tratamos de que gobierne la paz en el reino de la información, en el mundo en línea (on-line).

---

## AGENTES INTELIGENTES

---

Existen muchas definiciones de agentes; analizando algunas de ellas desde diferentes puntos de vista, es posible entender varios aspectos importantes de su mundo y aplicación.

Según las definiciones del diccionario [1], un agente es:

- Algo que produce o es capaz de producir un efecto.
- Algo que actúa para o en lugar de otro por autorización de él.
- Un medio o instrumento por el cual, mediante una inteligencia dada, se logra un resultado.

Los agentes inteligentes tienen un significado más amplio desde el punto de vista de las ciencias de la computación, especialmente de los investigadores en Inteligencia Artificial.

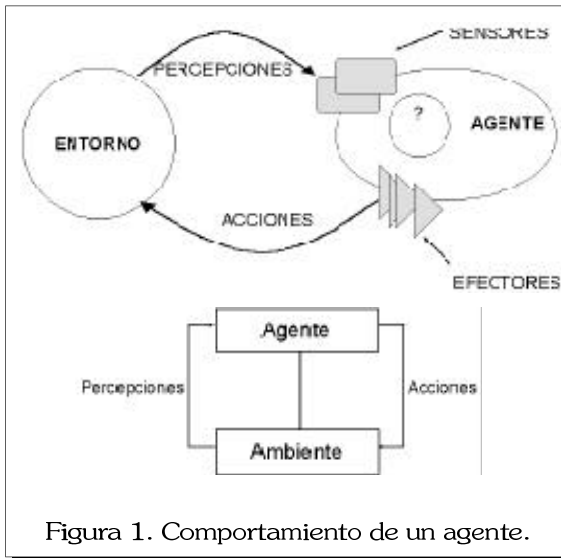
Algunos de estos conceptos son:

Un agente inteligente se considera como un sistema de computación que sustituye a una persona o proceso, para realizar una actividad o cumplir con un requerimiento. La entidad sustituta ofrece capacidades de toma de decisiones similares a las descritas por las intenciones de un humano. Un agente inteligente puede operar dentro de los límites de una necesidad general o representada con precisión, y entre los límites de un espacio de información dado [2].

También se define a los agentes inteligentes en tres formas, ordenadas por su grado de sofisticación o inteligencia. Con base en estas definiciones se revisarán, más adelante, aspectos importantes del diseño de agentes [3].

En sentido genérico, y considerando el comportamiento, un agente es algo que se puede ver percibiendo su ambiente a través de sensores y actuando sobre el mismo a través de efectores. (Figura 1).

Son programas encargados de establecer interfases y administrar la información, a fin de tratar de satisfacer las necesidades de los clientes o usuarios. Desde que el término agente se usó por primera vez, ha provocado una gran discusión en la comunidad de la Inteligencia Artificial.



to inmerso en el ambiente que ha sido construido por el diseñador. Wooldridge [4] afirma que la autonomía es la característica principal que identifica a los agentes inteligentes distinguiéndolos sobre cualquier sistema experto y, por lo tanto, define a un agente como un programa de computador capaz de realizar acciones autónomas flexibles en algún entorno, considerando la flexibilidad como el poseer las características de reactividad, proactividad y sociabilidad.

No se puede negar que el término agente es una palabra de moda, sobre todo en Internet. Los programadores no dudan en denominar como agente a cualquier programa que reúna ciertas características, abusando del término.

Teniendo en cuenta la racionalidad, para cada posible secuencia de percepciones, un agente racional ideal puede hacer cualquier cosa para maximizar su medida de rendimiento, sobre la base de la evidencia ofrecida por la secuencia de percepciones y cualquier conocimiento integrado que tenga.

Un agente racional es autónomo en la medida que sus acciones y preferencias dependan de su propia experiencia en lugar del conocimiento

Un sistema reactivo es aquel que mantiene una relación continua con su entorno y responde a los cambios en el mismo.

Un sistema proactivo dirige su comportamiento al logro de unas metas; por lo tanto, no sólo se maneja por eventos, sino que también puede tomar la iniciativa.

La sociabilidad es un aspecto importante en las comunidades de agentes. El mundo real es un entorno multiagente MA (figura 2) en donde no se puede ir tratando de alcanzar unas metas sin tener en cuenta otras. Algunas metas sólo se pueden lograr con la cooperación de otras. Demasiado

zeau [5] define los sistemas multiagente como un conjunto de agentes posibles organizados, que interactúan en un ambiente común y poseen los cuatro elementos fundamentales A E I O:

A (Agentes): arquitecturas internas de las entidades de procesamiento.

E (Entorno): elementos dependientes del dominio para construir interacciones externas entre entidades.

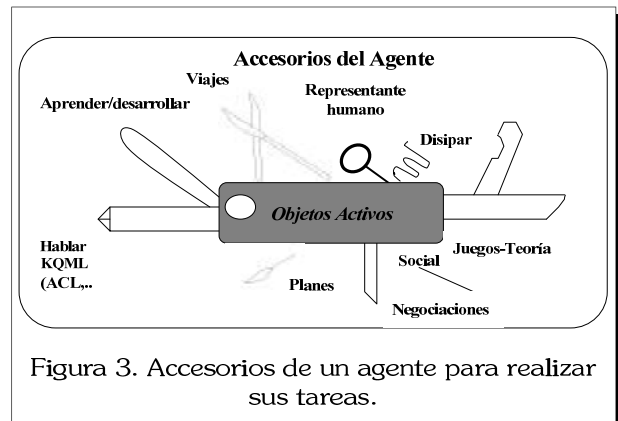
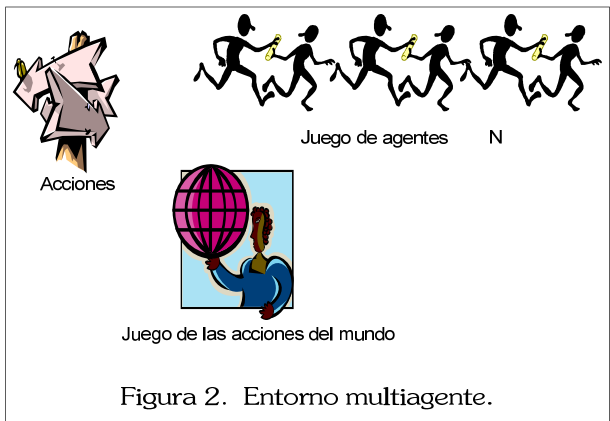
I (Interacciones): elementos para construir interacciones internas entre entidades.

O (Organizaciones): elementos para estructurar conjuntos de entidades de acuerdo a sus funciones en el sistema multiagente.

En la figura 3 se puede observar una representación de los diferentes accesorios con los que cuenta el agente para desempeñar sus funciones particulares, y en conjunto con otros agentes [7].

Dependiendo del área de aplicación de los agentes, Somers [8] también afirma que estos pueden exhibir las siguientes propiedades:

- Movilidad: la habilidad para moverse a través de la red.



- Veracidad: Un agente no comunicará información que considere falsa.
- Benevolencia: los agentes no tienen metas conflictivas y por lo tanto deben siempre tratar de hacer lo que se les ha encomendado.
- Racionalidad: los agentes deben actuar para lograr sus objetivos y no para prevenir que sus metas no se logren.
- Aprendizaje/Adaptación: los agentes incrementan su rendimiento sobre el tiempo (figura 4).

Atkinson [9] define con amplitud el concepto de agentes en términos de agencia e inteligencia:

Los agentes inteligentes son entidades de software que realizan un conjunto de operaciones sobre la conducta o comportamiento de un usuario u otro programa, con algún grado de independencia o autonomía. Para hacerlo, emplean algún conocimiento o representación de las metas, objetivos o deseos del usuario.

La agencia es el grado de autonomía y autoridad dado al agente y se puede medir al menos cualitativamente por la naturaleza de la interac-

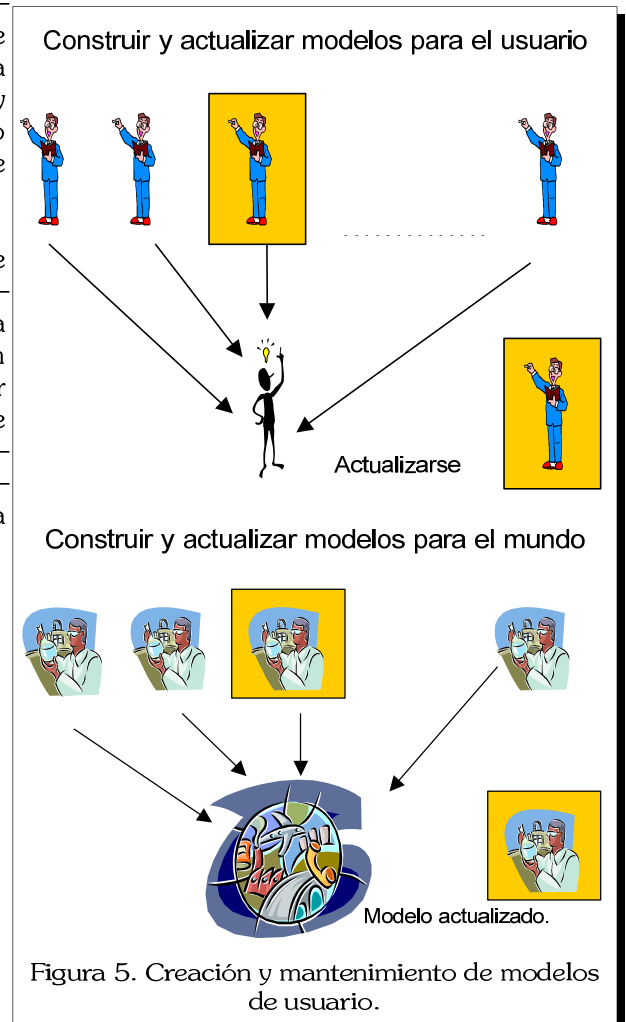
ción entre el agente y otras entidades del sistema. El grado de agencia aumenta si un agente representa al usuario de alguna forma. Un agente más avanzado puede interactuar con otras entidades tales como datos, aplicaciones y servicios, y puede colaborar y “negociar” con otros agentes.

La inteligencia es el grado de razonamiento y capacidad de aprendizaje, la habilidad del agente para aceptar las instrucciones de las metas del usuario y llevar a cabo la tarea delegada por él. Al menos deben existir algunas instrucciones de preferencias, tal vez en forma de reglas con un motor de inferencia o algún otro mecanismo que actúe sobre estas preferencias. Los niveles altos de inteligencia [7] incluyen un modelo de usuario o alguna otra forma de entender y razonar acerca de lo que el usuario quiere hacer (figura 5).

El término agente de software o simplemente agente, se ha vuelto muy popular en los últimos años, por su potente atractivo de servir de “mayordomos” en el “vagabundeo” por Internet para

cubrir las necesidades de información de las personas. Según Hendler [10], tres criterios principales identifican a los agentes de software:

- Comportamiento: un programa es un agente si exhibe un comportamiento de agente o asume responsabilidades de agente (tales como realizar tareas de alto nivel para su propietario).
- Comunicación: es agente si realiza todas las comunicaciones con sus pares en un lenguaje expresivo de comunicación de agentes.
- Implementación: es agente si mantiene y razona sobre la representación explícita de sus metas, patrones y capacidades.



El objetivo de los investigadores en inteligencia artificial y que trabajan con agentes inteligentes, es crear programas que puedan funcionar con efectividad en entornos de importancia para los humanos. Para los agentes físicos, esto incluye entornos donde es difícil la presencia humana (en el espacio exterior, en el fondo del mar, etc.) o donde es peligroso (como en el centro de un reactor nuclear).

Para los agentes de información, los dominios son mundos virtuales que no existen en el universo cognitivo o sensorial del humano, tales como el universo de información en Internet o el "cibespacio" interactivo descrito por la comunidad de la realidad virtual.

Hoy en día existen demasiados medios de información: al día se pueden recibir datos por diez o más conductos diferentes, la mayoría de ellos electrónicos. A la vez, no solamente están dispersos los medios de información, sino también sus fuentes: geográficamente, personalmente, por grupos de interés, etc.

En su versión más ambiciosa o fuerte, un agente sería un programa de inteligencia artificial capaz de desenvolverse autónomamente en las redes de computadoras para buscar información de un modo selectivo.

En su versión débil, la única que existe actualmente, un agente es un programa que se ejecuta en segundo plano (background) para realizar alguna acción cuando se cumplen condiciones predefinidas. Por ejemplo, un agente podría reagrupar los archivos del disco duro cuando la fragmentación supere, por ejemplo, el 20%.

Estos agentes se activan automáticamente cuando se dan ciertas condiciones, o bien mediante elecciones expresas de menú. Tales aplicaciones pueden lanzarse contra todos los

documentos de una base de datos, o bien contra aquellos documentos que satisfagan una determinada condición.

Otro ejemplo es el de un agente que, en un periodo de tiempo determinado, busca en los últimos documentos que han entrado a una base de datos para seleccionar los que correspondan a un perfil semántico determinado, y los muestra en un orden también determinado, haciendo así una labor de alerta sobre los temas de interés para el usuario.

La otra novedad es la integración del mundo Internet, de manera que se pueden almacenar documentos web con bases de datos, en un conjunto de utilidades y bases de datos denominados InterNotes.

Según Maes [11] un agente inteligente debe ser un programa autónomo, capaz de aprender de las experiencias y de adaptarse a nuevas situaciones.

Eso lo convierte en un componente ideal de la interfaz para el usuario en las aplicaciones modernas; así, en vez de tener que enfrentar dígitos parpadeantes de un video, habrá un agente, como el clip de Office, que irá preguntando: ¿Qué quiere hacer? ¿Quiere poner la hora? ¿Grabar un video? y cosas así, con el agente siempre latente en segundo plano esperando peticiones del usuario.

Somos nosotros, los productores de sistemas artificiales, los responsables de definir las pautas a seguir, a fin de lograr utilidad y crecimiento ordenado. Esta es una área donde los tecnólogos de la informática tendrán que concentrar su investigación, si es que se quiere avanzar en la importante cuestión de la producción de máquinas que se comprendan unas a otras.

A veces se le llama Knowbots a los programas que realizan acciones en nuestro nombre. Esta palabra fue acuñada por Robert Kahn, conocido por los protocolos TCP/IP, para establecer un paralelo con la noción de robots. También se les llama agentes inteligentes, lo que evoca la imagen de hábiles representantes que realizan nuestro trabajo con un fin determinado.

Desde un punto de vista amplio, un agente debe gozar de autonomía y sociabilidad, con capacidad de reacción e iniciativa. La autonomía se da en la medida en que un agente ha de actuar sin (o casi sin) la intervención directa externa.

Por su parte, los agentes de software son programas que desempeñan tareas para un usuario. Aunque esta definición podría hacer pensar en cualquier programa, lo cierto es que tanto los robots como los agentes de software deben tener ciertas propiedades especiales para considerarse realmente "inteligentes"

---

### CONCLUSIÓN

---

Es comúnmente aceptado que los agentes son entidades dentro de un entorno, y que pueden sentir y actuar (no necesariamente en este orden). Esto significa que los agentes no son entidades aisladas, sino que son capaces de comunicarse y colaborar con otras entidades. Simplemente, los agentes que no sean capaces de trabajar conjuntamente con otros agentes (humanos incluidos) están destinados a ser virtualmente inútiles.

Una vez que los agentes están a punto para empezar a colaborar, necesitan encontrar otros agentes con los que llevarán a cabo la colaboración. Ésta es una tarea fácil si saben exactamente qué agentes contactar y

dónde contactarlos. No obstante, es muy difícil que exista una distribución estática de los agentes: la gente está normalmente en movimiento y no siempre está disponible para interactuar con los demás.

Entonces, si repasamos algunas de las propiedades que caracterizan a cualquier agente, podemos comprobar que el agente inteligente es autónomo, funciona independientemente guiado por sus objetivos de diseño y dispone de un control sobre sus acciones y estado interno.

# BIBLIOGRAFÍA

- [1] Merriam-Webster's Collegiate Dictionary, 10th ed., 1993, p.22.
- [2] J.A.King, *Intelligent Agents: Bringing Good Things to Life*. AI Expert, February 1995, pp.17-19.
- [3] Stuart Russell, Peter Norvig, *Inteligencia Artificial un Enfoque Moderno*. Prentice Hall. 1997.
- [4] Mike Wooldridge, *Intelligent Agents: Introduction*. 2nd European Agent Systems Summer School, EASSS2000. August 14 - 18. Saarbrücken, Germany. 2000.
- [5] Yves Demazeau, *Foundations of Multi Agent Systems*. 2nd European Agent Systems Summer School, EASSS2000. August 14 - 18. Saarbrücken, Germany. 2000.
- [6] Sandip Sen, *Learning Agents*. 2nd European Agent Systems Summer School, EASSS2000. August 14 - 18. Saarbrücken, Germany. 2000.
- [7] Van Parunak, *Synthetic Ecosystems: A perspective for Multi-Agent Systems*. 2nd European Agent Systems Summer School, EASSS2000. August 14 - 18. Saarbrücken. 2000.
- [8] Fergal Somers, *HYBRID: Unifying Centralised and Distributed Network Management using Intelligent Agents*. IEEE/IFIP NOMS' 96, Network Operations and Management Symposium, April 1996.
- [9] B. Atkinson, et. al., *IBM Intelligent Agents*. Unicom Seminar on Agent Software, London, UK, May 25 1995.
- [10] James A. Hendler, *Intelligent Agents: Where AI meets Information Technology*. IEEE Expert, pp. 20-22. 1996.
- [11] Maes, P., *Designing Autonomous Agents*. Ed. P. Maes The MIT Press, Cambridge, MA. 1991.

# Algoritmo para Aproximar el Área Bajo la Curva de la Función Normal Estándar

M. en C. Víctor Manuel Silva García,  
M. en C. Eduardo Vega Alvarado,  
Profesores del CIDETEC-IPN  
Ing. Salvador Pérez Cárdenas,  
Ing. Miguel Ángel Jiménez Cruz,  
Profesores de ESIQUIE-IPN.

**L**a función normal definida como

$$f(x) = \left( \frac{1}{\sqrt{2\pi}\sigma} \right) e^{-\frac{(x-u)^2}{2\sigma^2}}$$

para  $-\infty < x < \infty$ , es una de las funciones más estudiadas en la teoría de la probabilidad, con importantes aplicaciones en la estadística. Prácticamente todos los libros que abordan las ramas de probabilidad o estadística tienen como apéndice una tabla, en la cual aparecen los valores de la integral definida de la función normal para un caso particular, a saber, para  $u=0$  y  $\sigma=1$ . Sin embargo, normalmente en ninguno de estos libros se explica la manera de cómo se obtienen los valores de la tabla; es claro que el área bajo la curva se puede aproximar utilizando las sumas de Riemann, pero este proceso puede ser inexacto para algunos casos. El presente trabajo propone otra forma de cálculo, cuya aproximación tiene un error que oscila entre las cotas  $-2.5 \cdot 10^{-14}$  y  $4.16 \cdot 10^{-14}$  para cada intervalo de longitud 0.01.

El descubrimiento de la función normal se debe al matemático alemán Carl Friedrich Gauss, a quién también se debe la demostración del teorema fundamental del álgebra, que afirma que todo polinomio de constantes complejas tiene al menos una raíz. Gauss demostró este importante teorema de cuatro formas diferentes a lo largo de su vida.

## PROCEDIMIENTO

Se partirá del hecho de que si  $f^{(4)}(x)$  es continua en el intervalo  $[a,b]$ , entonces:

$$\int_a^b f(x)dx = \left( \frac{b-a}{6} \right) \left( f(a) + 4f\left(\frac{a+b}{2}\right) + f(b) \right) - \frac{(b-a)^5}{2880} f^{(4)}(c) \quad \dots [1]$$

para un cierto  $c \in [a,b]$

La estrategia para demostrar esta igualdad se encuentra en la sección de problemas del capítulo 18 del libro *Calculus* de Spivak, M., Ed. Reverte, 1992. Cabe mencionar que el teorema de Rolle y la definición de la función  $F(t) = Q_1(x)[f(t)-Q(t)] + Q_1(t)[f(x)-Q(x)]$  juegan un papel importante en la demostración.

Aquí

$$Q_1(x) = (x-a)\left(x - \frac{a+b}{2}\right)^2(x-b)$$

y

$$Q(x) = P(x) + A(x-a)\left(x - \frac{a+b}{2}\right)(x-b)$$

Con A igual a una constante,  $P(x)$  es un polinomio de grado 2 con la siguiente característica:  $P(a)=f(a)$ ,

$$P\left(\frac{a+b}{2}\right) = f\left(\frac{a+b}{2}\right) \quad y$$

La función que se sustituirá en la expresión 1 es :

para  $-\infty < x < \infty$ , la cual se denomina función normal estándar. La cuarta derivada de la función normal estándar es continua en cualquier intervalo  $[a,b]$  en los números reales. Esto se debe básicamente a que la cuarta derivada

## Algoritmo para Aproximar el Área Bajo la Curva de la Función Normal Estándar

se puede expresar como el producto de  $f(x) \cdot P1(x)$ , en donde  $P1(x)$  es un polinomio de grado cuatro.

Si se aplica la fórmula de derivación a

$$f(x) = \left( \frac{1}{\sqrt{2\pi}} \right) e^{-\frac{x^2}{2}}$$

de manera sucesiva, es sencillo encontrar que:

$$f^{(4)}(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}} (x^4 - 6x^2 + 3)$$

Ahora se buscarán las cotas para la función  $f^{(4)}(x)$  con  $x \in [a, b]$  y así conocer aproximadamente el valor de

$$\frac{(b-a)^5}{2880} f^{(4)}(c)$$

Antes de seguir adelante es conveniente mencionar que, en general, en las tablas donde se proporcionan los valores de la integral definida de la normal estándar, se considera al intervalo  $[a, b]$  como  $[0, 3]$ , el cual a su vez se divide en subintervalos de longitud 0.01.

En este orden de ideas se analizará la función  $f^{(4)}(x)$  en dicho intervalo, con el fin de averiguar los valores máximo y mínimo de  $f^{(4)}(x)$  con  $x \in [a, b]$ .

Para esto se deben considerar tres aspectos, a saber:

1. Los puntos singulares de  $f^{(4)}(x)$  en  $(0, 3)$ .
2.  $f^{(4)}(0)$  y  $f^{(4)}(3)$ .
3. Los puntos de  $x \in [0, 3]$  tales que  $f^{(4)}(x)$  no es derivable.

Dado que la función  $f^{(4)}(x)$  es derivable en todo el intervalo  $[0, 3]$ , entonces se descarta el último punto. Para localizar los puntos singulares de  $f^{(4)}(x)$  en  $[0, 3]$ , se toma como base la igualdad  $f^{(5)}(x) = 0$ . A partir de ella se obtiene:

$$\frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}} (-x^5 + 10x^3 - 15x) = 0 \quad \dots [2]$$

Dado que

$$\frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}} > 0 \quad \text{para todo } x \in \mathbb{R}$$

La expresión 2 se puede escribir como:

$$x^5 - 10x^3 + 15x = 0 \quad \dots [3]$$

Utilizando [3] se pueden calcular todos los puntos singulares de  $f^{(4)}(x)$ , siendo estos:

$$x_{10} = 0, \quad x_{11} = \sqrt{\frac{10 + \sqrt{40}}{2}}, \quad x_{12} = -\sqrt{\frac{10 + \sqrt{40}}{2}}$$

$$x_{21} = \sqrt{\frac{10 - \sqrt{40}}{2}} \quad \text{y} \quad x_{22} = -\sqrt{\frac{10 - \sqrt{40}}{2}}$$

Sin embargo, dado que el intervalo de interés es  $(0, 3)$ , se tomará en cuenta únicamente:

$$x_{11} \cong 2.857 \quad \text{y} \quad x_{21} \cong 1.355$$

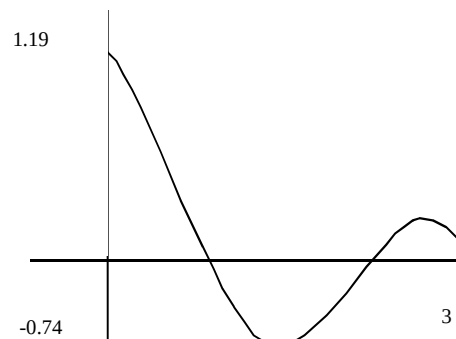
Entonces, los puntos de interés son:  $f^{(4)}(0)$ ,  $f^{(4)}(1.355)$ ,  $f^{(4)}(2.857)$  y  $f^{(4)}(3)$ .

Es simple observar que el valor máximo de  $f^{(4)}(x)$  lo toma en  $x=0$  y el mínimo en  $x=1.355$ , lo que conduce a las cotas siguientes:

$$-2.5(10^{-14}) \leq \frac{(0.01)^5 f(c)}{2880} \leq 4.16(10^{-14})$$

A continuación se presenta la gráfica de la función

$$f^{(4)}(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}} (x^4 - 6x^2 + 3) \quad \text{para } x \in [0, 3]$$



Las cotas que se muestran en la expresión, son válidas para cada uno de los subintervalos de longitud 0.01, que



## Algoritmo para Aproximar el Área Bajo la Curva de la Función Normal Estándar

se forman a partir del intervalo [0,3]. De aquí se puede concluir para fines prácticos que:

$$\frac{(0.01)^5 f(c)}{2880} \cong 0$$

Ahora bien, si se tiene en cuenta la conclusión anterior, y se divide el intervalo [0,3] en 300 subintervalos de longitud 0.01, entonces el área bajo la curva se puede aproximar de la siguiente manera:

$$\int_0^3 f(x)dx \cong \left[ \sum_{i=1}^{300} e^{-\frac{[(i-1)(0.01)]^2}{2}} + 4e^{-\frac{[(i-1/2)(0.01)]^2}{2}} + e^{-\frac{[i(0.01)]^2}{2}} \right]$$

multiplicado por  $\frac{0.01}{\sqrt{2\pi}}$

A continuación se presenta un programa en lenguaje Fortran 90/95 para aproximar el área bajo la curva en [0,3].

```
PROGRAM Tabla_Normal
```

```
!Propósito: Calcular los valores del área bajo la curva normal,&  
!en el intervalo [0,3].
```

```
IMPLICIT NONE
```

```
REAL :: C
```

```
CHARACTER (1en=30) Archivo_de_Salida
```

```
INTEGER :: Estado
```

```
REAL, DIMENSION (0 : 309) :: A=0.
```

```
INTEGER :: i
```

```
REAL, PARAMETER :: RAIZ_DO_PI=2.506606, INC=0.01
```

```
C= (INC)/(6*RAIZ_DO_PI)
```

```
DO i=1,309
```

```
  A(i)=(exp((-0.5)*((REAL(i)-1.)*(INC))**2))+  
    4*exp((-0.5)*(((REAL(i)-0.5)*(INC))**2))+  
    exp((-0.5) & *(((REAL(i))*(INC))**2)))*C+A(i-1)
```

```
END DO
```

```
WRITE(*,*)DAME NOMBRE DE Archivo_de_Salida
```

```
READ(*,*)Archivo_de_Salida
```

```
OPEN(20,FILE=Archivo_de_Salida,STATUS='NEW',
```

```
  AC& TION='WRITE', IOSTAT=Estado)
```

```
WRITE (20,100) (A(i), i=0,309)
```

```
100FORMAT (10(1x,F7.4))
```

```
END PROGRAM
```

### CONCLUSIONES

- \* Si se compara el método propuesto con el método de Riemann, para situaciones similares (tres sumandos por intervalo), se observa que el método de Riemann puede tener fallas en el tercer decimal (**tabla1**).
- \* Con el método propuesto se pueden obtener resultados confiables, hasta del orden de  $10^{-11}$ . De hecho, se realizó el cálculo de una tabla con cinco decimales (**tabla2**).
- \* Una aplicación de este procedimiento es de utilidad para los alumnos del área de ingeniería o del área de ciencias, ya que es importante que ellos puedan obtener su tabla de la función normal estándar de manera confiable.

### BIBLIOGRAFÍA

- [1] Stephen Chapman, *Fortran 90/95 for Scientists and Engineers*. Editorial Mc Graw Hill.
- [2] John E. Freud, Ronald E. Walpole, *Estadística Matemática*, Editorial Prentice Hall.
- [3] Michael Spivak, *Calculus*, Editorial Reverté.
- [4] D. Struik, *Historia Concisa de las Matemáticas*. Instituto Politécnico Nacional.

# Algoritmo para Aproximar el Área Bajo la Curva de la Función Normal Estándar

| AREA BAJO LA CURVA DE LA NORMAL ESTANDAR<br>MÉTODO PROPUESTO |        |        |        |        |        |        |        |        |        |        |
|--------------------------------------------------------------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| X                                                            | .00    | .01    | .02    | .03    | .04    | .05    | .06    | .07    | .08    | .09    |
| 0.0                                                          | 0.0000 | 0.0040 | 0.0080 | 0.0120 | 0.0160 | 0.0199 | 0.0239 | 0.0279 | 0.0319 | 0.0359 |
| 0.1                                                          | 0.0398 | 0.0438 | 0.0478 | 0.0517 | 0.0557 | 0.0596 | 0.0636 | 0.0675 | 0.0714 | 0.0753 |
| 0.2                                                          | 0.0793 | 0.0832 | 0.0871 | 0.0910 | 0.0948 | 0.0987 | 0.1026 | 0.1064 | 0.1103 | 0.1141 |
| 0.3                                                          | 0.1179 | 0.1217 | 0.1255 | 0.1293 | 0.1331 | 0.1368 | 0.1406 | 0.1443 | 0.1480 | 0.1517 |
| 0.4                                                          | 0.1554 | 0.1591 | 0.1628 | 0.1664 | 0.1700 | 0.1736 | 0.1772 | 0.1808 | 0.1844 | 0.1879 |
| 0.5                                                          | 0.1915 | 0.1950 | 0.1985 | 0.2019 | 0.2054 | 0.2088 | 0.2123 | 0.2157 | 0.2190 | 0.2224 |
| 0.6                                                          | 0.2257 | 0.2291 | 0.2324 | 0.2357 | 0.2389 | 0.2422 | 0.2454 | 0.2486 | 0.2517 | 0.2549 |
| 0.7                                                          | 0.2580 | 0.2612 | 0.2642 | 0.2673 | 0.2704 | 0.2734 | 0.2764 | 0.2794 | 0.2823 | 0.2852 |
| 0.8                                                          | 0.2881 | 0.2910 | 0.2939 | 0.2967 | 0.2995 | 0.3023 | 0.3051 | 0.3079 | 0.3106 | 0.3133 |
| 0.9                                                          | 0.3159 | 0.3186 | 0.3212 | 0.3238 | 0.3264 | 0.3289 | 0.3315 | 0.3340 | 0.3365 | 0.3389 |
| 1.0                                                          | 0.3413 | 0.3438 | 0.3461 | 0.3485 | 0.3508 | 0.3531 | 0.3554 | 0.3577 | 0.3599 | 0.3621 |
| 1.1                                                          | 0.3643 | 0.3665 | 0.3686 | 0.3708 | 0.3729 | 0.3749 | 0.3770 | 0.3790 | 0.3810 | 0.3830 |
| 1.2                                                          | 0.3849 | 0.3869 | 0.3888 | 0.3907 | 0.3925 | 0.3944 | 0.3962 | 0.3980 | 0.3997 | 0.4015 |
| 1.3                                                          | 0.4032 | 0.4049 | 0.4066 | 0.4082 | 0.4099 | 0.4115 | 0.4131 | 0.4147 | 0.4162 | 0.4177 |
| 1.4                                                          | 0.4192 | 0.4207 | 0.4222 | 0.4236 | 0.4251 | 0.4265 | 0.4279 | 0.4292 | 0.4306 | 0.4319 |
| 1.5                                                          | 0.4332 | 0.4345 | 0.4357 | 0.4370 | 0.4382 | 0.4394 | 0.4406 | 0.4418 | 0.4430 | 0.4441 |
| 1.6                                                          | 0.4452 | 0.4463 | 0.4474 | 0.4485 | 0.4495 | 0.4505 | 0.4515 | 0.4525 | 0.4535 | 0.4545 |
| 1.7                                                          | 0.4554 | 0.4564 | 0.4573 | 0.4582 | 0.4591 | 0.4599 | 0.4608 | 0.4616 | 0.4625 | 0.4633 |
| 1.8                                                          | 0.4641 | 0.4649 | 0.4656 | 0.4664 | 0.4671 | 0.4678 | 0.4686 | 0.4693 | 0.4700 | 0.4706 |
| 1.9                                                          | 0.4713 | 0.4719 | 0.4726 | 0.4732 | 0.4738 | 0.4744 | 0.4750 | 0.4756 | 0.4762 | 0.4767 |
| 2.0                                                          | 0.4773 | 0.4778 | 0.4783 | 0.4788 | 0.4793 | 0.4798 | 0.4803 | 0.4808 | 0.4812 | 0.4817 |
| 2.1                                                          | 0.4821 | 0.4826 | 0.4830 | 0.4834 | 0.4838 | 0.4842 | 0.4846 | 0.4850 | 0.4854 | 0.4857 |
| 2.2                                                          | 0.4861 | 0.4865 | 0.4868 | 0.4871 | 0.4875 | 0.4878 | 0.4881 | 0.4884 | 0.4887 | 0.4890 |
| 2.3                                                          | 0.4893 | 0.4896 | 0.4898 | 0.4901 | 0.4904 | 0.4906 | 0.4909 | 0.4911 | 0.4913 | 0.4916 |
| 2.4                                                          | 0.4918 | 0.4920 | 0.4922 | 0.4925 | 0.4927 | 0.4929 | 0.4931 | 0.4932 | 0.4934 | 0.4936 |
| 2.5                                                          | 0.4938 | 0.4940 | 0.4941 | 0.4943 | 0.4945 | 0.4946 | 0.4948 | 0.4949 | 0.4951 | 0.4952 |
| 2.6                                                          | 0.4953 | 0.4955 | 0.4956 | 0.4957 | 0.4959 | 0.4960 | 0.4961 | 0.4962 | 0.4963 | 0.4964 |
| 2.7                                                          | 0.4965 | 0.4966 | 0.4967 | 0.4968 | 0.4969 | 0.4970 | 0.4971 | 0.4972 | 0.4973 | 0.4974 |
| 2.8                                                          | 0.4974 | 0.4975 | 0.4976 | 0.4977 | 0.4977 | 0.4978 | 0.4979 | 0.4980 | 0.4980 | 0.4981 |
| 2.9                                                          | 0.4981 | 0.4982 | 0.4983 | 0.4983 | 0.4984 | 0.4984 | 0.4985 | 0.4985 | 0.4986 | 0.4986 |
| 3.0                                                          | 0.4987 | 0.4987 | 0.4987 | 0.4988 | 0.4988 | 0.4989 | 0.4989 | 0.4989 | 0.4990 | 0.4990 |

| AREA BAJO LA CURVA DE LA NORMAL ESTANDAR<br>MÉTODO DE RIEMANN |        |        |        |        |        |        |        |        |        |        |
|---------------------------------------------------------------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| X                                                             | .00    | .01    | .02    | .03    | .04    | .05    | .06    | .07    | .08    | .09    |
| 0.0                                                           | 0.0000 | 0.0040 | 0.0080 | 0.0120 | 0.0160 | 0.0199 | 0.0239 | 0.0279 | 0.0319 | 0.0359 |
| 0.1                                                           | 0.0398 | 0.0438 | 0.0478 | 0.0517 | 0.0557 | 0.0596 | 0.0636 | 0.0675 | 0.0714 | 0.0753 |
| 0.2                                                           | 0.0792 | 0.0832 | 0.0871 | 0.0909 | 0.0948 | 0.0987 | 0.1025 | 0.1064 | 0.1102 | 0.1141 |
| 0.3                                                           | 0.1179 | 0.1217 | 0.1255 | 0.1293 | 0.1330 | 0.1368 | 0.1405 | 0.1443 | 0.1480 | 0.1517 |
| 0.4                                                           | 0.1554 | 0.1590 | 0.1627 | 0.1663 | 0.1700 | 0.1736 | 0.1772 | 0.1808 | 0.1843 | 0.1879 |
| 0.5                                                           | 0.1914 | 0.1949 | 0.1984 | 0.2019 | 0.2053 | 0.2088 | 0.2122 | 0.2156 | 0.2189 | 0.2223 |
| 0.6                                                           | 0.2256 | 0.2290 | 0.2323 | 0.2355 | 0.2388 | 0.2420 | 0.2452 | 0.2484 | 0.2516 | 0.2548 |
| 0.7                                                           | 0.2579 | 0.2610 | 0.2641 | 0.2672 | 0.2702 | 0.2732 | 0.2762 | 0.2792 | 0.2821 | 0.2851 |
| 0.8                                                           | 0.2880 | 0.2908 | 0.2937 | 0.2965 | 0.2994 | 0.3021 | 0.3049 | 0.3076 | 0.3104 | 0.3131 |
| 0.9                                                           | 0.3157 | 0.3184 | 0.3210 | 0.3236 | 0.3262 | 0.3287 | 0.3312 | 0.3337 | 0.3362 | 0.3387 |
| 1.0                                                           | 0.3411 | 0.3435 | 0.3459 | 0.3482 | 0.3506 | 0.3529 | 0.3551 | 0.3574 | 0.3596 | 0.3619 |
| 1.1                                                           | 0.3640 | 0.3662 | 0.3683 | 0.3705 | 0.3725 | 0.3746 | 0.3767 | 0.3787 | 0.3807 | 0.3826 |
| 1.2                                                           | 0.3846 | 0.3865 | 0.3884 | 0.3903 | 0.3922 | 0.3940 | 0.3958 | 0.3976 | 0.3994 | 0.4011 |
| 1.3                                                           | 0.4028 | 0.4045 | 0.4062 | 0.4079 | 0.4095 | 0.4111 | 0.4127 | 0.4143 | 0.4158 | 0.4173 |
| 1.4                                                           | 0.4188 | 0.4203 | 0.4218 | 0.4232 | 0.4246 | 0.4260 | 0.4274 | 0.4288 | 0.4301 | 0.4314 |
| 1.5                                                           | 0.4327 | 0.4340 | 0.4353 | 0.4365 | 0.4378 | 0.4390 | 0.4402 | 0.4413 | 0.4425 | 0.4436 |
| 1.6                                                           | 0.4447 | 0.4458 | 0.4469 | 0.4480 | 0.4490 | 0.4500 | 0.4510 | 0.4520 | 0.4530 | 0.4540 |
| 1.7                                                           | 0.4549 | 0.4559 | 0.4568 | 0.4577 | 0.4586 | 0.4594 | 0.4603 | 0.4611 | 0.4619 | 0.4627 |
| 1.8                                                           | 0.4635 | 0.4643 | 0.4651 | 0.4658 | 0.4666 | 0.4673 | 0.4680 | 0.4687 | 0.4694 | 0.4701 |
| 1.9                                                           | 0.4707 | 0.4714 | 0.4720 | 0.4726 | 0.4732 | 0.4738 | 0.4744 | 0.4750 | 0.4756 | 0.4761 |
| 2.0                                                           | 0.4767 | 0.4772 | 0.4777 | 0.4782 | 0.4787 | 0.4792 | 0.4797 | 0.4802 | 0.4807 | 0.4811 |
| 2.1                                                           | 0.4815 | 0.4820 | 0.4824 | 0.4828 | 0.4832 | 0.4836 | 0.4840 | 0.4844 | 0.4848 | 0.4851 |
| 2.2                                                           | 0.4855 | 0.4858 | 0.4862 | 0.4865 | 0.4868 | 0.4872 | 0.4875 | 0.4878 | 0.4881 | 0.4884 |
| 2.3                                                           | 0.4887 | 0.4889 | 0.4892 | 0.4895 | 0.4897 | 0.4900 | 0.4902 | 0.4905 | 0.4907 | 0.4910 |
| 2.4                                                           | 0.4912 | 0.4914 | 0.4916 | 0.4918 | 0.4920 | 0.4922 | 0.4924 | 0.4926 | 0.4928 | 0.4930 |
| 2.5                                                           | 0.4932 | 0.4933 | 0.4935 | 0.4937 | 0.4938 | 0.4940 | 0.4941 | 0.4943 | 0.4944 | 0.4946 |
| 2.6                                                           | 0.4947 | 0.4948 | 0.4950 | 0.4951 | 0.4952 | 0.4953 | 0.4954 | 0.4956 | 0.4957 | 0.4958 |
| 2.7                                                           | 0.4959 | 0.4960 | 0.4961 | 0.4962 | 0.4963 | 0.4964 | 0.4965 | 0.4965 | 0.4966 | 0.4967 |
| 2.8                                                           | 0.4968 | 0.4969 | 0.4969 | 0.4970 | 0.4971 | 0.4972 | 0.4972 | 0.4973 | 0.4974 | 0.4974 |
| 2.9                                                           | 0.4975 | 0.4975 | 0.4976 | 0.4976 | 0.4977 | 0.4978 | 0.4978 | 0.4979 | 0.4979 | 0.4979 |
| 3.0                                                           | 0.4980 | 0.4980 | 0.4981 | 0.4981 | 0.4982 | 0.4982 | 0.4982 | 0.4983 | 0.4983 | 0.4983 |

Tabla 1

# Algoritmo para Aproximar el Área Bajo la Curva de la Función Normal Estándar

| ÁREA BAJO LA CURVA DE LA NORMAL ESTANDAR<br>CINCO CIFRAS DECIMALES |         |         |         |         |         |         |         |         |         |         |
|--------------------------------------------------------------------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|
| X                                                                  | .00     | .01     | .02     | .03     | .04     | .05     | .06     | .07     | .08     | .09     |
| 0.0                                                                | 0.00000 | 0.00399 | 0.00798 | 0.01197 | 0.01595 | 0.01994 | 0.02392 | 0.02790 | 0.03188 | 0.03586 |
| 0.1                                                                | 0.03983 | 0.04380 | 0.04776 | 0.05172 | 0.05567 | 0.05962 | 0.06356 | 0.06750 | 0.07142 | 0.07535 |
| 0.2                                                                | 0.07926 | 0.08317 | 0.08707 | 0.09095 | 0.09484 | 0.09871 | 0.10257 | 0.10642 | 0.11026 | 0.11409 |
| 0.3                                                                | 0.11791 | 0.12172 | 0.12552 | 0.12930 | 0.13307 | 0.13683 | 0.14058 | 0.14431 | 0.14803 | 0.15173 |
| 0.4                                                                | 0.15542 | 0.15910 | 0.16276 | 0.16640 | 0.17003 | 0.17365 | 0.17724 | 0.18082 | 0.18439 | 0.18793 |
| 0.5                                                                | 0.19146 | 0.19498 | 0.19847 | 0.20195 | 0.20540 | 0.20884 | 0.21226 | 0.21566 | 0.21904 | 0.22241 |
| 0.6                                                                | 0.22575 | 0.22907 | 0.23237 | 0.23565 | 0.23892 | 0.24216 | 0.24538 | 0.24857 | 0.25175 | 0.25491 |
| 0.7                                                                | 0.25804 | 0.26115 | 0.26424 | 0.26731 | 0.27035 | 0.27338 | 0.27638 | 0.27935 | 0.28231 | 0.28524 |
| 0.8                                                                | 0.28815 | 0.29103 | 0.29389 | 0.29673 | 0.29955 | 0.30234 | 0.30511 | 0.30785 | 0.31057 | 0.31327 |
| 0.9                                                                | 0.31594 | 0.31859 | 0.32122 | 0.32382 | 0.32639 | 0.32895 | 0.33148 | 0.33398 | 0.33646 | 0.33892 |
| 1.0                                                                | 0.34135 | 0.34376 | 0.34614 | 0.34850 | 0.35083 | 0.35314 | 0.35543 | 0.35769 | 0.35993 | 0.36215 |
| 1.1                                                                | 0.36434 | 0.36650 | 0.36865 | 0.37077 | 0.37286 | 0.37493 | 0.37698 | 0.37900 | 0.38100 | 0.38298 |
| 1.2                                                                | 0.38493 | 0.38686 | 0.38877 | 0.39065 | 0.39252 | 0.39435 | 0.39617 | 0.39796 | 0.39973 | 0.40148 |
| 1.3                                                                | 0.40320 | 0.40491 | 0.40659 | 0.40824 | 0.40988 | 0.41150 | 0.41309 | 0.41466 | 0.41621 | 0.41774 |
| 1.4                                                                | 0.41925 | 0.42073 | 0.42220 | 0.42365 | 0.42507 | 0.42647 | 0.42786 | 0.42922 | 0.43057 | 0.43189 |
| 1.5                                                                | 0.43320 | 0.43448 | 0.43575 | 0.43700 | 0.43822 | 0.43943 | 0.44062 | 0.44180 | 0.44295 | 0.44409 |
| 1.6                                                                | 0.44520 | 0.44630 | 0.44739 | 0.44845 | 0.44950 | 0.45053 | 0.45155 | 0.45254 | 0.45353 | 0.45449 |
| 1.7                                                                | 0.45544 | 0.45637 | 0.45729 | 0.45819 | 0.45907 | 0.45994 | 0.46080 | 0.46164 | 0.46247 | 0.46328 |
| 1.8                                                                | 0.46407 | 0.46486 | 0.46562 | 0.46638 | 0.46712 | 0.46785 | 0.46856 | 0.46926 | 0.46995 | 0.47063 |
| 1.9                                                                | 0.47129 | 0.47194 | 0.47258 | 0.47320 | 0.47381 | 0.47442 | 0.47501 | 0.47558 | 0.47615 | 0.47671 |
| 2.0                                                                | 0.47725 | 0.47779 | 0.47831 | 0.47883 | 0.47933 | 0.47982 | 0.48030 | 0.48078 | 0.48124 | 0.48170 |
| 2.1                                                                | 0.48214 | 0.48257 | 0.48300 | 0.48342 | 0.48383 | 0.48423 | 0.48462 | 0.48500 | 0.48538 | 0.48574 |
| 2.2                                                                | 0.48610 | 0.48645 | 0.48679 | 0.48713 | 0.48746 | 0.48778 | 0.48809 | 0.48840 | 0.48870 | 0.48899 |
| 2.3                                                                | 0.48928 | 0.48956 | 0.48983 | 0.49010 | 0.49036 | 0.49062 | 0.49087 | 0.49111 | 0.49135 | 0.49158 |
| 2.4                                                                | 0.49181 | 0.49203 | 0.49224 | 0.49245 | 0.49266 | 0.49286 | 0.49306 | 0.49325 | 0.49344 | 0.49362 |
| 2.5                                                                | 0.49379 | 0.49397 | 0.49414 | 0.49430 | 0.49446 | 0.49462 | 0.49477 | 0.49492 | 0.49506 | 0.49521 |
| 2.6                                                                | 0.49534 | 0.49548 | 0.49561 | 0.49574 | 0.49586 | 0.49598 | 0.49610 | 0.49621 | 0.49632 | 0.49643 |
| 2.7                                                                | 0.49654 | 0.49664 | 0.49674 | 0.49684 | 0.49693 | 0.49702 | 0.49711 | 0.49720 | 0.49729 | 0.49737 |
| 2.8                                                                | 0.49745 | 0.49753 | 0.49760 | 0.49768 | 0.49775 | 0.49782 | 0.49789 | 0.49795 | 0.49802 | 0.49808 |
| 2.9                                                                | 0.49814 | 0.49820 | 0.49825 | 0.49831 | 0.49836 | 0.49842 | 0.49847 | 0.49852 | 0.49856 | 0.49861 |
| 3.0                                                                | 0.49865 | 0.49870 | 0.49874 | 0.49878 | 0.49882 | 0.49886 | 0.49890 | 0.49893 | 0.49897 | 0.49900 |

Tabla 2

# El Problema de Adopción de las Tecnologías de Información en las PyME's

Elizabeth Acosta Gonzaga  
Profesora del CIDETEC-IPN  
Abraham Gordillo Mejía  
Profesor de UPIICSA-IPN

**E**ste artículo se basa en los resultados del proyecto de investigación No. 200933, de la CGPI del IPN, el cual se desarrolló mediante una investigación de campo en las pequeñas y medianas empresas (PyME) de la demarcación política de Iztacalco, D.F., para determinar el impacto que en su control interno han tenido las tecnologías de información (TI). Para esto fue necesario obtener datos sobre las TI que han adquirido, cuales tienen en operación, en que áreas internas funcionan, y los éxitos y fracasos generados de estas experiencias, a fin de que los resultados obtenidos puedan servir para el mejor diseño de Planes y Programas de estudio en el Instituto.

La composición de las PyME es muy variada y realmente compleja, por dedicarse a giros diversos y con objetivos de lucro familiar y social; por lo tanto, sus formas de organización y de trabajo no se sujetan a modelos teóricos preestablecidos, sino que toman formas sui géneris de acuerdo a como hayan sido concebidas, lo que sin duda incrementa sus problemas de percepción del mercado y de su permanencia en el mismo.

Estas PyME enfrentan muchas situaciones difíciles, tales como accesos de competencia lentos y con alto

grado de incertidumbre por falta de información veraz y confiable, además de sus propias limitaciones en la adquisición y manejo de tecnologías actuales.

Las PyME realmente desconocen el concepto de TI, dándole significados parciales y en ocasiones equivocados, a pesar de que un porcentaje alto de las empresas encuestadas cuenta con un área de sistemas. Es fácil encontrar auténticas “torres de babel”, es decir, existe una gran confusión por el crecimiento amorfo del uso de ciertas tecnologías, encontrándose computadoras personales (PC) de diferentes arquitecturas y configuraciones, redes sin explotarse en forma óptima, software de diferentes proveedores y para diferentes aplicaciones administrativas (que en muchas ocasiones no se usa), etc.

Todas estas situaciones que se mencionan de manera general sirven de introducción a los resultados más precisos y simplificados que arrojó esta investigación, en donde se investigaron cerca de 400 empresas como parte de la muestra estadística que se calculó para este proyecto, y que se presentan a continuación.

---

## RESULTADOS DEL PROYECTO

---

- El 57.5% de las empresas investigadas cuenta con una estructura orgánica definida e incluida en

algún manual de organización del negocio, o bien en algún documento, en donde se representan de manera gráfica todas las áreas de incumbencia del mismo, pero solo en un 63% de ellas existe un área de sistemas plenamente identificada y localizada en un segundo o tercer nivel jerárquico, haciéndola depender de diversas gerencias administrativas.

- Solo un 10% de la muestra manifiesta llevar actividades de Planeación Estratégica de Tecnologías de Información, que incluya su compra, instalación y operación.
- En este tipo de empresas prevalecen los equipos personales PC, manifestando un 89% de la muestra contar al menos con un equipo de este tipo, siendo mínima la presencia de Minicomputadoras con apenas un 4%, Mainframes con un 2% y redes con un 6%. Cabe mencionar que la existencia de Minicomputadoras y Mainframes obedece a que en la muestra se incluyeron por error de muestreo a 20 empresas medianas y grandes.
- Las compras de equipo se realizan con criterios de “moda y precio”, así como de imagen del proveedor o marca, sin considerar otros factores de selección de tecnología.
- Los equipos con que cuentan estas empresas, tienen en promedio

de vida de más de tres años, lo que vuelve la capacidad instalada obsoleta para enfrentar retos actuales del mercado.

- Existe un desconocimiento enorme sobre el significado real del software de desarrollo, limitándose al uso de la paquetería más difundida en el mercado, sin que exista una sola observación que manifieste el uso de este tipo de software.
- El software y los equipos se localizan principalmente en áreas de tipo administrativo, lo cual se manifestó en la muestra con un 80%, indicando la prioridad que sus ejecutivos establecen en sus procesos de toma de decisiones.
- Las áreas de producción, principalmente, están muy desprotegidas respecto de las TI, alcanzando solo un 12% de presencia en la muestra, correspondiendo a empresas que tienen algún tipo de equipo asignado a esas actividades, de donde se deduce que su control interno aún se lleva "a mano", desvalorizando o desconociendo el uso de herramientas de software y hardware.
- Como punto positivo en estas empresas, el 96% del software existente cuenta con licencia de uso, lo que manifiesta la conciencia clara que hay al respecto.
- El concepto de red es poco manejado, debido principalmente al desconocimiento técnico del tema, a pesar de que de forma general se conocen sus beneficios. Solo 42 empresas del total cuentan con algún tipo de red instalada y además funcionando.
- El uso de impresoras se restringe a equipos de matriz de puntos

(obsoletos), con un 4% de presencia en la muestra, el 95% corresponde a equipos de inyección de tinta, y solo el 1% representa equipos tipos láser.

- Al adquirir los equipos y tratar de instalarlos se detectaron situaciones "muy de este gremio de empresas", como son las siguientes: se encontraron 16 negocios que no supieron hacerlo, correspondientes a aquellas empresas que quieren hacer solas el trabajo, sin algún tipo de ayuda; también se encontraron 19 empresas que manifestaron que la alta gerencia, al fallar el arranque de los equipos, prefirió dejarlos así, sin funcionar, para no encontrar más problemas y continuar haciendo el trabajo de la forma tradicional. Una observación curiosa es que ninguna de las empresas encuestadas mencionó haber utilizado un servicio externo para la solución de sus problemas.
- A pesar de todas las condiciones anteriores, el equipo con que cuentan estas empresas es utilizado de alguna forma, y solo 82 equipos se reportaron como "no utilizados", siendo aquellos considerados como obsoletos.
- Con respecto a los recursos humanos, las empresas que cuentan con áreas de sistemas contratan personal de nivel profesional, situación que se deja de lado en las empresas que no cuentan con estas áreas.
- La mayoría de las empresas (90%) utilizan Internet con fines de consulta y correo, instalado en diversos equipos de los negocios, sin llegar aun a considerar plenamente el concepto de empresas e.com.

Estos fueron los resultados principales que generó el estudio de las

pequeñas y medianas empresas en la Delegación Iztacalco, las cuales sin duda no están preparadas para afrontar hoy en día y de manera integral, las exigencias competitivas del mercado. Ciertamente es que lo hacen parcialmente con estrategias de costos y precios, o bien mediante financiamientos y crecimientos, pero se olvidan o bien desconocen las bondades de las tecnologías de información.

Las necesidades de capacitación, asesoría y consultoría son evidentes en este sector de empresas, desde el conocimiento básico sobre las tecnologías de información, hasta los beneficios que proporcionan a las organizaciones, sin olvidar la asesoría técnica para la compra de tecnología, su instalación, adopción, uso y explotación.

## EL PROBLEMA DE SELECCIÓN DE SOFTWARE

Como ya se mencionó, estas empresas se dejan influenciar por las modas y las trampas de precio e imagen de proveedor, ya que el software con que cuentan no obedece a estrategias de planeación en sus adquisiciones de tecnología; el cuadro 1 indica el software que manejan de manera preferente.

| SISTEMAS               | FRECUENCIA | %    |
|------------------------|------------|------|
| Total System           | 7          | 3%   |
| Visual Basic           | 9          | 4%   |
| Informix               | 8          | 3%   |
| COI, NOI, SAE          | 64         | 25%  |
| PISA                   | 5          | 2%   |
| Otras (incluye Office) | 161        | 63%  |
| TOTAL                  | 254        | 100% |

Cuadro 1. Tipo de Software utilizado.

Del cuadro anterior se observa el predominio de paquetes de tipo administrativo, principalmente de los más populares, como serían los productos de ASPEL, COI, NOI y SAE; en el renglón de otros, las empresas encuestadas mencionaron diversos paquetes poco conocidos, pero con una característica predominante: casi todos dirigidos a las áreas administrativas y muy pocos a las de producción.

En cuanto a las áreas de empleo del software (figura 1), predominan con un 28% las administrativas (refiriéndose principalmente en empresas pequeñas a las áreas de nómina, contabilidad, y almacén), con algún tipo de aplicación funcionando; con un 22% el área de Finanzas (incluyendo en las empresas medianas funciones tales como contabilidad, presupuestos, tesorería, manejo de bancos, cuentas por cobrar y pagar); con un 15% las áreas de ventas y el manejo de facturación; con un 12% las áreas de producción, que consisten más bien en desarrollos de prototipos con programas especializados, o con suites de trabajo tales como Office o Lotus Smart; con un 9%

aparecen las áreas de Recursos Humanos, que en la empresa mediana corresponden a las funciones de nómina principalmente, con poca incidencia en sistemas de reclutamiento, selección y capacitación de personal; un 8% hizo referencia al área de Sistemas, pero realmente trataron de describir al trabajo centralizado que en algunas empresas se realiza en áreas de este tipo, donde se procesan todos los sistemas de información o aplicaciones que existan; con un 6% aparecen las áreas de mercadotecnia, (principalmente en las empresas medianas), en donde pudiera existir este concepto aplicado a la función ventas y logística de ventas, con algún tipo de aplicación funcionando.

En esta estadística también se puede apreciar el desconocimiento que existe de la jerga organizacional por la duplicidad o variedad de interpretación en la nomenclatura de algunas áreas; es decir, algunos le llaman ventas y otros mercadotecnia, recursos humanos y/o nómina, manufactura, producción, etc. Esta dualidad existe y sin duda crea confusión.

## AREAS EN LAS QUE SE UTILIZAN LOS SISTEMAS DE INFORMACION

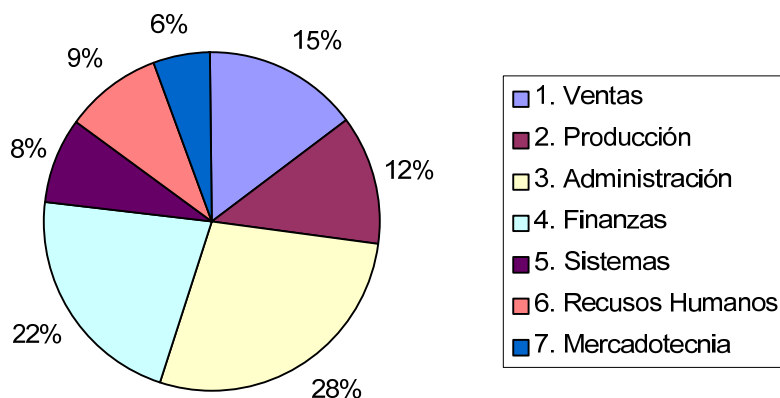


Figura 1. Distribución de áreas en que se utiliza software.

## EL PROBLEMA DE LA ADOPCIÓN

Este es uno de los principales problemas de las PyME, ya que al comprar una tecnología de información tienen problemas severos para instalarla o mejor dicho, para que funcione y sea operada por el personal encargado de ello. De la estadística correspondiente, se desprende que la principal causa que presentan las empresas encuestadas es un desconocimiento total de la Teoría de Difusión de Innovaciones y la Adopción de Tecnologías de Información, lo que provoca un gran choque de decisión entre ocupar las TI o no hacerlo. Para clarificar más esta idea, y a manera de ejemplo, en las PyME es muy común que al comprar una PC y un paquete para el proceso de la nómina del negocio, en los primeros ejercicios de cálculo y proceso existan retrasos sensibles en la emisión de pagos, lo que lleva a que algunos empresarios opinen que "es más rápido hacerlo a mano" o de manera tradicional según sea el caso; lo mismo sucede para las áreas de contabilidad, compras o almacén en empresas pequeñas, medianas e incluso las grandes. Este fenómeno frustra y desalienta el uso de las TI en cualquier organización, sin importar su giro o tamaño.

## ADOPCIÓN EN FORMA DE "S"

La teoría de la Difusión y Adopción postula que toda tecnología que se adquiere para ser utilizada por usuarios finales forma una "S", cuya gráfica puede generarse sobre la variable tiempo cuando se estudia su comportamiento, ya que se dice que su adopción por toda una organización empieza por pocos individuos y después se difunde de manera natural entre el resto de la población.

Este manejo del tiempo es lo que los empresarios no comprenden, ya



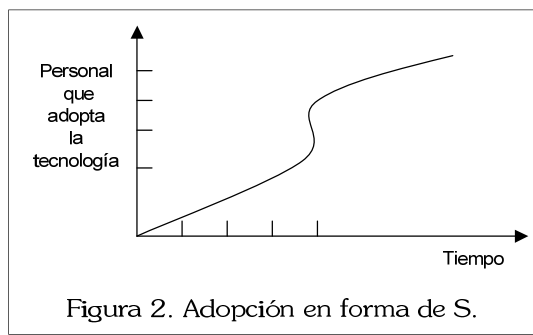


Figura 2. Adopción en forma de S.

que tienen una idea equivocada; tienen por entendido que si hoy compran la tecnología y se da un curso de capacitación para su uso, de inmediato deben darse los resultados de optimización y ahorro de recursos; sin embargo, y como ya se mencionó, esto no es así, realmente nunca ocurre porque el proceso de adopción es muy diferente. El comportamiento de este fenómeno se muestra en la figura 2.

La interpretación de la gráfica es la siguiente:

- 1) Al principio pocas personas adoptan una tecnología o idea nueva.
- 2) Después los pioneros demuestran que si funciona, y más personas desean intentarlo.
- 3) La tecnología se disemina en toda la organización y una mayoría temprana y tardía la utilizará.
- 4) Por último aparece una minoría tardía o laggards, que son los que se oponen hasta el último momento, por diversas causas, a los cambios de tecnología.

Los procesos de adopción en una organización, sobre todo cuando compran una TI, deben estar apoyados por toda una estrategia, la cual debe ser más individual que grupal, cubriendo los siguientes puntos:

- a) Dar a conocer la adquisición de la tecnología.
- b) Instrumentar actividades de persuasión y capacitación previas a su implantación.

c) Fijar, en base al éxito de las dos acciones anteriores, una fecha de arranque de operaciones para su uso y explotación.

d) Iniciar el uso de la tecnología, tomando como referencia una estrategia de implantación que pudiera ser paralela o que sustituya a los métodos originales.

Un punto muy importante al adoptar una tecnología es conocer las características del usuario final, o bien de los componentes o integrantes de las plantillas de personal de las áreas que ocuparan las tecnologías; algunas de estas características aparecen en los cuadros 2,3 y 4.

Estos cuadros sirven para mostrar a las empresas y negocios que la

## LAGGARDS

| CARACTERÍSTICAS                                       | IMPLICACIONES                                                                         |
|-------------------------------------------------------|---------------------------------------------------------------------------------------|
| 10 años mayor en promedio a la edad media             | Tradicionalista                                                                       |
| Gerente medio, con estudios de licenciatura o técnico | No trabajan con incertidumbre, y siempre sospechan de los beneficios de la tecnología |
| Leen muy poco                                         |                                                                                       |
| Poca participación fuera de la organización           | No se enteran mucho de los avances en el medio                                        |
| No son líderes de opinión                             | Son muy eficientes en su trabajo                                                      |

Cuadro 4. Características de los laggards en el uso de tecnología

compra, instalación y explotación de una TI está en función de muchos factores de personalidad humana, los cuales en muchos casos ayudan, pero normalmente estorban para la adopción de nuevas tecnologías; por ello se sugiere a los líderes o gerentes que realizan este tipo de trabajos el tener plenamente identificadas estas características de su personal.

## PIONEROS

| CARACTERÍSTICAS                                                                | IMPLICACIONES                           |
|--------------------------------------------------------------------------------|-----------------------------------------|
| Edad promedio (18-30 años)                                                     | Aventureros                             |
| Licenciatura o Posgrado                                                        | Tolerancia alta para la incertidumbre   |
| Leen mucho                                                                     |                                         |
| Activos profesionalmente, fuera de la organización                             | Muchas relaciones externas              |
| Platican frecuentemente con la gente de sistemas, ya que se interesan en la TI | Desean probar muchas tecnologías nuevas |

Cuadro 2. Características generales de los pioneros de tecnología

El caso de las PyME en Iztacalco no es la excepción a estas situaciones; cierto es que los equipos con los que cuentan se usan, pero tuvieron diversos problemas reales de adopción, y lo mismo sucedió con el software que adquirieron. También es verdad que muchas de ellas con el tiempo han resuelto sus problemas iniciales y de aceptación, y muchos ya "olvidaron estas situaciones"; los cuadros 5 al 8 muestran en porcentaje esta problemática. Estos cuadros, extraídos del Informe final del proyecto de investigación referido, muestran las dificultades que han enfrentado estas PyME para adoptar alguna tecnología de información, al mismo tiempo que se refuerza la idea del rezago que presenta el sector en esta materia.

## ADOPCIÓN TEMPRANA Y TARDIA

| CARACTERÍSTICAS                                                                         | IMPLICACIONES                                               |
|-----------------------------------------------------------------------------------------|-------------------------------------------------------------|
| Edad promedio (18-30 años)                                                              | Reducen la incertidumbre                                    |
| Licenciatura                                                                            | Solo hasta que ven que la tecnología sirve, la adoptan      |
| Hábitos comunes de lectura                                                              | Leen muchos libros de tipo novelas y de superación personal |
| Menos activos fuera de la organización                                                  | Logran muchos contactos internos, son líderes de opinión    |
| Hablan muy pocas veces con la gente de sistemas, no tienen confianza o les da vergüenza | Están poco interesados en aprender tecnología               |

Cuadro 3. Características de los que adoptan tecnología después de los pioneros

| TIPO DE EQUIPO                                                    | % de ocurrencia |
|-------------------------------------------------------------------|-----------------|
| PC 386                                                            | 24              |
| PC 486                                                            | 11              |
| Impresoras                                                        | 11              |
| Otros (tarjetas, cables, digitalizadores, servidores de red, etc) | 36              |

Cuadro 5. Tipos de equipos que no pudieron utilizar en su momento.

| CLAVE DE NO USO                        | % de ocurrencia |
|----------------------------------------|-----------------|
| 1. No se supo instalar                 | 16              |
| 2. Por obsolescencia                   | 18              |
| 3. No existen manuales                 | 13              |
| 4. No hay servicio de proveedor        | 5               |
| 5. No hay personal capacitado          | 3               |
| 6. Es cara la capacitación para usarla | 8               |
| 7. No hay interés por la gerencia      | 19              |
| 8. Otras                               | 14              |

Cuadro 6. Causas principales por las que no se utilizó el equipo indicado en el cuadro 5.

| TIPO DE SOFTWARE                                                                                                   | % de ocurrencia |
|--------------------------------------------------------------------------------------------------------------------|-----------------|
| Lotus                                                                                                              | 5               |
| D base                                                                                                             | 15              |
| Word Perfect                                                                                                       | 2               |
| Otros (Lotus SmartSuite, Correos, Lenguajes de programación, Administradores de bases de datos, graficadores, etc) | 30              |

Cuadro 7. Software que no se supo instalar en su momento.

| CLAVE DE NO USO                         | % de ocurrencia |
|-----------------------------------------|-----------------|
| Se compro sin criterio de adquisiciones | 10              |
| Es muy difícil de manejarlo             | 11              |
| No hay personal que sepa manejarlo      | 19              |
| Otras                                   | 29              |

Cuadro 8. Causas del no uso o de dificultades de instalación, del software mencionado en el cuadro 7.

## CONCLUSIONES

Las TI son sin duda una gran herramienta para cualquier empresa si se quieren crear ventajas competitivas, para transformar paradigmas sobre cómo una empresa compite, o bien para innovar los procesos de una organización. Sin embargo, estos son beneficios que se obtienen después de recorrer todo un proceso de automatización en el registro y uso de la información; de hecho, las PyME en Iztacalco están en este proceso, y de algunos años a la fecha han intentado adoptar tecnologías de información, recurriendo al método de ensayo y error en su instalación y explotación.

Sin duda se requieren acciones específicas que permitan a estas PyME avanzar en dicho proceso; entre ellas destacan contratar personal de mayor perfil de competencia en tecnologías de información, para minimizar los costos de adopción y arranque, o bien contratar servicios de asesoría especializada, lo que representa un nicho de oportunidades en este sector industrial y comercial.

El problema de la adopción de tecnología es un tema menospreciado por los empresarios, y aún por algunos especialistas del área; sin embargo, si no se considera dentro de los proyectos informáticos, se seguirá sufriendo de grandes problemas de instalación, arranque y uso, lo que provocará costos elevados de adquisición y operación, además de una tendencia a la obsolescencia en detrimento de la competitividad de las empresas.



# Una Perspectiva de las Bases de Datos Orientadas a Objetos con Respecto a las Relacionales

LAI. Eunice Echeverría Portillo.  
Profesora del CIDETEC -IPN.  
M. en C. Ivonne Rivera Santos.  
LAI. María del Carmen Celis Ortega.  
Profesoras del CIC-IPN.

**L**os actuales manejadores de Sistemas de Bases de Datos son inadecuados para el manejo de información inherentemente compleja, lo que ha generado la necesidad de nuevos tipos de sistemas, en respuesta al nuevo rango de requerimientos de las estructuras de información complejas, tales como: integración de código y datos, multimedia, versiones de datos, datos persistentes, etc. Las Bases de Datos Orientadas a Objetos (BDOO) ofrecen una arquitectura que soporta estas estructuras; aunque se encuentran actualmente en desarrollo, lo que representa algunas limitantes.

## INTRODUCCIÓN

Para adentrarse en el tema es conveniente analizar la definición de Sistema Manejador de Bases de Datos (SMBD), propuesta por Richard Cooper:

Es un software que debe permitir el manejo de grandes cantidades de datos estructurados, es decir, que cuenta con una forma previamente establecida por algún modo-

Por otro lado, Kim W. refiere que una BDOO es una colección de objetos en los que su estado, comportamiento y relaciones se definen de acuerdo a un modelo de datos.

Ambos conceptos indican que los datos a manipular tienen una composición lógica, previamente diseñada, y que existen relaciones entre ellos.

Un SMBD no es un simple repositorio de datos, dado que posee una arquitectura, un modelo de datos y un conjunto de lenguajes especiales para definir, manipular y consultar los datos almacenados (ver figura 1); la

finalidad del almacenamiento, es recuperar información valiosa, del contenido o de la infraestructura. El valor de un sistema de información radica en la potencia de sus consultas y el manejo de sus datos.

Con el rápido avance de la tecnología en computación, la definición anterior ha extendido su semántica a otros tipos de sistemas de información, de donde se derivan los siguientes conceptos:

**Sistema Manejador de Bases de Datos (SMBD):** Paquete de Software que permite manejar gran-

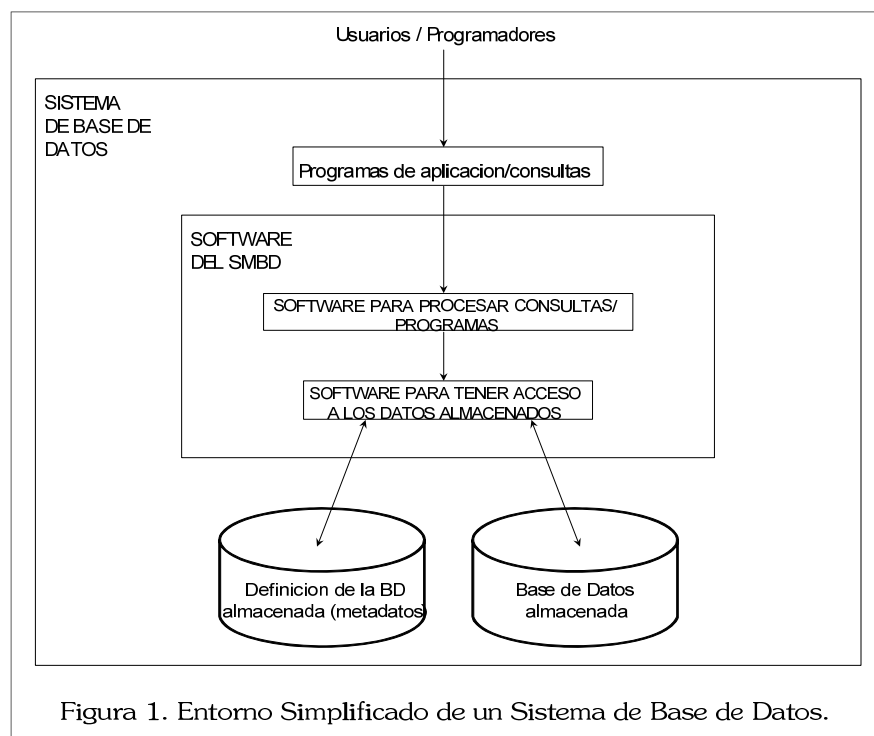


Figura 1. Entorno Simplificado de un Sistema de Base de Datos.

des cantidades de datos estructurados.

**Sistema de Objetos o Sistema basado en Objetos (SO):** Sistema que soporta el modelado de datos como entidades abstractas con identidad.

**Sistema Orientado a Objetos (SOO):** Sistema de objetos donde todos los datos se crean como instancias de clase.

**Sistema Manejador de Bases de Datos Orientadas a Objetos (SMBDOO):** Sistema de base de datos que permite la definición y manipulación de una base de datos orientada a objetos.

**Bases de Datos Orientadas a Objetos (BDOO):** Es una base de datos formada por objetos y manejada por un SMBDOO.

En todas estas definiciones se encuentra un concepto común: el modelo de datos de una aplicación. El Modelo de Datos es la estructura básica con que se construyen todas las entidades en el sistema (especificaciones denominadas esquema). Por ejemplo, el modelo relacional está basado en los conceptos registro, columna, llave, tabla y relación. Todas las construcciones de los sistemas basados en este modelo, emplean esta estructura para modelar un problema.

En el caso del modelo de objetos, son los conceptos objeto, clase, herencia y demás componentes del paradigma, los que estarán presentes en tal modelado.

Los Sistemas de Objetos a los que se refiere la definición anterior incluyen a los Sistemas de Objetos Complejos, Modelos Semánticos de Datos y Lenguajes de Programación Orientados a Objetos (LPOO).

En los SOO la creación de toda entidad se realiza basándose en la instanciación de una clase, dando como resultado un objeto que encapsula estado y comportamiento, en donde dicha clase forma parte de una jerarquía de herencia.

Un SMBD además de definir, manipular y consultar grandes cantidades de datos, debe proporcionar ciertas facilidades identificadas en el cuadro sinóptico de la figura 2, las cuales se describen brevemente a continuación.

## • GENERALIDAD DE APLICACIÓN

Implica que el Sistema debe ser capaz de manejar datos de diferentes áreas de aplicación de igual manera, empleando las mismas técnicas. Es decir, no importa la naturaleza de los datos ni el tipo de sistema de información que los emplea para desarrollar una aplicación en el manejador. Por ejemplo, se diseña el SMBD para que manipule un sistema de nómina de una empresa, y por igual, para manejar un sistema de registro y seguimiento para pacientes de un hospital.

## • ACCESO EFICIENTE A DATOS

Se refiere a que los datos se almacenarán de forma tal que puedan accederse de la manera más rápida posible. Esta velocidad se logra usando estructuras de datos diseñadas especialmente para indexamiento, las cuales aceleran la búsqueda de un dato y su acceso ordenado (familia de los árboles B, B+, B\*). Esto es, búsquedas en un orden de  $O(\lg n)$ .

Por otro lado, se emplean técnicas de almacenamiento de datos basándose en clusters, las cuales consisten en utilizar pequeñas áreas en disco (un conjunto de "x" bloques lógicamente contiguos) donde se almacenan datos relacionados. Esto garantiza que las transferencias de lectura y escritura para datos que se encuentran en la misma área se realicen en bloques, lo que acelera dichas operaciones.

Como ejemplo de ello se tiene que, al leer la relación entre los registros de dos tablas relacionadas, almacenados en una misma área, existe la garantía de que los registros implicados se recupera-

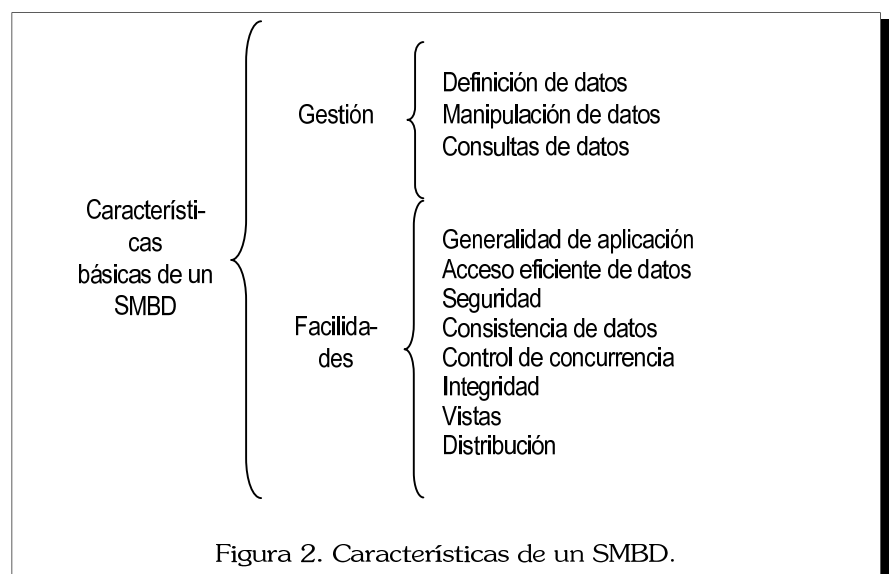


Figura 2. Características de un SMBD.

rán más rápidamente que si estuvieran dispersos en el disco. Por otro lado, el almacenamiento de todos los objetos de una clase en una sola área, permite optimizar los algoritmos de actualización del recolector de basura.

- **SEGURIDAD**

Para un SMBD, la seguridad implica la existencia de mecanismos de privacidad que garanticen la confidencialidad de los datos almacenados. No todos los usuarios deben tener acceso a todos los datos, de ahí que el empleo de dichos mecanismos incluya desde asignar niveles de acceso a una base de datos, permisos de lectura y escritura a tablas, hasta el empleo del principio de encapsulación de los objetos, donde sólo se acceden las propiedades que la interfaz del objeto muestra al exterior.

- **CONSISTENCIA DE DATOS**

Otro concepto importante empleado en los SMBD es la **transacción**, que consiste en uno o más cambios sobre los datos de la base, lo cual la cambia de un estado consistente a otro.

Mediante la consistencia de datos se garantiza que las transacciones a realizar se efectúen de forma tal que, en cualquier momento que se consulte la base de datos existirán datos consistentes. Esto está estrechamente relacionado con la **resistencia o tolerancia a fallas**, ya sea por parte del software o hardware, lo cual implica que, después de la falla, los datos regresarán al último estado consistente anterior a la modificación que provocó dicho fallo.

Uno de los mecanismos para asegurar la consistencia es organizar los accesos a la base de datos en

transacciones, garantizándose la Atomicidad, Consistencia, Independencia y Durabilidad del manejo de transacciones (propiedades ACID), para los datos almacenados.

- **CONTROL DE CONCURRENCIA**

Si el Sistema cuenta con un mecanismo de bloqueo, entonces soporta un ambiente multiusuario; esto es, permite el acceso de más de un usuario a la misma base de datos y garantiza que no habrá interferencias en el procesamiento. Esto conlleva un nivel de granularidad de bloqueo en el SMBD, el cual puede ser a nivel de la base de datos, de tabla, de columna o de registro (Modelo Relacional) o de base de datos extensión u objeto (Modelo de Objetos).

- **INTEGRIDAD**

Existe la facilidad de imponer restricciones para mantener la coherencia de los datos almacenados, así como establecer rangos de dominios para los valores de las columnas de los registros o del estado de un objeto. Dichas restricciones son sentencias que se verifican en cada modificación a los datos almacenados; en el caso de infringirse, generan excepciones o disparan otros procedimientos almacenados.

- **VISTAS**

Son diferentes representaciones de los datos almacenados, las cuales se generan con la finalidad de restringir o seleccionar cierta información que se presentará a los diferentes tipos de usuario. Conocidas también como **interfaces de usuario**, son la cara que presenta el SMBD a los usuarios para el manejo de la aplicación y sus

diferentes herramientas. Estas pueden ser paramétricas, de consulta, gráficas, de programación o administrativas.

- **DISTRIBUCIÓN**

Un SMBD que permite trabajar con bases de datos que se encuentran físicamente particionadas o replicadas en diferentes lugares, las cuales funcionan como una sola base de datos a través de una comunicación vía red.

En resumen, los SMBD poseen estas facilidades a fin de manejar los datos mediante una forma sencilla: el modelo de datos lógico.

Otro tipo de modelo de datos empleado actualmente para trabajar grandes cantidades de información es el Sistema basado en el Manejo de Registros

Estos Sistemas se caracterizan por el uso del registro como entidad básica o estructura principal de datos, el cual está compuesto por campos o columnas que poseen un tipo de dato previamente definido y con naturaleza atómica.

Existen dos tipos principales de SMBD basadas en archivos: de Red y Relacionales.

Los SMBD de Red organizan las colecciones de registros del mismo tipo como listas ligadas, las cuales cuentan con una cabecera que describe la estructura del tipo de registro que mantiene.

Por su parte, en los SMBD Relacionales, donde las colecciones de registros del mismo tipo reciben el nombre de tabla, se maneja el concepto de llave primaria, cuyo fin es el de garantizar una identidad única a cada registro. Las relaciones entre colecciones de registros tienen dife-

rente representación según el tipo del SMBD. Por ejemplo, en los de tipo relacional se emplea la llave foránea y en los de tipo red se agrega un campo más al registro, el cual contiene un apuntador a la otra lista o listas con las que se relaciona.

Estos SMBD manejan enormes cantidades de datos eficientemente, gracias al almacenamiento simple en registros con campos atómicos. Sin embargo, el gran éxito de estos sistemas ha motivado a un amplio rango de nuevos clientes potenciales a buscar alternativas que satisfagan sus necesidades ante el manejo de datos más complejos, ya que en estos manejadores relacionales los datos no pueden transformarse fácilmente dentro de sus estructuras simples.

Hasta hace algunos años, el uso de los Manejadores Basados en Registros había sido suficiente para desarrollar aplicaciones comerciales robustas; no obstante, el desarrollo de nuevas líneas de investigación en software ha generado necesidades que no cubren los manejadores tradicionales. Su empleo en aplicaciones avanzadas ha significado que el diseño y su adaptación se tome extremadamente problemático.

A continuación se presentan brevemente algunas de estas aplicaciones avanzadas:

**Diseño Asistido por Computadora (CAD)**, la Ingeniería Asistida por Computadora (CAE) y la Manufactura Integrada por Computadora (CIM), las cuales implican la creación y el mantenimiento de estructuras complejas que describen objetos técnicos que requieren un trabajo cooperativo, tales como los diagramas.

Igualmente, se encuentra la Ingeniería de Software Asistido por Computadora (CASE), la cual me-

jora el proceso de diseño y mantenimiento de software, proporcionando mecanismos para almacenar, buscar y recuperar componentes de software de forma sistemática, los cuales requieren estar organizados en estructuras complejas.

Estas dos áreas de aplicación son usualmente referidas como aplicaciones de diseño, por lo que comparten características comunes y manejan diseños complejos de información.

Por otra parte, están los Sistemas de Información de Oficina (OIS), los cuales permiten manejar la información tradicional de una oficina (documentos tales como: mensajes, listas, etc.), pero no cuentan con una estructura definida, y requieren de almacenar datos junto a los procesos que los manejan.

Los Sistemas de Información Geográfica (GIS), almacenan información que se deriva de tomas satelitales e inspecciones, con relaciones espaciales y temporales, así como complejas formas de visualización. Juegan un papel importante en los sistemas de transportación, en la organización de herramientas de telecomunicaciones, tierras de cultivo e irrigación.

El World Wide Web (WWW) maneja sistemas hipermedia con datos de texto integrado, numéricos, gráficos y sonido, empleando funciones que permiten recorrer tales estructuras. Utiliza técnicas de compresión para tipos grandes de datos. Sin embargo, la WWW se ha desarrollado con poca tecnología de bases de datos.

Las características de los SMBD relacionales aplicados con gran éxito en los negocios, dieron pie a que se les considerara con similar importancia en aplicaciones avanzadas o complejas; sin embargo, algunas de estas

características necesitaban una transformación radical, mientras que otras realmente eran requerimientos completamente nuevos. No se considera al modelo relacional como la arquitectura idónea para el desarrollo de las nuevas áreas, debido a que presenta algunas carencias y desventajas para sus nuevas necesidades. A continuación se describen aquellas características más relevantes que requirieron de alguna modificación:

## • TRANSACCIONES CORTAS

Los manejadores relacionales están diseñados para trabajar con transacciones cortas y no soportan las de largo tiempo. Una transacción larga, como en las aplicaciones avanzadas, puede durar el tiempo necesario para actualizar una pieza o diseñar un componente de software (horas, días). El aislamiento es un aspecto vital de la transacción desde el momento en que se compite por los recursos. El Aislamiento y el Control de concurrencia de una transacción tradicional (ACID) son características válidas en una transacción de corto tiempo, pero no en una transacción de periodo largo. Las transacciones largas requieren un trabajo colaborativo y cooperativo más que competitivo. Cuando una transacción entra en ejecución y requiere de un largo tiempo en espera para finalizar, se cede la ejecución a otras transacciones. Entonces se requiere de una estrategia que permita envolver nuevos tipos y compartir el acceso.

## • ESTRUCTURAS SIMPLES

Para almacenar una estructura compleja en el modelo relacional es necesaria su conversión a una estructura simple (uso de las formas normales), donde sus campos están atomizados y el rango de valores para los tipos que se ma-

nejan utiliza texto y formato numérico. Por igual, la identificación única del registro se basa en el valor de uno o más campos del mismo. Los desarrollos para las nuevas áreas, por el contrario, requieren el uso de estructuras que no restrinjan su representación a simple texto o números, o por la atomicidad de sus campos, donde la identidad de las entidades que manejan no dependa de la información que contienen.

Las aplicaciones avanzadas requieren facilidades para trabajar con diferentes versiones de los datos y el manejo de restricciones no sólo en la descripción estructural, lo cual tendría que ser añadido a un manejador convencional.

- **INFORMACIÓN ADICIONAL**

Cuando un manejador relacional crea relaciones entre entidades, se agrega una gran cantidad de datos a la tabla, complicando su manejo. No existe una representación adecuada de la identidad de un registro, debido a que se basa en el valor de alguno de sus campos originales o en uno que se creó para dicho fin, lo cual se hace visible en el esquema.

- **SOBRECARGA SEMÁNTICA**

Esto se debe a que su estructura básica (tabla) tiene más de un sentido de creación. Una tabla sirve tanto para agrupar una colección de registros del mismo tipo como para crear una relación entre tablas diferentes. Las relaciones entre dos registros son el resultado de modelar las propiedades multivaluadas de una entidad o la unión de los componentes de un valor complejo. La razón por la cual los valores de dos columnas se encuentran relacionados, se debe a que ambas

columnas pertenecen al mismo registro o bien están de registros diferentes conectados por algún tipo de relación.

Este sobreuso de las estructuras básicas con más de un propósito, hace imposible descifrar la naturaleza de una aplicación con sólo observar su esquema relacional.

- **IMPEDANCIA DE CORRESPONDENCIA**

Esta se presenta cuando se requiere extraer datos del manejador para realizar cálculos complejos mediante un lenguaje de programación; su empleo se debe a la limitada cantidad de operaciones que el manejador puede realizar sobre los datos.

Por lo general, el uso de un lenguaje orientado a bases de datos (tal como SQL), sirve para recuperar los datos del manejador, procesarlos con las instrucciones del lenguaje de programación y que finalmente se regresen modificados al manejador o los resultados de las operaciones. Sin embargo, la transferencia de datos entre ambas aplicaciones (Base de Datos - Lenguaje de Programación) no siempre se realiza de forma directa, ¿cuál es la causa? Simplemente que ambas aplicaciones están basadas en un modelo de datos diferente y los tipos que maneja cada aplicación no siempre serán equivalentes o compatibles; así, se presenta una resistencia a la navegación directa de los datos de una aplicación a otra, dado que existen tipos desconocidos o con diferente representación.

Al suceder lo anterior, es necesario un mecanismo de conversión para tipos diferentes de datos. La estrategia para solucionar este problema sería unificar los tipos de datos empleados en ambas apli-

caciones, es decir, el modelo de datos de la base y los tipos en el lenguaje de programación, permitiendo de esta forma que sean persistentes y realizar cálculos complejos sobre ellos.

- **ALMACENAMIENTO DE CÓDIGO**

Algunas veces, la complejidad y naturaleza dinámica de un dato puede ser tan grande, que podría ser preferible su representación mediante código; sin embargo, los manejadores relacionales (SMBDR) no pueden almacenar código.

A continuación se presentan tres posibles formas de solucionar esta carencia en los SMBDR:

- ? Agregar una Extensión a la estructura de la base de datos, incluyendo con esto formas de integrar el código y los datos empleados.
- Adicionar una Estructura de Almacenamiento de Aplicaciones al SMBD, a fin de que el código de la aplicación se encuentre dentro del mismo, pero separado de los datos. Por lo regular, ésta es la técnica empleada cuando se actualiza un SMBD prediseñado.
- Considerar al código como Otro Tipo de Datos, el cual pueda reconocer el modelo de datos del SMBD y almacenarlo tal cual.

El almacenamiento de código y datos sin algún tipo de relación, genera falta de integridad e inconsistencia, lo cual resulta de las alteraciones en alguna de las dos partes sin el conocimiento de la otra. Por tal motivo, en este tipo de aplicaciones se emplean extensivamente las restricciones de integridad, y en virtud de que la tarea de modelado de aplicaciones es más compleja, se requiere producir interfaces de usuario más amigables para

aquellos usuarios no especialistas en el tema.

De acuerdo a las necesidades que presentan las nuevas áreas de desarrollo surgen los SMBD avanzados, en donde destacan los siguientes:

- 1.-SMBD Históricos: Se caracterizan por proporcionar facilidades para conservar todos los estados pasados de una base de datos. Cada vez que se actualiza o elimina un dato, el valor anterior no se remueve de la base, sino que se marca con una estampa de tiempo, la cual presenta información tal como la fecha de creación y destrucción de ese dato, permitiendo así recuperar todas las entidades que han estado presentes en algún momento.

Este almacenamiento de datos se hace con el fin de soportar la coexistencia de varias versiones de un dato, aunque sólo es válida una versión a la vez. Como ejemplo de este tipo de sistemas se encuentra Chronolog, que trabaja con Oracle.

- 2.-SMBD de Objetos Complejos: Conocidos también como relacionales sin primera forma normal, estos sistemas son relacionales cuyo valor en un campo de la tabla puede ser un registro completo; como ejemplo de estos está PostgreSQL.

- 3.-SMBD Deductivos: Estos poseen la capacidad de definir reglas con las cuales deducir e inferir información adicional a partir de datos almacenados en la base. Centran su importancia en los datos derivados y las restricciones, así como en proporcionar mecanismos para describir los datos en términos de estas últimas. La mayoría de estos sistemas se desarrollan como formas persistentes del lenguaje Pro-

log/Datalog, el cual permite la descripción de los datos como simples hechos y reglas para generar otros hechos. Existe una amplia variedad de desarrollos tanto académicos como comerciales de estos sistemas; entre los primeros destaca el proyecto CORAL de la Universidad de Wisconsin.

- 4.-Lenguajes de Programación Persistentes: La persistencia de los datos es una de las características más relevantes en un SMBD. Dicha persistencia es la cualidad de algunos objetos de mantener su identidad y relaciones con otros objetos, con independencia del sistema o proceso que los creó. En la práctica, se implementa proporcionando a los objetos un mecanismo cuyo objetivo es tanto el almacenamiento en memoria como la recuperación posterior de éstos. Así, el papel que juegan los lenguajes de programación persistente consiste en eliminar el problema de impedancia de correspondencia, entre los datos persistentes y los lenguajes de programación. Este tipo de lenguajes parte de un lenguaje de programación normal y lo extiende con capacidades de base de datos, y se basan en dos principios:

? Los valores de cualquier tipo de datos del lenguaje de programación se pueden almacenar y recuperar posteriormente (persistentes), incluyendo valores procedurales y multimedia.

? Para almacenar los datos no es necesario cambiar su estructura física.

Como principales ejemplos se encuentran Napier88 y el proyecto Forest.

- 5.-SMBD Activos: Implican que el manejador contiene algún componente de comportamiento, usualmente del tipo de restricciones; por ejemplo, actualización de la base de datos o una salida de esta por parte del usuario. Frecuentemente esto es una actividad almacenada que puede iniciarse por un cambio en la base de datos, o por un evento detectado automáticamente, como podría ser el transcurso de un lapso de tiempo. En este tipo de desarrollos sobresale el proyecto IDEA.

- 6.-Sistemas de Dominio Específico: Son una respuesta a las necesidades de áreas específicas de investigación, tales como las Bases de Datos Espaciales (sistemas de información geográfica), y las Bases de Datos Temporales. Estos tipos de SMBD intentan proporcionar nuevas estructuras dentro del modelo de datos, así como facilidades para emplearlas.

La mayoría de las investigaciones realizadas en estas áreas, han dado la pauta para el empleo del paradigma orientado a objetos, esperando que en un futuro cercano emerjan componentes espaciales y temporales mediante los Sistemas Manejadores de Bases de Datos Orientadas a Objetos (SMBDOO).

Un SMBDOO soporta la creación y manipulación directa de los objetos desde un Lenguaje de Programación Orientado a Objetos (LPOO). Este tipo de manejador incluye toda la funcionalidad de un SMBD más las construcciones para la descripción de datos que le proporciona el paradigma de la Orientación a Objetos, como son: Identidad de objetos, Clasificación, Encapsulación, Herencia, Redefinición y Sobrecarga (figura 3). A continuación se explica brevemente cada uno de ellos:

## IDENTIDAD DE OBJETOS

Es el principio que establece que cada entidad del mundo real será representada por una estructura indivisible llamada objeto, el cual posee una identidad que es independiente de los valores que almacenan sus atributos. La identidad del objeto lo distingue de las demás entidades creadas por la aplicación, y es inherente al objeto por el resto de su vida. Existen varias formas de asignar el OID (Object Identifier) al objeto, siendo obligatorio que la aplicación lo establezca en el momento de la creación y se emplee exclusivamente por el sistema, sin que el desarrollador pueda modificarlo.

## CLASIFICACIÓN

Los SMBDOO soportan tipos de datos definidos por el usuario, de tal forma que los desarrolladores pueden crear el tipo de dato que requieran, empleando el principio de abstracción y modelándolo como un Tipo de Dato Abstracto (ADT), considerando a este como sinónimo de Clase o Tipo. Una vez que se define una clase, es posible hacer tantas instanciaciones de ésta como se requiera (crear objetos de dicha clase).

## ENCAPSULACIÓN

Cada clase tiene definido un conjunto de atributos que representan su estado, y un conjunto de métodos (propiedades) con los que se manipulan los objetos de la clase, es decir, desarrollan un comportamiento. La descripción de la clase se encuentra dentro de una interfaz, la cual permite la interacción con el mundo exterior; en ella se enumeran los atributos y propiedades con sus respectivas firmas. Esta interfaz representa la división entre la abstracción de la clase y su

implementación, comúnmente denominada encapsulación.

Mediante la encapsulación, algunos atributos y propiedades de la clase se toman inaccesibles (poseen modificadores de acceso) para otras clases dentro de la aplicación. El establecimiento de estos modificadores tiene políticas propias para cada lenguaje de programación.

## HERENCIA

Las clases definidas toman parte en una Jerarquía de Herencia, de tal forma que se puede definir una clase nueva a partir de una existente, la cual heredará los atributos y propiedades de su antecesora, siendo posible adicionar comportamiento y/o atributos a la clase recién creada.

## REDEFINICIÓN

Al utilizar el mecanismo de herencia para definir nuevas clases, es posible que se desee redefinir la funcionalidad de algún comportamiento heredado, proporcionando una nueva implementación a la

firma de dicho método, con lo que se oculta o se Redefine la operación original. Así, dependiendo del objeto que invoque este comportamiento, se ejecutará el código indicado.

## SOBRECARGA

Otro uso del polimorfismo consiste en declarar una nueva funcionalidad bajo el nombre de un método existente, pero con diferencias en la firma; a esto se le denomina sobrecarga.

## REDES DE OBJETOS

Este tipo de manejadores también soporta el almacenamiento y navegación de estructuras tipo red, las cuales se constituyen cuando los objetos mantienen referencias a otros objetos (posiblemente creados en diferentes sesiones de la aplicación) estableciendo relaciones entre ellos. Cada objeto en la red debe tener una referencia a los identificadores de sus objetos vecinos. El LPOO en que se basa el manejador debe soportar el acceso a los atributos de un objeto, así como a sus métodos, a través de

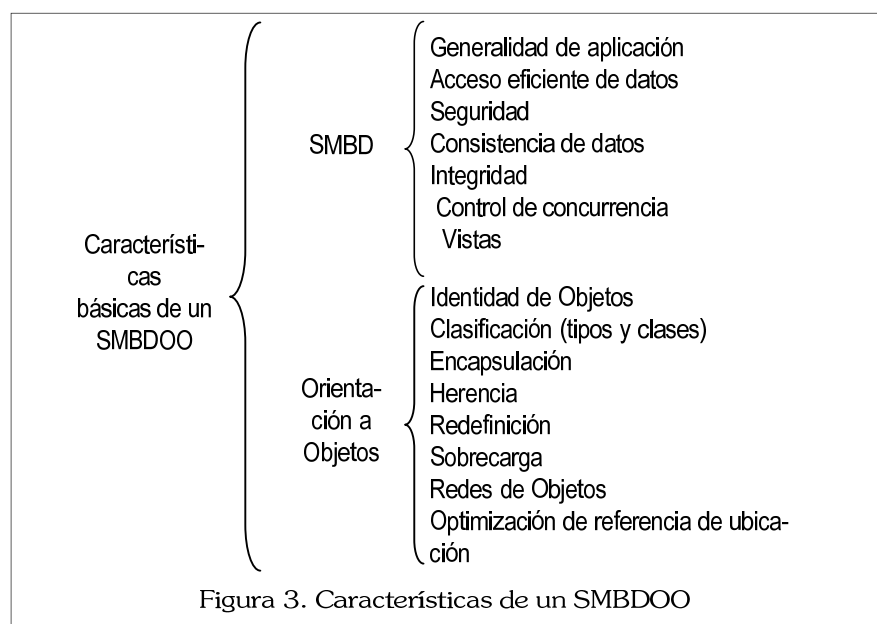


Figura 3. Características de un SMBDOO

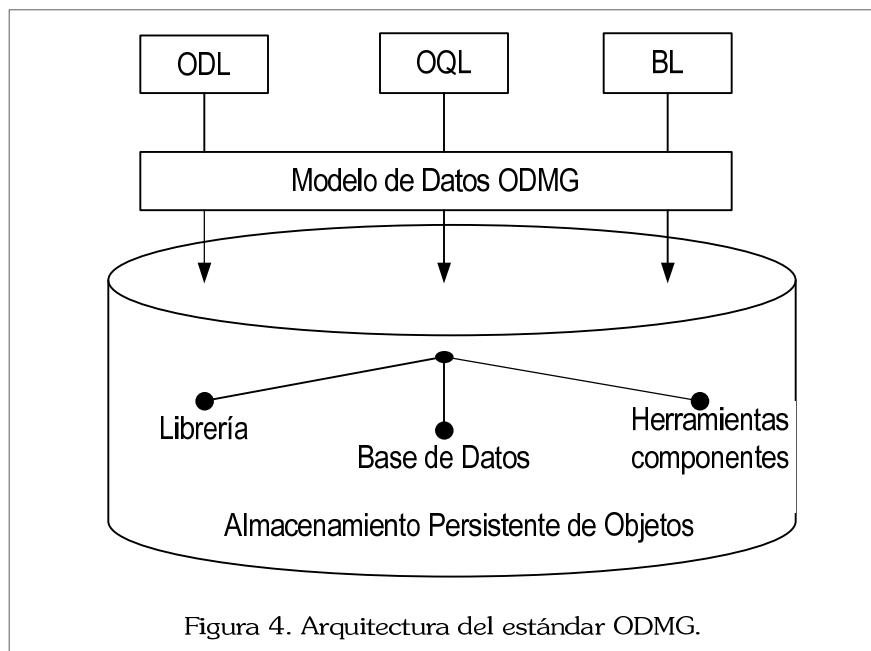


Figura 4. Arquitectura del estándar ODMG.

estas referencias. Comúnmente este soporte incluye el manejo de asociaciones con diferente cardinalidad (uno a uno, uno a muchos y muchos a muchos) basadas en los OID de los objetos implicados.

#### UBICACIÓN

Esta característica se refiere a la mejora en el desempeño de las tareas de almacenamiento y recuperación de objetos, conocida como optimización, la cual aprovecha las estructuras formadas de red. Esta optimización se desarrolla mediante tres técnicas, la de generación de clusters en disco, para almacenar objetos relacionados, la técnica de bloqueo (bloques de objetos), en cuanto al manejo en memoria, y la técnica del manejo de cachés de objetos, basada en las relaciones en que éstos se encuentran implicados.

En el campo de los SBDOO comerciales existen varias diferencias, que van desde la definición del concepto de Base de Datos Orientada a

Objetos, hasta las características que ésta debe poseer. Ante esto, surge el Object Data Management Group (ODMG), organización de empresas desarrolladoras de SMBDOO, dedicada al establecimiento de estándares para BDOO.

Dado que los desarrollos comerciales propietarios no soportaban la portabilidad entre plataformas, se planteó el establecimiento de un estándar que creará un ambiente donde los SMBDOO compartiesen muchas características, similar a las interfaces SQL de los manejadores relacionales. Esta organización se ideó basándose en grupos de trabajo para cada capítulo del estándar a desarrollar, y las especificaciones adoptadas en cada grupo de trabajo debían pasar por un consenso final.

Algunos de los miembros de ODMG son: Sun Microsystems, Excellon Corp., Winward Solutions, Barry & Associates, American Management Systems, Wichita State University, in Line Software, Ardent Software, NEC, Hitachi, Lucent Technologies, Unidata, POET Software, Loc-

kheed Martin, UniSQL, IBEX ObjectSystems, Object Design, GemStone Systems, Versant Object Technology, y MATISSE Software.

El primer resultado de este comité fue el estándar ODMG-93, cuya segunda versión se liberó en 1997, y la tercera versión, 3.0, se publicó en enero del 2000.

---

#### ARQUITECTURA DE UN SMBDOO

---

El estándar ODMG versión 3.0 establece una arquitectura, proporcionando algo más que un lenguaje de alto nivel como SQL para la definición, manipulación restringida y consulta de datos. Esta arquitectura en particular establece que un SMBDOO integra transparentemente las capacidades de una base de datos con las de un lenguaje de programación, dentro de un ambiente portable (figura 4). Esto es, la definición de datos está implícita en la construcción de la aplicación orientada a objetos, y la manipulación de datos se realiza a través de las operaciones del tipo.

El estándar coloca como componentes principales de esta arquitectura los siguientes: especificación de un repositorio de objetos, el modelo de objetos (Object Model) y los lenguajes de bases de datos tales como: lenguaje de especificación o definición de objetos (Object Definition Language, ODL), lenguaje de consulta (Object Query Language, OQL) y lenguaje de enlace o ligado (Language Binding).

A continuación se describen brevemente el modelo de objetos y los lenguajes de bases de datos.

#### MODELO DE OBJETOS (OM)

Es el componente central de la arquitectura, en virtud de que es la



jerarquía de clases en la que se basa el desarrollo de los esquemas de las aplicaciones, y la estructura de organización con la cual se almacena toda la información. ODMG soporta un Modelo de Objetos unificado para compartir datos entre diferentes LPOO. Todos los datos que maneja el SMBDOO están estructurados en términos de las construcciones del Modelo de Datos.

### LENGUAJE DE ESPECIFICACIÓN O DEFINICIÓN DE OBJETOS (ODL)

El programador debe especificar los esquemas de la aplicación y de la implementación de las entidades definidas. Los esquemas pueden desarrollarse con una extensión de la sintaxis del Lenguaje de Ligado con el que se esté trabajando (Language Binding), o con un ODL independiente del Lenguaje de Ligado, por ejemplo UML. Los esquemas se describen en términos del ODL, como un conjunto de interfaces de objetos, es decir, las propiedades y las relaciones de los tipos junto con las firmas de sus métodos. El Lenguaje de Especificación no es completamente un lenguaje de programación, dado que las implementaciones de los tipos definidos deben realizarse con el Lenguaje de Ligado.

Por lo tanto, se dice que el ODL es un lenguaje virtual, el cual no está implementado obligatoriamente en los productos basados en el estándar ODMG; en su lugar, es posible que tengan lenguajes de definición equivalentes, los que poseen la funcionalidad que ofrece un ODL, pero desarrollados a la medida de las especificaciones de un SMBDOO en particular.

### LENGUAJE DE LIGADO (BINDING)

La implementación de los métodos definidos en los esquemas y

las aplicaciones se lleva a cabo con un LPOO. Sin embargo, estos Lenguajes de Programación no son persistentes por naturaleza, así que es necesario extenderlos para que puedan interactuar con datos persistentes. Es entonces cuando se interactúa con el Lenguaje de Ligado, que es el puente de integración entre un LPOO y el Modelo de Objetos de la arquitectura del SMBDOO.

El tipo de extensiones que permiten que un Lenguaje de Programación Orientado a Objetos trabaje con Objetos Persistentes son construcciones adicionales al lenguaje o bibliotecas, las cuales proporcionan el Lenguaje de Ligado específico para ese lenguaje de programación. El SMBDOO puede proporcionar más de un Lenguaje de Ligado con el fin de usar diferentes LPOO como Lenguajes de Manipulación de sus datos. El estándar ODMG 3.0, establece el Lenguaje de Ligado para Smalltalk, C++ y Java. Un SMBDOO sólo puede trabajar con un Lenguaje de Ligado a la vez; dicho lenguaje está diseñado a la medida de un LPOO específico, con el fin de proporcionar un ambiente de programación único e integrado para la manipulación de datos. Los esquemas y la implementación se compilan y ligan junto al SMBDOO para producir la aplicación ejecutable. Esta aplicación accede a una base de datos, los cuales corresponden a los tipos declarados en los esquemas, y los manipula mediante el paso de mensajes a los métodos definidos e implementados para cada clase.

El acceso a las bases de datos puede ser compartido entre varias aplicaciones, debido a que el SMBDOO proporciona un servicio concurrente con base en transacciones y manejo de bloqueos.

### LENGUAJE DE CONSULTA DE OBJETOS (OQL)

OQL es el Lenguaje de Consulta que forma parte del estándar ODMG. Es similar al SQL en el sentido de que puede utilizarse como un Lenguaje de Consulta por sí solo o puede estar integrado en un LPOO, pero a diferencia de SQL, OQL sólo contiene comandos de consulta y carece de comandos para definición o manipulación de datos. Este lenguaje es de uso común en los diferentes lenguajes de programación con los que puede trabajar un SMBDOO.

A pesar del potencial de los SMBDOO ante las aplicaciones complejas o nuevas, aún presentan varios problemas, los cuales abarcan diversas líneas de investigación:

### LA MIGRACIÓN DE DATOS

Un gran número de empresas en el mundo mantienen actualmente cantidades considerables invertidas en Sistemas de Bases de Datos, Sistemas Expertos y Sistemas de Cómputo, principalmente en Datos Heredados (Data Legacy), siendo éstos últimos los datos sobre los que corren las aplicaciones ya mencionadas; esto es, los datos históricos de la empresa que han sido almacenados y procesados en el transcurso de los años y donde reside el valor de una institución: su información. Por lo tanto, es natural que a pesar de que la compañía decida migrar hacia una nueva plataforma, será necesario tomar en cuenta los costos de transferencia o compatibilidad de tal plataforma con los datos existentes; esta problemática generó la línea de investigación denominada Migración de Datos. De entre los resultados hasta ahora obtenidos para dar solución a esta

problemática se encuentran las envolturas (wrappers), para manejadores relacionales, y el mapeo objeto relacional, entre otros.

## COMPLEJIDAD DEL MODELO ORIENTADO A OBJETOS

Más de 20 años de investigación en los SMBD basados en registros han otorgado una alta eficiencia, a diferencia de los SMBDOO, los cuales reflejan una tecnología inmadura. A pesar de que los SMBDOO hacen uso de técnicas desarrolladas especialmente para los SMBD basados en registros, es necesario desarrollar líneas de investigación especiales para cubrir las necesidades particulares de esta tecnología.

## COMPLEJIDAD DEL MODELO ORIENTADO A OBJETOS CON RESPECTO AL DE REGISTROS

Una jerarquía de clases diseñada apropiadamente implica un bajo costo de mantenimiento. Sin embargo, esta jerarquía no se puede modificar fácilmente, sobre todo si ya se han creado objetos persistentes. Es por ello que se debe tener un gran cuidado al identificar las clases y las relaciones entre ellas, de tal forma que todas ellas se identifiquen plenamente.

El tiempo dedicado a las fases de análisis y diseño en una BDOO es mucho mayor que el de una BD relacional, aunque se ve compensado por los costos de mantenimiento mucho menores. Otro aspecto de la complejidad es encontrar personal capacitado y experimentado para estos desarrollos.

## FALTA DE ESTÁNDARES

Es provocada por la motivación natural de los vendedores para mantener productos propietarios,

situación muy común en la industria del software, donde los vendedores de bases de datos se esfuerzan por asegurar su clientela de tal forma que no le sea fácil migrar el procesamiento de datos a otros manejadores.

El término Modelo de Datos Orientado a Objetos no implica una definición de estructuras universalmente aceptadas. La Tecnología de Objetos cuenta con diferentes metodologías de soporte para las etapas de análisis y diseño de sistemas de cómputo; quizá las cuatro más populares son las de Rumbaugh, Booch, Coard-Yourdan y la de Schaer-Mellor. En la mayoría de los casos, estas metodologías son muy similares entre sí, teniendo cada una su manera de representar gráficamente las entidades y aplicar los principios del paradigma. Sin embargo, es posible integrar conceptos de un método con conceptos de otra técnica, creando una metodología mixta; ante esta problemática, surge el Unified Modeling Language (UML) que es un lenguaje para especificar, visualizar, construir y documentar los componentes de sistemas de software.

El UML representa una colección de las mejores prácticas de ingeniería (Booch, OMT y OOSE) que han probado su éxito en el modelado de sistemas grandes y complejos, adoptándose como un estándar OMG. Debido a la falta de estándares para arquitecturas de BDOO, la mayoría de los vendedores de SMBDOO han creado productos propietarios, los cuales difieren bastante entre sí, principalmente en lo referente a sus estructuras internas.

## CONCLUSIONES

El empleo de los SMBD relacionales ha tenido sin duda un gran éxito en el manejo de datos sencillos, de sistemas tales como: nómina, compras, ventas, almacén, renta de automóviles, reservaciones de vuelos, etc. Por otro lado, el adaptar este modelo relacional al manejo de estructuras complejas de datos, resultado del avance tecnológico en la computación y de las nuevas líneas de investigación en Bases de Datos, ha sido problemático e inadecuado; debido principalmente a que su modelo simple de datos carece de un lenguaje de programación completamente computacional, que permita tanto el acceso directo a los datos como la integración de código y datos dentro del control de almacenamiento del SMBD.

Esto ha dado pauta a la investigación de nuevos paradigmas, modelos, técnicas, estándares de modelado y lenguajes, etc., para aplicaciones avanzadas que permitan manipular datos o información compleja, tales como: datos multimedia, requerimientos para múltiples versiones de datos, bases de datos distribuidas en ambiente de redes e Internet, datos espaciales y temporales, habilidad para manejar tipos de restricciones más poderosas, al igual que la integración de persistencia con tipos de datos complejos; es decir, la necesidad apremiante de integrar de una mejor forma código y datos, siendo esto último lo más importante.

Por lo mismo, las Bases de Datos Orientadas a Objetos (BDOO) representan el paso siguiente en esta evolución, para integrar el análisis, diseño y programación Orientada a Objetos; además, permiten el desarrollo y mantenimiento de aplicaciones complejas con un costo menor, haciendo posible que el mismo modelo concep-

tual se aplique al análisis, diseño, programación, definición y acceso a la base de datos, reduciendo el problema de traducción entre diferentes modelos a través de todo el ciclo de vida de los sistemas.

Todo este potencial, brindado por el Modelo de Datos Orientado a Objetos, el cual extiende el modelo semántico para integrar código con la manipulación de datos en una sola estructura, es empleado por los Sistemas Manejadores de Bases de Datos Orientados a Objetos (SMBDOO), como un modelo lógico para el SMBD o como un LPOO persistente.

Por otro lado, este modelo aun joven está teniendo éxito con resultados satisfactorios en la atención de las nuevas necesidades, aún cuando pre-

senta algunas limitantes tales como: inmadurez, la necesidad de innovación de las técnicas y aspectos del manejo de datos complejos, la escasez de diseñadores y desarrolladores con experiencia, la falta de apego a los estándares por parte de fabricantes y desarrolladores de software y el rendimiento limitado de la computadora en aplicaciones de tiempo real, entre otros.

Las Bases Orientadas a Objetos coexistirán con las Bases de Datos Relacionales durante los próximos años, puesto que se utiliza un modelo relacional como forma de estructura de datos dentro de las BDOO.

---

### BIBLIOGRAFÍA

---

- [1] Cooper, Richard. "Object Databases An ODMG Approach". Thomson Computers Press. ITP An International Thomson Publishing Company. 1997.
- [2] Catell, R.G.G.. Barry K. Douglas. "The Object Data Standard: ODMG 3.0". Morgan Kaufmann Publishers, an Imprint of Academic Press. 2000.
- [3] Eckel, Bruce. "Thinking in Java". Second edition. Prentice Hall PTR. 2000.
- [4] Martínez Prieto, Ana Belén. Tesis para obtener el grado de Doctor en Informática. "Un sistema de Gestión de Bases de Datos Orientadas a Objetos Sobre una Máquina Abstracta Persistente". Universidad de Oviedo. Mayo de 2001.
- [5] Elmasri Ramez, Navathe Shantikant B. "Sistemas de Bases de Datos Conceptos Fundamentales". Segunda edición. Addison-Wesley Iberoamericana. 1998.

# Interpretación y Maquinado de Archivos Gerber

Ing. Agustín Cruz Contreras.  
M. en C. J. Carlos Herrera Lozada.  
M. en C. J. Carlos González Robles.  
 Profesores del CIDETEC-IPN.

**E**n este artículo se describe el desarrollo de un sistema CAM (Computer-Aided Manufacturing, Manufactura Asistida por Computadora) de tres ejes, dedicado a la elaboración de circuitos impresos por desbaste de material; el impreso se genera partiendo de la información contenida en archivos Gerber con formato RS-274X.

## INTRODUCCIÓN

Todas las áreas de la electrónica requieren de circuitos impresos para plasmar sus desarrollos. Los prototipos de circuito impreso se pueden elaborar empleando técnicas artesanales, utilizando sustancias químicas que tienen un costo considerable, son contaminantes, afectan la salud y requieren de cierta destreza manual para su uso. El perforado es un trabajo adicional al grabado, y puede volverse tedioso en circuitos con una gran cantidad de perforaciones, además que no se garantiza la precisión del perforado.

También es posible contratar el servicio profesional de fabricantes de circuitos impresos; aunque el costo de los prototipos es muy alto y el tiempo

de entrega considerable, debido al bajo volumen de producción.

El sistema que se presenta en este trabajo es, esencialmente, un graficador (Plotter) robusto de precisión, con herramienta de corte. La programación del sistema controla los ejes X, Y, Z e interpreta la información de los archivos Gerber generados con un programa CAD (Computer-Aided Design, Diseño Asistido por Computadora), Orcad en este caso; los archivos Gerber describen la trayectoria de una herramienta de corte, y se integran por diversas directivas y códigos.

El proceso incluye las siguientes etapas:

- Perforación.
- Interpretación de archivos Gerber.
- Determinación de contornos.
- Seguimiento de contornos.

## MECANISMO

Este trabajo es la continuación del sistema XYZ para la perforación automática de circuitos impresos, presentado en el Congreso Electro 2000 (Instituto Tecnológico de Chihuahua, memoria Electro 2000, página 311).

Para este trabajo se tiene un motor de 400 pasos/vuelta y un husillo con un paso de 8 mm, por lo que el avance por paso es  $8/400 = 0.02$  mm o 0.787 milésimas de pulgada.

## PERFORACIONES

El primer paso en la elaboración del impreso es la perforación. Se requieren las coordenadas X-Y con el diámetro de perforación, las cuales se obtienen interpretando el archivo de diámetros y coordenadas que genera el programa CAD. Este archivo es de tipo texto en forma de lista, el listado No. 1 muestra la información correspondiente al circuito de la figura 1. A simple vista se observan 14 marcas de perforación (pad) en el circuito impreso de dicha figura, y su relación

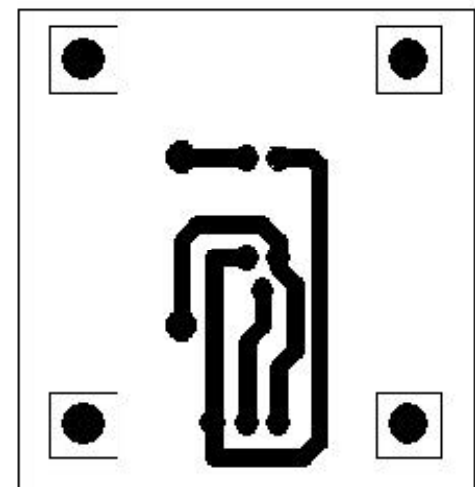


Figura 1. Ejemplo de Circuito Impreso.

```
%
T1C0.031F200S100
X005000Y005000
X006000Y002000
X005000Y010000
X007000Y002000
X007500Y006000
X007000Y007000
X007000Y010000
X008000Y002000
X008000Y007000
X008000Y010000
T2C0.125F200S100
X002000Y002000
X012000Y002000
X012000Y013000
X002000Y013000
M30
```

inicio

coordenadas

inicio del segundo bloque de perforación

a estas coordenadas les corresponde un diámetro de perforación de 0.125".

fin

Listado No.1. Coordenadas y diámetros de perforación.

directa con 14 coordenadas en el listado, en donde los campos precedidos por X y Y corresponden al valor de la coordenada en milésimas de pulgada, el carácter % marca el inicio del programa, M30 es código que marca el final y los campos precedidos por T, indican el diámetro de perforación para el siguiente bloque de coordenadas.

Para interpretar cada coordenada se guardan los caracteres del campo en un arreglo para formar una cadena, y se convierte a valor entero.

El siguiente bloque de programa muestra la forma de obtener el valor entero para la coordenada X a partir de la cadena correspondiente.

```
X018000Y002000
if(letra=='X'){
for(i=0;letra!='Y';i++){
letra=getc(in);
dato[i]=letra;
}
dato[i]='\0';
nx=atoi(dato)/10;
```

Cadena con el valor de X

Con el valor de las coordenadas se determina el número de pasos que debe avanzar cada eje; por ejemplo,

si la coordenada es de 125 milésimas se tiene  $125/0.78=160.25$ , redondeando se requieren 160 pasos para llegar a la posición deseada.

## ARCHIVOS GERBER

La lista siguiente describe los códigos del listado No.2, el cual corresponde al impreso de la figura 1.

G04 Comentario  
G54 Prepara herramienta  
G74 Deshabilita interpolación circular de 360°  
D01 Herramienta abajo  
D02 Herramienta arriba  
D03 Corte sin desplazamiento  
M02 Fin de programa  
IN Nombre  
FS Declaración de formato  
\* Fin de Bloque  
MO Unidades  
IP Polaridad  
AS Selección de eje  
AD Descripción de apertura  
C Círculo  
R Rectángulo

Con el desplazamiento de un círculo de diámetro predeterminado se dibujan las pistas, texto y todas las formas que integran el dibujo de un circuito impreso (ver figura 2).

Analizando los archivos Gerber, estos se pueden considerar formados por tres bloques de información:

- Bloque de aperturas; contiene el tipo y dimensión de las herramientas de corte, que en este caso corresponden a formas geométricas tales como círculos y rectángulos.
- Bloque de coordenadas para el trazo de pistas, texto, planos de tierra, etc.

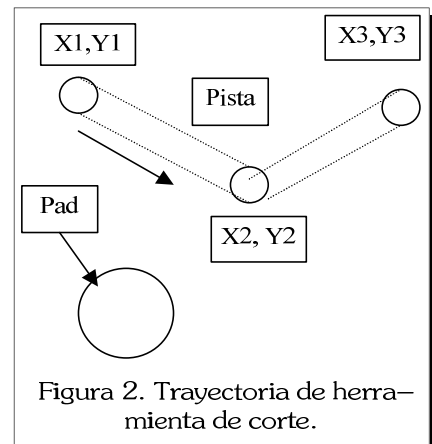


Figura 2. Trayectoria de herramienta de corte.

Comentarios

G04 Mass Parameters \*\*\*

\* G04 Image \*\*\*

%INC:\FOTOINTER.BOT\*%  
%ICAS\*%  
%MOIN\*%  
%IPPOS\*%  
%ASAXBY\*%  
G74\*%FSLAN2X34Y34\*%

Nombre

Unidades (IN=pulgadas)

Polaridad positiva

Selección de ejes

L = omitir ceros  
A = coordenadas absolutas  
N = dos dígitos en código  
X34 = 3 enteros 4 decimales

\* G04 Aperture Definitions \*\*\*

\* %ADD10C,0.1250\*%  
Apertura en círculo con apertura en 0.0125"

%ADD11C,0.0800\*%  
%ADD12R,0.0800X0.0800\*%  
%ADD13C,0.1000\*%  
%ADD14C,0.0010\*%  
%ADD15C,0.0080\*%  
%ADD16C,0.0600\*%  
%ADD17C,0.0500\*%  
%ADD18C,0.0900\*%  
%ADD19R,0.0900X0.0900\*%  
%ADD20C,0.1100\*%  
%ADD21C,0.0100\*%

Aperturas

Listado No.2 (primera parte)

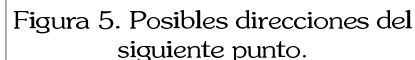


- ## DETERMINACIÓN DE CONTORNOS

Con la información del archivo Gerber se dibuja el circuito impreso en pantalla para verificar que la interpretación de los códigos sea correcta.

## SEGUIMIENTO DE CONTORNOS

Después de determinar los contornos, lo siguiente es localizarlos y recorrerlos en la matriz; el recorrido se refleja en el mecanismo avanzando un paso por cada punto en la matriz. Para ello se busca en la matriz, de izquierda a derecha y de arriba abajo, al elemento que los identifica (un 4); hasta este punto el mecanismo se desplaza con la herramienta retraída, y al llegar al inicio del contorno esta se baja y se mantiene así durante el resto del recorrido. Para encontrar los siguientes puntos del contorno se revisan las ocho posibles direcciones, figura 5.



Si se encuentra otro punto, se avanza a éste, marcando el anterior con una identificación para indicar que ya ha sido considerado; en caso contrario, el contorno se ha terminado. Se repite el proceso en busca del siguiente contorno hasta abarcar toda la matriz.

En la figura 6 se muestra el proceso concluido, apreciándose las perforaciones y contornos grabados.

## CONCLUSIONES

El sistema desarrollado ha permitido comprobar las ideas y algoritmos contenidos en el planteamiento de este trabajo; para tener un prototipo comparable con los equipos comerciales se requiere mejorar el mecanismo y desarrollar una interfaz de usuario. Con respecto al mecanismo, se requiere un bastidor rígido y compacto, un motor de 50,000 r.p.m. en la herramienta de corte, reducir el paso de los husillos de 8 mm a 4 mm, colocar servomotores con codificador en los ejes e implementar el cambio automático de herramienta.

## BIBLIOGRAFÍA

- [1] Barco Gerber System Corporation, "RS-274X Format User's Guide".
- [2] Chris H. Papas, William H. Murray, "Manual de Borland C", Osborne/Mcgraw-Hill.

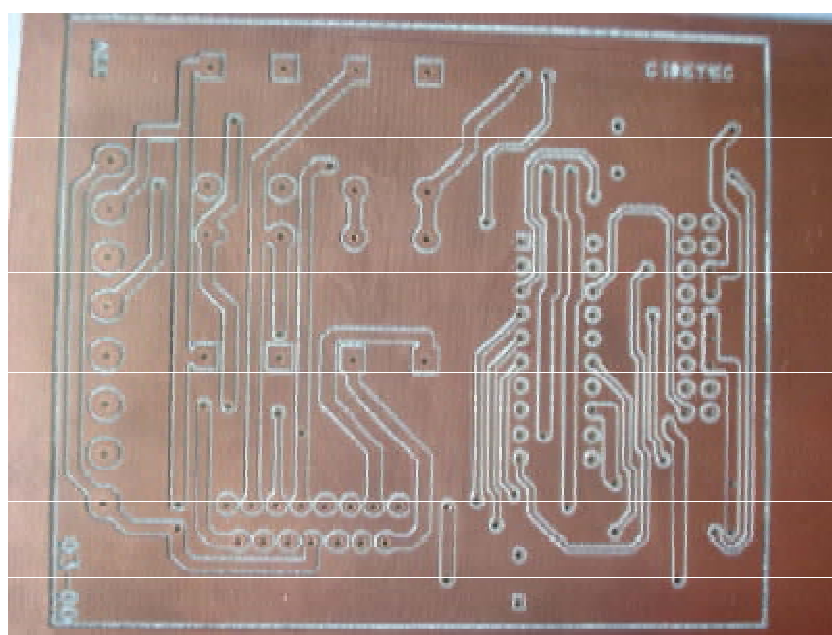


Figura 6. Perforaciones y grabado de contornos.

# Técnicas, Estrategias y Herramientas para el Desarrollo de Software Educativo

**Ing. Patricia Pérez Romero**  
**Profesora del CIDETEC-IPN.**

**A**ctualmente, el mundo se expresa y comunica a través de un universo de imágenes que, dentro de un fenómeno creciente de integración de medios, proponen al sentido de la visión como predominante para los espacios informativos, de entretenimiento y aprendizaje.

La educación no ha quedado fuera; la red Internet, como proveedor de material altamente interactivo, la tecnología multimedia, como ámbito propiciatorio de descubrimiento y aprendizaje con un alto compromiso de los sentidos, y la inteligencia artificial, como disciplina capaz de generar aportes en el tratamiento del lenguaje y la resolución de sus problemas asociados, presentan a la computadora como una herramienta de comunicación en la cual la interactividad adopta la forma del lenguaje humano.

En lo que a capacitación se refiere se está gestando una revolución: están apareciendo las primeras "aulas virtuales", que permiten que los temas sean presentados por especialistas en forma directa e interactiva, sin importar la ubicación geográfica de los alumnos, logrando así la ecuación ideal entre los resultados de la capacitación y la disminución de costos.

Dentro de este marco de profundo cambio, se percibe como estratégico el **"aprender a aprender"**.

Es así, como:

- libros,
- películas y videos,
- cassettes, discos compactos de audio y DVD,
- software educativo,
- kits de experimentación para arte, ciencia y tecnología,
- el acceso a Internet,

sumados a los espacios de observación de la realidad, aparecen como espacios complementarios de comunicación y aprendizaje, donde cada medio tiene la posibilidad de dar de sí lo mejor y encontrar en el otro un factor dinámico de sus propias posibilidades de uso.

Partiendo de esto, encontramos las siguientes cuestiones:

**¿Cómo combinar el poder de la computadora con la filosofía, los conceptos y los métodos de desarrollo de materiales educativos basados en el uso de tecnología informática?**

**¿Cómo redefinir el papel del docente, dentro del contexto de la Era de la Información, como agente orientador en la construcción de conceptos y articulador de experiencias?**

Uno de los factores claves en esta concepción es la identificación explícita del "profesor como el agente del

cambio crítico" para posibilitar una transferencia de tecnología, desde los materiales tradicionales basados en la impresión, a otros medios tecnológicos.

Para que los profesores sigan desarrollando su función como elementos clave en la enseñanza, con la capacidad de adaptación necesaria ante las nuevas herramientas que la tecnología está poniendo en sus manos, deben mantener una actitud abierta para incorporar estos desarrollos tecnológicos, ya que en caso contrario se puede correr el riesgo de "enseñar a los estudiantes de hoy a resolver los problemas del mañana con las herramientas de ayer".

Desde una concepción pedagógica, en el desarrollo de software educativo debemos considerar los siguientes aspectos:

- ¿Cuál es el punto de partida?
- ¿Cuáles son las perspectivas de abordaje posibles?

En el punto de partida se engloban los siguientes aspectos:

El reconocimiento de la disciplina del conocimiento comprometida en modo predominante en el material educativo a desarrollar, contempla:

- su naturaleza
- la identificación de los objetos de estudio



- las relaciones entre los objetos
- los métodos asociados

### FUNDAMENTOS Y MÉTODOS DEL CONOCIMIENTO CIENTÍFICO DEL DISEÑO

- descubrir el conocimiento, a partir de un cierto espacio de exploración dinámica,
- justificar analíticamente el conocimiento, a partir de los objetos que componen un campo problemático y las operaciones y relaciones que ellos permiten establecer.

### MARCO CONCEPTUAL DEL DISEÑO

Se trata de tomar un trozo de realidad y modelarlo según la lógica computacional, caracterizada por: la modelación, el cálculo, la simulación y la interactividad, para transformarlo en un producto escrito, numérico y calculable.

En el desarrollo de software educativo podemos considerar a la computadora como:

- objeto de estudio,
- instrumento de operación,
- extensión de la mente humana,
- herramienta de comunicación, donde la interactividad adopta la forma del lenguaje humano.

### MARCO ESTRATÉGICO DEL DISEÑO

- los temas centrales y su presencia a través de espacios informativos, de trabajo y evaluación,
- el modelo de interactividad,
- la interfaz hombre - máquina.

### LOS TEMAS CENTRALES PUEDEN PRESENTARSE COMO DESARROLLOS DENTRO DE:

- una misma disciplina del conocimiento,
- un marco integrador y multidisciplinario.

En este planteamiento es necesario pensar que cada disciplina tiene una naturaleza y objetos de conocimiento propios, una pertinencia en su red de relaciones, también propia, y métodos particulares de análisis y producción de conceptos y de resolución de problemas.

### LOS MODELOS DE INTERACTIVIDAD

Pensando en la interacción humana como la actividad recíproca y simultánea de dos participantes de una conversación, se debe tener en cuenta que:

*Para que haya interactividad debe permitirse la interrupción.*

Sin embargo, es necesario determinar ¿Cuál es el elemento mínimo de este sistema interactivo, la palabra o la frase?

Ahí aparece un cierto intervalo humano, para que el interlocutor entienda que el otro no lo ignora.

Pensemos ahora en los modelos de interactividad presentes en un material educativo basado en el uso de tecnología informática; aquí encontramos estos tipos de escenarios :

- cerrados, donde la interacción posible es la que la computadora puede estrictamente validar,
- abiertos, con capacidad de generar espacios de producción y aprendizaje.

El desarrollo del tema que aborda el programa puede plantear como modelo de interactividad:

- el diálogo, suscribiendo entonces el modelo de la conversación (modelo de interrupciones recíprocas y simultáneas)
- la conferencia (modelo alternante), que se corresponde con una

forma de comunicación de tipo “cambio y fuera” (*walkie - talkie*).

### LA INTERFAZ HOMBRE - MÁQUINA

¿Qué parte de un programa es la que le permite al usuario comunicarse con una computadora?

El concepto de interfaz hombre - máquina, pensada como superficie de contacto o espacio de comunicación entre el usuario y la computadora, se presenta como un “punto de encuentro entre los bits y la gente”.

Cuando el concepto de interfaz comenzó a emerger, se entendía que la interfaz hombre - máquina era el hardware y el software a través del cual el usuario y la computadora se podían comunicar.

Al evolucionar, hoy se incluyen también los aspectos cognitivos y emocionales que el usuario experimenta al utilizar un programa; desde una concepción tecnológica tenemos:

- Interfases alfanuméricas
- Interfases gráficas de usuario (GUI)
- Interfases gráficas multimediales (MUI)
- Interfases de usuario inteligente (IUI)

En el diseño de interfases, ¿Cuál es el eje de análisis para la toma de decisiones?:

¿Diseñar interfases centradas en el problema o en el usuario?

Es bien conocido que las características de las interfases hombre - máquina actuales en los materiales educativos hipermediales son muy importantes, sin embargo:

- muchas veces son difíciles de usar,
- no resuelven el problema de la desorientación,

- provocan sobresaturación de información, (“*overhead* cognitivo”),
- confunden al usuario, dado que la interfase tiende a ser en sí misma tanto o más importante que el contenido.

El construir una interfaz adecuada está fuertemente relacionado con una mirada hacia el contenido, por una parte, ya que no se debe ignorar en qué disciplina está encuadrado un desarrollo; pero por otra parte, se requiere de un análisis significativo al usuario potencial, hacia el cual va dirigido el material.

La interfaz debe permitir:

- la navegación a través del material según el propio camino y el propio tiempo,
- evitar el problema de la desorientación, construyendo, por ejemplo, mapas conceptuales alternativos a partir de la propia navegación del usuario,
- comunicar directamente los espacios fuertemente relacionados desde el contenido o desde la propuesta de producción, sin viajes innecesarios a través de uniones superfluas.

Ante todo, debe generarse un espacio de comunicación donde todo lo ya enumerado ocurre, y es en ese espacio donde se ponen en juego también los aspectos cognitivos y emocionales, imprescindibles para generar una verdadera “situación de aprendizaje”.

### HERRAMIENTAS Y TÉCNICAS DE DESARROLLO

La disponibilidad de múltiples entornos de desarrollo de software permite la elección de la herramienta adecuada para cada caso específico.

La primer gran decisión a tomar es qué se va a usar como:

- entornos de programación (C++, Visual Basic, Borland Delphi, Logo, etc.)
- entornos de autor que manejan hipermedios en un sentido amplio (Hypercard, Toolbook, Oracle Media Object, Macromedia Director, etc.)
- entornos de autor que manejan hipermedios con un propósito específico (Wincalis, MathKit, diseñadores de sistemas expertos, etc.)
- marcos de autor de propósito general (Authorware, Iconauthor, etc.)

Esta decisión está estrechamente vinculada con:

- el objetivo que alienta el desarrollo del material,
- el factor de escala (pequeña, mediana o grande), en el cual está encuadrada la aplicación a desarrollar,
- la disciplina a la que pertenece el tema central a desarrollar,
- las acciones, desde todo punto de vista, a que van a ser sometidos los objetos de la temática.

El factor de escala se determina por la amplitud y complejidad del proyecto y los recursos económicos, humanos y tecnológicos involucrados en el mismo.

### ¿CÓMO SE INTEGRA UN GRUPO DE TRABAJO?

Un grupo de trabajo para desarrollar software multimedia debe estar compuesto por:

- especialistas en informática,
- especialistas en contenidos (docentes y externos consultados),

- un grupo de diseño gráfico para generar interfases y procesar imágenes,
- un grupo que genere animaciones simples y grabe y/o capture video,
- un fotógrafo,
- un músico.

### ¿QUÉ RECURSOS SE NECESITAN?

- Al menos dos computadoras conectadas en red; una de ellas dedicada a la gestión de gráficos y animación, con:

- tarjeta de captura de video,
- programa de captura y edición de video,
- un digitalizador,
- un programa de OCR (reconocimiento de caracteres),
- programas para procesamiento de imágenes,
- dispositivos para guardar información en discos ópticos, CD-ROM.

- El segundo equipo estará dedicado a la programación de las aplicaciones, con las herramientas adicionales de programación de base necesarias para generar un espacio de trabajo dinámico y seguro.

- Software con bases de datos, incluyendo videos, imágenes y sonidos, indizados debidamente con descripciones precisas.

- Un acceso eficiente a Internet, para acceder a información, bancos de imágenes, software de base y soporte para los problemas técnicos que puedan presentarse.

Si los materiales se desarrollan con fines educativos y no comerciales, pueden utilizarse recursos de diversas fuentes, haciendo referencia a las mismas.

Si los materiales se van a exponer o comercializar es importante conocer los derechos y alcances de la Propiedad Intelectual asociada.

## ¿QUÉ METODOLOGÍA SE APLICA?

Primeramente se debe definir con claridad y precisión el alcance y objetivos del proyecto a desarrollar, elaborando un modelo con base en:

- la definición de categorías en el dominio del problema: entidades y sus relaciones,
- la definición de estructuras de navegación de alto nivel,
- llenar los contenidos de los nodos,
- definir la apariencia de los nodos.

Analizar, en cada caso:

- ejemplos de animación,
- video digital con relato,
- video digital musicalizado,
- imagen fija.

## CONCLUSIONES

Las aplicaciones de software educativo tienden a crear espacios altamente interactivos, donde dicha interactividad toma la forma del lenguaje humano.

El concepto de computadora como máquina de procesamiento de información está cambiando hacia el concepto de máquina de procesamiento del lenguaje humano. El software educativo debe basarse en este concepto y crear verdaderos espacios informativos y de trabajo, que contemplen la formulación de modelos alternativos de aprendizaje que permitan, a su vez, la adquisición del conocimiento como un proceso social.

El factor humano es decisivo, ya que es el que plantea los problemas más serios y cercanos relacionados con aquellos aspectos de la tecnología que no han sido totalmente comprendidos y optimizados para propósitos de comunicación.

## BIBLIOGRAFÍA

- [1] Álvarez Sáiz, E., Alvaro González, J.I., *Crear Multimedia con PC*. ToolBook, 1996.
- [2] Ausubel, D.P., Novak, J.D. & Hanesian, H., *Psicología Educativa: un punto de vista cognoscitivo*. México, Editorial Trillas, 2da. Edición, 1983.
- [3] Baeza Oleza, L., (1996). *Elaboración de documentos hipertextuales. Reflexión sobre experiencias y retos*. Disponible en: [www.uib.es/depart/gte/baeza.html](http://www.uib.es/depart/gte/baeza.html),
- [4] Bou, Guillem, *El guión multimedia*. Editorial Anaya, 1997.
- [5] Cacheiro, M.L., *El diseño multimedia orientado al proceso de enseñanza-aprendizaje*. 1995.
- [6] Vaughan, T., *Multimedia: making it work*. Macromedia, 1994.
- [7] Cabero, J., (1996). *Navegando, construyendo: la utilización de los hipertextos en la enseñanza*. Disponible en: [www.doe.d5.ub.es/te/any95/cabero\\_hiper-text](http://www.doe.d5.ub.es/te/any95/cabero_hiper-text).
- [8] Días, P., Catanazzi, N., Aedo, I., *De la Multimedia a la Hipermidia*. Editorial RA-MA, 1996.
- [9] Landow George P., *Teoría del hipertexto*. Editorial Paidós, 1997.

# Morfología Matemática en Tiempo Real para Imágenes Binarias (B&N)

Ing. Agustín Cruz Contreras  
Profesor del CIDETEC-IPN.  
Email: [acruz@ipn.mx](mailto:acruz@ipn.mx)

**E**n este artículo se describe la implementación en tiempo real de las operaciones básicas de la Morfología Matemática, *Dilatación* y *Erosión*, aplicada a imágenes en blanco y negro y apegada a la temporización de la norma de video NTSC.

## INTRODUCCIÓN

Los procesamientos lineales surgidos a mediados de los años 50, como consecuencia de la carrera espacial, resolvieron con éxito la eliminación de ruido y otras funciones importantes para las que fueron creados, pero presentaban tiempos considerables de procesamiento, alto costo del hardware y una serie de limitaciones en cuanto a la degradación de las imágenes procesadas.

Como respuesta a estas limitaciones nace la *Morfología Matemática*, teoría surgida de las investigaciones de **G. Matheron** y **J. Serra** en los años 60, con fundamento en el uso de procesos no lineales derivados de la comparación de la imagen con un patrón geométrico sencillo, denominado *Elemento Estructurante*, aportando las ventajas siguientes:

- Baja complejidad computacional, basándose tan sólo en operaciones lógicas.
- Se conservan las características geométricas de la imagen procesada.
- Dos primitivas básicas, *Dilatación* y *Erosión*; con combinaciones de las mismas se construyen todos los operadores.

Por lo anterior, la Morfología Matemática (MM) resulta una teoría muy atractiva para su implementación en tiempo real.

## MORFOLOGÍA MATEMÁTICA

La MM es una herramienta para extraer componentes de una imagen, útiles en la representación y descripción de regiones, contornos, esqueletos y formas convexas; su lenguaje es la teoría de conjuntos, los cuales representan las formas de los objetos en una imagen. La MM estudia las transformaciones morfológicas de la imagen, como son la erosión, dilatación y sus combinaciones, bajo la aplicación de operadores locales denominados elementos estructurantes (EE), que actúan como un parámetro de las transformaciones morfológicas.

Tomando como base estas dos primitivas se construyen otras más complejas, tales como la apertura (dilatarse el resultado de una erosión) y

su operación dual, la cerradura (erosionar el resultado de una dilatación), para implementar filtros básicos.

Matemáticamente, la dilatación se define en la **Ecuación 1**, la cual se interpreta como la unión de las traslaciones de A por los elementos b que pertenecen a B.

$$C = A \oplus B = \bigcup_{b \in B} A_b$$

Ecuación 1. Dilatación

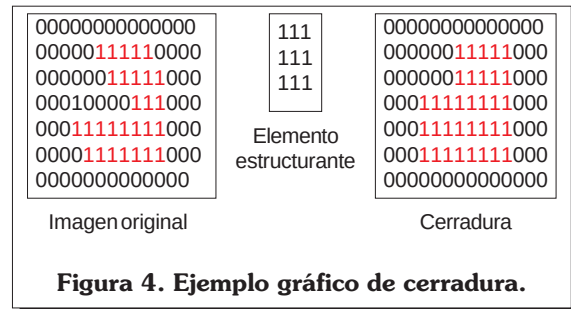
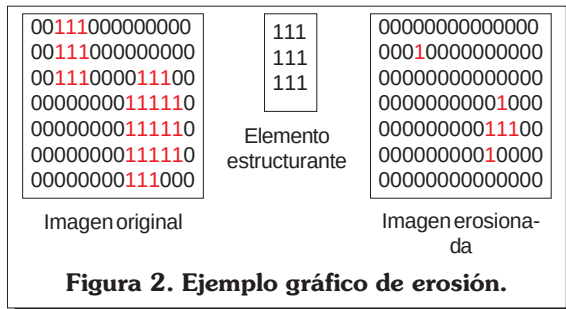
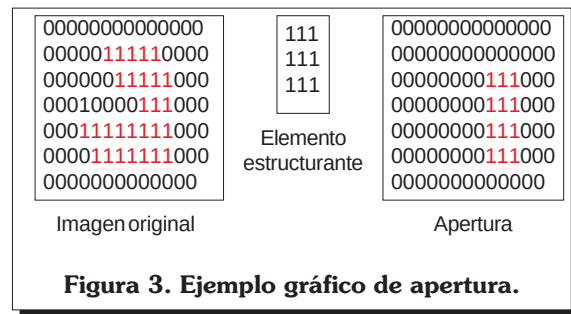
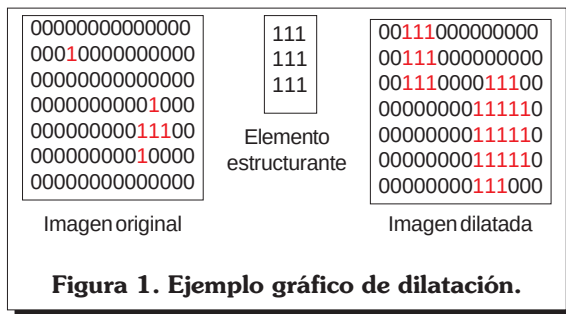
La erosión se define en la **Ecuación 2**, y se interpreta como la intersección de las traslaciones de A por los elementos b que pertenecen a B.

$$C = A \ominus B = \bigcap_{b \in B} A_b$$

Ecuación 2. Erosión

La **figura 1** representa gráficamente la dilatación de una imagen sintética, donde cada uno de sus puntos se representa con el valor uno y el fondo con cero; la dilatación consiste en poner a uno los puntos vecinos (conforme al EE) de cada punto de la imagen original.

En la erosión, cada punto de la imagen original, representado con uno, tendrá el valor de uno en la imagen erosionada, si los puntos vecinos también son uno; de lo contrario será cero. La **figura 2** muestra la imagen original y su erosión con relación al EE.



La apertura se define en la **Ecuación 3**, y se interpreta como la dilatación por B, de la erosión de A con respecto a B.

$$A \circ B = (A \ominus B) \oplus B$$

Ecuación 3

La **Ecuación 4** define a la cerradura, siendo esta la erosión por B, de la dilatación de A con B.

$$A \bullet B = (A \oplus B) \ominus B$$

Ecuación 4

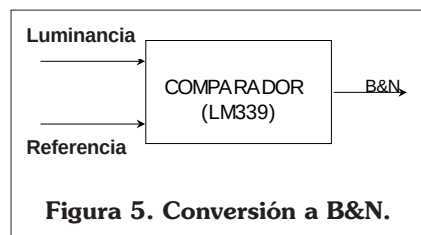
En las **figuras 3 y 4** se muestran ejemplos gráficos para la apertura y la cerradura.

## NORMA DE VIDEO NTSC

La norma de video NTSC (*National Television System Committee*) adoptada en Estados Unidos a finales de 1953, maneja 525 líneas por imagen entrelazada, repartida en 263 y 262 líneas respectivamente para cada campo, con una frecuencia de cuadro de 60 Hz y una duración de línea de 64 ms. La señal de Video

Compuesto, es una señal completa de video, que integra los pulsos de sincronía y la información de la imagen; para el color se agrega una subportadora que contiene la información del mismo, manejando una amplitud de 1 v pico a pico, siendo un estándar en los equipos que manejan señales de video. La información de la imagen monocromática corresponde al brillo y se conoce como luminancia; su valor mínimo corresponde al negro y el máximo al blanco.

En este trabajo las imágenes de interés son en Blanco y Negro (B&N), para ello se requiere convertir la señal monocromática a B&N. El proceso de conversión consiste en tomar un valor de referencia entre el mínimo y el máximo de la señal de luminancia; a los valores menores a la referencia se les asigna 0 (0 v), y a los mayores 1 (5 v), por medio de un comparador, tal como se muestra en la **figura 5**.



El valor de referencia es variable y se conoce como umbral de la imagen de B&N.

## DIGITALIZACIÓN

Se determina una resolución de 320 x 240, esto en base a que se utiliza un solo campo de la imagen; el campo tiene 262 líneas, de las cuales las primeras 21 no tienen información de imagen. Se descartan 22 líneas,  $262 - 22 = 240$ , y con el propósito de mantener la proporción de la imagen se determinan 320 puntos por línea.

El proceso de conversión a B&N tiene implícita la digitalización de la imagen; para sincronizar este proceso de conversión se requieren los pulsos vertical y horizontal, por lo que se emplea el separador de sincronía LM1881. Este circuito integrado recibe la señal de video compuesto, entregando los pulsos horizontal y vertical, y el campo presente (par e impar, **figura 6**).

Para el muestreo de 320 puntos por línea se requiere una señal de





**Figura 6. Separador de sincronía.**

reloj que genere 320 pulsos en el tiempo de duración de la línea, sincronizado con el pulso horizontal. La línea tiene una duración de 64 ms, por lo tanto el periodo aproximado del oscilador es:  $64\text{ ms}/320=0.2\text{ ms}$ .

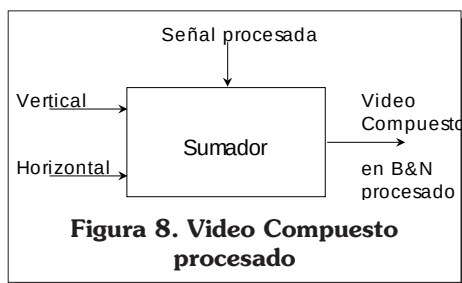
## PROCESAMIENTO

Como se ha descrito, las operaciones morfológicas básicas se pueden realizar a nivel local empleando un punto y su vecindad; la figura 8 muestra la arquitectura propuesta para las operaciones de erosión y dilatación. Para procesar un punto con una vecindad de otro punto se requieren por lo menos dos líneas de video completas y tres puntos de la tercera; las líneas de video se almacenan en registros de corrimiento.

Para la obtención de la dilatación se aplica la operación lógica OR al punto y su vecindad, en el caso de la

errosión se aplica AND (**Figura 7**). Como puede apreciarse, el tiempo de procesamiento, es igual al tiempo de propagación de la compuerta lógica, y el retraso corresponde al tiempo de dos líneas de video y tres puntos de una tercera. Los registros son considerablemente grandes (320 bits cada uno), por lo que se descartó su implementación de manera discreta y se optó por usar un dispositivo FPGA, el XC4010XL de XILINX; cabe aclarar que se empleó este FPGA por contar con él, y no por ser el más apropiado.

La imagen analizada se genera nuevamente a su forma de video compuesto, sumando las señales de sincronía provenientes del dispositivo LM1881 con la señal procesada proveniente del FPGA (**Figura 8**).



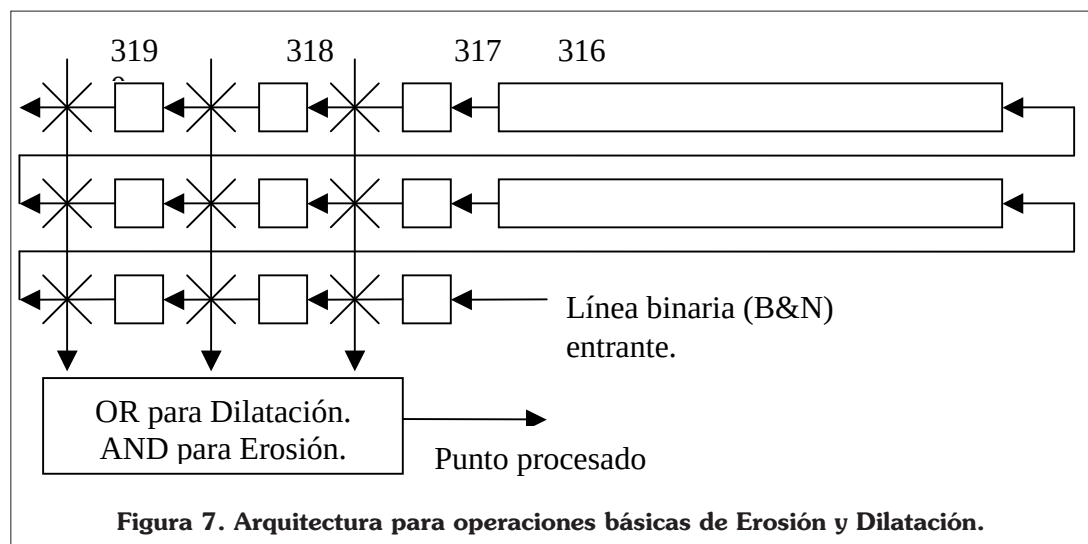
**Figura 8. Video Compuesto  
procesado**

## CONCLUSIONES

Este trabajo es un primer paso en el procesamiento de imágenes en tiempo real, aplicando Matemática Morfológica. En esta etapa se implementan las primitivas básicas de la MM, generando una experiencia que sirve de apoyo para continuar con el desarrollo de la infraestructura que permita la aplicación de algoritmos morfológicos en imágenes B&N, tonos de gris y color. La captura y almacenamiento de las imágenes es otra tarea a realizar; el puerto USB puede ser útil para transferir imágenes en B&N y tonos de gris, mientras que para las imágenes en color se requiere de la velocidad del bus PCI.

## BIBLIOGRAFÍA

- [1] González, R. and Woods R. *"Digital Image Processing"*. Addison-Wesley. 1993.
- [2] Ruiz Vassallo F. *"Televisión"*. ceac. 1985.



**Figura 7. Arquitectura para operaciones básicas de Erosión y Dilatación.**