8

Análisis y Diseño de Sistemas

KENDALL & KENDALL

PEARSON

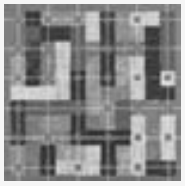
CAPÍTULO 13

Diseño de bases de datos

OBJETIVOS DE APRENDIZAJE

Al completar este capítulo usted podrá:

1. **Comprender los conceptos de bases de datos.**
2. **Usar la normalización para almacenar los datos con eficiencia en una base de datos.**
3. **Usar bases de datos para presentar la información.**
4. **Comprender el concepto de los almacenes de datos corporativos.**
5. **Comprender la utilidad de publicar las bases de datos en la Web.**



Algunos consideran que el almacenamiento de datos es el corazón de un sistema de información. En primer lugar, los datos tienen que estar disponibles cuando el usuario desee utilizarlos; en segundo, deben ser precisos y consistentes (es decir, deben poseer integridad). Además de estos requerimientos, los objetivos del diseño de bases de datos incluyen asegurar la eficiencia en el almacenamiento, la actualización y la recuperación de los datos. Por último, es necesario que la recuperación de información tenga un propósito. La información que se obtiene de los datos almacenados debe estar en una forma que sea útil para la administración, la planeación, el control o la toma de decisiones.

Hay dos metodologías para el almacenamiento de datos en un sistema basado en computadora. La primera es almacenar los datos en archivos individuales, cada uno de los cuales es único para cierta aplicación específica. La segunda implica la creación de una base de datos, es decir, un almacén de datos formalmente definido y controlado centralmente, creado para emplearse en distintas aplicaciones.

A menudo los archivos individuales se diseñan sólo para satisfacer una necesidad inmediata, por lo que es importante que las consultas al sistema se realicen con base en una combinación de algunos de los atributos, que pueden estar contenidos en archivos separados o ni siquiera existan. Las bases de datos son planificadas, por lo que los datos se organizan de manera que se puedan almacenar y recuperar con eficiencia. Los almacenes de datos corporativos son bases de datos muy grandes que alojan datos sintetizados relacionados con un tema específico, para poder responder a las consultas de una manera muy eficiente.

BASES DE DATOS

Las bases de datos no son sólo una colección de archivos. Una base de datos es una fuente central de datos con el fin de que varios usuarios la compartan para su uso en varias aplicaciones. El corazón de una base de datos es el sistema de administración de bases de datos (DBMS), el cual permite crear, modificar y actualizar la base de datos, la recuperación de los datos y la generación de informes y pantallas. A la persona que asegura que la base de datos cumpla con sus objetivos se le conoce como administrador de bases de datos.

Los objetivos de efectividad de la base de datos incluyen lo siguiente:

1. Asegurar que los datos se puedan compartir entre los usuarios y en varias aplicaciones.
2. Mantener datos precisos y consistentes.
3. Asegurar que todos los datos requeridos para las aplicaciones actuales y futuras estén siempre disponibles.

OPORTUNIDAD DE CONSULTORÍA 13.1

Enganche su carrito de limpieza a una estrella

La empresa Marc Schneider Janitorial Supply Company ha solicitado su ayuda para limpiar su almacén de datos. Tan pronto como usted empieza a cuestionar a Marc Schneider con preguntas detalladas sobre su base de datos, el rostro de su interlocutor se ruboriza. "En realidad no tenemos una base de datos de la manera en que la describe", dice con un poco de vergüenza. "Siempre he querido limpiar nuestros registros, pero no pude encontrar una persona capaz de hacerse cargo de esa tarea".

Después de hablar con el Sr. Schneider, usted camina por el pasillo hacia la oficina del tamaño de un clóset que pertenece a Stan Lessink, el programador en jefe. Stan lo pone al tanto del desarrollo histórico del actual sistema de información. "La historia de la empresa Marc Schneider Janitorial Supply Company es el típico relato de mendigo a millonario", comenta Stan. "El primer trabajo del Sr. Schneider fue como conserje en un boliche. Ahorró el suficiente dinero para comprar algunos productos y empezó a venderlos a otros boliches. Pronto decidió expandir el negocio de provisiones de limpieza. Descubrió que, a medida que crecía su negocio, tenía más líneas de productos y tipos de clientes. Los vendedores en la empresa están asignados a las distintas líneas de productos principales (tiendas y oficinas, por ejemplo); algunos están en ventas dentro de la empresa y otros se especializan en equipo pesado, como las pulidoras y enceradoras de pisos. Los registros se mantenían en archivos separados".

Usted recuerda que el Sr. Schneider dijo: "El problema es que no tenemos manera de comparar las ganancias de cada división. Nos gustaría establecer programas de incentivos para los vendedores y proveer un mejor balance al momento de asignar vendedores a cada línea de productos".

Sin embargo, al estar hablando con Stan, él agrega: "Cada división tiene su propio sistema de incentivos. Las comisiones varían. No veo cómo podríamos tener un sistema común. Además, puedo obtener nuestros informes con rapidez debido a que nuestros archivos están configurados como queremos. Nunca hemos emitido un cheque de nómina atrasado".

Describa cómo analizaría las necesidades de almacenamiento de datos de la empresa Marc Schneider Janitorial Supply Company. ¿Desecharía el sistema antiguo o simplemente lo puliría un poco? Describa las implicaciones de su decisión en dos párrafos.

4. Permitir que la base de datos evolucione a medida que aumenten las necesidades de los usuarios.
5. Permitir que los usuarios construyan su propia vista personal de los datos sin preocuparse por la forma en que éstos se almacenan físicamente.

La anterior lista de objetivos constituye un recordatorio de las ventajas y desventajas de la metodología de las bases de datos. En primer lugar, compartir los datos significa que éstos se deben guardar sólo una vez, lo que a su vez ayuda a asegurar su integridad, ya que las modificaciones a éstos se realizan con más facilidad y confiabilidad si los datos aparecen sólo una vez y no en muchos archivos distintos.

Cuando un usuario necesita ciertos datos específicos, una base de datos bien diseñada se anticipa a la necesidad de dichos datos (o tal vez ya se hayan utilizado en otra aplicación). En consecuencia, los datos tienen una mayor probabilidad de estar disponibles en una base de datos que en un sistema de archivos convencional. Una base de datos bien diseñada también puede ser más flexible que varios archivos separados; es decir, una base de datos puede evolucionar a medida que cambian las necesidades de los usuarios y las aplicaciones.

Por último, la metodología de las bases de datos tiene la ventaja de permitir que los usuarios tengan su propia vista de los datos. Los usuarios no necesitan preocuparse de la estructura de la base de datos ni con su almacenamiento físico.

Muchos usuarios extraen partes de la base de datos central de las mainframes y las descargan en PC o dispositivos portátiles. Después, estas bases de datos reducidas se utilizan para generar informes o responder a las consultas específicas para el usuario final.

Las bases de datos relacionales para las PC han mejorado considerablemente durante los últimos años. Uno de los principales cambios tecnológicos ha sido el diseño de software de base de datos que aprovecha la GUI. Con la llegada de programas como Microsoft Access, los usuarios pueden arrastrar y soltar campos entre dos o más tablas. El desarrollo de bases de datos relacionales con estas herramientas se ha facilitado mucho.

CONCEPTOS DE DATOS

Es importante comprender cómo se representan los datos antes de considerar el uso de archivos o de la metodología de las bases de datos. En esta sección veremos algunas definiciones críticas, incluyendo la abstracción de los datos del mundo real hasta el almacenamiento de los datos en tablas, y las relaciones de bases de datos.

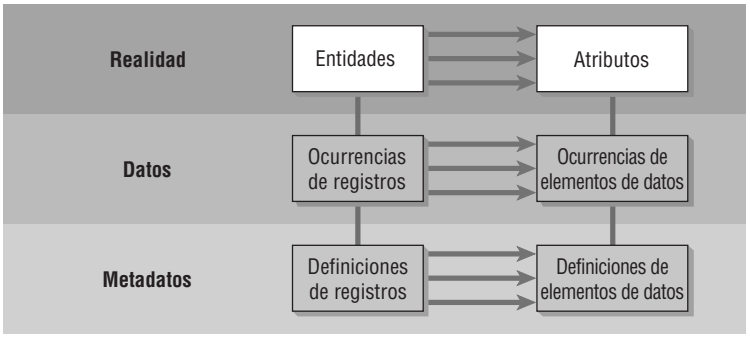


FIGURA 13.1
Realidad, datos y metadatos.

Realidad, datos y metadatos

Nos referiremos al mundo real como la realidad. Los datos que se recopilen sobre personas, lugares o eventos en la realidad se almacenarán en un momento dado en un archivo o en una base de datos. Para comprender la forma y estructura de los datos, se requiere información sobre los datos en sí. La información que describe a los datos se denomina metadatos.

En la figura 13.1 se muestra la relación entre realidad, datos y metadatos. Dentro del ámbito de la realidad están las entidades y los atributos; dentro del ámbito de los datos reales están las ocurrencias de registros y las ocurrencias de elementos de datos; y dentro del ámbito de los metadatos están las definiciones de registros y las definiciones de elementos de datos. En las siguientes subsecciones hablaremos sobre el significado de estos términos.

ENTIDADES Cualquier objeto o evento sobre el que alguien decida recolectar datos es una entidad. También puede ser una persona, lugar o cosa (por ejemplo, un vendedor, una ciudad o un producto). Una entidad puede ser también un evento o unidad de tiempo, como la descompostura de una máquina, una venta, un mes o un año. Además de las entidades que vimos en el capítulo 2 hay una entidad menor adicional, conocida como subtipo de entidad; se representa con un rectángulo más pequeño dentro del rectángulo de entidad.

Un subtipo de entidad es una relación especial de uno a uno empleada para representar los atributos (campos) adicionales de otra entidad que tal vez no estén presentes en todos los registros de la primera entidad. Los subtipos de entidades eliminan la situación en la que una entidad puede tener campos nulos almacenados en tablas de bases de datos.

Un ejemplo es la entidad primaria de un cliente. Los clientes preferidos pueden tener campos especiales que contengan información de descuento; esta información se coloca en un subtipo de entidad. Otro ejemplo son los estudiantes que tienen prácticas. El ARCHIVO MAESTRO DE ESTUDIANTES no debería contener información sobre las prácticas para cada estudiante, ya que tal vez sólo un pequeño número de ellos las tengan.

RELACIONES Las relaciones son asociaciones entre entidades (algunas veces se les conoce como asociaciones de datos). La figura 13.2 es un diagrama entidad-relación (E-R) que muestra varios tipos de relaciones.

El primer tipo de relación es una relación de uno a uno (se designa como 1:1). El diagrama muestra que sólo hay un PAQUETE DE PRODUCTO para cada PRODUCTO. La segunda relación de uno a uno muestra que cada EMPLEADO tiene una OFICINA única. Cabe mencionar que todas esas entidades se pueden describir con más detalle (un precio de producto no sería una entidad; tampoco lo sería una extensión telefónica).

Otro tipo de relación es la asociación de uno a muchos (1:M) o de muchos a uno. Como se muestra en la figura, a un MÉDICO en una organización al cuidado de la salud se le asignan muchos PACIENTES, pero un PACIENTE se asigna a un solo MÉDICO. Otro ejemplo muestra que un EMPLEADO es miembro de sólo un DEPARTAMENTO, pero cada DEPARTAMENTO tiene muchos EMPLEADOS.

Por último, una relación de muchos a muchos (se designa como M:N) describe la posibilidad de que las entidades puedan tener muchas asociaciones en cualquier dirección. Por ejemplo, un ESTUDIANTE puede tener muchos CURSO(s) y al mismo tiempo un CURSO puede tener muchos ESTUDIANTE(s) inscritos en él. El segundo ejemplo muestra que un VENDEDOR puede llamar a muchas CIUDAD(es) y una CIUDAD puede ser un área de ventas para muchos VENDEDOR(es).

La figura 13.3 muestras los símbolos estándar para la notación de pata de cuervo, la explicación oficial de los símbolos y lo que en realidad significan. Observe que el símbolo para una entidad es un rectángulo. Una entidad se define como una clase de persona, lugar o cosa. Un rectángulo con un diamante en su interior representa a una entidad asociativa, la cual se utiliza para unir dos entidades. Un rectángulo con un óvalo en su interior representa a una entidad atributiva, la cual se utiliza para repetir grupos.

Las demás notaciones necesarias para dibujar diagramas E-R son las conexiones, de las cuales hay cinco tipos distintos. En la porción inferior de la figura se explica el significado de la notación. Cuando una línea recta

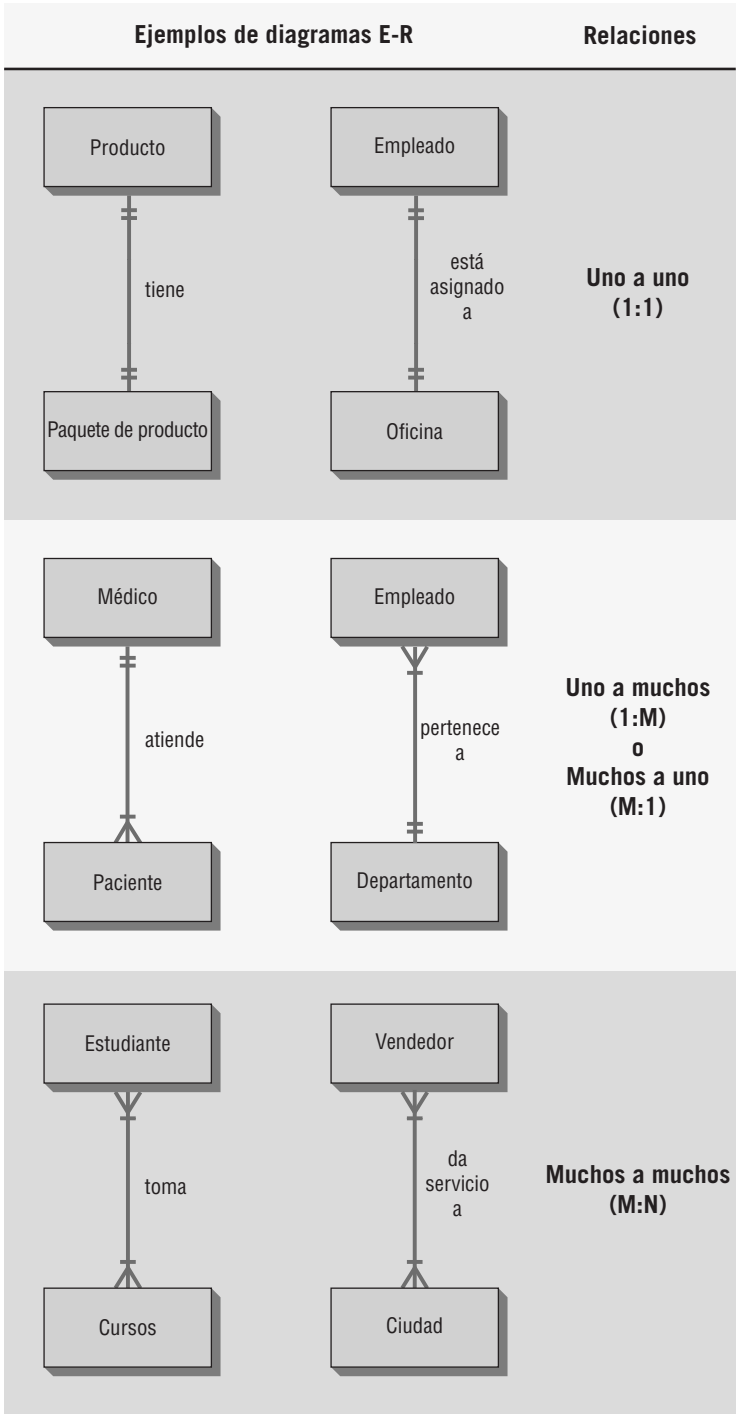


FIGURA 13.2
Los diagramas de entidad-relación (E-R) pueden mostrar asociaciones de uno a uno, de uno a muchos, de muchos a uno o de muchos a muchos.

conecta dos entidades simples y los extremos de la línea están marcados con dos rayas pequeñas (||), existe una relación de uno a uno. Después observará una pata de cuervo con una raya pequeña (|); cuando esta notación vincula a las entidades, indica una relación de uno a uno o de uno a muchos (a uno o más).

Las entidades vinculadas con una línea recta más una raya corta (|) y un cero (que parece más un círculo, O) describen una relación de uno a cero o de uno a uno (sólo cero o uno). Un cuarto tipo de vínculo para relacionar entidades se dibuja con una línea recta marcada en un extremo con un cero (O) seguido de una pata de cuervo. Este tipo muestra una relación de cero a cero, de cero a uno o de cero a muchos. Por último, una línea recta que vincula entidades con una pata de cuervo en un extremo describe una relación a más de uno.

Una entidad puede tener una relación que se conecta a sí misma. Este tipo de relación se denomina relación de autounión; la implicación es que debe haber una forma de vincular un registro de un archivo a otro registro

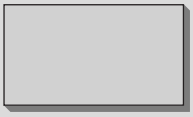
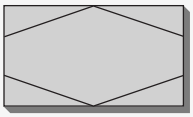
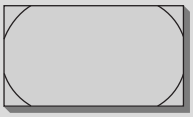
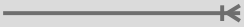
Símbolo	Explicación oficial	Lo que realmente significa
	Entidad	Una clase de personas, lugares o cosas
	Entidad asociativa	Se utiliza para unir dos entidades
	Entidad atributiva	Se usa para repetir grupos
	Relación a 1	Exactamente uno
	Relación a muchos	Uno o más
	Relación a 0 o 1	Sólo cero o uno
	Relación a 0 o más	Puede ser cero, uno o más
	Relación a más de 1	Mayor de uno

FIGURA 13.3
Los símbolos del diagrama entidad-relación y sus significados.

del mismo archivo. Podemos encontrar un ejemplo de relación de autounión en las simulaciones de HyperCase que se incluyen en estos capítulos. Una tarea puede tener una tarea anterior (es decir, una que se deba completar antes de poder empezar la tarea actual). En esta situación, un registro (la tarea actual) apunta a otro registro (la tarea anterior) en el mismo archivo.

Las relaciones en palabras se pueden escribir a lo largo de la parte superior o lateral de cada línea conectora. En la práctica se ve la relación en una dirección, aunque podemos escribir relaciones en ambos lados de la línea, cada una de las cuales representa el punto de vista de una de las dos entidades (vea el capítulo 2 para obtener más detalles sobre cómo dibujar diagramas E-R).

UN EJEMPLO DE ENTIDAD-RELACIÓN En la figura 13.4 podemos ver un ejemplo de un diagrama entidad-relación que contiene muchas entidades, muchos tipos distintos de relaciones y numerosos atributos. El diagrama E-R tiene que ver con un sistema de facturación, en especial con la parte del sistema relacionada con la prescripción (por cuestión de simpleza vamos a suponer que las visitas al consultorio se manejan de una manera distinta y están fuera del alcance de este sistema).

Las entidades son PRESCRIPCIÓN, MÉDICO, PACIENTE y COMPAÑÍA DE SEGUROS. La entidad TRATAMIENTO no es importante para el sistema de facturación, pero forma parte del diagrama E-R debido a que se utiliza como puente para PRESCRIPCIÓN y PACIENTE. Por lo tanto la dibujamos como entidad asociativa en la figura.

Aquí, un MÉDICO atiende a muchos PACIENTE(s) (1:M), cada uno de los cuales está suscrito a una COMPAÑÍA DE SEGUROS individual. Desde luego que el PACIENTE es sólo uno de los muchos pacientes que se suscriben a esa COMPAÑÍA DE SEGUROS (M:1) específica.

Para completar los registros del MÉDICO, éste necesita mantener información sobre los tratamientos que ha llevado un PACIENTE. Muchos PACIENTE(s) experimentan muchos TRATAMIENTO(s), lo cual la con-

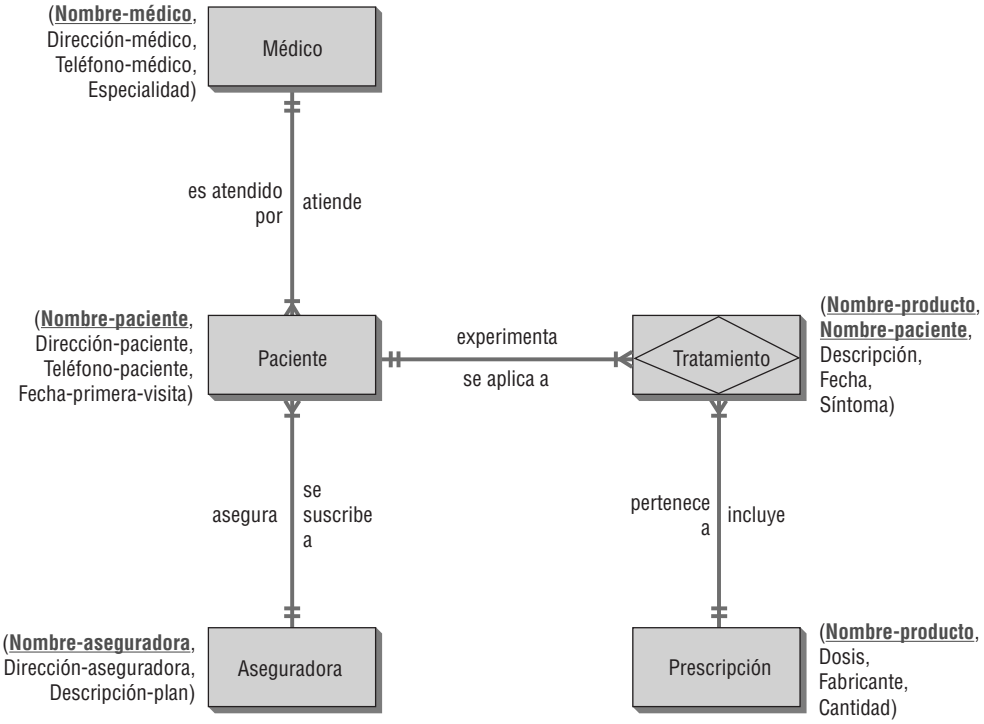


FIGURA 13.4
El diagrama entidad-relación para el tratamiento de pacientes. Los atributos se pueden enlistar en un lado de las entidades. En cada caso, la clave está subrayada.

vierte en una relación de muchos a muchos (M:N). El TRATAMIENTO se representa como una entidad asociativa debido a que no es importante en nuestro sistema de facturación por sí solo. Los TRATAMIENTO(s) pueden incluir tomar PRESCRIPCIÓN(es), por lo cual también es una relación M:N debido a que muchos tratamientos pueden requerir combinaciones de farmacéuticos y muchos fármacos pueden funcionar para muchos tratamientos.

Después se llenan ciertos detalles para los atributos. Éstos se enlistan a un lado de cada una de las entidades y se subraya la clave. Por ejemplo, la entidad PRESCRIPCIÓN tiene NOMBRE PRODUCTO, DOSIS, FABRICANTE y CANTIDAD. En teoría sería benéfico diseñar una base de datos de esta forma, utilizando diagramas entidad-relación y después completando los detalles concernientes a los atributos. Esta metodología arriba-abajo es conveniente, pero algunas veces es difícil de lograr.

ATRIBUTOS Un atributo es cierta característica de una entidad. Puede haber muchos atributos para cada entidad. Por ejemplo, un paciente (entidad) puede tener muchos atributos tales como apellido paterno, primer nombre, dirección, ciudad, estado, etcétera; la fecha de la última visita del paciente, así como los detalles de las prescripciones también son atributos. Cuando construimos el diccionario de datos en el capítulo 8, el elemento más pequeño que describimos se denominó elemento de información. Al hablar sobre archivos y bases de datos, estos elementos de datos se denominan comúnmente elementos de datos. De hecho, los elementos de datos son las unidades más pequeñas en un archivo o base de datos. El término *elemento de datos* se puede intercambiar con la palabra *atributo*.

Los elementos de datos pueden tener valores. Estos valores pueden ser de longitud fija o variable; pueden ser caracteres alfabéticos, numéricos, especiales o alfanuméricos. En la figura 13.5 encontrará ejemplos de elementos de datos y sus valores.

Algunas veces a un elemento de datos también se le conoce como campo. Sin embargo, un campo representa algo físico, no lógico. Por lo tanto, muchos elementos de datos se pueden empaquetar en un campo; el campo se puede leer y convertir en varios elementos de datos. Un ejemplo común de esto es guardar la fecha en un solo campo como MM/DD/AAAA. Para ordenar el archivo por fecha, se extraen tres elementos de datos separados del campo y se ordenan primero por AAAA, después por MM y finalmente por DD.

REGISTROS Un registro es una colección de elementos de datos que tienen algo en común con la entidad descrita. La figura 13.6 es una ilustración de un registro con muchos elementos de datos relacionados. El registro que se muestra es para un pedido realizado con una empresa de pedidos por correo. PEDIDO-#, APELLIDO PATERNO, INICIAL, DIRECCIÓN CALLE, CIUDAD, ESTADO y TARJETA CRÉDITO son todos atributos. La mayoría de los registros son de longitud fija, por lo que no hay necesidad de determinar la longitud del registro cada vez que se utiliza.

Entidad	Elemento de datos	Valor
Vendedor	Número vendedor	87254
	Nombre vendedor	Kaytell
	Nombre empresa	Music Unlimited
	Dirección	45 Arpeum Circle
	Ventas	$20,765
Paquete	Anchura	2
	Altura	16
	Longitud	16
	Peso	3
	Dirección postal	765 Dulcinea Drive
	Dirección retorno	P.O. Box 341, Spring Valley, MN
Pedido	Producto(s)	B521
	Descripción(es)	Disco compacto “Mi Bella Dama”
	Cantidad ordenada	1
	Apellido paterno del cliente	Kiley
	Primera inicial	R.
	Dirección calle	765 Dulcinea Drive
	Ciudad	La Mancha
	Estado	CA
	Código postal	93407
	Número tarjeta crédito	65-8798-87
	Fecha en que se colocó el pedido	01/03/2010
	Cantidad	$6.99
	Estado	Pedido pendiente

FIGURA 13.5

Los valores típicos asignados a los elementos de datos pueden ser números, caracteres alfabéticos, caracteres especiales y combinaciones de los tres.

Bajo ciertas circunstancias (por ejemplo, cuando el espacio es una prioridad) se utilizan registros de longitud variable. Este tipo de registro se utiliza como alternativa para reservar una gran cantidad de espacio para el registro más largo posible, como el número máximo de veces que haya visitado un paciente a un médico. Cada visita contendría muchos elementos de datos que formarían parte del registro completo del paciente (o carpeta de archivo en un sistema manual). Más adelante en el capítulo hablaremos sobre la normalización de una relación. La normalización es un proceso que elimina a los grupos repetitivos que se encuentran en los registros de longitud variable.

CLAVES Una clave es uno de los elementos de datos en un registro que se utiliza para identificarlo. Cuando una clave identifica a un registro en forma única, se le llama clave primaria. Por ejemplo, PEDIDO-# puede ser una clave primaria, ya que sólo se asigna un número a cada pedido del cliente. De esta forma, la clave primaria identifica a la entidad del mundo real (pedido del cliente).

Hay que tener cuidado especial a la hora de diseñar la clave primaria. A menudo es un número secuencial o un número secuencial con un número autoverificable (llamado dígito de verificación) al final. A veces hay cierto significado integrado en la clave primaria, pero definir una clave primaria con base en un atributo se considera un riesgo: si cambia el atributo, la clave primaria también cambiará y se creará una dependencia entre la clave primaria y los datos.

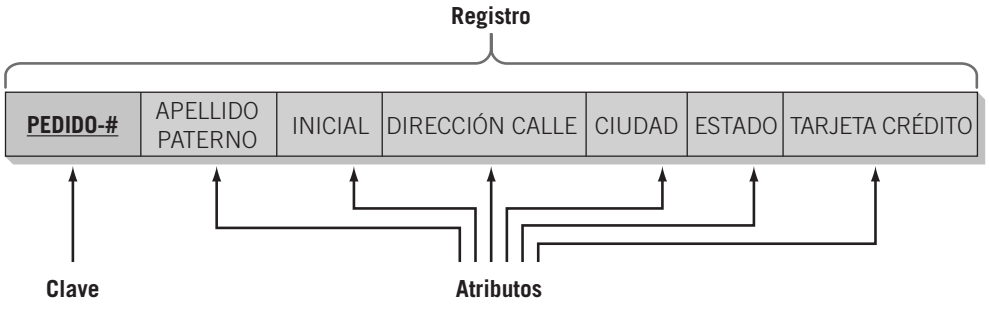


FIGURA 13.6

Un registro tiene una clave primaria y puede tener muchos atributos.

Un ejemplo de clave primaria basada en datos es el uso de una abreviatura de estado para el nombre de un estado o el código de equipaje de una aerolínea para el nombre de un aeropuerto. A un atributo o colección de atributos que pueden servir como clave primaria se le denomina clave candidata. Una clave primaria también debe ser mínima y no debe contener atributos adicionales que sean necesarios para identificar a un registro.

A una clave se le denomina clave secundaria si no puede identificar a un registro en forma única. Las claves secundarias pueden ser únicas o pueden identificar a varios registros en una base de datos. Las claves secundarias se pueden utilizar para seleccionar un grupo de registros que pertenezcan a un conjunto (por ejemplo, pedidos del estado de Virginia).

Cuando no es posible identificar a un registro en forma única mediante el uso de los elementos de datos encontrados en un registro, podemos construir una clave al elegir dos o más elementos de datos y combinarlos. A esta clave se le denomina clave concatenada o compuesta. Cuando se utiliza un elemento de datos como clave en un registro, la descripción se subraya. Por lo tanto, en el registro de PEDIDO (PEDIDO-# APELLIDO PATERNO, INICIAL, DIRECCIÓN CALLE, CIUDAD, ESTADO, TARJETA CRÉDITO) la clave es PEDIDO-#. Si un atributo es una clave en otro archivo, se debe subrayar con una línea punteada.

Algunas bases de datos permiten al desarrollador usar un identificador de objetos (OID), una clave única para cada registro en la base de datos, no sólo en una tabla. Dado un identificador de objetos, se obtendrá un registro sin importar la tabla en la que exista. Éste se puede incluir con un pedido o una confirmación de pago, junto con un mensaje como: "Éste es su número de confirmación".

METADATOS Los metadatos son datos sobre los datos del archivo o base de datos. Los metadatos describen el nombre proporcionado y la longitud asignada a cada elemento de datos. Los metadatos también describen la longitud y la composición de cada uno de los registros.

La figura 13.7 es un ejemplo de metadatos para una base de datos para cierto software genérico. La longitud de cada elemento de datos se indica de acuerdo con una convención, donde 7.2 significa que se reservan siete espacios para el número, dos de los cuales están a la derecha del punto decimal. La letra N significa "numérico" y la A significa "alfanumérico". La D representa a la "fecha" y se establece de manera automática en la forma MM/DD/AAAA. Algunos programas como Microsoft Access usan inglés o español simple para los metadatos, por lo que se utilizan palabras como *text (texto)*, *currency (moneda)* y *number (número)*. Microsoft Access provee un valor predeterminado de 50 caracteres como longitud de campo para los nombres, lo cual está bien si trabajamos con sistemas pequeños, sin embargo, si va a trabajar con una base de datos extensa para un banco o una compañía de servicios púbicos, por ejemplo, no es conveniente que dedique todo ese espacio a ese campo, pues de lo contrario la base de datos crecería mucho y se llenaría de espacio desperdiciado. Aquí es cuando podemos usar metadatos para planear con anticipación y diseñar una base de datos más eficiente.

Archivos

Un archivo contiene grupos de registros que se utilizan para proveer información para operaciones, planeación, administración y toma de decisiones. Primero hablaremos sobre los tipos de archivos utilizados y después describiremos las diversas formas en que se pueden organizar los archivos convencionales.

TIPOS DE ARCHIVOS Podemos usar los archivos para guardar datos durante un periodo indefinido o almacenarlos provisionalmente con un propósito específico. Los archivos maestros y los archivos de tablas se utilizan para almacenar datos por un periodo prolongado. Los archivos temporales por lo general se denominan archivos de transacciones, archivos de trabajo o archivos de informes.

Archivos maestros Los archivos maestros contienen registros para un grupo de entidades. Estos atributos se pueden actualizar con frecuencia, pero los registros en sí son relativamente permanentes. Estos archivos tienden a tener registros extensos que contienen toda la información sobre una entidad de datos. Por lo general cada registro contiene una clave primaria y varias claves secundarias.

Aunque el analista tiene la libertad de ordenar los elementos de datos en un archivo maestro en cualquier orden, una disposición habitual es colocar el campo de la clave primaria primero, seguido de los elementos descriptivos y al último los elementos que cambian con frecuencia con base en las actividades de negocios. Ejemplos de un archivo maestro son los registros de pacientes, los registros de clientes, un archivo de personal y un archivo de inventario de piezas.

Archivos de tablas Un archivo de tablas contiene los datos que se utilizan para calcular más datos o medidas de desempeño. Un ejemplo es una tabla de las tarifas postales empleada para determinar los costos de envío de un paquete; otro ejemplo es una tabla de impuestos. Por lo general sólo un programa lee los archivos de tablas.

Archivos de transacciones Un archivo de transacciones se utiliza para introducir las modificaciones que actualizan el archivo maestro y producen informes. Suponga que el archivo maestro de suscriptores de un periódico necesita actualizarse; el archivo de transacciones contendría el número de suscriptor y un código de transacción, como E para extender la suscripción, C para cancelarla o D para cambiar la dirección. Así sólo hay

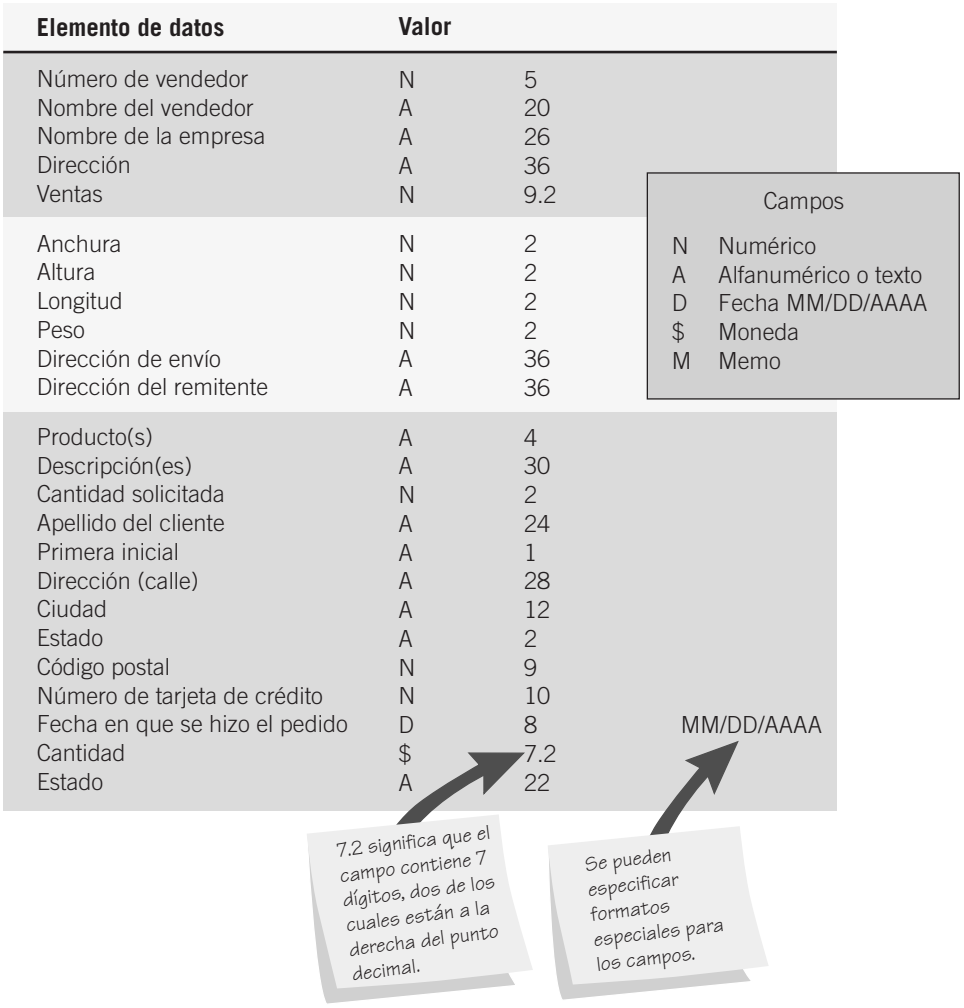


FIGURA 13.7
Los metadatos incluyen una descripción de la apariencia que debe tener el valor de cada elemento de datos.

que introducir la información relevante a las necesidades de actualización; es decir, la longitud de la renovación si es E y la dirección si es D. No se requeriría información adicional si se cancelara la suscripción. El resto de la información ya existe en el archivo maestro. Los archivos de transacciones pueden contener varios tipos de registros, como los tres que se utilizan para actualizar el archivo maestro de suscripciones al periódico, con un código en el archivo de transacciones para indicar el tipo de transacción.

Archivos de informes Cuando es necesario imprimir un informe y no hay una impresora disponible (por ejemplo, cuando está ocupada imprimiendo otros trabajos), se utiliza un archivo de informes. Al proceso de enviar la salida a un archivo en vez de enviarla a una impresora se le conoce como puesta en cola, o "spooling". Después, cuando el dispositivo está listo se puede imprimir el documento. Los archivos de informes son muy útiles, ya que los usuarios pueden llevar los archivos a otros sistemas computarizados y enviarlos como salida a dispositivos especializados.

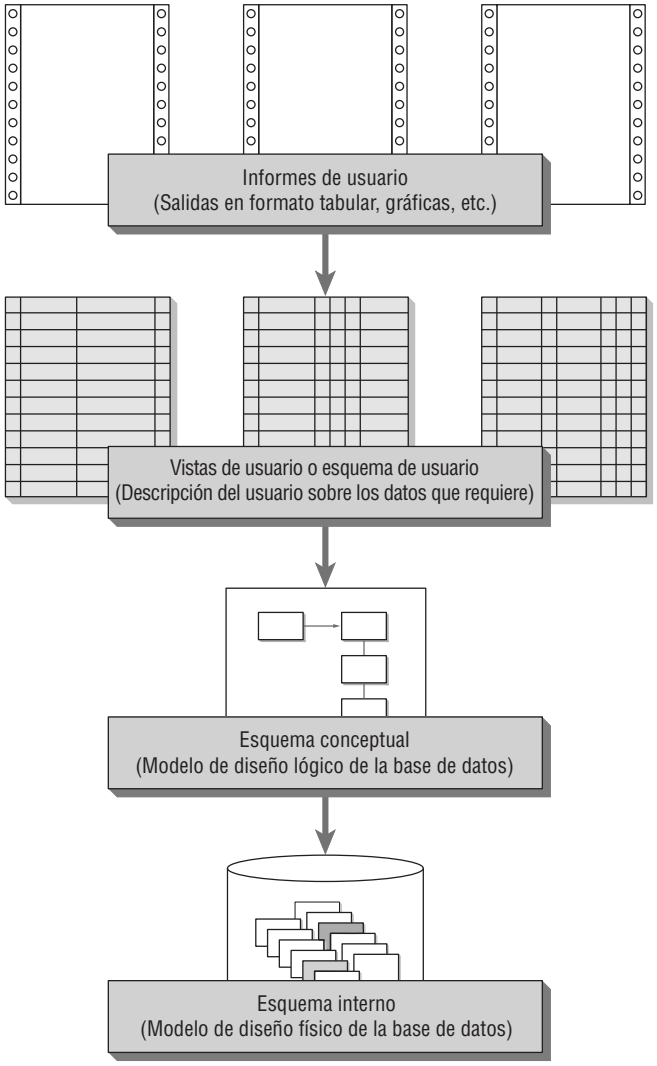
Bases de datos relacionales

Las bases de datos se pueden organizar en varias formas; el tipo más común es la base de datos relacional. Una base de datos relacional está organizada en tablas representativas, lo cual minimiza la repetición de los datos, lo cual a su vez reduce los errores y el espacio de almacenamiento.

VISTAS LÓGICAS Y FÍSICAS DE LOS DATOS A diferencia de un archivo, una base de datos está diseñada para que la compartan muchos usuarios. Está claro que todos los usuarios ven los datos en distintas formas. Nos referiremos a la manera en que un usuario imagina y ve los datos como vista de usuario; esto genera un inconveniente: distintos usuarios tienen distintas vistas de usuarios. El analista de sistemas necesita examinar estas vistas y desarrollar un modelo lógico general de la base de datos. Por último, hay que transformar el modelo lógico de la base de datos en el correspondiente diseño de la base de datos física. El diseño físico se relaciona con la forma en que se almacenan y relacionan los datos, así como la manera en que se accede a ellos.

FIGURA 13.8

El diseño de bases de datos incluye sintetizar los informes de los usuarios, las vistas de usuario y los diseños lógicos y físicos.



En la literatura de bases de datos, las vistas se conocen como esquema. La figura 13.8 muestra cómo están relacionados los informes y las vistas de usuario (esquema de usuario) con el modelo lógico (esquema conceptual) y el diseño físico (esquema interno).

Hay tres tipos principales de bases de datos estructuradas en forma lógica: jerárquicas, de redes y relacionales. Podemos encontrar los primeros dos tipos en sistemas heredados (antiguos). En la actualidad, el analista comúnmente diseña una base de datos relacional.

ESTRUCTURAS DE DATOS RELACIONALES Una estructura de datos relacional consiste en uno o más tablas bidimensionales, las cuales se conocen como relaciones. Las filas de la tabla representan los registros y las columnas contienen atributos.

La figura 13.9 muestra la estructura relacional para una base de datos de pedidos de CD musicales. Aquí se necesitan tres tablas para: 1) describir los artículos y llevar el registro del precio actual de CD (PRECIO ARTICULO), 2) describir los detalles del pedido (PEDIDO) y 3) identificar el estado del pedido (ESTADO ARTICULO).

Para determinar el precio de un artículo necesitamos conocer el número de artículo para encontrarlo en la relación PRECIO ARTICULO. Para actualizar el número de tarjeta de crédito de G. MacRae, podemos buscar la relación PEDIDO para MacRae y corregirla sólo una vez, incluso cuando él pidió muchos CD. Sin embargo, para averiguar el estado de parte de un pedido debemos conocer ARTICULO-# y PEDIDO-#; después debemos localizar esa información en la relación ESTADO PEDIDO.

La labor de mantenimiento de las tablas en una estructura relacional es bastante simple, si se le compara con la labor de mantenimiento de una estructura jerárquica o de red. Una de las principales ventajas de las estructuras relacionales es que las consultas ad hoc se manejan con eficiencia.

PRECIO ARTICULO

ARTICULO-#	TITULO	PRECIO
B235	Guys and Dolls	8.99
B521	My Fair Lady	6.99
B894	42nd Street	10.99
B992	A Chorus Line	10.99

PEDIDO

PEDIDO-#	APELLIDO PATERNO	I	DIRECCIÓN CALLE	CIUDAD	EDO	CTA COBRO
10784	MacRae	G	2314 Curly Circle	Lincoln	NE	45-4654-76
10796	Jones	S	34 Dream Lane	Oklahoma City	OK	44-9876-74
11821	Preston	R	1008 Madison Ave.	River City	IA	34-7642-64
11845	Channing	C	454 Harmonia St.	Nueva York	NY	34-0876-87
11872	Kiley	R	765 Dulcinea Drive	La Mancha	CA	65-8798-87

ESTADO ARTICULO

ARTICULO-#	PEDIDO-#	ESTADO
B235	10784	Se envió el 5/12
B235	19796	Se envió el 5/14
B235	11872	En proceso
B521	11821	En proceso
B894	11845	Pedido pendiente
B894	11872	Se envió el 5/12
B992	10784	Se envió el 5/12

FIGURA 13.9

En una estructura de datos relacional, los datos se almacenan en muchas tablas.

Cuando se analizan estructuras relacionales en la literatura sobre bases de datos, se utiliza con frecuencia distinta terminología. A un archivo se le llama tabla o relación, a un registro por lo general se le denomina tupla y al conjunto de valores de atributos se le conoce como dominio.

Para que las estructuras relacionales sean útiles y manejables, primero hay que normalizar las tablas relacionales. En la siguiente sección veremos la normalización con detalles.

NORMALIZACIÓN

La normalización es la transformación de las vistas de usuario y almacenes de datos complejos en un conjunto de estructuras de datos estables y más pequeñas. Además de ser más simples y estables, las estructuras de datos normalizadas se pueden mantener con más facilidad que las demás estructuras.

Los tres pasos de la normalización

Empezando con una vista de usuario o con un almacén de datos desarrollado para un diccionario de datos (vea el capítulo 8), el analista normaliza una estructura de datos en tres pasos, como se muestra en la figura 13.10. Cada paso implica un procedimiento importante que simplifique la estructura de datos.

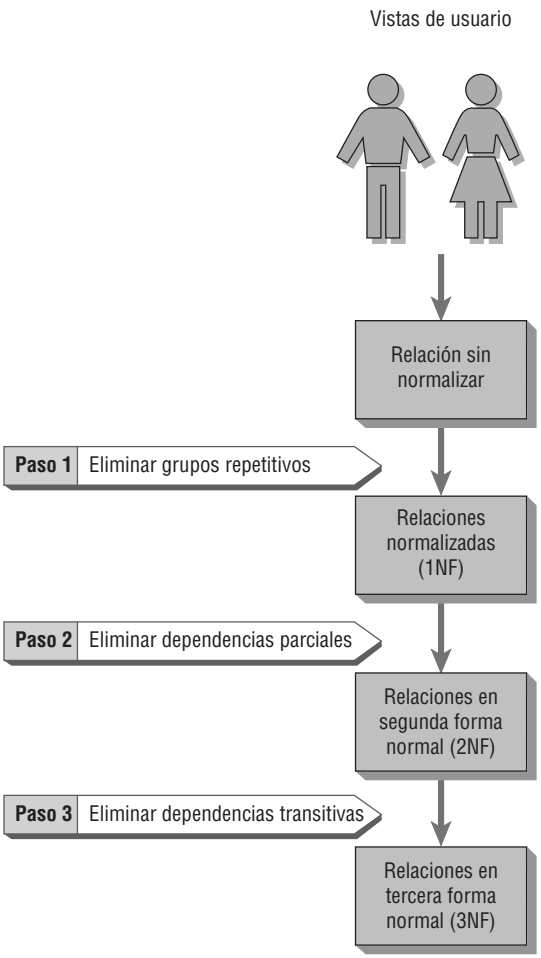
Es probable que la relación que se deriva de la vista de usuario o almacén de datos esté desnormalizada. El primer paso del proceso incluye eliminar todos los grupos repetitivos e identificar la clave primaria. Para ello, la relación necesita descomponerse en dos o más relaciones. En este punto, las relaciones tal vez ya estén en la tercera forma normal, pero es probable que se necesiten más pasos para transformar las relaciones a la tercera forma normal.

El segundo paso asegura que todos los atributos que no sean claves dependan por completo de la clave primaria. Se eliminan todas las dependencias parciales y se colocan en otra relación.

El tercer paso elimina las dependencias transitivas. En una dependencia transitiva los atributos que no son claves dependen de otros atributos que tampoco son claves.

FIGURA 13.10

La normalización de una relación se lleva a cabo en tres pasos principales.



Un ejemplo de normalización

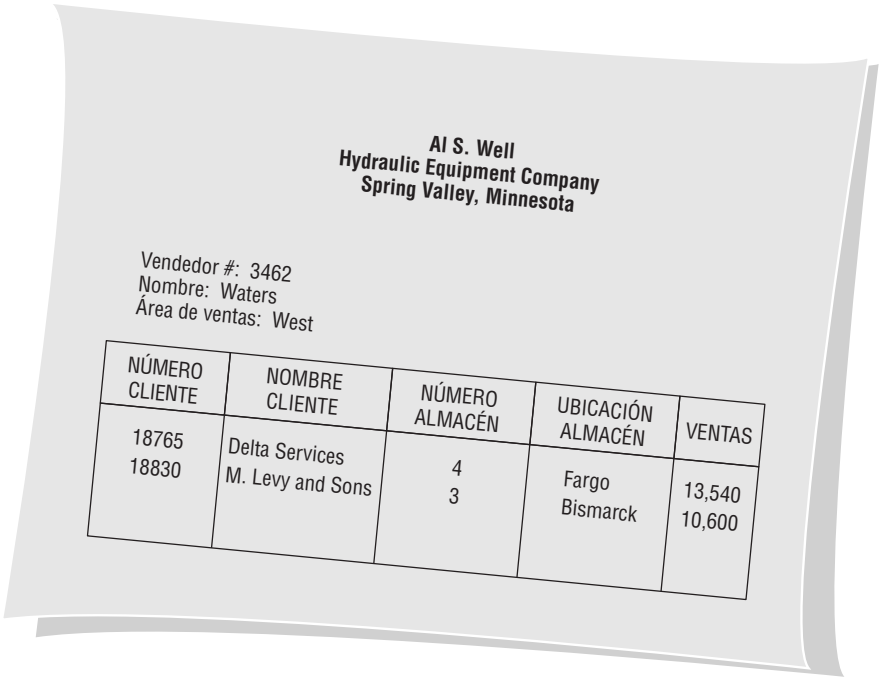
La figura 13.11 es una vista de usuario para la empresa Al S. Well Hydraulic Equipment Company. El informe muestra los siguientes elementos: 1) NUMERO-VENDEDOR, 2) NOMBRE-VENDEDOR y 3) AREA-VENTAS. El cuerpo del informe muestra el 4) NUMERO-CLIENTE y el 5) NOMBRE-CLIENTE. A continuación está el 6) NUMERO-ALMACEN que dará servicio al cliente, seguido de la 7) UBICACION-ALMACEN, que es la ciudad en la que se encuentra la empresa. La información final contenida en la vista de usuario es 8) MONTO-VENTAS. Las filas (una para cada cliente) en la vista de usuario muestran que los elementos 4 a 8 forman un grupo repetido.

Si el analista estuviera usando una metodología de flujo de datos/diccionario de datos, la misma información en la vista de usuario aparecería en una estructura de datos. La figura 13.12 muestra cómo aparecería la estructura de datos en la etapa del análisis correspondiente al diccionario de datos. El grupo repetitivo también se indica en la estructura de datos mediante un asterisco (*) y el uso de sangría.

Antes de continuar, observe las asociaciones de datos de los elementos de datos de la figura 13.13. A este tipo de ilustración se le denomina diagrama de burbuja o diagrama de modelo de datos. Cada entidad está encerrada en una elipse; se utilizan flechas para mostrar las relaciones. Aunque es posible dibujar estas relaciones con un diagrama E-R, algunas veces es más fácil usar el diagrama de burbuja más simple para modelar los datos.

En este ejemplo hay sólo un NUMERO-VENDEDOR asignado a cada NOMBRE-VENDEDOR y esa persona sólo cubrirá un AREA-VENTAS, pero cada AREA-VENTAS se puede asignar a muchos vendedores: de aquí que se utilice la notación de doble flecha de AREA-VENTAS a NUMERO-VENDEDOR. Para cada NUMERO-VENDEDOR puede haber muchos NUMERO(s)-VENDEDOR(es).

También habría una correspondencia de uno a uno entre NUMERO-CLIENTE y NOMBRE-CLIENTE; lo mismo se aplica para NUMERO-ALMACEN y UBICACION-ALMACEN. NUMERO-CLIENTE(s) sólo tendrá un NUMERO-ALMACEN y una UBICACION-ALMACEN, pero cada NUMERO-ALMACEN o UBICACION-ALMACEN puede atender a muchos NUMERO(s)-CLIENTE(s). Por último, para determinar



Al S. Well
Hydraulic Equipment Company
Spring Valley, Minnesota

Vendedor #: 3462
Nombre: Waters
Área de ventas: West

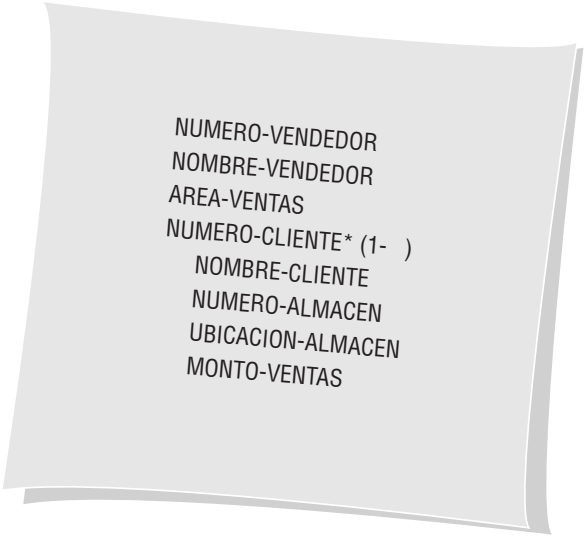
NÚMERO CLIENTE	NOMBRE CLIENTE	NÚMERO ALMACÉN	UBICACIÓN ALMACÉN	VENTAS
18765 18830	Delta Services M. Levy and Sons	4 3	Fargo Bismarck	13,540 10,600

FIGURA 13.11
Un reporte de usuario para la empresa Al S. Well Hydraulic Equipment Company.

el MONTO-VENTAS de las llamadas de un vendedor a una empresa específica, es necesario conocer tanto el NUMERO-VENDEDOR como el NUMERO-CLIENTE.

El principal objetivo del proceso de normalización es simplificar todos los elementos de datos complejos que se encuentran con frecuencia en las vistas de usuario. Por ejemplo, si el analista tomara la vista de usuario antes descrita y tratara de crear una tabla relacional con base en ella, su apariencia sería como en la figura 13.14. Como esta relación se basa en nuestra vista de usuario inicial, nos referimos a ella como INFORME-VENTAS.

INFORME-VENTAS es una relación sin normalizar, ya que tiene grupos repetidos. También es importante observar que un solo atributo tal como NUMERO-VENDEDOR no puede servir como la clave. La razón es clara si examinamos las relaciones entre NUMERO-VENDEDOR y los otros atributos en la figura 13.15. Aunque hay una correspondencia de uno a uno entre NUMERO-VENDEDOR y dos atributos (NOMBRE-VENDEDOR y AREA-VENTAS), existe una relación de uno a muchos entre NUMERO-VENDEDOR y los otros cinco atribu-

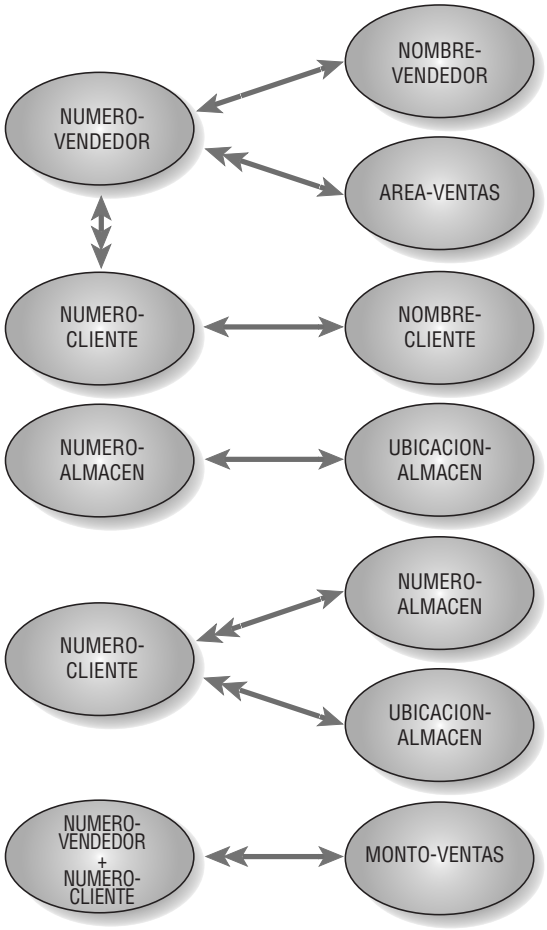


NUMERO-VENDEDOR
NOMBRE-VENDEDOR
AREA-VENTAS
NUMERO-CLIENTE* (1-)
NOMBRE-CLIENTE
NUMERO-ALMACEN
UBICACION-ALMACEN
MONTO-VENTAS

FIGURA 13.12
Para el analista sería útil una estructura de datos (de un diccionario de datos) al desarrollar una base de datos.

FIGURA 13.13

Algunas veces dibujar diagramas de modelos de datos para las asociaciones de datos ayudan a los analistas a apreciar la complejidad del almacenamiento de datos.



tos (NUMERO-CLIENTE, NOMBRE-CLIENTE, NUMERO-ALMACEN, UBICACION-ALMACEN y MONTO-VENTAS).

Podemos expresar el INFORME-VENTAS mediante la siguiente notación abreviada:

VENDEDOR-CLIENTE (NUMERO-VENDEDOR,
NOMBRE-VENDEDOR, AREA
DE VENTAS
NUMERO-CLIENTE,
NOMBRE-CLIENTE,
NUMERO-ALMACEN,
UBICACION-ALMACEN,
MONTO-VENTAS)

donde el conjunto interno de paréntesis representa el grupo repetido.

FIGURA 13.14

Si los datos se listaran en una tabla sin normalizar, podría haber grupos repetitivos.

NUMERO VENDEDOR	NOMBRE VENDEDOR	AREA VENTAS	NUMERO CLIENTE	NOMBRE CLIENTE	NUMERO ALMACEN	UBICACION ALMACEN	MONTO VENTAS
3462	Waters	West	18765	Delta Systems	4	Fargo	13540
			18830	A. Levy and Sons	3	Bismarck	10600
			19242	Ranier Company	3	Bismarck	9700
3593	Dryne	East	18841	R. W. Flood Inc.	2	Superior	11560
			18899	Seward Systems	2	Superior	2590
			19565	Stodola's Inc.	1	Plymouth	8800
etc.							

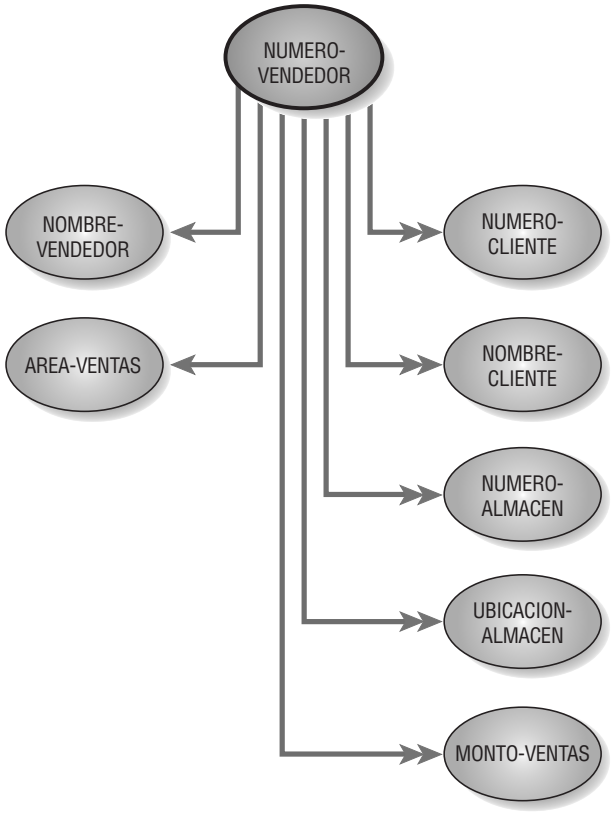


FIGURA 13.15
Un diagrama de modelo de datos muestra que en la relación sin normalizar, NUMERO-VENDEDOR tiene una asociación de uno a muchos con algunos atributos.

PRIMERA FORMA NORMAL (1NF) El primer paso para normalizar una relación es eliminar los grupos repetidos. En nuestro ejemplo, la relación sin normalizar INFORME-VENTAS se descompondrá en dos relaciones separadas. Estas nuevas relaciones se llamarán VENDEDOR y VENDEDOR-CLIENTE.

La figura 13.16 muestra cómo se normaliza la relación INFORME-VENTAS al separar la relación en dos nuevas relaciones. Cabe mencionar que la relación VENDEDOR contiene la clave primaria NUMERO-VENDEDOR y todos los atributos que no se repetían (NOMBRE-VENDEDOR y AREA-VENTAS).

La segunda relación, VENDEDOR-CLIENTE, contiene la clave primaria de la relación VENDEDOR (la clave primaria de VENDEDOR es NUMERO-VENDEDOR) así como los atributos que formaban parte del grupo repetido (NUMERO-CLIENTE, NOMBRE-CLIENTE, NUMERO-ALMACEN, UBICACION-ALMACEN y MONTO-VENTAS). Sin embargo, conocer el NUMERO-VENDEDOR no implica que conoceremos automáticamente el NOMBRE-CLIENTE, MONTO-VENTAS, UBICACION-ALMACEN, etcétera. En esta relación debemos usar una clave concatenada (tanto NUMERO-VENDEDOR como NUMERO-CLIENTE) para acceder al resto de la información. Es posible escribir las relaciones en notación abreviada, como se muestra a continuación:

VENDEDOR (NUMERO VENDEDOR,
NOMBRE-VENDEDOR, AREA VENTAS)

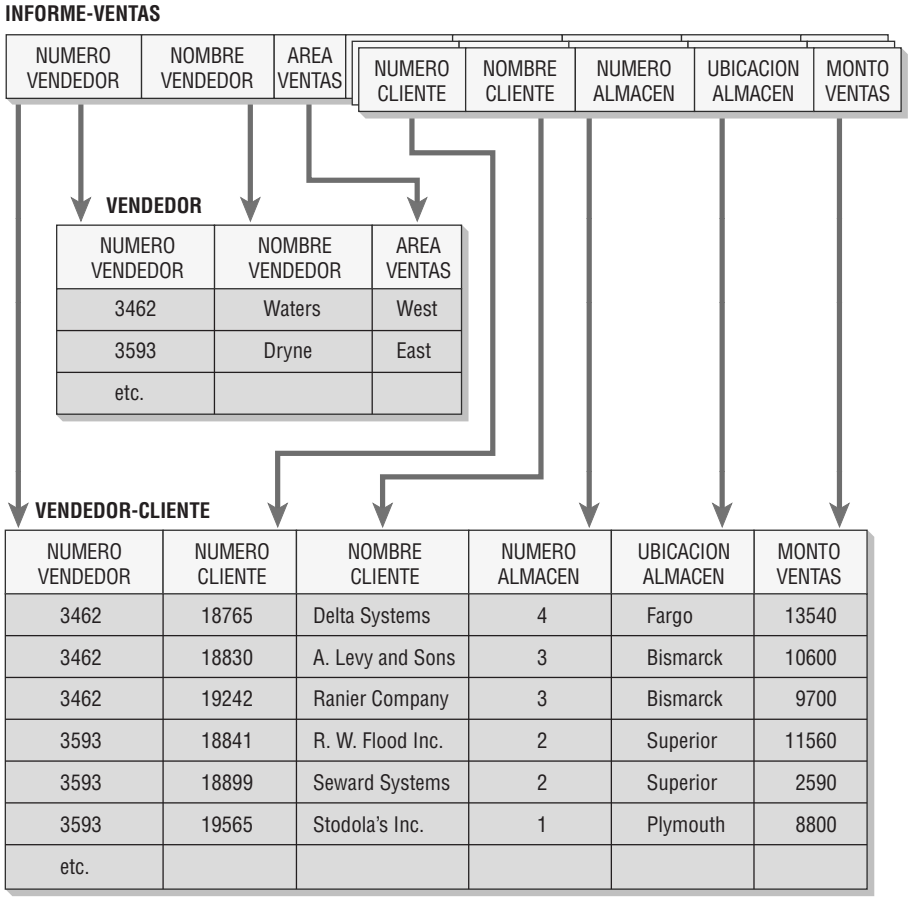
y

VENDEDOR-CLIENTE (NUMERO-VENDEDOR,
NUMERO-CLIENTE,
NOMBRE-CLIENTE,
NUMERO-ALMACEN,
UBICACION-ALMACEN,
MONTO-VENTAS)

La relación VENDEDOR-CLIENTE es una relación en primera forma normal, pero no está en su forma ideal. Surgen problemas debido a que algunos de los atributos no son funcionalmente dependientes de la clave prima-

FIGURA 13.16

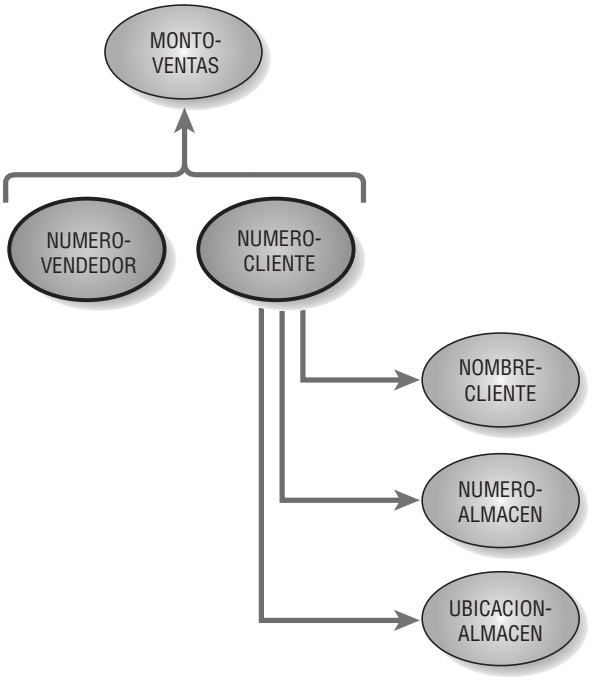
La relación original sin normalizar INFORME-VENTAS se separa en dos relaciones, VENDEDOR (3NF) y VENDEDOR-CLIENTE (1NF).



ria (es decir, NUMERO-VENDEDOR, NUMERO-CLIENTE). En otras palabras, algunos de los atributos que no son claves dependen sólo de NUMERO CLIENTE y no de la clave concatenada. El diagrama de modelo de datos en la figura 13.17 muestra que MONTO-VENTAS depende de NUMERO-VENDEDOR y de NUMERO-CLIENTE, pero los otros tres atributos dependen sólo de NUMERO-CLIENTE.

FIGURA 13.17

Un diagrama de modelo de datos muestra que tres atributos son dependientes de NOMBRE-CLIENTE, por lo que la relación aún no está normalizada. Tanto NUMERO-VENDEDOR como NUMERO-CLIENTE tienen que buscar el MONTO-VENTAS.



SEGUNDA FORMA NORMAL (2NF) En la segunda forma normal, todos los atributos serán funcionalmente dependientes de la clave primaria. Por lo tanto, el siguiente paso es eliminar todos los atributos parcialmente dependientes y colocarlos en otra relación. La figura 13.18 muestra cómo se divide la relación VENDEDOR-CLIENTE en dos nuevas relaciones: VENTAS y CLIENTE-ALMACEN. Estas relaciones también se pueden expresar de la siguiente manera:

VENTAS (NUMERO-VENDEDOR, NUMERO-CLIENTE, MONTO-VENTAS

y

ALMACEN CLIENTE (NUMERO-CLIENTE, NOMBRE-CLIENTE, NUMERO-ALMACEN, UBICACION-ALMACEN)

La relación CLIENTE-ALMACEN está en la segunda forma normal. Todavía se puede simplificar más debido a que hay dependencias adicionales en la relación. Algunos de los atributos que no son claves dependen no sólo de la clave primaria, sino también de un atributo que no es clave. Esta dependencia se denomina dependencia transitiva.

La figura 13.19 muestra las dependencias en la relación CLIENTE-ALMACEN. Para que la relación asuma la segunda forma normal, todos los atributos deben ser dependientes de la clave primaria NUMERO-CLIENTE, como se muestra en el diagrama. Sin embargo, UBICACION-ALMACEN también depende obviamente de NUMERO-ALMACEN. Para simplificar esta relación se requiere otro paso.

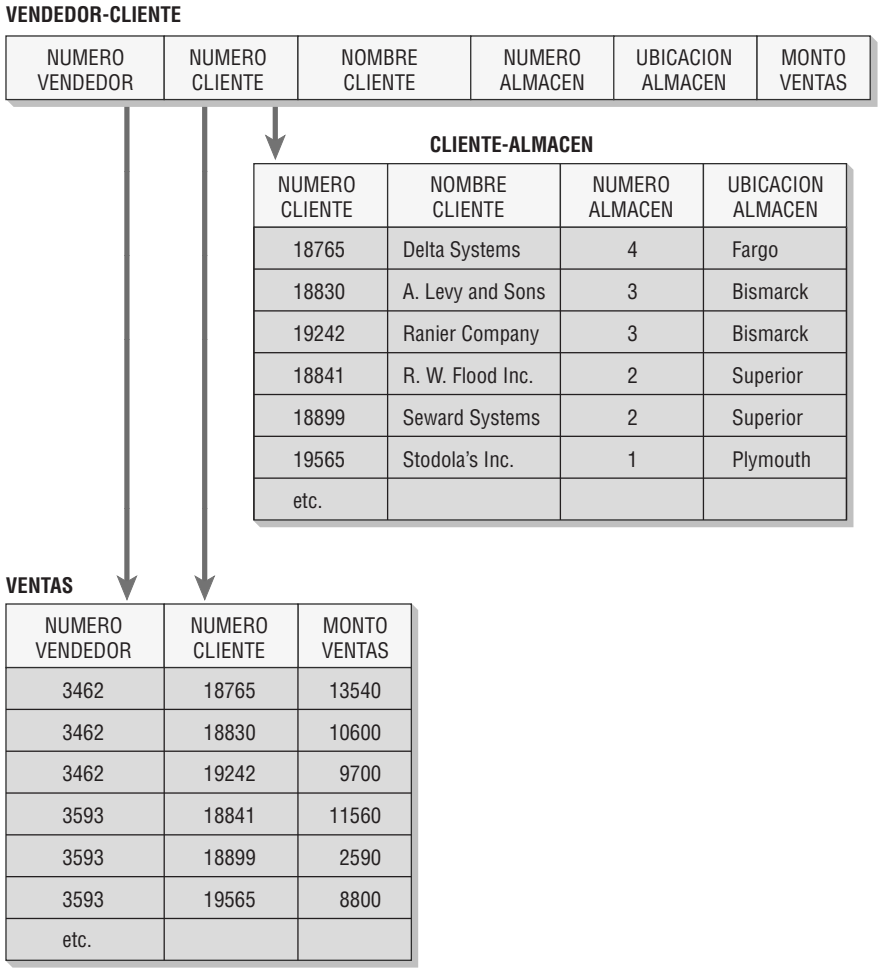
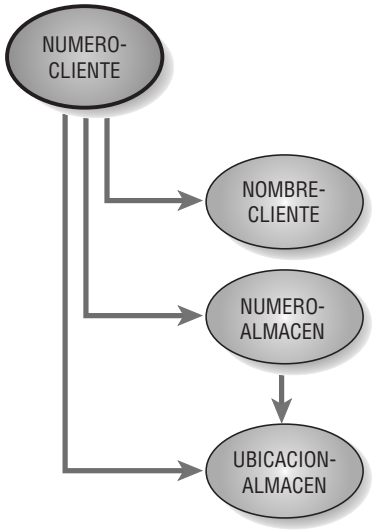


FIGURA 13.18
La relación VENDEDOR-CLIENTE se separa en una relación llamada CLIENTE-ALMACEN (2NF) y una relación llamada VENTAS (1NF).

FIGURA 13.19

Un diagrama de modelo de datos muestra que existe una dependencia transitiva entre NUMERO-ALMACEN y UBICACION-ALMACEN.



TERCERA FORMA NORMAL (3NF) Una relación normalizada está en la tercera forma normal si todos los atributos que no son claves son completa y funcionalmente dependientes de la clave primaria y no hay dependencias transitivas (no claves). Al igual que en los pasos anteriores, es posible descomponer la relación CLIENTE-ALMACEN en dos, como se muestra en la figura 13.20.

Las dos nuevas relaciones se llaman CLIENTE y ALMACEN, y se pueden escribir de la siguiente manera:

CLIENTE (NUMERO-CLIENTE, NOMBRE-CLIENTE, NUMERO-ALMACEN)

y

ALMACEN (NUMERO-ALMACEN, UBICACION-ALMACEN)

La clave primaria para la relación CLIENTE es NUMERO-CLIENTE y la clave primaria para la relación ALMACEN es NUMERO-ALMACEN.

Además de estas claves primarias, podemos identificar a NUMERO-ALMACEN como una clave externa en la relación CLIENTE. Una clave externa es cualquier atributo que no sea clave en una relación, pero que sea

FIGURA 13.20

La relación CLIENTE-ALMACEN se separa en dos relaciones llamadas CLIENTES (1NF) y ALMACEN (1NF).

CLIENTE-ALMACEN

NUMERO CLIENTE	NOMBRE CLIENTE	NUMERO ALMACEN	UBICACION ALMACEN

CLIENTE

NUMERO CLIENTE	NOMBRE CLIENTE	NUMERO ALMACEN
18765	Delta Systems	4
18830	A. Levy and Sons	3
19242	Ranier Company	3
18841	R. W. Flood Inc.	2
18899	Seward Systems	2
19565	Stodola's Inc.	1
etc.		

ALMACEN

NUMERO ALMACEN	UBICACION ALMACEN
4	Fargo
3	Bismarck
2	Superior
1	Plymouth
etc.	

clave primaria en otra. Para designar a NUMERO-ALMACEN como clave externa en la notación anterior y en las figuras, la pondremos subrayada: _____.

Finalmente, la relación original sin normalizar INFORME-VENTAS se ha transformado en cuatro relaciones 3NF. Al revisar las relaciones que se muestran en la figura 13.21, podemos ver que la relación individual INFORME-VENTAS se transformó en las siguientes cuatro relaciones:

VENDEDOR (NUMERO-VENDEDOR, NOMBRE-VENDEDOR, AREA-VENTAS)

VENTAS (NUMERO-VENDEDOR, NUMERO-CLIENTE, MONTO-VENTAS)

CLIENTE (NUMERO-CLIENTE, NOMBRE-CLIENTE, NUMERO-ALMACEN)

y

ALMACEN (NUMERO-ALMACEN, UBICACION-ALMACEN)

La tercera forma normal es adecuada para la mayoría de los problemas de diseño de bases de datos. La simplificación que se obtiene al transformar una relación sin normalizar en un conjunto de relaciones en 3NF constituye un enorme beneficio en cuanto al tiempo que se requiere para insertar, eliminar y actualizar información en la base de datos.

En la figura 13.22 se muestra un diagrama E-R para la base de datos. Un VENDEDOR atiende a muchos CLIENTE(s), quienes generan VENTAS y reciben sus artículos de un ALMACEN (el ALMACEN más cercano a su ubicación). Tome su tiempo para observar cómo se relacionan las entidades y atributos con la base de datos.

Uso del diagrama entidad-relación para determinar las claves de los registros

Podemos usar el diagrama E-R para determinar las claves requeridas para un registro o una relación de la base de datos. El primer paso es construir el diagrama E-R e identificar una clave única (primaria) para cada entidad

VENDEDOR

NUMERO VENDEDOR	NOMBRE VENDEDOR	AREA VENTAS
3462	Waters	Oeste
3593	Dryne	Este
etc.		

VENTAS

NUMERO VENDEDOR	NUMERO CLIENTE	MONTO VENTAS
3462	18765	13540
3462	18830	10600
3462	19242	9700
3593	18841	11560
3593	18899	2590
3593	19565	8800
etc.		

CLIENTE

NUMERO CLIENTE	NOMBRE CLIENTE	NOMBRE ALMACEN
18765	Delta Systems	4
18830	A. Levy and Sons	3
19242	Ranier Company	3
18841	R. W. Flood Inc.	2
18899	Seward Systems	2
19565	Stodola's Inc.	1
etc.		

ALMACEN

NOMBRE ALMACEN	UBICACION ALMACEN
4	Fargo
3	Bismarck
2	Superior
1	Plymouth
etc.	

FIGURA 13.21

La base de datos completa consiste en cuatro relaciones en 1NF llamadas VENDEDOR, VENTAS, CLIENTE y ALMACEN.

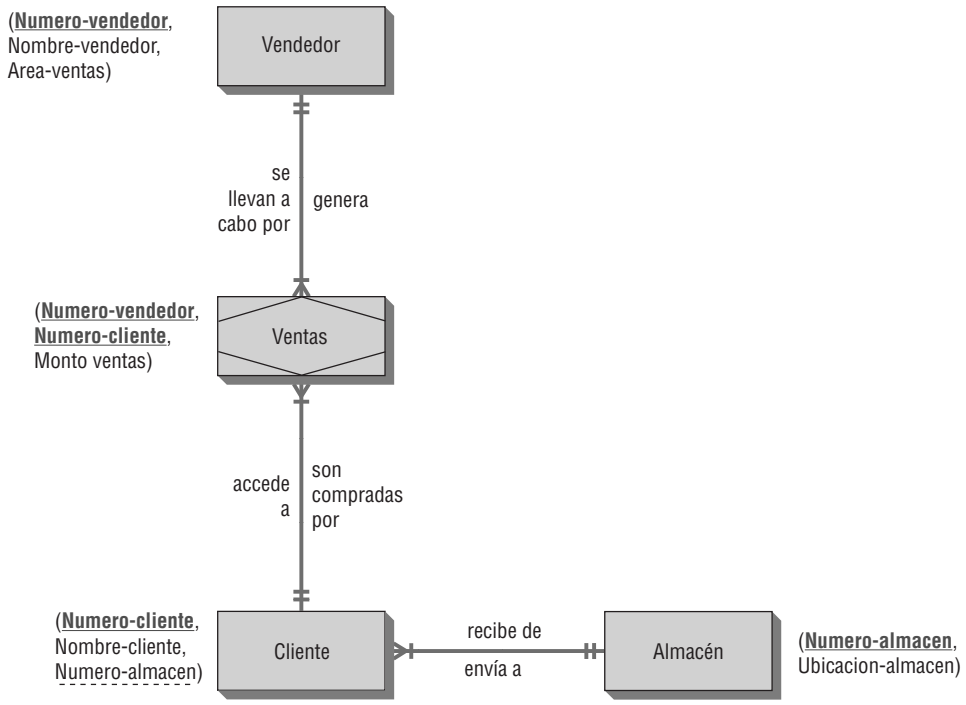


FIGURA 13.22
Un diagrama entidad-relación para la base de datos de la empresa Al S. Well Hydraulic Company.

de datos. La figura 13.23 muestra un diagrama E-R para un sistema de pedidos de clientes. Hay tres entidades de datos: CLIENTE, con la clave primaria NUMERO-CLIENTE; PEDIDO, con la clave primaria NUMERO-PEDIDO, y ARTICULO, con NUMERO-ARTICULO como clave primaria. Un CLIENTE puede colocar muchos pedidos, pero cada PEDIDO puede ser colocado sólo por un CLIENTE, por lo que la relación es de uno a muchos. Cada PEDIDO puede contener muchos ARTICULO(s), y cada ARTICULO puede estar contenido en muchos PEDIDO(s), por lo que la relación PEDIDO-ARTICULO es de muchos a muchos.

Sin embargo, una clave externa es un campo de datos de cierto archivo que es la clave primaria de otro archivo maestro. Por ejemplo, un NUMERO-DEPARTAMENTO que indica que la especialidad de un estudiante puede existir en la tabla MAESTRA DE ESTUDIANTES. NUMERO-DEPARTAMENTO también podría ser la clave única para la tabla MAESTRA DE DEPARTAMENTOS.

Relación de uno a muchos

Una relación de uno a muchos es el tipo más común de relación, ya que todas las relaciones de muchos a muchos se deben descomponer en dos relaciones de uno a muchos. Cuando ocurra una relación de uno a muchos, coloque la clave primaria de la tabla en el extremo de uno de la relación como una clave externa para la tabla en el extremo de muchos de la relación. Por ejemplo, como un cliente puede tener muchos pedidos, coloque el número de cliente en el registro de pedidos.

Es fácil construir el diseño de páginas Web, pantallas o informes que incluyan información de sólo un registro de la relación de muchos, junto con la información del extremo de uno de la relación. La pantalla no tendrá información repetida. Un ejemplo es la consulta de un pedido mediante el uso de un número de pedido para buscar un pedido individual. Como el pedido es para un cliente, el resultado consistiría en los campos del pedido y un solo cliente.

Es más complicado diseñar el inverso, ya que la tabla en el extremo de uno de la relación puede tener muchos registros para el extremo de muchos. Éstos se implementan en varias formas. Para una pantalla de visualización simple, la información del extremo de uno se muestra con un número repetitivo de grupos de información del extremo de muchos de la relación. En Microsoft Access esto podría ser un formulario con un subformulario,

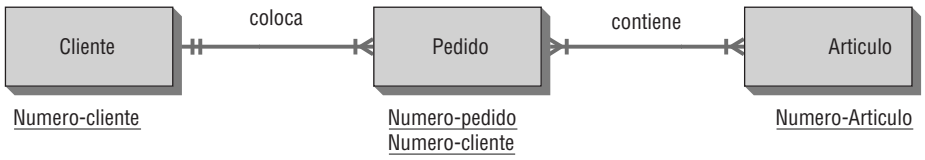


FIGURA 13.23
Un diagrama entidad-relación para los pedidos de los clientes.

como un cliente con un subformulario de todos los pedidos del cliente. Si hubiera un extenso número de registros del extremo de muchos, aparecerían barras de desplazamiento.

En situaciones simples, la relación también se podría implementar mediante el uso de una lista desplegable, donde cada registro del extremo de muchos se convertiría en una entrada en el extremo de uno; un ejemplo es la pantalla de un automóvil junto con una lista desplegable que contenga todos los modelos de ese automóvil. Al diseñar sitios Web, la información del extremo de uno podría estar en la parte superior de la página, con varios grupos de datos debajo de ésta o varios vínculos a los datos. Un ejemplo es un tema en un motor de búsqueda que produce como resultado muchos vínculos coincidentes, o un género de música y muchos artistas que coinciden con ese género.

Relación de muchos a muchos

Cuando la relación es de muchos a muchos se requieren tres tablas: una para cada entidad de datos y una para la relación. Las entidades PEDIDO y ARTICULO en nuestro ejemplo tienen una relación de muchos a muchos. La clave primaria de cada entidad de datos se almacena como una clave externa de la tabla relacional. Esta tabla relacional puede contener sólo las claves primarias para cada entidad de datos, o puede contener datos adicionales, como la calificación recibida por un curso o la cantidad de un artículo ordenado. Consulte la disposición de la tabla que se muestra en la figura 13.24. La tabla ARTICULO PEDIDO contiene información sobre los artículos que contiene cada pedido y provee un vínculo entre la tabla PEDIDO y la tabla MAESTRA DE ARTICULOS.

La tabla de relación se debe indexar en cada clave externa (una para cada tabla en la relación) y puede tener una clave primaria que consista en una combinación de las dos claves externas. A menudo las empresas usan una clave única, como un número de secuencia, como la clave primaria para la tabla relacional. Para buscar muchos registros de una segunda tabla dada la primera, lea directamente la tabla relacional en busca de la clave deseada. Localice el registro que coincida en la segunda tabla de *muchos*. Siga iterando por la tabla relacional hasta que ya no encuentre la clave deseada. Por ejemplo, para buscar los registros en la tabla MAESTRA DE ARTICULOS para un registro específico en la tabla PEDIDO, lea directamente la tabla PEDIDO-ARTICULO usando el NUMERO-PEDIDO como índice. Los registros se secuencian en forma lógica con base en los datos del índice, por lo que todos los registros para el mismo NUMERO-PEDIDO se agrupan entre sí. Para cada registro de PEDIDO-ARTICULO que coincida con el NUMERO-PEDIDO deseado, lea directamente la tabla MAESTRA DE ARTICULOS usando el NUMERO-ARTICULO como índice.

La lógica es la misma para la situación inversa, como buscar en todos los pedidos un artículo pendiente que se haya recibido. Use el NUMERO-ARTICULO deseado para leer la tabla PEDIDO-ARTICULO directamente. El índice de PEDIDO-ARTICULO se establece en el NUMERO-ARTICULO. Para todos los registros de PEDIDO-ARTICULO que coincidan, use el NUMERO-PEDIDO para leer la tabla PEDIDO directamente. Por último, lea la tabla MAESTRA DE CLIENTES directamente para obtener el NOMBRE-CLIENTE y DIRECCION mediante el uso del NUMERO-CLIENTE en la tabla PEDIDO.

Las tablas relacionales pueden tener relaciones con más tablas en la base de datos además de las dos con las que se conectan directamente. Por ejemplo, podría haber una tabla relacional llamada Clase o Seccion para vincular estudiantes y cursos, ya que cada estudiante puede tomar muchos cursos y cada curso puede tener muchos estudiantes. La tabla Seccion puede tener una relación con el Librotexto o Instructor para esa sección.

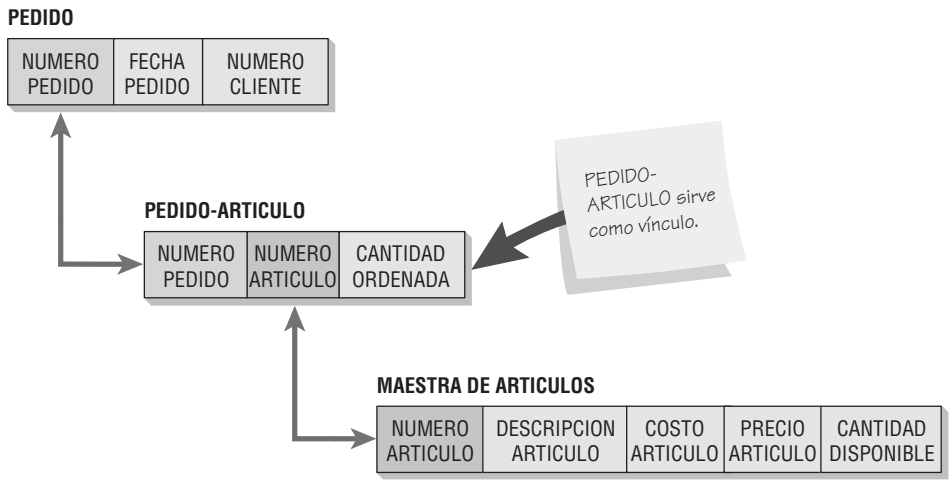


FIGURA 13.24

Cuando la relación es de muchos a muchos se necesitan tres archivos.

LINEAMIENTOS PARA EL DISEÑO DE RELACIONES DE ARCHIVOS MAESTROS/BASES DE DATOS

Hay que tener en consideración los siguientes lineamientos al diseñar relaciones de archivos maestros o bases de datos:

1. Cada entidad de datos separada debe crear una tabla maestra de base de datos. No debe combinar dos entidades distintas en un archivo. Por ejemplo, los artículos se compran de los distribuidores. La tabla MAESTRA DE ARTICULOS debe contener sólo información de los artículos y la tabla MAESTRA DE DISTRIBUIDORES debe contener sólo información de los distribuidores.
2. Un campo de datos específico debe existir sólo en una tabla maestra. Por ejemplo, el NOMBRE CLIENTE podría existir sólo en la tabla MAESTRA DE CLIENTES, no en la tabla PEDIDO o en ninguna otra tabla maestra. Las excepciones a este lineamiento son los campos claves o índices, que pueden estar en todas las tablas que sea necesario. Si un informe o pantalla necesita información de muchas tablas, los índices deben proveer los vínculos para obtener los registros requeridos.
3. Cada relación de tabla maestra o base de datos debe tener programas para crear (**C**), leer (**R**), actualizar (**U**) y eliminar (**D**) registros (lo cual se abrevia como CRUD). En teoría sólo un programa debería agregar nuevos registros y sólo un programa debería eliminar registros especificados. Sin embargo, muchos programas pueden ser responsables de modificar los campos de datos en el curso de las actividades de negocios normales. Por ejemplo, un archivo MAESTRO DE CLIENTES podría tener un campo SALDO ACTUAL que aumente en base al TOTAL PEDIDO en el programa de procesamiento de pedidos y disminuya con base en un MONTO PAGO o a un MONTO DEVUELTO que provengan de dos programas adicionales.

Restricciones de integridad

Las restricciones de integridad son reglas que gobiernan las acciones de modificar y eliminar registros, y que ayudan a mantener la precisión de los datos en la base de datos. Podemos aplicar tres tipos de restricciones de integridad a una base de datos:

1. Integridad de entidad.
2. Integridad referencial.
3. Integridad de dominio.

Las restricciones de integridad de entidad son reglas que gobiernan la composición de claves primarias. La clave primaria no puede tener un valor nulo; si la clave primaria es una clave compuesta, ninguno de los campos que componen la clave puede contener un valor nulo. Algunas bases de datos nos permiten definir una restricción única o una clave única. Esta clave única sólo identifica un registro que no es una clave primaria. La diferencia ente una clave única y una clave primaria es que una clave única puede contener un valor nulo.

La integridad referencial gobierna la naturaleza de los registros en una relación de uno a muchos. La tabla que está conectada en el extremo de uno de la relación se llama padre. La tabla conectada al extremo de muchos de la relación se llama tabla hija. Integridad referencial significa que todas las claves externas en la tabla de muchos (la tabla hija) deben tener un registro que coincida en la tabla padre. Por ende, no podemos agregar un registro en la tabla hija (muchos) sin un registro que coincida en la tabla padre.

Una segunda implicación es que no podemos cambiar una clave primaria que tenga registros que coincidan en la tabla hija. Si pudiéramos cambiar el registro padre, el resultado sería un registro hijo que tendría un distinto registro padre o un registro huérfano, o un registro hijo sin un registro padre. Algunos ejemplos son un registro CALIFICACION para un estudiante que no estuviera en la tabla MAESTRA DE ESTUDIANTES y un registro PEDIDO para un NUMERO CLIENTE inexistente. La última implicación de la integridad referencial es que no podemos eliminar un registro padre que tenga registros hijos. Eso también produciría los registros huérfanos que mencionamos antes.

La integridad referencial se implementa de dos formas. La primera es tener una base de datos restringida, en la cual el sistema pueda actualizar o eliminar un registro padre sólo si no hay registros hijos que coincidan. Una base de datos en cascada eliminará o actualizará todos los registros hijos cuando se elimine o modifique un registro padre (el padre desencadena los cambios).

Es mejor una relación restringida al eliminar registros. ¡No sería conveniente eliminar un registro de un cliente y eliminar también todas las facturas pendientes! La metodología en cascada es mejor a la hora de modificar registros. Si se modifica la clave primaria del registro de un estudiante, se modificarían también las claves externas (el NUMERO ESTUDIANTE en la tabla MAESTRA DE CURSOS) en todos los registros de cursos para ese estudiante.

Las reglas de integridad de dominio se utilizan para validar los datos, como las comprobaciones de validación de tabla, límite, rango, etcétera. En el capítulo 15 explicaremos estas reglas con más detalle. Por lo gene-

ATRACTIVO DE LA MAC

Aunque Microsoft Word, Excel y PowerPoint están disponibles para el sistema operativo de la Mac, la única forma de ejecutar Microsoft Access es ejecutar Windows en modo de virtualización o iniciar directamente en Windows. Existen otras dos opciones para la Mac: Bento y FileMaker Pro. Bento es una base de datos personal que permite a los usuarios recopilar información en forma eficiente de la libreta de direcciones, la aplicación de calendario, Apple Mail y Microsoft Excel; después podemos agregar campos al instante para crear una base de datos personalizable.

Tal vez algunos usuarios encuentren a Bento algo limitante; para estos casos está el hermano mayor de Bento, FileMaker Pro. Éste es un programa completo de bases de datos relacionales que proporciona acceso directo a bases de datos de SQL. Su característica distintiva es que las pantallas, los formularios y los informes para acceder a la base de datos están completamente integrados con el motor de bases de datos.

FIGURA 13.MAC

Ejemplo de pantalla de Bento, una base de datos personal. Esta pantalla se utilizó con permiso de FileMaker, Inc.

ral, las reglas de dominio de integridad se almacenan en la estructura de la base de datos en una de dos formas. Las restricciones de verificación se definen a nivel de tabla y pueden hacer referencia a uno o más campos en la tabla. Un ejemplo es que la FECHA DE COMPRA siempre es menor o igual a la fecha actual. Las reglas se definen a nivel de base de datos con objetos separados y se pueden utilizar con varios campos. Un ejemplo es un valor mayor a cero que se utilice para validar varios elementos.

Anomalías

Pueden ocurrir cuatro anomalías al crear tablas de bases de datos:

1. Redundancia de datos.
2. Anomalía de inserción.
3. Anomalía de eliminación.
4. Anomalía de actualización.

La redundancia de los datos ocurre cuando se almacenan los mismos datos en más de un lugar en la base de datos (exceptuando las claves primarias que se almacenan como claves externas). Para resolver este problema hay que crear tablas que estén en 3NF.

Una anomalía de inserción ocurre cuando no se conoce toda la clave primaria y la base de datos no puede insertar un nuevo registro, el cual violaría la integridad de entidad. Este problema ocurre por lo general cuando la clave primaria es una clave compuesta que contiene varios atributos más pequeños. Para evitar una anomalía de inserción hay que utilizar un número de secuencia para la clave primaria.

Una anomalía de eliminación ocurre cuando se elimina un registro y como resultado se pierden otros datos relacionados. Un ejemplo es cuando un artículo tiene un número de distribuidor y ese artículo específico es la única referencia a cierto distribuidor. Si se elimina ese artículo, no habría referencia al registro del distribuidor.

Una anomalía de actualización se produce cuando un cambio en el valor de un atributo ocasiona que la base de datos contenga datos inconsistentes o haya que modificar varios registros. Un ejemplo es cuando cambia el nombre de una calle en una ciudad. Podríamos cambiar algunos de los nombres de las calles y otros no, o tendríamos que asegurarnos que se hayan cambiado los nombres de todas las calles. Esto puede ocurrir cuando hay dependencias transitivas y podemos prevenirlo mediante la creación de tablas que estén en 3NF (aunque en el ejemplo de la calle, tal vez los datos estén en 3NF).

USO DE LA BASE DE DATOS

Hay varios pasos que debemos llevar a cabo en orden secuencial para asegurar que la base de datos sea útil para presentar los datos.

Pasos para recuperar y presentar los datos

Hay ocho pasos en el proceso de recuperación y presentación de los datos:

1. Seleccionar una relación de la base de datos.
2. Unir las relaciones.
3. Proyectar las columnas de la relación.
4. Seleccionar filas de la relación.
5. Derivar nuevos atributos.
6. Indexar u ordenar filas.
7. Calcular totales y medidas de desempeño.
8. Presentar los datos.

Hay que llevar a cabo los pasos primero y último, pero los seis pasos intermedios son opcionales, dependiendo de la forma en que se vayan a usar los datos. La figura 13.25 es una guía visual para los pasos.

El paso final en la recuperación de los datos es la presentación. Podemos presentar los datos que se abstraen de la base de datos en muchas formas. Algunas veces se presentan en formato tabular, algunas veces en gráficos y otras como una respuesta de una sola palabra en pantalla. El diseño de la salida, como vimos en el capítulo 11, provee un análisis más detallado sobre los objetivos, formas y métodos de presentación.

DESNORMALIZACIÓN

Una de las principales razones de la normalización es organizar los datos de manera que se reduzcan los datos redundantes. Si no tenemos que almacenar los mismos datos una y otra vez, podemos ahorrar mucho espacio. Dicha organización permite al analista reducir la cantidad de almacenamiento necesario, algo que era muy importante cuando el almacenamiento era costoso.

En la última sección aprendimos que para usar datos normalizados tuvimos que progresar por una serie de pasos que involucran la unión, el ordenamiento y la síntesis. Cuando la velocidad de consultar la base de datos (es decir, hacer una pregunta y requerir una respuesta rápida) es crítica, tal vez sea importante almacenar los datos de otras formas.

La desnormalización es el proceso de tomar el modelo de datos lógico y transformarlo en un modelo físico que sea eficiente para las tareas requeridas con más frecuencia. Estas tareas pueden incluir la generación de informes, pero también pueden implicar consultas más eficientes. Las consultas complejas como el procesamiento analítico en línea (OLAP), así como los procesos de minería de datos y descubrimiento de conocimiento en bases de datos (KDD), también pueden usar bases de datos que estén desnormalizadas.

Podemos llevar a cabo la desnormalización de varias formas. La figura 13.26 muestra uno de los métodos. Primero podemos tomar una relación de muchos a muchos, como la de VENDEDOR y CLIENTE, que comparten la entidad asociativa VENTAS. Al combinar los atributos de VENDEDOR y VENTAS podemos evitar uno de los procesos de unión. Esto puede producir una cantidad considerable de duplicación de datos, pero mejora la eficiencia de las consultas sobre los patrones de ventas.

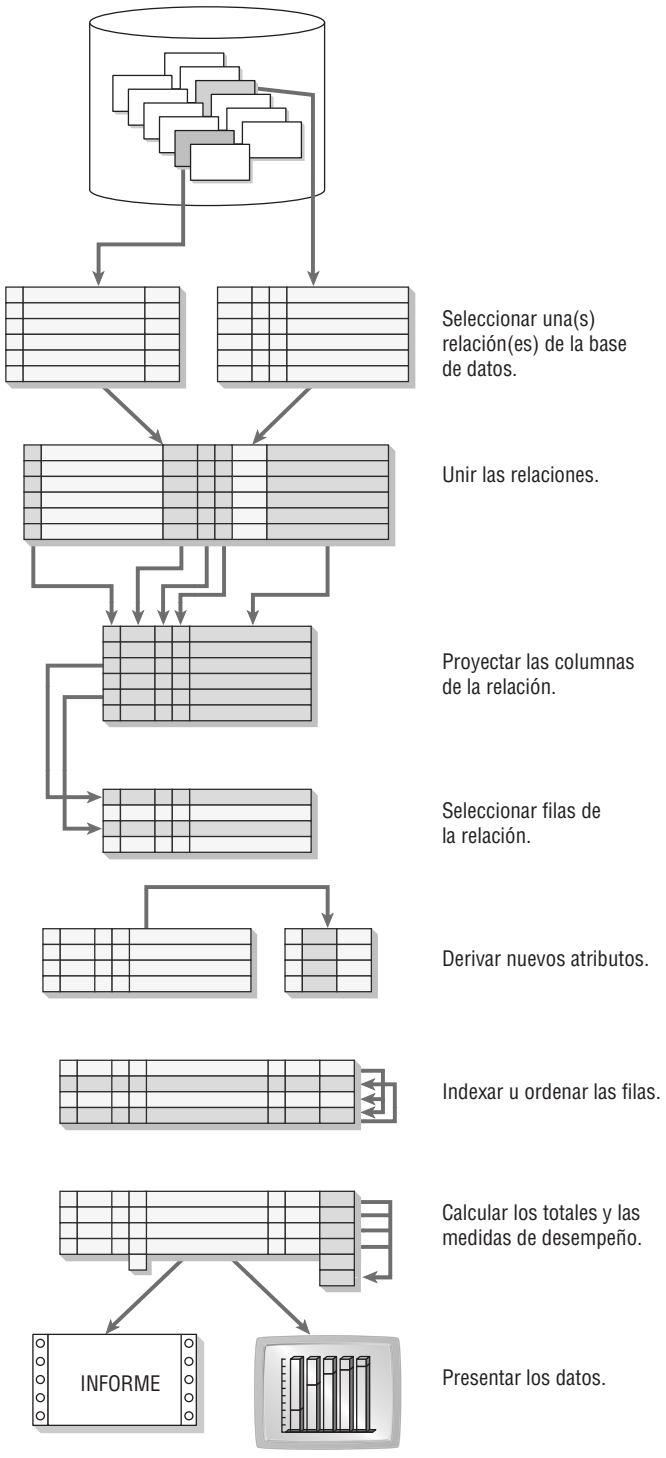


FIGURA 13.25
Los datos se obtienen y presentan en ocho pasos.

Otra de las razones de desnormalizar es evitar la referencia repetida a una tabla de búsqueda. Tal vez sea más eficiente repetir la misma información (por ejemplo: ciudad, estado y código postal) incluso si por lo general esta información se puede almacenar sólo como un código postal. Por ende, en el ejemplo de ventas podríamos combinar CLIENTE y ALMACEN.

Finalmente veremos las relaciones de uno a uno, ya que es muy probable que se combinen por razones prácticas. Si descubrimos que muchas de las consultas relacionadas con los pedidos también se interesan en saber cómo se envió el pedido, sería conveniente combinar, o desnormalizar. Así, en el ejemplo algunos de los detalles pueden aparecer tanto en DETALLES-PEDIDO como en DETALLES-ENVIO al llevar a cabo la desnormalización.

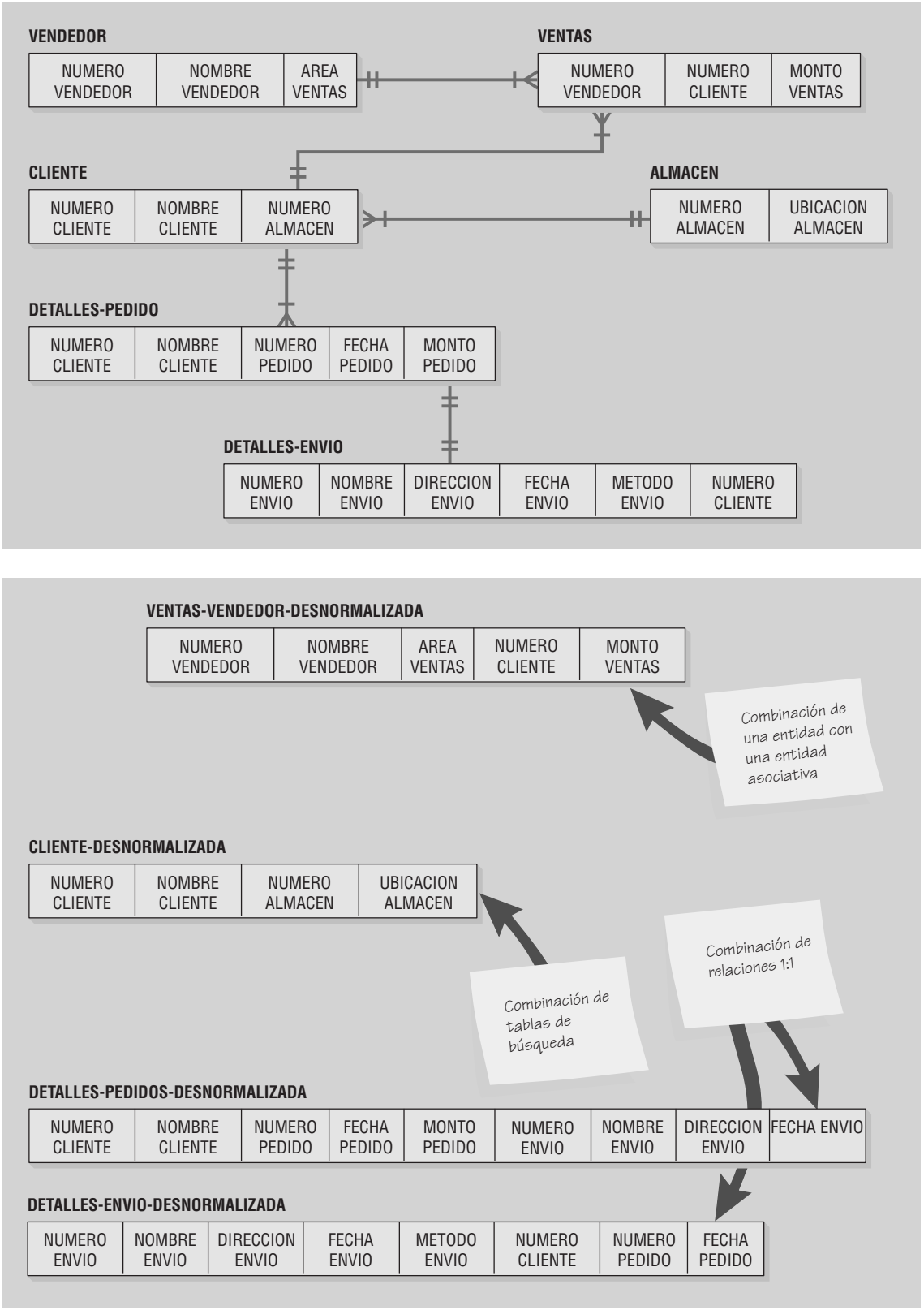


FIGURA 13.26

Tres ejemplos de desnormalización para hacer el acceso más eficiente.

ALMACENES CORPORATIVOS DE DATOS

Los almacenes corporativos de datos son distintos de las bases de datos tradicionales. El propósito de un almacén corporativo de datos es organizar la información para consultas rápidas y efectivas. Es cierto que almacenan datos desnormalizados, pero van un paso más allá. Organizan los datos con base en los temas. En la mayoría de los casos el almacén corporativo de datos es más que una base de datos que se procesa de manera que los datos se representen en maneras uniformes. Por lo tanto, los datos almacenados en los almacenes de datos provienen de distintas fuentes, por lo general bases de datos que se establecieron para distintos fines.

El concepto de almacén corporativo de datos es único. Las diferencias entre los almacenes corporativos de datos y las bases de datos tradicionales incluyen lo siguiente:

1. En un almacén corporativo de datos, los datos se organizan con base en los temas principales en vez de las transacciones individuales.
2. Por lo general, los datos en un almacén corporativo de datos se organizan como datos sintetizados en vez de los datos puros detallados que se encuentran en una base de datos orientada a transacciones.
3. Los datos de un almacén corporativo de datos cubren un marco de tiempo mucho mayor que los datos en las bases de datos tradicionales orientadas a transacciones, ya que comúnmente las consultas se relacionan con la toma de decisiones a largo plazo, en vez de los detalles diarios de las transacciones.
4. La mayoría de los almacenes corporativos de datos se organizan para consultas rápidas, mientras que las bases de datos más tradicionales están normalizadas y estructuradas de tal forma que provean un almacenamiento eficiente de la información.
5. Por lo general, los almacenes corporativos de datos están optimizados para responder consultas complejas de los gerentes y analistas conocidas como OLAP, en vez de las consultas simples que se realizan en forma repetida.
6. Los almacenes corporativos de datos permiten un fácil acceso a través de software de minería de datos (conocido como *siftware*), el cual busca patrones y puede identificar las relaciones que los humanos encargados de tomar decisiones no imaginan.
7. Los almacenes corporativos de datos no sólo incluyen una, sino varias bases de datos que se han procesado de manera que los datos del almacén corporativo se definan con uniformidad. Estas bases de datos se conocen como datos limpios.
8. Comúnmente los almacenes corporativos de datos incluyen información de fuentes externas (como un informe industrial, el archivo de la empresa en la Comisión de Bolsa y Valores, o incluso información sobre los productos de los competidores), así como datos generados para uso interno.

El proceso de crear un almacén corporativo de datos es una tarea monumental. El analista necesita recopilar datos de una variedad de fuentes y traducir esos datos a una forma común. Por ejemplo, tal vez una base de datos almacene información sobre los géneros como "Masculino" y "Femenino", otra la almacene como "M" y "F", y una tercera la almacene como "1" y "0". El analista necesita establecer un estándar y convertir todos los datos al mismo formato.

Una vez que están limpios los datos, el analista debe decidir cómo sintetizarlos. Una vez sintetizados se pierden los detalles, por lo que el analista tiene que predecir el tipo de consultas que se podrían realizar.

Después, el analista necesita diseñar el almacén corporativo de datos mediante la organización lógica (y tal vez el agrupamiento físico) de los datos por tema, lo cual requiere de mucho análisis y diseño. El analista necesita conocer mucho sobre la empresa.

El tamaño de los almacenes corporativos de datos ordinarios tiende a estar entre los 50 gigabytes hasta decenas de terabytes. Como son grandes, también son costosos. La mayoría de los almacenes corporativos de datos cuestan millones de dólares.

Procesamiento analítico en línea

El procesamiento analítico en línea (OLAP) fue introducido por E. F. Codd en 1993. Su objetivo era responder a las preguntas complejas de los encargados de tomar decisiones. Codd concluyó que el encargado de tomar decisiones tenía que analizar los datos en distintas formas. Por lo tanto, la base de datos en sí tenía que ser multidimensional. Muchas personas consideran que OLAP es el Cubo de Rubik de los datos. Se pueden ver los datos de todos los distintos lados y también se pueden manipular al girarlos o torcerlos, para que tengan sentido.

Esta metodología OLAP validó el concepto de los almacenes corporativos de datos. Después surgió la necesidad de organizar los datos en formas que permitieran consultas eficientes. Desde luego que OLAP implica el procesamiento de los datos por medio de manipulación, síntesis y cálculo, por lo que hay más de un almacén corporativo de datos involucrado.

Minería de datos

La minería de datos puede identificar patrones que un humano no puede. O el encargado de tomar decisiones no puede ver un patrón, o tal vez no puede pensar en preguntar si existe ese patrón. Los algoritmos de minería de datos buscan patrones en los almacenes corporativos de datos mediante el uso de algoritmos. La figura 13.27 ilustra el concepto de minería de datos.

OPORTUNIDAD DE CONSULTORÍA 13.2

Almacenamiento de minerales para la salud, minería de datos

Esther See, una de las empleadas de Marathon Vitamin Shop, se acerca al propietario Bill Berry para comentarle: "He observado que nuestros clientes tienen distintos hábitos. Algunos vienen con regularidad y otros son menos predecibles", dice Esther. "Cuando veo a un cliente regular, me enorgullezco de anticipar lo que el cliente va a comprar e incluso hasta puedo sugerirle otras vitaminas de su agrado. Creo que genero más ventas de esa forma. Además, el cliente está más contento".

Esther continúa: "Aunque desearía poder ayudar mejor a algunos de los clientes que vienen con menos frecuencia".

"Es una actitud muy favorable, Esther, además de que ayuda a nuestra tienda", le responde Bill. "Sé que podemos beneficiarnos de otras formas si logramos manejar mejor los patrones de los clientes. Por ejemplo, podemos estar seguros de tener un artículo en existencia".

Ester asiente y agrega: "No es sólo el tipo de vitamina de lo que estoy hablando. Algunos clientes prefieren una marca en vez de otra. No sé si depende de su nivel de ingreso o de los intereses que tengan en actividades de esparcimiento. Como los deportes, por ejemplo".

"Ya veo, Srita. See" dice Bill sonriendo, "¿tiene algo en mente?".

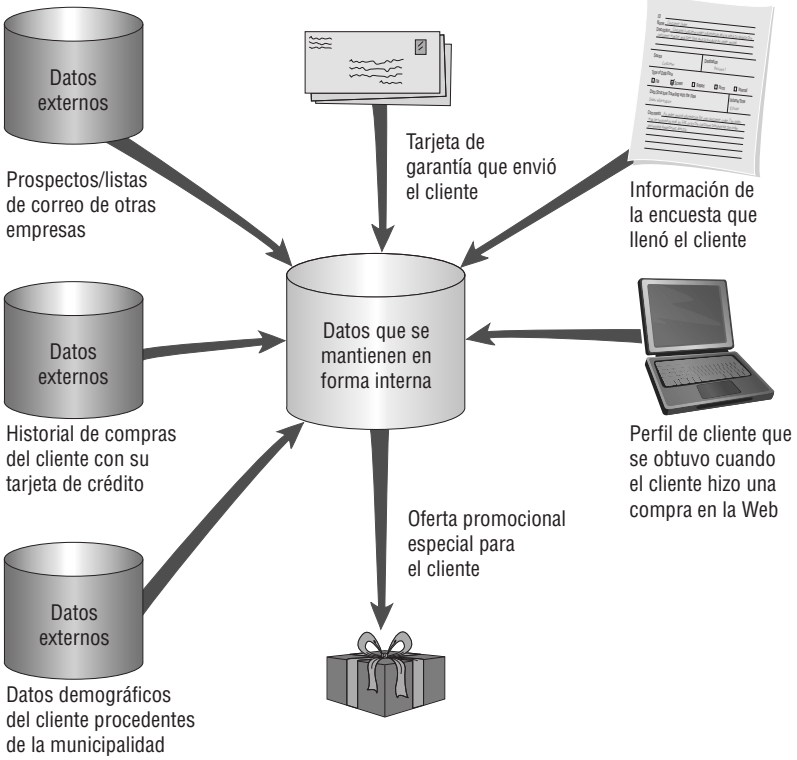
"Sí, Sr. Berry", dice en un tono más formal. "Deberíamos organizar los datos que tenemos sobre nuestros clientes mediante el concepto de un almacén corporativo de datos. Podemos fusionar los datos que tenemos con la información de otras fuentes. Así podemos buscar patrones en nuestros datos. Tal vez podamos identificar los patrones existentes y predecir nuevas tendencias".

Piense cómo organizaría un almacén corporativo de datos para Marathon Vitamin Shops. ¿Qué otras bases de datos le gustaría fusionar en el almacén de datos? ¿Qué tipo de patrones debería estar buscando Bill Berry? Identifique estos patrones por tipo (asociaciones, secuencias, agrupamiento o tendencias) y descríbalos en uno o dos párrafos.

La minería de datos se conoce también por otro nombre: descubrimiento de conocimiento en bases de datos (KDD). Algunos piensan que KDD es distinto a la minería de datos, debido a que su objetivo es ayudar a los encargados de tomar decisiones a encontrar patrones, en vez de ceder el control a un algoritmo para que los encuentre. Las ayudas disponibles para las decisiones se conocen como siftware; incluyen el análisis estadístico, los árboles de decisión, las redes neurales, los agentes inteligentes, la lógica difusa y la visualización de datos.

FIGURA 13.27

La minería de datos recopila información personal sobre los clientes, en un esfuerzo por ser más específicos al interpretar y anticiparse a sus preferencias.



OPORTUNIDAD DE CONSULTORÍA 13.3

Pérdida de prospectos

"La participación en el mercado puede ser un verdadero problema", dice Ryan Taylor, director de sistemas de marketing para una aseguradora médica de la Costa Este de Estados Unidos. "Uno de los mayores retos al que hacemos frente es cómo identificar buenos prospectos para nuestros vendedores. Con más del 50 por ciento de participación en el mercado, debemos eliminar los nombres de la mayoría de los prospectos que compramos antes de llenar nuestra base de datos de marketing; es imprescindible que lo hagamos bien, ya que ésta es parte crucial de nuestro arsenal de herramientas de información estratégica de la empresa".

Ryan explica a Chandler, uno de los miembros del equipo de analistas de sistemas que usted dirige: "Una base de datos de marketing, o MDB para abreviar, es una poderosa base de datos relacional que representa el corazón de los sistemas de marketing. Nuestra base de datos de marketing se utiliza para proveer información para todos los sistemas de marketing. En estos sistemas se incluyen las herramientas de productividad, como nuestra Automatización de las fuerzas de ventas y nuestros Sistemas de correo masivo, diseñados para ayudar a los vendedores a administrar el ciclo de ventas. También se incluyen las herramientas analíticas, como nuestros sistemas de información geográfica (GIS) o las herramientas del lenguaje gráfico de consulta (CQL), que están diseñadas para proveer soporte para las decisiones.

"Pero la principal función de una base de datos de marketing es rastrear la información sobre nuestros clientes y prospectos. En la actualidad rastreamos la información geográfica, demográfica y psicográfica o, como me gusta decirlo, dónde viven, quiénes son y cómo piensan".

"Las bases de datos de marketing más simples se pueden crear a partir de sólo tres archivos: Perfil de prospectos, Perfil de clientes e Historial de compras y pagos".

"Una vez diseñada la base de datos de marketing, el siguiente reto es decidir cómo llenarla. En la actualidad compramos nuestra información de prospectos a un distribuidor de listas. Como la estrategia de marketing de nuestra empresa se basa en el marketing en masa, compramos todos los negocios en nuestra área. Debido a este volumen, pagamos menos de diez centavos de dólar por cada prospecto. No obstante, si una empresa practica la diferenciación de productos, es probable que su base de prospectos sea más definida. Esta empresa probablemente pagaría una prima por datos más detallados que se validaron con cuidado", explica Ryan.

"Nos enfrentamos a un verdadero reto. Si tuviera un dólar por cada vez que un representante se queja conmigo sobre la dirección equivocada de un prospecto, podría retirarme y mudarme a Florida", bromea Ryan. "Mi trabajo es identificar cuáles prospectos son malos. No es muy difícil si sólo hay unos cuantos miles, pero ¿qué hace uno cuando hay más de un cuarto de millón?".

Ryan continúa, "Como usamos estos datos con frecuencia para enviar muchos correos, es muy importante para nosotros asegurar que los nombres y direcciones en ese archivo tengan la mayor exactitud posible. Por ejemplo, deben conformarse a los estándares postales y no debe haber duplicados."

"Para lograr esto utilizamos una técnica llamada higiene de datos. ¿Qué tal suena ese término especializado? Por lo general logramos la higiene de los datos mediante software especializado, el cual se utiliza para determinar la validez de una dirección. Este software relaciona la dirección de la base de datos con su propia base de datos interna de calles y rangos de números válidos en una ciudad o código postal específico".

Ryan prosigue: "Uno de los otros retos de datos a los que se enfrentan los que trabajan en marketing es la eliminación de registros duplicados en la base de datos de marketing. Hay dos tipos de duplicados a buscar: duplicados internos, que consisten en la existencia de varios registros del mismo cliente o prospecto, y los duplicados externos, que representan nuestra incapacidad de eliminar clientes de nuestros datos de prospectos".

"Los duplicados internos crean problemas en los informes e incrementan los costos del servicio de correo. Los duplicados externos son incluso peores: son tanto costosos como vergonzosos". Ryan explica: "Una de las cosas más vergonzosas para un representante de ventas es realizar una llamada a un prospecto sólo para descubrir que ya es nuestro cliente. Por lo general, el cliente se queda con la sensación de que es sólo un número en una de nuestras computadoras. Además de la mala impresión, desperdicia el valioso tiempo y los recursos".

Describa en dos párrafos algunas técnicas que podría usar Ryan para ayudar a identificar los duplicados internos y externos en la base de datos de marketing de su empresa. Describa cómo crearía una base de datos de marketing para minimizar los duplicados (use un párrafo). ¿Hay métodos operacionales que podrían reducir este problema? Haga una lista de ellos. ¿Quién más en la organización podría ayudar con este proceso? Proporcione una lista breve. Recomiende en un párrafo los métodos para Chandler y los demás miembros de su equipo de análisis de sistemas que se puedan usar para ayudar a reclutar y asegurar la asistencia de otros miembros relevantes de la organización.

Los tipos de patrones que los encargados de tomar decisiones tratan de identificar incluyen las asociaciones, las secuencias, el agrupamiento y las tendencias. Las asociaciones son patrones que ocurren en conjunto al mismo tiempo. Por ejemplo, una persona que compra cereal por lo general compra leche para acompañarlo. Por otro lado, las secuencias son patrones de acciones que se llevan a cabo durante un periodo de tiempo. Por ejemplo, si una familia compra una casa este año, es muy probable que vayan a comprar bienes de consumo duraderos (un refrigerador, o lavadora y secadora) el próximo. El agrupamiento es el patrón que se desarrolla entre un grupo de personas. Por ejemplo, los clientes que viven en un código postal específico tal vez tiendan a comprar un automóvil específico. Por último, las tendencias son patrones que se observan durante un periodo de tiempo. Por ejemplo, tal vez los consumidores pasen de comprar artículos genéricos a productos premium.

El concepto de minería de datos surgió del deseo de usar una base de datos para una focalización más selectiva de los clientes. Las primeras metodologías para el correo directo incluían el uso de la información del código

postal como una forma de determinar cuál podría ser el ingreso de una familia (suponiendo que ésta debe generar suficiente ingreso como para poder vivir en el prestigioso código postal 90210 de Beverly Hills o en algún otro vecindario acomodado). Era una manera (imperfecta, desde luego) para limitar el número de catálogos enviados.

La minería de datos lleva este concepto un paso más allá. Si suponemos que el comportamiento anterior es un buen vaticinador de las compras futuras, se recopila una gran cantidad de datos sobre una persona específica a partir de las compras con su tarjeta de crédito. La empresa puede identificar en qué tiendas compramos, qué hemos comprado y cuánto pagamos por un artículo, además de saber cuándo y con qué frecuencia viajamos. Los datos también se introducen, almacenan y utilizan para una variedad de propósitos cuando llenamos las garantías, solicitamos una licencia de conductor, respondemos a una oferta gratuita o solicitamos una tarjeta de membresía en una tienda de renta de videos. Lo que es más: las empresas comparten estos datos y a menudo hacen dinero al venderlos a terceros.

American Express ha sido líder en minería de datos para fines de marketing. American Express le envía cupones de descuento para nuevas tiendas o entretenimiento cuando le envía la factura de su tarjeta de crédito, habiendo determinado que usted compró en tiendas similares o asistió a eventos similares. General Motors ofrece una tarjeta MasterCard que permite a los clientes acumular puntos de bonificación para la compra de un nuevo automóvil, y después envía información sobre los nuevos vehículos en el momento en que sería más probable que el consumidor esté interesado en comprar un auto nuevo.

Sin embargo, la metodología de minería de datos también presenta sus problemas. En primer lugar, los costos pueden ser demasiado altos como para poder justificar la minería de datos, algo que tal vez se descubra sólo hasta después de haber acumulado enormes costos de preparación. En segundo lugar, la minería de datos tiene que estar coordinada de manera que varios departamentos o subsidiarias no traten de contactar al cliente al mismo tiempo. Además, los clientes tal vez piensen que se ha invadido su privacidad y resientan las ofertas que reciben. Por último, los clientes podrían pensar que los perfiles creados únicamente con base en sus compras con tarjeta de crédito presentan una imagen bastante distorsionada de quiénes son en realidad.

Los analistas deben asumir responsabilidad en cuanto a los aspectos éticos de cualquier proyecto de minería de datos propuesto. Hay que considerar y averiguar, junto con el cliente, todos los detalles sobre el periodo que se mantendrá el material de los perfiles, la confidencialidad del mismo, las medidas de protección de la privacidad incluidas y los usos que se le darán. Las oportunidades de abuso son evidentes, por lo que debemos estar protegidos. Para los consumidores, la minería de datos es sólo otra tecnología más que se les impone, y si ellos no desean que pase esto, los esfuerzos de minería de datos fracasarán.

RESUMEN

Una decisión importante en el diseño de un sistema de información es la forma en que se almacenan los datos. Hay dos metodologías para ello: la primera consiste en almacenarlos en archivos individuales, usando un archivo para cada aplicación; la segunda, en desarrollar una base de datos que puedan compartir muchos usuarios para varias aplicaciones, a medida que surja la necesidad.

Para comprender el almacenamiento de los datos hay que captar tres ámbitos: realidad, datos y metadatos. Una entidad es un objeto o evento por el que estamos dispuestos a recolectar y almacenar datos. Los atributos son las características reales de estas entidades. Los elementos de datos pueden tener valores y se pueden organizar en registros a los que se puede acceder mediante una clave. Los metadatos describen a los datos y pueden contener restricciones sobre el valor de un elemento de datos (por ejemplo, que sea sólo numérico).

Algunos ejemplos de archivos convencionales son los archivos maestros, los archivos de tablas, los archivos de transacciones, los archivos de trabajo y los archivos de informes. Por lo general, las bases de datos se construyen con una estructura relacional. Sin embargo, los sistemas heredados pueden tener estructuras jerárquicas o de redes.

La normalización es el proceso que toma las vistas de usuario y las transforma en estructuras menos complejas conocidas como relaciones normalizadas. Hay tres pasos en el proceso de normalización. En primer lugar se eliminan todos los grupos repetitivos. En segundo lugar se eliminan todas las dependencias parciales. Por último se quitan las dependencias transitivas. Una vez que se completan estos tres pasos, el resultado es la creación de numerosas relaciones que están en la tercera forma normal (3NF).

Podemos usar el diagrama de entidad-relación para determinar las claves requeridas para un registro o una relación de una base de datos. Los tres lineamientos a seguir al diseñar tablas maestras o relaciones de bases de datos son: 1) cada entidad de datos separada debe crear una tabla maestra (no combinar dos entidades distintas dentro de una tabla); 2) un campo de datos específico debe existir en sólo una tabla maestra; y 3) cada tabla maestra o relación de base de datos debe tener programas para crear (C), leer (R), actualizar (U) y eliminar (E).

El proceso de recuperar datos puede involucrar hasta ocho pasos: 1) seleccionar una relación, 2) unir dos relaciones, 3) proyectar (seleccionar) las columnas, 4) seleccionar las filas relevantes, 5) derivar nuevos atributos, 6) ordenar o indexar filas, 7) calcular totales y medidas de desempeño, y finalmente 8) presentar los resultados al usuario.

La desnormalización es un proceso que toma el modelo de datos lógico y lo transforma en un modelo físico que sea eficiente para las tareas más necesarias. Los almacenes corporativos de datos difieren de las bases de datos tradicionales en muchos aspectos; uno de ellos es que almacenan datos desnormalizados, los cuales están organizados en base a temas. Los almacenes cor-

EXPERIENCIA DE HYPERCASE® 13

"Escuché muchas cosas buenas sobre su equipo de las personas en Sistemas administrativos. Incluso se ganaron un elogio bien merecido de Tom Ketcham, a quien no es muy fácil complacer en estos días. Hasta *él* está viendo algunas posibilidades. Creo que hasta ahora han logrado mantenernos unidos… a menos que nos vayamos en distintas direcciones otra vez. Sólo estoy bromeando. Les dije que pensaran acerca de si somos una familia, un zoológico o una zona de guerra. Ahora es el momento de empezar a diseñar sistemas que se adapten a nosotros. Ya ha estado usted aquí el tiempo suficiente como para poder formarse esas opiniones. Espero que sean favorables. Creo que nuestra famosa hospitalidad sureña y sentido del humor notablemente británico deberían ayudar a influenciarlo, ¿o no? Estuve tan ocupado tratando de persuadirlo de que valemos la pena el esfuerzo que olvidé decirle: Tom y Snowden han acordado pensar en cambiar a una base de datos de algún tipo. ¿Podría tener esto listo en las próximas dos semanas? Tom está en una conferencia en Minneapolis, pero cuando regrese sería excelente que usted tuviera algunas ideas al respecto para que Snowden y Tom discutan sobre ellas. Siga perseverando".

Preguntas de HYPERCASE

1. Suponga que los miembros de su equipo han usado el Informe de características de los clientes de la Unidad de capacitación (Training Unit Client Characteristics Report) para diseñar una tabla de base de datos que almacene la información contenida en este informe, con el siguiente resultado:
2. Aplique la normalización a la tabla que su equipo desarrolló para eliminar los grupos repetitivos. Muestre sus resultados.
3. Elimine las dependencias transitivas de su tabla y muestre la tabla de datos resultante.

Nombre de tabla: TABLA CLIENTES

Nombre de columna	Descripción
ID CLIENTE (clave primaria)	Nemónico inventado por los usuarios, como STHSP para State Hospital
NOMBRE CLIENTE	El nombre actual completo del cliente
DIRECCION	La dirección del cliente
CONTACTO	El nombre de la persona de contacto
NUMERO TELEFONICO	El número telefónico de la persona de contacto
CLASE	El tipo de institución (Hospital de administración de veteranos, clínica, otro)
TAMANIO-PERSONAL	Tamaño del personal del cliente (número)
NIVEL CAPACITACION	Mínimo nivel de experiencia requerido del personal (según lo definido por la clase)
CTD-EQUIP	El número de equipos médicos que tiene el cliente
TIPO EQUIP	El tipo de equipos médicos (por ejemplo, rayos X, IRM, TAC)
MODELO-ANIO EQUIP	El modelo y año de cada equipo médico

porativos de datos permiten un fácil acceso mediante el software de minería de datos también conocido como software, el cual busca patrones e identifica las relaciones que no pueden imaginar los humanos encargados de tomar decisiones.

La minería de datos involucra el uso de una base de datos para una focalización más selectiva de los clientes. Suponiendo que el comportamiento anterior es un buen vaticinador de las compras futuras, las empresas recopilan datos sobre una persona a partir de las compras con su tarjeta de crédito, las solicitudes de licencia de conductor, las tarjetas de garantías, etcétera. La minería de datos puede ser poderosa, pero también puede ser costosa y debe estar coordinada. Además, puede infringir la privacidad del consumidor o incluso los derechos civiles de una persona.

PALABRAS CLAVE Y FRASES

administrador de base de datos
almacén de datos
almacenamiento de datos
anomalía de actualización
anomalía de eliminación
archivo convencional
archivo de informes
archivo de tablas
archivo de trabajo
archivo de transacciones
archivo maestro
atributo
base de datos
caracteres especiales
clave concatenada
clave primaria
clave secundaria
CRUD (crear, leer, actualizar y eliminar)
datos limpios
dependencias parciales
dependencias transitivas
desnormalización

diagrama de burbuja
diagrama de modelo de datos
diagrama entidad-relación (E-R)
elemento de datos
elemento de información
entidad
estructura de datos de red
estructura de datos jerárquica
estructura de datos relacional
grupo repetido
identificador de objetos (OID)
integridad de dominio
integridad referencial
minería de datos
normalización
primera forma normal (1NF)
procesamiento analítico en línea (OLAP)
realidad, datos y metadatos
recuperación
recuperación de información
registro
relación
relación sin normalizar
restricción de integridad de entidad
segunda forma normal (2NF)
siftware
sistema de administración de bases de datos (DBMS)
subtipo de entidad
tercera forma normal (3NF)
vista física
vista lógica

PREGUNTAS DE REPASO

1. ¿Cuáles son las ventajas de organizar el almacenamiento de datos como archivos separados?
2. ¿Cuáles son las ventajas de organizar el almacenamiento de datos mediante una metodología de base de datos?
3. ¿Cuáles son las medidas de efectividad del diseño de bases de datos?
4. Mencione algunos ejemplos de entidades y sus atributos.
5. ¿Cuál es la diferencia entre una clave primaria y un identificador de objetos?
6. Defina el término *metadatos*. ¿Cuál es el propósito de los metadatos?
7. Mencione los tipos de archivos convencionales de uso común. ¿Cuáles de éstos son archivos temporales?
8. Mencione los tres tipos principales de organización de bases de datos.
9. Defina el término *normalización*.
10. ¿Qué se elimina cuando una relación se convierte a la primera forma normal?
11. ¿Qué se elimina cuando una relación se convierte de 1NF a 2NF?
12. ¿Qué se elimina cuando una relación se convierte de 2NF a 3NF?
13. Mencione las tres restricciones de entidad. Describa en un enunciado el significado de cada restricción de entidad.
14. Describa las cuatro anomalías que pueden ocurrir al crear tablas de bases de datos.
15. Mencione los ocho pasos para recuperar, preordenar y presentar los datos.
16. ¿Qué hace la unión? ¿Qué es la proyección? ¿Qué es la selección?
17. Defina desnormalización.
18. Explique las diferencias entre bases de datos tradicionales y almacenes corporativos de datos.
19. Defina qué hace el software cuando se utiliza en minería de datos.

PROBLEMAS

1. Dado el siguiente archivo de arrendatarios:

Número de registro	*Apellido paterno*	*Número de apartamento*	*Renta*	*Expiración del arrendamiento*
41	Warkentin	102	550	4/30
42	Buffington	204	600	4/30
43	Schuldt	103	550	4/30
44	Tang	209	600	5/31
45	Cho	203	550	5/31
46	Yoo	203	550	6/30
47	Pyle	101	500	6/30

   a. Muestre un ejemplo de proyección.
   b. Muestre un ejemplo de selección.
   c. Muestre dos ejemplos de cómo ordenar filas.
   d. Muestre un ejemplo de cómo calcular los totales.

2. El siguiente es un ejemplo de una boleta de calificaciones para dos estudiantes en la University of Southern New Jersey:

USNJ - Boleta de calificaciones – Primavera – Semestre 2010

Nombre: I. M. Smarte *Especialidad: MIS*

Estudiante: 053-06929-24 *Estado: Avanzado*

Número de curso	*Título del curso*	*Profesor*	*Departamento del profesor*	*Calificación*
MIS 403	Análisis de sistemas	Diggs, T.	MIS	A
MIS 411	Fundamentos conceptuales	Barre, G.	MIS	A
MIS 420	Factores humanos en SI	Barre, G.	MIS	B
CIS 412	Diseño de bases de datos	Menzel, I.	CIS	A
DESC 353	Modelos de administración	Murney, J.	MIS	A

USNJ - Boleta de calificaciones – Primavera – Semestre 2010

Nombre: E. Z. Grayed *Especialidad: MIS*

Estudiante: 472-6124-59 *Estado: Avanzado*

Número de curso	*Título del curso*	*Profesor*	*Departamento del profesor*	*Calificación*
MIS 403	Análisis de sistemas	Diggs, T.	MIS	B
MIS 411	Fundamentos conceptuales	Barre, G.	MIS	A

Dibuje un diagrama de modelo de datos con asociaciones para la vista de usuario.

3. Convierta la vista de usuario del problema 2 a una relación 3NF. Muestre cada paso del proceso.

4. ¿Qué problema podría surgir al utilizar el número de curso como clave primaria para los datos en el problema 2? (*Sugerencia*: Piense sobre lo que ocurriría si cambia el Nombre del departamento [no se muestra en los datos]).

5. Dibuje un diagrama de entidad-relación para la siguiente situación: muchos estudiantes practican muchos deportes. Una persona denominada entrenador en jefe asume el papel de entrenador en todos estos deportes. Cada una de las entidades tiene un número y un nombre (haga las suposiciones necesarias para completar un diagrama razonable. Haga una lista de sus suposiciones).

6. El diagrama de entidad-relación que dibujó en el problema 5 representa las entidades de datos necesarias para implementar un sistema para rastrear a los estudiantes y los deportes que practican. Haga una lista de las tablas necesarias para implementar el sistema, junto con las claves primarias, secundarias y externas que se requieren para vincular las tablas.

7. Dibuje un diagrama de entidad-relación para la siguiente situación: una panadería comercial fabrica muchos productos: panes, postres, pasteles especiales y muchos otros productos horneados. La panadería compara a varios distribuidores los ingredientes como harina, especias y leche. Algunas veces compran un ingrediente a un solo distribuidor y otras veces lo compran a varios. La panadería tiene clientes comerciales, como escuelas y restaurantes, que hacen pedidos de productos horneados con regularidad. Cada producto horneado tiene un especialista que supervisa la organización de la operación de la panadería e inspecciona el producto terminado.

8. Haga una lista de las tablas y claves que se necesitan para implementar el sistema de la panadería comercial.

9. Dibuje un diagrama E-R para el sistema de pedidos de la figura 13.24.

10. Dibuje un diagrama de flujo de datos para colocar un pedido. Base su diagrama de flujo de datos en el diagrama E-R.

11. Cree un diagrama de entidad-relación para un paquete de software de genealogía llamado “PeopleTree” con el que se puede llevar un registro de los ancestros. Suponga que cada persona estará en una tabla Persona y que una persona puede tener padres (un padre y una madre) biológicos y padres (un padre y una madre) adoptivos. Las madres y padres se deben almacenar en la tabla Persona también. Cada persona sólo debe tener un lugar de nacimiento, el cual se debe almacenar en la tabla Persona. Muchas personas pueden nacer en el mismo lugar.

12. Defina la clave primaria utilizada para las tablas Persona y Lugar.

13. GaiaOrganix es una cooperativa de mayoreo de comida orgánica que vincula a productores y consumidores. GaiaOrganix negocia las compras que las tiendas de abarrotes y demás tiendas relacionadas hacen a los granjeros que cultivan una variedad de cosechas, como frutas, vegetales y granos. Cada granjero puede producir varias cosechas y varios granjeros pueden producir una cosecha. Para proveer el nivel más alto de productos frescos, éstos se envían directamente de la granja a la tienda. Cada tienda puede comprar de muchas granjas y cada granja puede vender a muchas tiendas. Dibuje un diagrama de entidad-relación en la tercera forma normal que muestre la relación entre el productor (granjas) y el vendedor (tiendas).
14. ArticleIndex.com es una empresa que produce índices de artículos de revistas y periódicos para una disciplina específica. Un usuario Web debe tener la capacidad de introducir el tema o los autores de un artículo para recibir una lista detallada de todos los artículos y periódicos en los que se encontró el tema. Cada artículo puede tener muchos autores y cada autor puede escribir muchos artículos. Un artículo puede existir sólo en un periódico, pero cada periódico por lo general contiene muchos artículos. Cada artículo puede tener muchos temas y cada tema puede estar en muchos artículos. Dibuje un diagrama de entidad-relación en la tercera forma normal para los artículos, autores, periódicos y temas.
15. Identifique las claves primarias y externas para el diagrama de entidad-relación creado en el problema 14.

PROYECTO EN GRUPO

1. Gregg Baker compró boletos para dos conciertos a través de la Web. Sus pedidos se procesaron, las ubicaciones exactas se asignaron y los boletos se enviaron por separado. Uno de los conjuntos de boletos se perdió en el correo. Cuando llamó al número de servicio no recordó la fecha ni los números de los asientos, pero la agencia pudo localizar sus boletos con rapidez debido a que desnormalizó la relación. Describa el sistema de pedidos de boletos mediante un listado de los elementos de datos que se mantienen en el formulario de pedido y el formulario de envío. ¿Qué información proporcionó Gregg a la agencia de boletos para recuperar la información?

BIBLIOGRAFÍA SELECCIONADA

Agrawal, R., A. Ailamaki, P. A. Bernstein, E. A. Brewer, M. J. Carey, S. Chaudhuri, et al. “The Claremont Report on Database Research”. *Communications of the ACM,* Vol. 52, Núm. 6, 2009, pp. 56-65.

Avison, D. E. *Information Systems Development: A Database Approach*, 2ª. Edición. Londres: Blackwell Scientific, 1992.

Codd, E. F. “A Relational Model of Data for Large Shared Data Banks”. *Communications of the ACM*, Vol. 13, Núm. 6, 1970, pp. 377-387.

Deitel, H. M., P. J. Deitel y T. R. Nieto. *E-Business and E-Commerce: How to Program*. Upper Saddle River, NJ: Prentice Hall, 2001.

Gane, C. y T. Sarson. *Structured Systems Analysis: Tools and Techniques*. Englewood Cliffs, NJ: Prentice Hall, 1979.

Gray, P. “Data Warehousing: Three Major Applications and Their Significance”. En *Emerging Information Technologies, Improving Decision, Cooperation, and Infrastructure*. Editado por K. E. Kendall, Thousand Oaks, CA: Sage Publications, 1999.

Hoffer, J., A. Prescott y H. Topi. *Modern Database Management*, 9ª. Edición. Upper Saddle River, NJ: Prentice Hall, 2009.

Sanders, G. L. *Data Modeling*. Nueva York: International Thomson Publishing, 1995.

Shin, S. K. y G. L. Sanders. “Denormalization Strategies for Data Retrieval from Data Warehouses”. *Decision Support Systems*, Vol. 42, Núm, 1, 2006, pp. 267-282.

EPISODIO 13

CASO DE LA CPU

ALLEN SCHMIDT, JULIE E. KENDALL Y KENNETH E. KENDALL

De vuelta a los fundamentos de datos

Después de realizar numerosas entrevistas, prototipos, diagramas de flujo de datos, entradas en el diccionario de datos y diagramas de UML, Anna y Chip empiezan a trabajar en el modelo de entidad-relación. "Yo seré responsable de crear las relaciones de las tablas de Microsoft Access", afirma Anna. Chip se ofrece a completar un diagrama de entidad-relación. "Cuando terminemos, vamos a comparar los dos diagramas en cuanto a precisión y consistencia", sugiere Anna, y así lo hacen.

La figura E13.1 muestra el diagrama de entidad-relación para el sistema de inventario de computadoras. Visible Analyst llama a cada uno de los rectángulos una entidad. Cada entidad representa una tabla de base de datos de información almacenada en el sistema, correspondiente a un almacén de datos en el diagrama de flujo de datos o a una clase de entidad en un diagrama de secuencia o de clases. Cada uno de los rectángulos en forma de diamante representa una entidad asociativa que representa una relación entre las entidades de datos. Un rectángulo con un óvalo en su interior representa a una entidad asociativa que no puede existir sin la entidad conectora. Por lo general éstos son elementos repetitivos. Microsoft Visio usa rectángulos para representar tanto una entidad asociativa como una entidad atributiva.

"Terminé de crear el diagrama de entidad-relación, empezando con las porciones más simples del sistema", Chip dice a Anna. "Las primeras entidades de datos creadas son SOFTWARE y COMPUTADORA. La relación es que el software se instala en la computadora. Después determiné la cardinalidad de la relación. Como un paquete de software se puede instalar en muchas computadoras, esta relación es de uno a muchos. Cada computadora también puede tener varios paquetes de software instalados en ella, por lo que también provee una relación de uno a muchos. Como hay una relación de uno a muchos para cada una de las entidades de datos, la relación completa entre ellas se vuelve de muchos a muchos".

Chip continúa y dice: "Esta primera vista está lejos de ser normalizada. Observa que NUMERO INVENTARIO SOFTWARE es un elemento repetitivo en la entidad HARDWARE. Voy a tener que crear varias entidades para cada uno de ellos". Poco después, Chip revisa su trabajo con Anna. Se quitó el NUMERO INVENTARIO SOFTWARE y se colocó en una entidad relacional. Consulte el diagrama de entidad-relación que se muestra en la figura E13.2. "Esto coloca a los datos en la primera forma normal", comenta Chip. "Además, no hay elementos que dependan sólo de una parte de la clave, por lo que los datos también se encuentran en la segunda forma normal. Sin embargo, hay elementos que no forman parte de la entidad que se representa en el diagrama y habrá que eliminarlos. Por ejemplo, da un vistazo a SISTEMA OPERATIVO y a EDIFICIO CAMPUS. Estos elementos no son parte del hardware de computadora, pero están instalados en la computadora o la computadora está instalada en una sala del campus. Deben tener su propia entidad. Esto facilita el proceso de cambiar la versión de un sistema operativo. En vez de tener que cambiar la versión del sistema operativo en muchos de los registros de COMPUTADORA, tendríamos que cambiarlo sólo una vez".

Anna asiente y comenta: "Es realmente una buena valoración de la situación. Facilitará la implementación de las tablas de Microsoft Access".

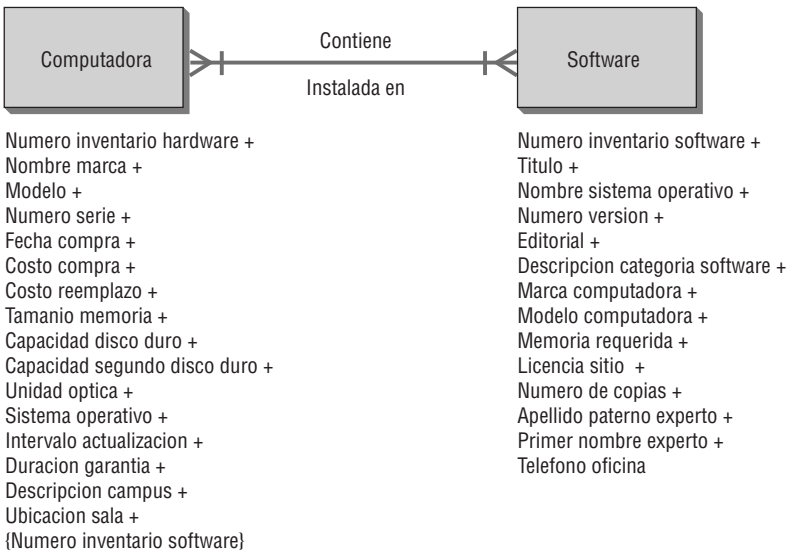


FIGURA E13.1
Diagrama de entidad-relación sin normalizar para el sistema computarizado. La relación de muchos a muchos se tendrá que definir como una entidad asociativa.

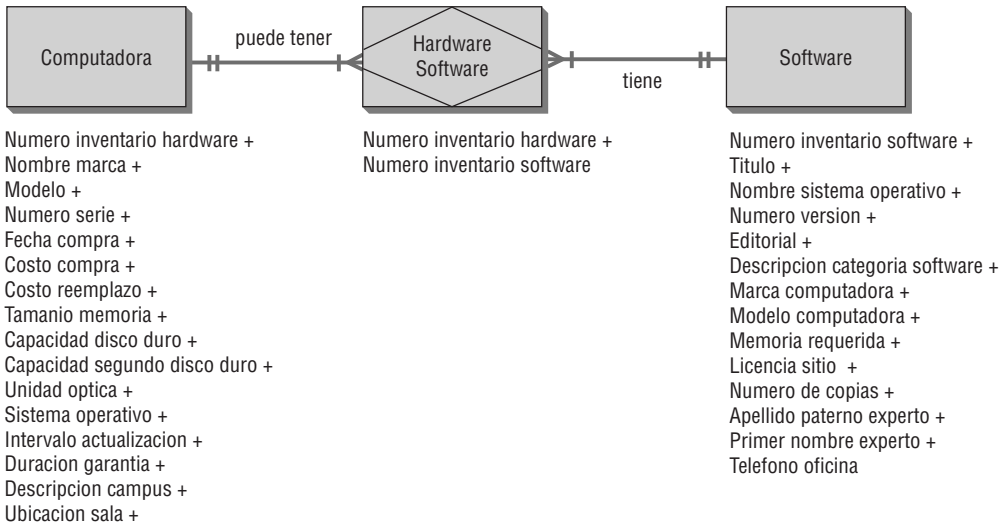


FIGURA E13.2

El diagrama de entidad-relación del sistema computarizado en primera forma normal.

Chip continúa trabajando en el diagrama de entidad-relación. Después de unas cuantas horas exclama: "Creo que está listo. ¿Podrías dar un vistazo a la versión final?". En la figura 13.3 se muestra la versión final. Se describieron todas las entidades y relaciones en el repositorio.

Anna revisa la versión final y exclama, "¡Se ve excelente! Tuviste razón en mover SISTEMA OPERATIVO y EDIFICIO CAMPUS a sus propias entidades. Buena idea, ya que el edificio no es parte de la computadora. Además, en definitiva el EXPERTO DE SOFTWARE no es parte de la entidad SOFTWARE. ¿Qué hay sobre CATEGORIA SOFTWARE?".

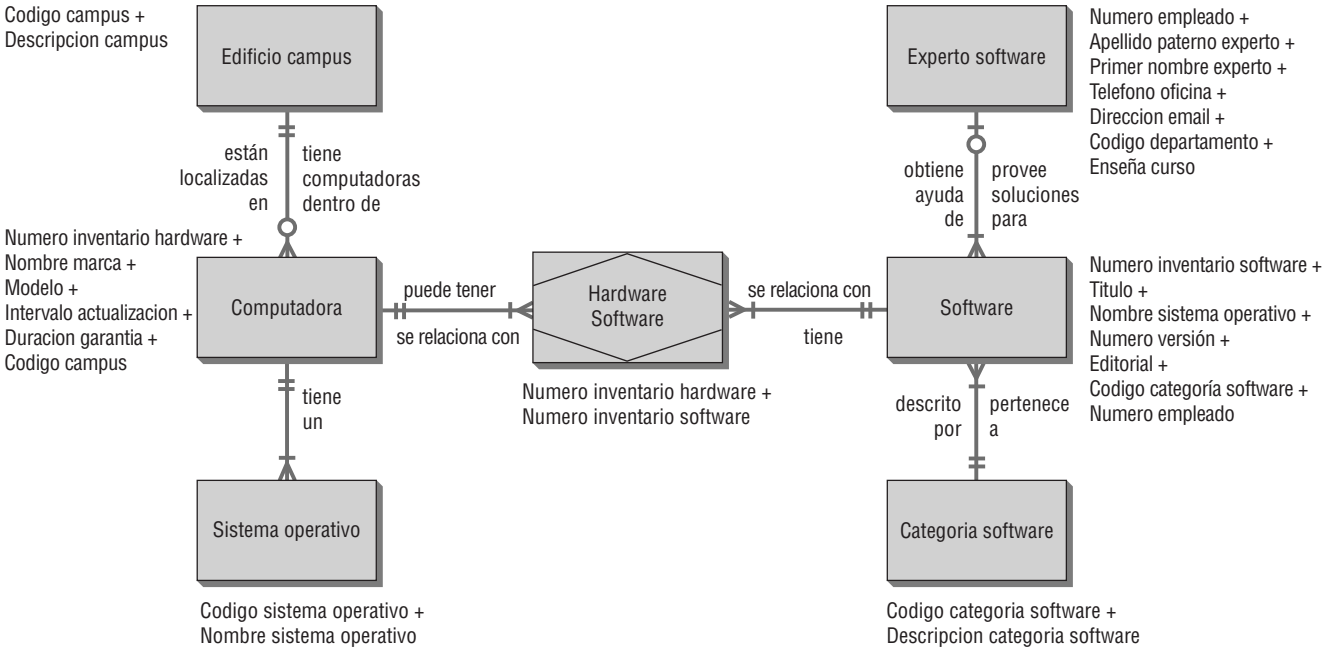
"Moví la CATEGORIA SOFTWARE a su propia entidad para ahorrar espacio en los archivos maestros cuando se construyan", responde Chip. "En realidad es una tabla de códigos y facilita el proceso de cambiar la categoría sin tener que cambiar todos los registros de software. Además vamos a almacenar un pequeño código en vez de una descripción extensa en cada registro de software. ¿Por qué no revisas dos veces las diversas claves en el diagrama? Cada entidad relacionada en el extremo de muchos debería tener una clave externa que coincida con la clave primaria de la entidad en el extremo de uno".

Anna examina el diagrama por un instante y comenta, "Se ve bien, en mi opinión. Tal vez debamos definir las entradas en el repositorio".

"Mira esta entrada en el repositorio". Chip abre el diagrama de entidad-relación (figura E13.4) y hace doble clic en la entidad COMPUTERS (COMPUTADORAS) para mostrar su entrada en el repositorio. Se definieron la clave primaria (la notación [pk] de Microsoft Visio y Visible Analyst frente al elemento NUMERO INVENTARIO HARDWARE [HARDWARE

FIGURA E13.3

Diagrama de entidad-relación final para el sistema computarizado.



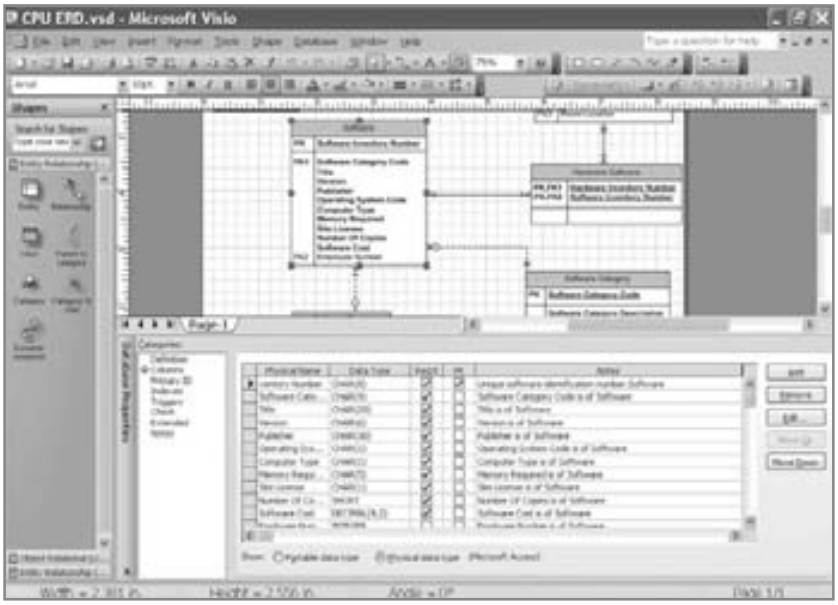


FIGURA E13.4

El diagrama de entidad-relación normalizado de la CPU se muestra en Microsoft Visio con la definición del repositorio para Software.

INVENTORY NUMBER] en el área de composición), las claves externas (FK1, FK2 y así en lo sucesivo) y varias claves alternas ([Ak1],[Ak2] y [Ak3]). Después de invertir tiempo en examinar el diagrama, así como las entradas en el repositorio y los informes de análisis, tanto Anna como Chip quedan satisfechos de que las relaciones entre los datos estén representadas en forma precisa. A continuación deciden cómo diseñar la base de datos a partir de los diagramas.

Primero analizan la relación HARDWARE/SOFTWARE. Como hay una relación de muchos a muchos entre estas dos entidades de datos, se puede implementar mediante el uso de tres tablas de base de datos:

1. Una tabla MAESTRA DE HARDWARE (HARDWARE MASTER).
2. Una tabla MAESTRA DE SOFTWARE (SOFTWARE MASTER).
3. Una tabla RELACION HARDWARE/SOFTWARE (HARDWARE/SOFTWARE RELATIONSHIP), que debe contener los campos clave de las tablas maestras HARDWARE y SOFTWARE para todo el software instalado en todas las máquinas. También podría funcionar una clave de primaria de autosecuencia, siempre y cuando se incluyan las claves externas en la tabla.

"Creo que es mi turno de trabajar en las relaciones", dice Anna mientras saca una copia del diagrama de entidad-relación. Voy a modificar las tablas de Microsoft Access de las sesiones de los prototipos".

Para empezar, Anna establece las claves primarias para cada una de las tablas. Cuando éstas se encuentran en su forma final, crea las relaciones entre ellas. En la figura E13.5 se muestra el diagrama de relaciones de Microsoft Access. Los rectán-

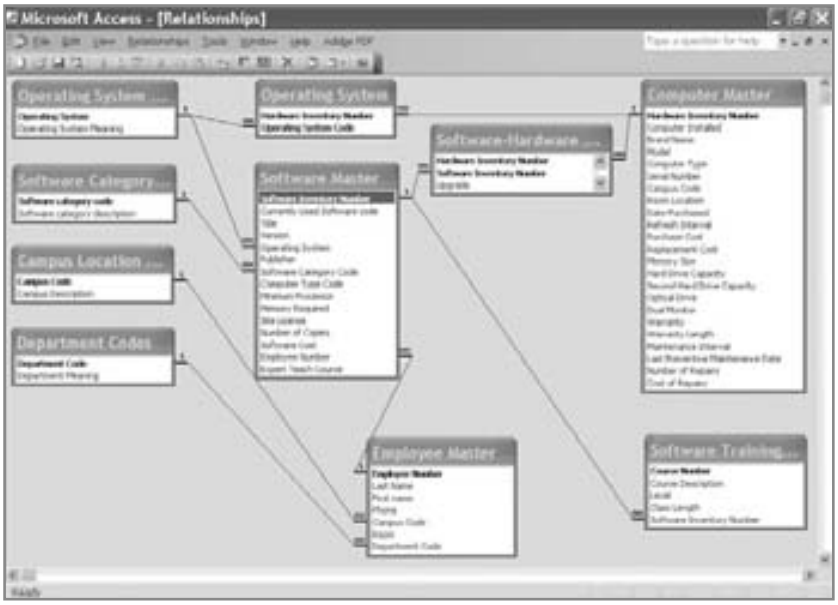


FIGURA E13.5

Un diagrama de RELACIONES de Microsoft Access. Observe los símbolos de infinito que representan el extremo de muchos de una relación.

gulos en el diagrama representan las tablas de la base de datos y corresponden a los diversos tipos de entidades que se encuentran en el diagrama de entidad-relación. Tenga en cuenta que la cardinalidad se representa mediante "1" y el símbolo de infinito. Los campos que son claves primarias se listan como el primer campo de cada rectángulo; también se muestran en negrita. Las claves externas se muestran unidas al otro extremo de la línea de la relación, si la clave externa es visible en el rectángulo de la tabla. Las claves se arrastran de una tabla a otra para establecer una relación; aparece un cuadro de diálogo para determinar las propiedades de la relación.

EJERCICIOS

E-1. Use Microsoft Visio o Visible Analyst para ver los diagramas de entidad-relación sin normalizar y en primera forma normal para el sistema computarizado. Haga doble clic en las entidades para ver la información del repositorio. (En Microsoft Visio y en Visible Analyst; en Microsoft Visio la entrada del repositorio está en un área en la parte inferior de la pantalla. Tal vez tenga que arrastrar el borde que separa el diagrama del repositorio hacia arriba. Haga clic en la entrada Columnas (Columns) que está en el área Categorias (Categories) del lado izquierdo del repositorio para ver los atributos de entidad). El nombre del diagrama es SISTEMA COMPUTARIZADO – SIN NORMALIZAR (COMPUTER SYSTEM – UNNORMALIZED).

E-2. Use Microsoft Visio o Visible Analyst para ver el diagrama de entidad-relación del sistema computarizado. Haga doble clic en las entidades para ver la información del repositorio (en Microsoft Visio y en Visible Analyst). El nombre del diagrama es COMPUTADORA (COMPUTER).

E-3. Agregue la entidad DISTRIBUIDOR (VENDOR) al diagrama. El distribuidor garantiza las computadoras; la relación entre DISTRIBUIDOR y COMPUTADORA es que un VENDEDOR puede garantizar muchas COMPUTADORA(s). Agregue las claves primarias. Microsoft Visio creará de manera automática las claves externas. En Visible Analyst, seleccione Sincronización de claves (Key Synchronization) del menú Repositorio (Repository).

E-4 Agregue la entidad MANTENIMIENTO (MAINTENANCE) al diagrama. Las reparaciones de mantenimiento se realizan en las computadoras; la relación entre MANTENIMIENTO y COMPUTADORA(s) es tal que una COMPUTADORA puede tener muchos registros de MANTENIMIENTO. Use el repositorio para definir NUMERO PEDIDO MANTENIMIENTO (MAINTENANCE ORDER NUMBER). Establezca este atributo como clave primaria para la entidad MANTENIMIENTO y genere la clave externa.

E-5. Describa la entidad CATEGORIA SOFTWARE (SOFTWARE CATEGORY) en el repositorio. Incluya los elementos que se encuentran en el diagrama de entidad-relación debajo de CATEGORIA SOFTWARE en el área **Composición (Composition)**.

E-6. Describa la entidad MANTENIMIENTO (MAINTENANCE) en el repositorio. Los elementos son los siguientes:

a. NUMERO ORDEN MANTENIMIENTO (MAINTENANCE ORDER NUMBER)
b. NUMERO INVENTARIO HARDWARE (HARDWARE INVENTORY NUMBER)
c. FECHA MANTENIMIENTO (MAINTENANCE DATE).
d. TIPO DE MANTENIMIENTO (TYPE OF MAINTENANCE).
e. COSTO DE MANTENIMIENTO (COST OF MAINTENANCE).
f. MANTENIMIENTO CUBIERTO POR GARANTIA (MAINTENANCE COVERED BY WARRANTY).

E-7. Describa la entidad DISTRIBUIDOR (VENDOR). Los elementos son los siguientes:

a. NUMERO DISTRIBUIDOR (VENDOR NUMBER).
b. NOMBRE DISTRIBUIDOR (VENDOR NAME).
c. CALLE (STREET).
d. CIUDAD (CITY).
e. ESTADO (STATE).
f. CODIGO POSTAL (ZIP CODE).
g. NUMERO TELEFONICO (TELEPHONE NUMBER).
h. FECHA ENVIO ULTIMO PEDIDO (DATE LAST ORDER SENT).
i. MONTO TOTAL COMPRA DISTRIBUIDOR (TOTAL AMOUNT PURCHASED FROM VENDOR).
j. NUMERO TOTAL DE PEDIDOS ENVIADOS AL DISTRIBUIDOR (TOTAL NUMBER OF ORDERS SENT TO VENDOR).

E-8. Cada computadora puede tener más de un sistema operativo y cada sistema operativo puede estar instalado en más de una computadora. Agregue una entidad asociativa llamada SISTEMA OPERATIVO COMPUTADORA (COMPUTER OPERATING SYSTEM) entre COMPUTADORA (COMPUTER) y SISTEMA OPERATIVO (OPERATING SYSTEM). Incluya las claves primarias y externas en el repositorio, ya sea en Microsoft Visio o en Visible Analyst.

E-9. Explique en un párrafo la relación entre una clave externa y una clave primaria, y por qué es necesario tenerlas en entidades separadas cuando hay una relación entre las entidades.

Los ejercicios en los que se antepone un icono www indican material de valor agregado disponible en el sitio Web www.pearsonhighered.com/kendall. Los estudiantes pueden descargar un archivo de ejemplo de Microsoft Visio, Visible Analyst, Microsoft Project o Microsoft Access que pueden usar para completar los ejercicios.

CAPÍTULO 15

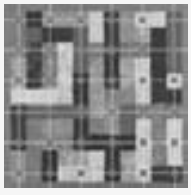
PARTE V
Aseguramiento e implementación de la calidad

Diseño de procedimientos precisos de entrada de datos

OBJETIVOS DE APRENDIZAJE

Al completar este capítulo usted podrá:

1. **Comprender los usos de la codificación efectiva para apoyar a los usuarios en la realización de sus tareas.**
2. **Diseñar metodologías efectivas y eficientes de captura de datos para las personas y los sistemas.**
3. **Reconocer cómo asegurar la calidad de los datos por medio de la validación.**
4. **Articular las ventajas de la precisión de la entrada de los usuarios en los sitios Web de comercio electrónico.**



Es de suma importancia asegurar que los usuarios puedan introducir datos en el sistema con precisión. A estas alturas es claro que la calidad de la entrada de datos determina la calidad de la salida de información. El analista de sistemas puede apoyar la entrada de datos con precisión por medio de la obtención de cuatro objetivos generales: 1) crear una codificación significativa para los datos, 2) diseñar metodologías eficientes de captura de datos, 3) asegurar una captura de datos completa y efectiva, y 4) asegurar la calidad de los datos por medio de la validación.

La calidad de los datos es una medida de la consistencia y precisión de los datos dentro de ciertos límites preestablecidos. Los datos efectivamente codificados facilitan al usuario su introducción precisa, reduciendo al mínimo la cantidad de datos y por ende, el tiempo requerido para introducirlos.

Cuando los usuarios introducen datos con eficiencia, la entrada cumple con ciertas medidas predeterminadas de desempeño basadas en la relación entre el tiempo invertido y el número de datos introducidos. La codificación efectiva, la captura e introducción de datos efectiva y eficiente, y la acción de asegurar la calidad de los datos por medio de procedimientos de validación son todos objetivos de entrada de datos que cubriremos en este capítulo.

CODIFICACIÓN EFECTIVA

Una de las formas en que se pueden introducir datos con más precisión y eficiencia es mediante el empleo informado de varios códigos. Convertir datos ambiguos o voluminosos en dígitos cortos que se introduzcan con facilidad se denomina codificación (no debemos confundirlo con la escritura de código, o codificación, de programas).

La codificación ayuda al analista de sistemas a alcanzar el objetivo de eficiencia, pues los datos codificados requieren menos tiempo para que las personas los introduzcan y, por ende, se reduce el número de elementos introducidos. La codificación también puede ayudar a ordenar los datos en forma apropiada en un punto posterior del proceso de transformación de los datos; además pueden ahorrar valioso espacio de memoria y almacenamiento. En resumen, la codificación es una manera de ser elocuente y a la vez conciso en la captura. Además de proveer precisión y eficiencia, los códigos deben tener un propósito para apoyar a los usuarios. Los tipos específicos de códigos nos permiten tratar los datos de una manera específica. Los propósitos humanos para la codificación incluyen lo siguiente:

1. Mantener el registro de algo.
2. Clasificar información.
3. Ocultar información.

FIGURA 15.1

Uso de un código de secuencia simple para indicar la secuencia en que los pedidos entran a una tienda de muebles a la medida.

Pedido #	Producto	Cliente
5676	Silla mecedora con piel	Arthur Hook, Jr.
5677	Silla de comedor/tapizada	Millie Monice
5678	Love Seat/tapizado	J. & D. Pare
5679	Silla mecedora para niño/calcomanías	Lucinda Morely

4. Revelar información.
5. Solicitar la acción apropiada.

Hablaremos sobre cada uno de estos propósitos de codificación en las siguientes secciones, junto con algunos ejemplos de códigos.

Mantener el registro de algo

Algunas veces simplemente queremos identificar una persona, lugar o cosa sólo para llevar un registro. Por ejemplo, un taller de muebles a la medida necesita asignar un número de trabajo a un proyecto. El vendedor necesita conocer el nombre y la dirección del cliente, pero el gerente del taller o los trabajadores que ensamblan los muebles no. En consecuencia, se asigna un número arbitrario al trabajo. El número puede ser aleatorio o secuencial, como se describe en la siguiente subsección.

CÓDIGOS DE SECUENCIA SIMPLE El código de secuencia simple es un número que se asigna a algo si necesita estar enumerado. Por lo tanto, no tiene relación con los datos en sí. La figura 15.1 muestra cómo se asigna un número a los pedidos de un fabricante de muebles. Con este medio de fácil referencia, la empresa puede llevar el registro del pedido en proceso. Es más eficiente introducir el trabajo "5676" que "esa silla mecedora color café con negro, con el asiento de piel para Arthur Hook, Jr.".

Usar un código secuencial en vez de un número aleatorio tiene sus ventajas. En primer lugar, elimina la posibilidad de asignar el mismo número; en segundo, da a los usuarios una aproximación de cuándo se recibió el pedido.

Hay que usar códigos de secuencia cuando el orden de procesamiento requiera de un conocimiento de la secuencia en la que los elementos entran al sistema, o el orden en el que se desarrollen los eventos. Podemos encontrar un ejemplo en la situación de un banco que ofrece una promoción especial en la que es importante saber cuándo presentó una persona su solicitud para un préstamo para vivienda especial de bajo interés, ya que (siendo todos los demás factores iguales) los préstamos hipotecarios especiales se otorgarán según el orden en el que se hayan presentado las solicitudes. En este caso es importante asignar un código de secuencia correcto a cada solicitante.

CÓDIGOS DE DERIVACIÓN ALFABÉTICA A veces no es conveniente usar códigos de secuencia. La situación más obvia es cuando *no* deseamos que alguien lea el código para averiguar cuántos números se han asignado; otro caso es cuando sea conveniente un código más complejo para evitar un costoso error. Un posible error sería agregar un pago a la cuenta 223 cuando en realidad queríamos agregarlo a la cuenta 224, y todo por introducir el dígito incorrecto.

El código de derivación alfabética es una metodología común para identificar un número de cuenta. El ejemplo de la figura 15.2 proviene de una etiqueta de correo para una revista. El código se convierte en el número de cuenta. Los primeros cinco dígitos provienen de los primeros cinco dígitos del código postal del suscriptor; los siguientes tres dígitos son las primeras tres consonantes en el nombre del suscriptor, los siguientes cuatro números son de la dirección y los últimos tres representan el código de la revista. El principal propósito de este código es identificar una cuenta.

FIGURA 15.2

Identificación de la cuenta del suscriptor de una revista mediante un código de derivación alfabética.

Código	Explicación del código
68506KND7533TVG	99999XXX9999XXX Abreviatura de la revista Cuatro dígitos de la dirección Primeras tres consonantes del apellido Primeros cinco dígitos del código postal

Un segundo propósito es el de imprimir etiquetas de correo. Al diseñar este código, el código postal es la primera parte del número de cuenta. Por lo general, los registros de suscriptores se actualizan sólo una vez al año, pero el principal propósito de los registros es imprimir etiquetas de correo una vez al mes o una vez por semana. Tener el código postal como la primera parte del campo de clave primaria significa que los registros no se tienen que ordenar por código postal para enviarlos en masa, ya que los registros de un archivo se almacenan en secuencia por clave primaria. Cabe mencionar que la fecha de expiración no es parte del número de cuenta, ya que ese número puede cambiar con más frecuencia que los otros datos.

Una desventaja de un código de derivación alfabética ocurre cuando la porción alfabética es pequeña (por ejemplo, el nombre Po) o cuando el nombre contiene menos consonantes de las que el código requiere. El nombre Roe tiene sólo una consonante y se tendría que derivar como RXX, o por medio de algún otro esquema. Otra desventaja es que algunos de los datos pueden cambiar. Al modificar la dirección o el nombre de un suscriptor cambiaría la clave primaria para el archivo.

Clasificar la información

La clasificación ofrece la posibilidad de diferenciar entre las clases de elementos. Las clasificaciones son necesarias para muchos fines, como reflejar las partes del plan de seguro médico con las que cuenta un empleado, o mostrar cuál estudiante completó los requerimientos básicos de su programa de estudios.

Para que sean útiles, las clases deben ser mutuamente excluyentes. Por ejemplo, si un estudiante está en la clase F (que significa estudiante de primer año) y completó de 0 a 36 horas crédito, no hay que clasificarlo también como estudiante de segundo año (S). Las clases superpuestas serían F = 0 − 36 horas crédito, S = 32 − 64 horas crédito, etcétera. Los datos no son claros y no se pueden interpretar con tanta facilidad cuando las clases de codificación no son mutuamente excluyentes.

CÓDIGOS DE CLASIFICACIÓN Los códigos de clasificación se utilizan para diferenciar un grupo de datos con características especiales de otro. Los códigos de clasificación pueden consistir de una sola letra o número. Son una manera abreviada de describir una persona, un lugar, una cosa o un evento.

Los códigos de clasificación se listan en los manuales o se publican de manera que los usuarios puedan localizarlos con facilidad. Muchas veces los usuarios se familiarizan tanto con los códigos de uso frecuente que los memorizan. Un usuario clasifica un elemento y después introduce su código directamente en un sistema en línea.

Un ejemplo de codificación de clasificación es la forma en que sería conveniente agrupar los rubros deducibles de impuestos para poder llenar una declaración de impuestos. La figura 15.3 muestra cómo se desarrollan códigos para elementos tales como Pagos de intereses, Gastos médicos, contribuciones, etcétera. El sistema de codificación es simple: se toma la primera letra de cada una de las categorías; las contribuciones son C, los pagos de intereses son P y los suministros son S.

Todo marcha bien hasta que llegamos a otras categorías (como elementos de computadora, pagos de seguros y suscripciones) que empiezan con las mismas letras utilizadas previamente. La figura 15.4 demuestra lo que ocurre en este caso. Extendimos la codificación de manera que pudiéramos usar P para “ComPutadora”, A para “PAgos de seguros” y B para “SuBscripciones”. Sin duda esta situación está lejos de ser perfecta. Una manera de evitar este tipo de confusión es permitir códigos mayores de una letra, lo cual veremos más adelante en este capítulo bajo el subtítulo de códigos mnemónicos. Los menús desplegables en un sistema de GUI utilizan con frecuencia códigos de clasificación como método abreviado para seleccionar las características del menú, como **Alt-A** para el menú **Archivo**.

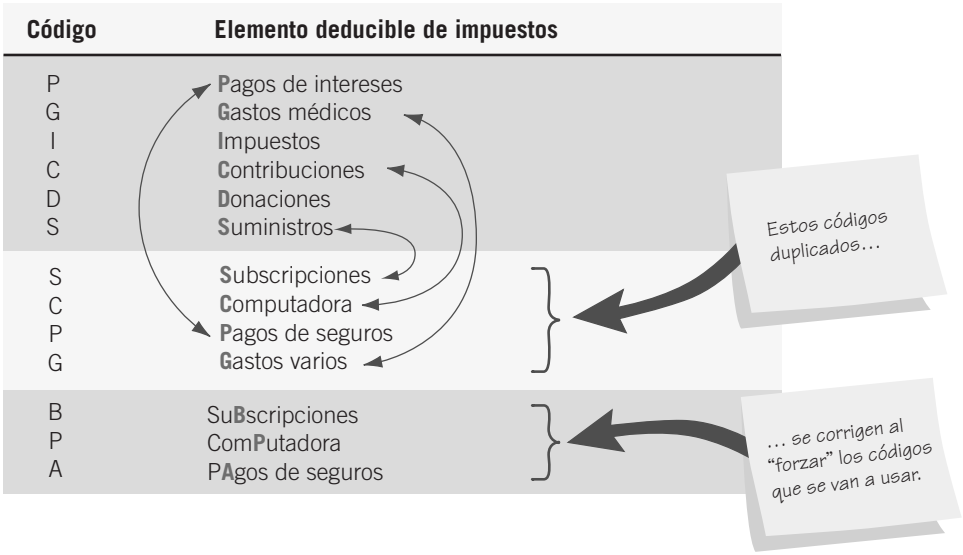
CÓDIGOS DE SECUENCIA EN BLOQUE Anteriormente vimos los códigos de secuencia; el código de secuencia en bloque es una extensión. La figura 15.5 muestra cómo el usuario de una empresa asigna números al software de computadora. Las principales categorías de software son navegadores, paquetes de bases de datos y diseño Web. A estas categorías se les asignaron números secuenciales en los siguientes “bloques” o rangos: navegador, 100-199, base de datos, 200-299, y así en lo sucesivo. La ventaja del código de secuencia en bloque es que los datos se agrupan de acuerdo con las características comunes, al tiempo que aprovechan la simpleza de asignar el siguiente número disponible (dentro del bloque, desde luego) al siguiente elemento que necesite identificación.

Código	Elemento deducible de impuestos
P	**P**agos de intereses
G	**G**astos médicos
I	**I**mpuestos
C	**C**ontribuciones
D	**D**onaciones
S	**S**uministros

FIGURA 15.3
Cómo agrupar los elementos deducibles de impuestos por medio de un código de clasificación de una letra.

FIGURA 15.4

Los problemas al usar un código de clasificación de una letra ocurren cuando las categorías comparten la misma letra.



Ocultar información

Podemos usar códigos para ocultar o disfrazar la información que otras personas no deban conocer. Hay muchas razones por las que una empresa podría querer hacer esto. Por ejemplo, tal vez una corporación no desee que los empleados capturistas de datos accedan a la información en un archivo de personal. Tal vez una tienda quiera que sus vendedores conozcan el precio de mayoreo para mostrarles cuál es el precio más bajo que puedan negociar, pero podría codificarlo en las etiquetas de los precios para evitar que los clientes lo descubran. Tal vez un restaurante quiera capturar la información sobre el servicio sin dejar que el cliente sepa el nombre del mesero. El ocultamiento de la información y la seguridad se han convertido en temas muy importantes durante los últimos años. Las corporaciones han empezado a permitir a los distribuidores y consumidores el acceso directo a sus bases de datos, y el hecho de manejar las transacciones comerciales a través de Internet ha provocado que sea necesario desarrollar estrictos esquemas de cifrado. En la siguiente subsección veremos un ejemplo de cómo ocultar información por medio de códigos.

CÓDIGOS DE CIFRADO Tal vez el método de codificación más simple sea la sustitución directa de una letra por otra, un número por otro o una letra por un número. Como ejemplo de sustitución de letras podríamos mencionar un tipo popular de acertijo conocido como criptograma. La figura 15.6 muestra un ejemplo de un código de cifrado tomado de una tienda de departamentos de Buffalo, Nueva York, la cual codificó todos los precios rebajados con las palabras BLEACH MIND. Nadie recordó realmente por qué se eligieron estas palabras, pero todos los empleados las conocían de corazón y eso contribuyó al éxito del código de cifrado. Podemos observar en esa figura que un artículo con un precio de venta de $25.00 podría tener un precio rebajado de BIMC, o de $18.75 cuando se decodifica letra por letra.

FIGURA 15.5

Uso de un código de secuencia en bloque para agrupar paquetes de software similares.

Código	Nombre del paquete de software	Tipo
100	Apple Safari	Navegador
101	Mozilla Firefox	
102	Microsoft Internet Explorer	
.		
.		
.		
200	Microsoft Access	Base de datos
201	MySQL	
202	Oracle	
.		
.		
.		
300	Adobe Dreamweaver	Diseño Web
301	Freeway Pro	
302	Microsoft Web Expression	

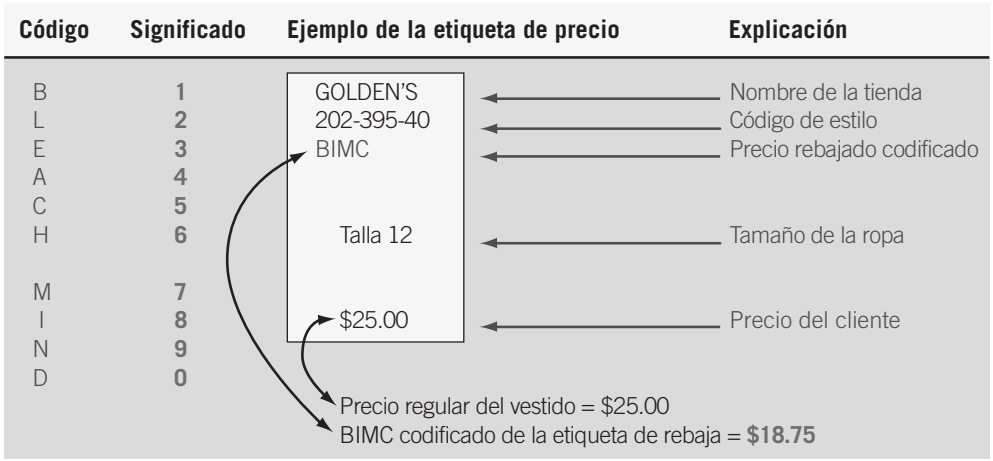


FIGURA 15.6

Codificar los precios rebajados mediante un código de cifrado es una forma de ocultar la información de los precios a los clientes.

Revelar información

Algunas veces es conveniente revelar información a usuarios específicos por medio de un código. En una tienda de ropa, la información sobre el departamento, producto, color y tamaño se imprime junto con el precio en el recibo para cada artículo. Esta información ayuda a los vendedores y al personal de existencias a localizar el lugar para la mercancía.

Otra razón de revelar información por medio de códigos es para que la entrada de datos sea más significativa para los humanos. Un número de pieza, nombre o descripción familiares producen una mayor entrada de datos. Los ejemplos de los códigos en la siguiente subsección explican cómo se pueden llevar a cabo estos conceptos.

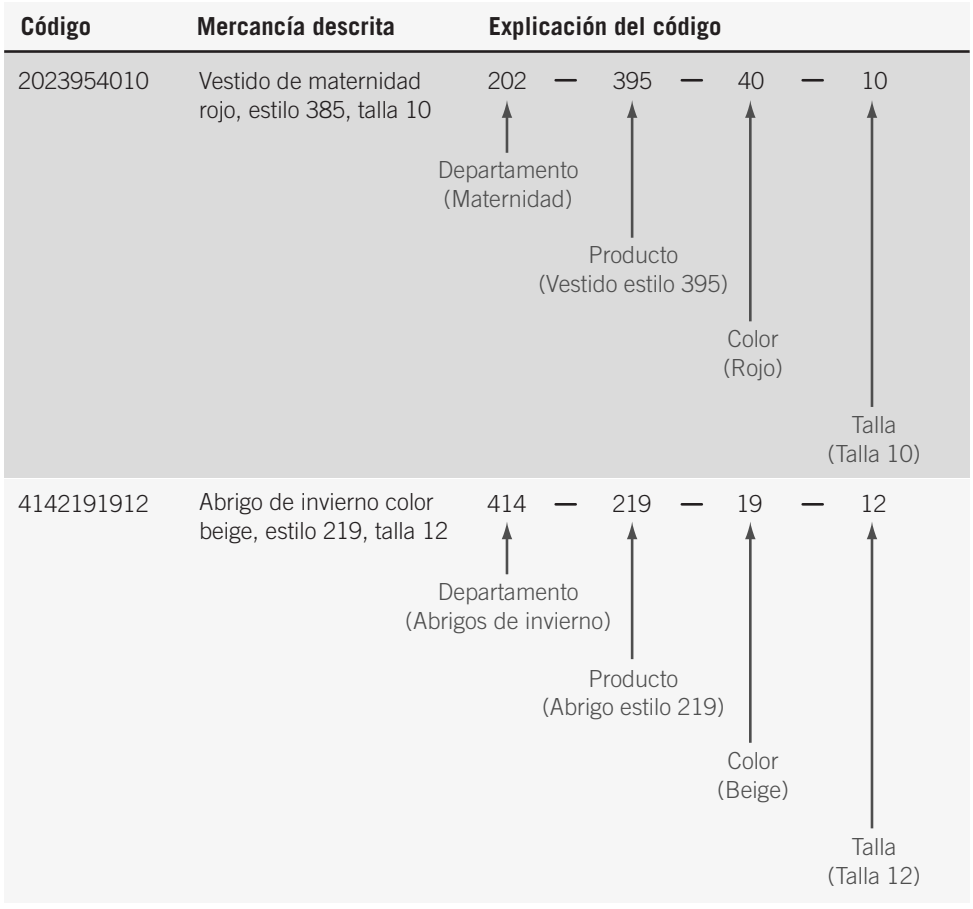


FIGURA 15.7

El uso de un código de subconjunto de dígitos significativos ayuda a los empleados a localizar los artículos que corresponden a un departamento específico.

CÓDIGOS DE SUBCONJUNTO DE DÍGITOS SIGNIFICATIVOS Cuando es posible describir un producto en virtud de su membresía en varios subgrupos, podemos usar un código de subconjunto de dígitos significativos para ayudar a describirlo. El ejemplo de la etiqueta de precios de la tienda de ropa en la figura 15.7 es un ejemplo efectivo de un código de subconjunto de dígitos significativos.

Para el cliente o espectador casual, la descripción del artículo parece ser un largo número; para los vendedores, el número está formado de varios números más pequeños, cada uno de los cuales tiene su propio significado. Los primeros tres dígitos representan el departamento, los siguientes tres el producto, los siguientes dos el color y los últimos dos el tamaño.

Los códigos de subconjunto de dígitos significativos pueden consistir de la información que describe realmente al producto (por ejemplo, el número 10 representa a la talla 10) o de números que se asignan en forma arbitraria (por ejemplo, se asigna un 202 para indicar el departamento de maternidad). En este caso, la ventaja de usar un código de subconjunto de dígitos significativos es que es posible localizar los artículos que pertenecen a cierto grupo o clase. Por ejemplo, si el gerente de la tienda decidió rebajar toda la mercancía de invierno para una venta próxima, los vendedores podrían localizar todos los artículos que pertenezcan a los departamentos del 310 al 449, el bloque de códigos que se utilizan para designar "invierno" en general.

CÓDIGOS MNEMÓNICOS Un mnemónico es una ayuda para la memoria humana. Se considera mnemónico cualquier código que ayuda a una persona a introducir los datos o a un usuario a recordar cómo usar la información. Al utilizar una combinación de letras y símbolos obtenemos una manera muy clara de codificar un producto, de tal forma que el código se pueda ver y comprender con facilidad.

Los códigos de hospitales de la ciudad utilizados por el Banco de Sangre Regional de Buffalo eran mnemónicos, como se muestra en la figura 15.8. Los códigos simples se inventaron precisamente debido a que los administradores y analistas de sistemas del banco de sangre querían asegurar que los códigos de los hospitales fueran fáciles de memorizar y recordar. Los códigos mnemónicos para los hospitales ayudaron a reducir la posibilidad de enviar sangre al hospital incorrecto.

Unicode

Los códigos nos permiten revelar caracteres que por lo general no podemos introducir o ver. Los teclados tradicionales proporcionan conjuntos de caracteres conocidos para las personas que utilizan caracteres alfabéticos occidentales (conocidos como caracteres del latín), pero muchos lenguajes como el griego, japonés, chino o hebreo no utilizan el alfabeto occidental. Estos lenguajes pueden usar letras griegas, glifos o símbolos que representan sílabas o palabras completas. La Organización Internacional para la Estandarización (ISO) definió el conjunto de caracteres Unicode, el cual incluye todos los símbolos de lenguajes estándar y tiene espacio para 65,535 caracteres. Podemos mostrar páginas Web escritas en otros alfabetos si descargamos un editor de métodos de entrada de Microsoft.

Los símbolos de glifos se representan mediante una notación "&#xnnnn;", en la que nnnn representa a una letra o símbolo específico y x significa que se utiliza la notación hexadecimal, o numeración base 16, para representar los caracteres Unicode. Por ejemplo, &#30B3 representa el símbolo *ko* en japonés katakana. El código que se utiliza para la palabra japonesa hola, konichiwa, es こにち&#x308F. En japonés, la palabra se ve así:

こにちわ
ko ni chi wa
hello

El conjunto completo de caracteres Unicode se agrupa por lenguaje; puede encontrarlo en www.unicode.org.

FIGURA 15.8

Los códigos mnemónicos funcionan como ayudas de memoria mediante el uso de una combinación significativa de letras y números.

Código	Hospitales de la ciudad
BGH	**B**uffalo **G**eneral **H**ospital
ROS	**Ros**well Park Memorial Institute
KEN	**Ken**more Mercy
DEA	**Dea**coness Hospital
SIS	**Sis**ters of Charity
STF	**S**aint **F**rancis Hospital
STJ	**S**aint **J**oseph's Hospital
OLV	**O**ur **L**ady of **V**ictory Hospital

Código	Función
1	Entregado
2	Vendido
3	Echado a perder
4	Perdido o robado
5	Devuelto
6	Enviado por transferencia
7	Recibido por transferencia
8	Entrada en el diario (Sumar)
9	Entrada en el diario (Restar)

FIGURA 15.9

Los códigos de función capturan en forma compacta las funciones que debe realizar la computadora.

Solicitar la acción apropiada

A menudo se requieren códigos para instruir a la computadora o al encargado de tomar decisiones sobre la acción que deben tomar. Por lo general, dichos códigos se conocen como códigos de función y comúnmente toman la forma de códigos de secuencia o mnemónicos.

CÓDIGOS DE FUNCIÓN Las funciones, que el analista o programador desea que la computadora realice con los datos, se capturan en códigos de función. La acción de explicar con detalle las actividades a realizar se traduce en un código corto numérico o alfanumérico.

La figura 15.9 muestra ejemplos de un código de función para actualizar el inventario. Suponga que administra un departamento de productos lácteos; si se echara a perder una caja de yogurt, utilizaría el código 3 para indicar este evento. Desde luego que los datos requeridos para la entrada dependen de la función que se necesite. Por ejemplo, para adjuntar datos a un registro o actualizarlo se requeriría sólo la clave del registro y al código de función, mientras que para agregar un registro habría que introducir todos los elementos de datos, incluyendo el código de función.

Lineamientos generales para la codificación

En las secciones anteriores examinamos los propósitos de usar distintos tipos de códigos cuando los humanos y las máquinas introducen y almacenan datos. A continuación examinaremos unas cuantas heurísticas para establecer un sistema de codificación. En la figura 15.10 se resaltan estas reglas.

SEA CONCISO Los códigos deben ser concisos. Los códigos demasiado extensos requieren más pulsaciones de tecla y, en consecuencia, puede haber más errores. Los códigos extensos también implican que el proceso de almacenar la información en una base de datos requerirá más memoria.

Los códigos cortos son más fáciles de recordar para las personas y más fáciles de introducir que los códigos extensos. Si hay que usar códigos extensos, sería conveniente descomponerlos en sub-códigos. Por ejemplo, 5678923453127 se podría descomponer mediante guiones de la siguiente manera: 5678-923-453-127. Esta metodología es mucho más manejable y aprovecha la forma en que sabemos que las personas procesan la información: en trozos pequeños. Algunas veces, los códigos se hacen más extensos de lo necesario por alguna razón. En general los números de tarjetas de crédito son extensos para evitar que las personas adivinen el número de una tarjeta de crédito. Visa y MasterCard utilizan números de 16 dígitos; esto permite generar números exclusivos para nueve billones de personas. Como los números no se asignan en forma secuencial, las probabilidades de adivinar el número de una tarjeta de crédito son muy pocas.

FIGURA 15.10

Hay ocho lineamientos generales para establecer un sistema de codificación.

Para establecer un sistema de codificación, el analista debe:
Mantener los códigos concisos
Mantener los códigos estables
Crear códigos únicos
Permitir ordenar los códigos
Evitar códigos confusos
Mantener los códigos uniformes
Permitir modificar los códigos
Que los códigos sean significativos

OPORTUNIDAD DE CONSULTORÍA 15.1

Aquí parece una jungla

"No puedo soportar esto. He estado buscando este sombrero durante los últimos 45 minutos", se queja Davey mientras balancea una gorra de mapache agarrándola por la cola y pasándola sobre su cabeza. Él es uno de los nuevos trabajadores del almacén para Crokett's, una gran empresa de ventas por catálogo y comercio electrónico. "La hoja del catálogo la llama 'Coo m5-9w/t'. Lo bueno fue que me dijiste que 'Coo' representa la palabra mapache. Después, claro está, pensé en las gorras y busqué por aquí. La encontré aquí, en este bote etiquetado como NIÑOS/GORRA. ¿No sería más fácil si el catálogo y las páginas Web coincidieran con los botes? Para mí, esta factura dice: "Utensilios de cocina metálicos, conjunto de 5-9 piezas con Teflón'. He estado varado en los conjuntos de cocina todo el tiempo".

Daniel, uno de sus compañeros, apenas si lo escucha mientras saca apurado artículos de los botes para surtir otro pedido. "Te acostumbrarás a esto. Deben tenerlo de esta manera para que las computadoras puedan entender la factura después. La mayor parte del tiempo busco el número de la página del catálogo en la factura, después lo busco en el libro y hago algo así como una traducción de vuelta acá… a menos que recuerde haberlo encontrado antes", explica Daniel.

Davey se empeña en decir: "Pero las computadoras son inteligentes y tenemos que surtir demasiados pedidos. Podríamos decir a los de facturación los nombres que tenemos en nuestros botes".

Daniel responde con cinismo: "Sí, claro. Se están muriendo por saber nuestra opinión". Después continúa con un tono más calmado: "Sabes, solíamos tenerlo así, pero cuando llegaron las nuevas computadoras y pasamos a los pedidos telefónicos las 24 horas y a través de Internet, todo cambió. Dijeron que los operadores (y usuarios) tenían que saber más sobre lo que estaban vendiendo (o comprando), por lo que cambiaron sus códigos para que fueran más como una historia".

Davey, sorprendido de la revelación de Daniel, pregunta: "¿Cuál es la historia del código con el que he estado trabajando?".

Después de inspeccionar el código en la factura de la gorra, Daniel responde: "El código con el que estabas trabajando era 'Coo m5-9w/tl'. Después de buscarlo con una rapidez increíble en la computadora, el operador puede decir al cliente: 'Es una gorra de mapache (Coo) para niños (m de masculino) con edades de 5 a 9, con una cola real (w/tl)'. Los árboles no nos dejan ver el bosque debido a sus códigos, pero ya sabe como son en Crockett's. Tienen que lograr la venta".

¿Qué tan importante es el hecho de que los botes y las facturas del almacén se codifiquen en forma inconsistente? Responda en un párrafo. ¿Cuáles son algunos de los problemas que surgen cuando un código parece ser mnemónico pero los empleados nunca reciben una "clave" apropiada para decodificarlo? Describa su respuesta en dos párrafos. ¿Qué cambios realizaría en la codificación de las facturas/el almacén para Crockett's? Documente sus cambios, identifique el tipo de código que usaría y use el código en un ejemplo de un producto que se podría vender en Crockett's. Recuerde descifrarlo también.

MANTENER LOS CÓDIGOS ESTABLES Estabilidad significa que el código de identificación para un cliente no debe cambiar cada vez que se reciban nuevos datos. Anteriormente presentamos un código de derivación alfabética para la lista de suscriptores de una revista. La fecha de expiración no formaba parte del código de identificación del suscriptor debido a que es probable que vaya a cambiar.

No cambie las abreviaturas de código en un sistema mnemónico. Una vez que haya elegido las abreviaturas del código, no trate de modificarlas ya que será muy difícil que el personal de captura de datos se adapte.

ASEGURAR QUE LOS CÓDIGOS SEAN ÚNICOS Para que los códigos funcionen, deben ser únicos. Tome nota de todos los códigos utilizados en el sistema para asegurar que no vaya a asignar el mismo número de código o nombre a los mismos artículos. Los números de código y los nombres son una parte esencial de las entradas en los diccionarios de datos, como vimos en el capítulo 8.

PERMITIR ORDENAR LOS CÓDIGOS Si va a manipular los datos en forma útil, los códigos deben poder ordenarse. Por ejemplo, si desea realizar una búsqueda de texto en los meses del año en orden ascendente, los meses que empiezan con "J" estarían fuera de orden (Julio y después Junio). Los diccionarios se ordenan de esta manera, una letra a la vez y de izquierda a derecha. Por lo tanto, si ordenara en base a MMMDDAAAA, en donde MMM representa el mes, DD el día y AAAA el año, el resultado sería incorrecto.

La figura 15.11 muestra lo que ocurriría si se realizara una búsqueda de texto en distintos formatos de fecha. La tercera columna muestra un problema que formaba parte de la crisis del año 2000 (Y2K) que provocó cierto grado de alarma e incluso apareció en la portada de la revista *Time*.

Una de las lecciones aprendidas es que debemos asegurarnos de que los usuarios puedan hacer lo que queríamos que hicieran con los códigos que creamos. Es mucho más fácil ordenar códigos numéricos que los alfanuméricos; por lo tanto, considere la opción de convertir los códigos a numéricos siempre que sea práctico.

EVITAR CÓDIGOS CONFUSOS Trate de evitar el uso de caracteres de codificación que se vean o suenen igual. Los caracteres O (la letra o) y 0 (el número cero) se confunden con facilidad, al igual que la letra I y el número 1, y también la letra Z y el número 2. Por ello, los códigos tales como B1C y 280Z no son satisfactorios.

Orden incorrecto al usar MMM-DD-AAAA	Orden incorrecto al usar MM-DD-AAAA	Orden incorrecto (problema del año 2000) AA-MM-DD	Orden correcto al usar AAAA-MM-DD
Dic-25-1998	06-04-1998	00-06-11	1997-06-12
Dic-31-1997	06-11-2000	97-06-12	1997-12-31
Jul-04-1999	06-12-1997	97-12-31	1998-06-04
Jun-04-1998	07-04-1999	98-06-04	1998-10-24
Jun-11-2000	10-24-1998	98-10-24	1998-12-25
Jun-12-1997	12-25-1998	98-12-25	1999-07-04
Oct-24-1998	12-31-1997	99-07-04	2000-06-11

FIGURA 15.11

Planifique para poder hacer algo útil con los datos que se hayan introducido. En este ejemplo, la persona que creó los códigos no tomó en cuenta que se tendrían que ordenar.

Un ejemplo de un código potencialmente confuso es el Código Postal Canadiense, como se muestra en la figura 15.12. El formato del código es X9X 9X9, en donde X representa una letra y 9 representa un número. Una ventaja de usar letras en el código es que se permiten más datos en un código de seis dígitos (hay 26 letras y sólo 10 números). Como los canadienses utilizan el código con regularidad, tiene mucho sentido para ellos. Sin embargo, para los extranjeros que envían correo a Canadá puede ser muy difícil saber si el penúltimo símbolo es una Z o un 2.

MANTENER LOS CÓDIGOS UNIFORMES Para que sean efectivos y eficientes para los humanos, los códigos necesitan conformarse a formas que se perciban rápidamente la mayor parte del tiempo. Los códigos que se usan en conjunto, como BUF-234 y KU-3456, son malos debido a que el primero contiene tres letras y tres números, mientras que el segundo sólo tiene dos letras seguidas de cuatro números.

Cuando tenga que agregar fechas, trate de evitar usar los códigos MMDDAAAA en una aplicación, AAAADDMM en otra aplicación y MMDDAA en una tercera aplicación. Es importante mantener la uniformidad en los códigos, así como dentro de los programas.

PERMITIR LA MODIFICACIÓN DE LOS CÓDIGOS. La capacidad de adaptación es una característica clave de un buen código. El analista debe tener en cuenta que el sistema evolucionará con el tiempo, por lo que el sistema de codificación debe ser capaz de abarcar el cambio. El número de clientes debe aumentar, los clientes cambiarán de nombres y los proveedores modificarán la forma en que enumeran sus productos. El analista debe ser capaz de pronosticar los cambios predecibles que los usuarios de la empresa deseen; además se debe anticipar a un amplio rango de necesidades a futuro a la hora de diseñar los códigos.

HACER LOS CÓDIGOS SIGNIFICATIVOS A menos que el analista dese ocultar información de manera intencional, los códigos deben ser significativos. Los códigos efectivos no sólo contienen información; también tienen mucho sentido para las personas que los utilizan. Los códigos significativos son fáciles de entender y recordar; también es muy fácil trabajar con ellos. El trabajo de capturar los datos se vuelve más interesante cuando se trabaja con códigos significativos en vez de introducir sólo una serie de números sin significado.

USAR LOS CÓDIGOS Podemos usar los códigos de varias formas. En los programas de validación, los datos de entrada se comparan con una lista de códigos para asegurar que sólo se hayan introducido códigos válidos. En los programas de informes y consultas, un código almacenado en un archivo se transforma en el significado del código. Los informes y las pantallas no deben mostrar o imprimir el código real. Si lo hicieran, el usuario tendría que memorizar los significados del código o buscarlos en un manual. Los códigos se utilizan en los programas de GUI para crear listas desplegables.

Formato de código para el Código Postal Canadiense X9X 9X9

Código manuscrito	Código real	Ciudad, Provincia	Problema
L8S 4M4	L8S 4M4	Hamilton, Ontario	La S parece un 5
T3A ZE5	T3A 2E5	Calgary, Alberta	El 2 parece una Z El 5 parece una S
LOS 1JO	L0S 1J0	Niagara-on-the-Lake, Ontario	El cero y la o se ven casi idénticos La S parece un 5 El 1 parece una I

FIGURA 15.12

Combinar caracteres similares en los códigos puede provocar errores.

OPORTUNIDAD DE CONSULTORÍA 15.2

Pescar un código de verano

Vicky retira los dedos del teclado y se inclina en su estación de trabajo para verificar las letras de las facturas apiladas en frente de ella. "¡Qué diantres!", Vicky pregunta en voz alta mientras escudriña las letras que codifican las ciudades a donde se van a enviar los pedidos.

Shelly Overseer, su supervisora, que por lo general se sienta a un par de estaciones de trabajo de distancia, pasa por ahí y ve la consternación de Vicky. "¿Cuál es el problema? ¿Acaso olvidaron los representantes de ventas escribir el código de la ciudad otra vez?".

Vicky gira en su silla para ver a Shelly de frente. "No, los códigos están aquí, pero están raros. Por lo general utilizamos un código de tres letras, ¿verdad? Como CIN para Cincinnati, SEA para Seattle, MIN para Minneapolis, BUF para Buffalo. Pero aquí hay sólo códigos de *cinco* letras".

"Mira", dice Vicky mientras levanta la factura para mostrarla a Shelly. "CINNC, SEATT, MINNE. Me tomará todo el día introducir estos códigos. En serio, me estoy tardando mucho. Tal vez haya un error. ¿No puedo simplemente usar el estándar?".

Shelly retrocede de la estación de trabajo de Vicky como si el problema fuera contagioso. Se disculpa y dice: "Son los de medio tiempo. Están aprendiendo sobre ventas ahora, y a la gerencia le preocupaba que hicieran un desorden con sus ciudades. Creo que tiene algo que ver con haber mezclado Newark y Nueva Orleans en los últimos pedidos. Por lo tanto, un comité decidió agregar dos letras para que las ciudades se pudieran reconocer con más facilidad. Esos chicos no pueden aprender todo lo que sabemos en una noche, incluso cuando lo intentan. Pero es sólo hasta agosto 19, cuando los empleados de medio tiempo regresan a la escuela.

Mientras Vicky regresa con desánimo a su teclado, Shelly sacude su cabeza con compasión y dice: "Se que es estresante y que te sientes miserable, pero no te preocupes. Ya se te pasará. Es sólo un código de verano".

¿Qué lineamientos generales de codificación ha pasado por alto la gerencia en su decisión de usar un código de verano para las ciudades? Haga una lista. ¿Cuál es el efecto sobre el personal de captura de datos de tiempo completo al cambiar los códigos para facilitar el trabajo al personal de ayuda temporal? Responda en dos párrafos. ¿Qué impacto a futuro podría tener el cambio temporal en los códigos en cuanto a ordenar y recuperar los datos introducidos durante el periodo de verano? Describa en dos párrafos estas implicaciones. ¿Qué cambios puede usted sugerir de manera que los empleados de medio tiempo no se confundan con los códigos a corto plazo? En un memo para el supervisor de este grupo de trabajo, haga una lista de cinco a siete cambios en los procedimientos de captura o introducción de datos que se puedan realizar para adaptarse a las contrataciones de corto plazo sin perturbar las actividades comerciales regulares. Indique en un párrafo cómo se puede lograr esta meta sin comprometer la productividad del personal que introduce los datos.

CAPTURA DE DATOS EFECTIVA Y EFICIENTE

Para asegurar la calidad de los datos que los usuarios introducen al sistema, es importante capturar los datos con efectividad. La captura de datos ha recibido cada vez más atención como el punto en el procesamiento de la información en el que se pueden obtener excelentes ganancias en productividad. El proceso de capturar datos ha mejorado de manera considerable en las últimas cuatro décadas, a medida que avanzamos de los sistemas de varios pasos, lentos y propensos a errores como las tarjetas perforadas, hasta el uso de sistemas sofisticados como el reconocimiento de caracteres (OCR), los códigos de barras y las terminales de punto de venta.

Qué capturar

La decisión sobre lo que se debe capturar es más importante que la interacción del usuario con el sistema. En definitiva es imprescindible para hacer que la interfaz final valga la pena, dado que aún es válido el adagio "entra basura, sale basura".

Los analistas de sistemas y los usuarios son los que se encargan de las decisiones sobre los datos que se deben capturar para introducirlos al sistema. Gran parte de lo que se captura es específico para la empresa en particular. Capturar datos, introducirlos, almacenarlos y recuperarlos son todos procesos costosos. Con todos estos factores en mente, determinar qué es lo que se debe capturar se convierte en una decisión importante.

Hay dos tipos de datos a introducir: los datos que *cambian* o *varían* con cada transacción y los datos que *diferencian* en forma concisa el elemento específico que se está procesando de los demás elementos.

Un ejemplo de datos que se pueden modificar es la cantidad de provisiones que se compran cada vez que una empresa de publicidad coloca un pedido con el mayorista de artículos de oficina. Como las cantidades cambian dependiendo del número de empleados en la empresa de publicidad y de cuántas cuentas estén manejando, hay que introducir datos sobre la cantidad cada vez que se coloca un pedido.

Un ejemplo de datos de diferenciación es incluir el número de Seguro Social y las primeras tres letras del nombre de un/una paciente en su registro. De esta manera, el/la paciente se diferencia en forma única de los demás pacientes en el mismo sistema.

Dejar que la computadora haga el resto

Al considerar qué datos es necesario capturar para cada transacción y qué datos hay que dejar para que el sistema los introduzca, el analista de sistemas debe aprovechar lo que las computadoras hacen mejor. En el anterior ejemplo de la agencia de publicidad y el pedido de artículos de oficina, no es necesario que el operador que introduce el pedido de papelería vuelva a capturar la descripción de cada artículo siempre que se reciba un pedido. La computadora puede almacenar y acceder a esta información con facilidad.

Las computadoras pueden manejar de manera automática las tareas repetitivas, como registrar el tiempo de la transacción, calcular nuevos valores a partir de la entrada, además de almacenar y recuperar datos bajo demanda. Al emplear las mejores características de las computadoras, el diseño de la captura de datos eficiente evita la entrada de datos innecesarios, que a su vez alivia muchos de los errores humanos y el aburrimiento, además de que permite a las personas concentrarse en tareas creativas o de más alto nivel. Podemos escribir software para pedir al usuario que introduzca la fecha de hoy o capturar la fecha del reloj interno de la computadora. Una vez introducido este dato, el sistema procede a usar esa fecha en todas las transacciones procesadas en esa sesión de entrada de datos.

Un ejemplo de primera sobre reutilización de datos que se introducen una vez es el del centro de bibliotecas de cómputo en línea (OCLC) que utilizan miles de bibliotecas en los Estados Unidos. OCLC se creó a partir de la idea de que cada elemento comprado por una biblioteca sólo debería catalogarse una vez en toda su vida. Después de introducir un elemento, la información de clasificación va a la enorme base de datos de OCLC y se comparte con las bibliotecas participantes. En este caso, la implementación del concepto simple de introducir datos sólo una vez ahorra una enorme cantidad de tiempo al introducir los datos.

También hay que tener en cuenta el poder de cálculo de la computadora al decidir qué *no* debemos volver a introducir. Las computadoras son adeptas a los cálculos extensos, en los que se utilizan datos que ya se introdujeron antes.

Por ejemplo, la persona que introduce los datos puede introducir los números de vuelo y el número de cuenta de un viaje aéreo realizado por un cliente que pertenezca a un programa de incentivos para viajeros frecuentes. Después, la computadora calcula el número de millas acumuladas para cada vuelo, lo agrega a las millas que ya están en la cuenta del cliente y actualiza el total de millas acumuladas en la cuenta. La computadora también puede marcar una cuenta que, en virtud de la gran cantidad de millas acumuladas, sea elegible para un premio. Aunque toda esta información puede aparecer en la cuenta actualizada del cliente, los únicos datos nuevos introducidos fueron los números de los vuelos.

En sistemas que utilizan una interfaz gráfica de usuario (GUI), a menudo los códigos se almacenan como una función o como una tabla separada en la base de datos. Hay desventajas al crear demasiadas tablas, ya que el software debe buscar los registros que coincidan de cada tabla, lo cual podría ocasionar un acceso lento. Si los códigos son relativamente estables y cambian con poca frecuencia, tal vez se puedan almacenar como una función de la base de datos. Si los códigos cambian con frecuencia, se almacenan en una tabla para que se puedan actualizar con facilidad.

La figura 15.13 muestra cómo se utiliza una lista desplegable para seleccionar los códigos para agregar o modificar un registro en la tabla CLIENTE (CUSTOMER). Cabe mencionar que el código está almacenado, pero

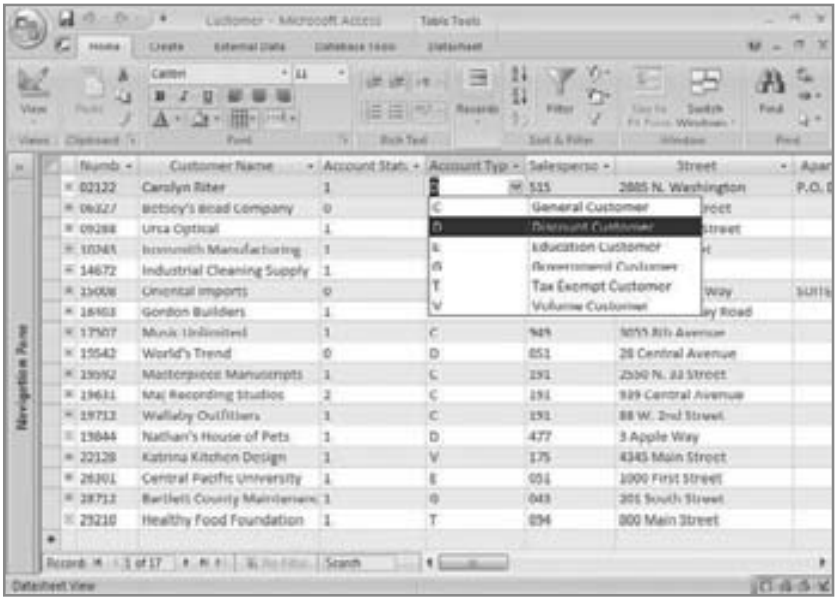


FIGURA 15.13

Uso de una tabla de códigos en una lista desplegable. Esta lista se utiliza para seleccionar un código para agregar o modificar un elemento en un registro.

la lista desplegable muestra tanto el código como su significado. Este método ayuda a asegurar la precisión, ya que el usuario no tiene que adivinar el significado del código y no hay probabilidad de que escriba uno inválido.

Cómo evitar cuellos de botella y pasos adicionales

En la terminología relacionada con la entrada de datos, un cuello de botella describe gráficamente el efecto del angostamiento de la botella en el flujo del contenido: los datos se vierten con rapidez en la amplia boca del sistema, sólo para desacelerar drásticamente en su "cuello" debido a la incapacidad del sistema para procesar el volumen o detalle de los datos que se están introduciendo. Una manera en la que podemos evitar un cuello de botella es asegurarnos que haya la capacidad suficiente para manejar los datos que se están introduciendo.

Las formas de evitar los pasos adicionales se determinan no sólo durante el análisis, sino también cuando los usuarios empiezan a interactuar con los prototipos del sistema. Entre menos pasos se requieran para introducir datos, menor será la probabilidad de introducir errores. Por lo tanto, además del ahorrar en mano de obra, el hecho de evitar pasos adicionales también es una forma de preservar la calidad de los datos. Una vez más, usar un sistema en tiempo real en línea que capture los datos de los clientes sin necesitar la complejidad de un formulario es un excelente ejemplo de cómo ahorrar pasos en el proceso de entrada de datos.

Empezar con un buen formulario

Podremos lograr una captura de datos efectiva sólo si consideramos con anticipación lo que debe contener el documento de origen. El operador introduce los datos del documento de origen (que por lo general es un tipo de formulario); este documento es la fuente de una gran cantidad de todos los datos del sistema. Los sistemas en línea (o los métodos especiales de entrada de datos, como los códigos de barras) pueden sortear la necesidad de un documento de origen, pero es común crear de todas formas algún tipo de formulario en papel, como un recibo.

Cuando se cuenta con formularios efectivos, no es necesario volver a introducir la información que la computadora ya haya almacenado, ni datos como la fecha y la hora que la computadora puede determinar de manera automática. En el capítulo 11 vimos con detalle cómo debemos diseñar un formulario o documento de origen para maximizar su utilidad para capturar datos y minimizar el tiempo que necesitan invertir los usuarios para introducir datos a partir de ese formulario o documento.

Elegir un método de entrada de datos

Hay varios métodos eficientes de entrada de datos disponibles; para elegir uno de ellos hay que tener en cuenta diversos factores, incluyendo la necesidad de velocidad, precisión y capacitación de los usuarios; el costo del método de entrada de datos (si requiere de muchos materiales o de mucha mano de obra); y los métodos que se están usando actualmente en la organización.

TECLADOS El teclado es el método más antiguo de entrada de datos; sin duda es con el que los miembros de las organizaciones están más familiarizados. Se han realizado algunas mejoras en los teclados con el paso de los años. Las características más recientes incluyen teclas de función especial para abrir programas, teclas que se utilizan para desplazarse por las pantallas y explorar la Web, y teclas que se pueden programar con macros para reducir el número de pulsaciones de teclas requeridas. Los teclados ergonómicos e infrarrojos, junto con los teclados y ratones habilitados con Bluetooth, son también mejoras considerables.

RECONOCIMIENTO ÓPTICO DE CARACTERES El reconocimiento óptico de caracteres (OCR) permite a un usuario leer la entrada desde un documento de origen mediante un escáner óptico en vez de usar los medios magnéticos que hemos visto hasta ahora. El uso de dispositivos OCR puede agilizar la entrada de datos de un 60 a un 90 por ciento, en comparación con algunos métodos de tecleo.

El aumento en la velocidad del OCR se debe a que no hay que codificar ni teclear los datos que provienen de los documentos de origen. Esto elimina muchos de los pasos que se requieren en otros dispositivos de entrada en los que se consume mucho tiempo y puede haber muchos errores. El método del OCR demanda pocas habilidades del empleado, lo que significa que su capacitación no tiene que ser muy especializada; esto resulta en menos errores y menos tiempo invertido por los empleados en esfuerzos redundantes. Además, descentraliza la responsabilidad de obtener datos de calidad para asignarla directamente a la unidad que los está generando. El OCR, que ahora está disponible para todos, tiene un uso adicional y muy práctico: transformar los faxes en documentos que se puedan editar.

OTROS MÉTODOS DE ENTRADA DE DATOS Hay otros métodos de entrada de datos que se están empleando cada vez con más frecuencia. La mayoría de estos métodos reducen los costos de mano de obra al requerir menos habilidades del operador o poca capacitación, acercan cada vez más el proceso de entrada de datos a su origen y eliminan la necesidad de un documento de origen. Al hacer esto se han convertido en métodos de entrada de datos veloces y muy confiables. Los métodos de entrada de datos que veremos en las siguientes subsecciones incluyen el reconocimiento de caracteres de tinta magnética, los formularios de detección de marcas, los formularios perforados y las tiras de datos.

Reconocimiento de caracteres de tinta magnética Los caracteres de tinta magnética se encuentran en la parte inferior de los cheques bancarios y algunas facturas de tarjetas de crédito. Este método es similar al OCR en cuanto a que se leen caracteres especiales, pero su uso es limitado. La entrada de datos por medio del reconocimiento de caracteres de tinta magnética (MICR) se realiza a través de una máquina que lee e interpreta una sola línea de material codificado con tinta formada de partículas magnéticas.

Algunas ventajas del uso de MICR son: 1) es un método confiable y de alta velocidad que no es susceptible a confundir las marcas fuera de lugar, ya que no están codificadas en forma magnética; 2) si se requiere en todos los cheques para retirar fondos, sirve como una medida de seguridad contra los cheques falsos, y 3) el personal de entrada de datos puede ver los números que componen el código, si es necesario verificarlo.

Formularios de detección de marcas Estos formularios permiten introducir datos por medio de un escáner que detecta en dónde se hicieron las marcas en formularios especiales. Un uso común es para calificar las hojas de respuestas de los cuestionarios a las encuestas, como se muestra en la figura 15.14. El personal requiere de una capacitación muy básica; además se puede procesar un alto volumen de formularios con rapidez.

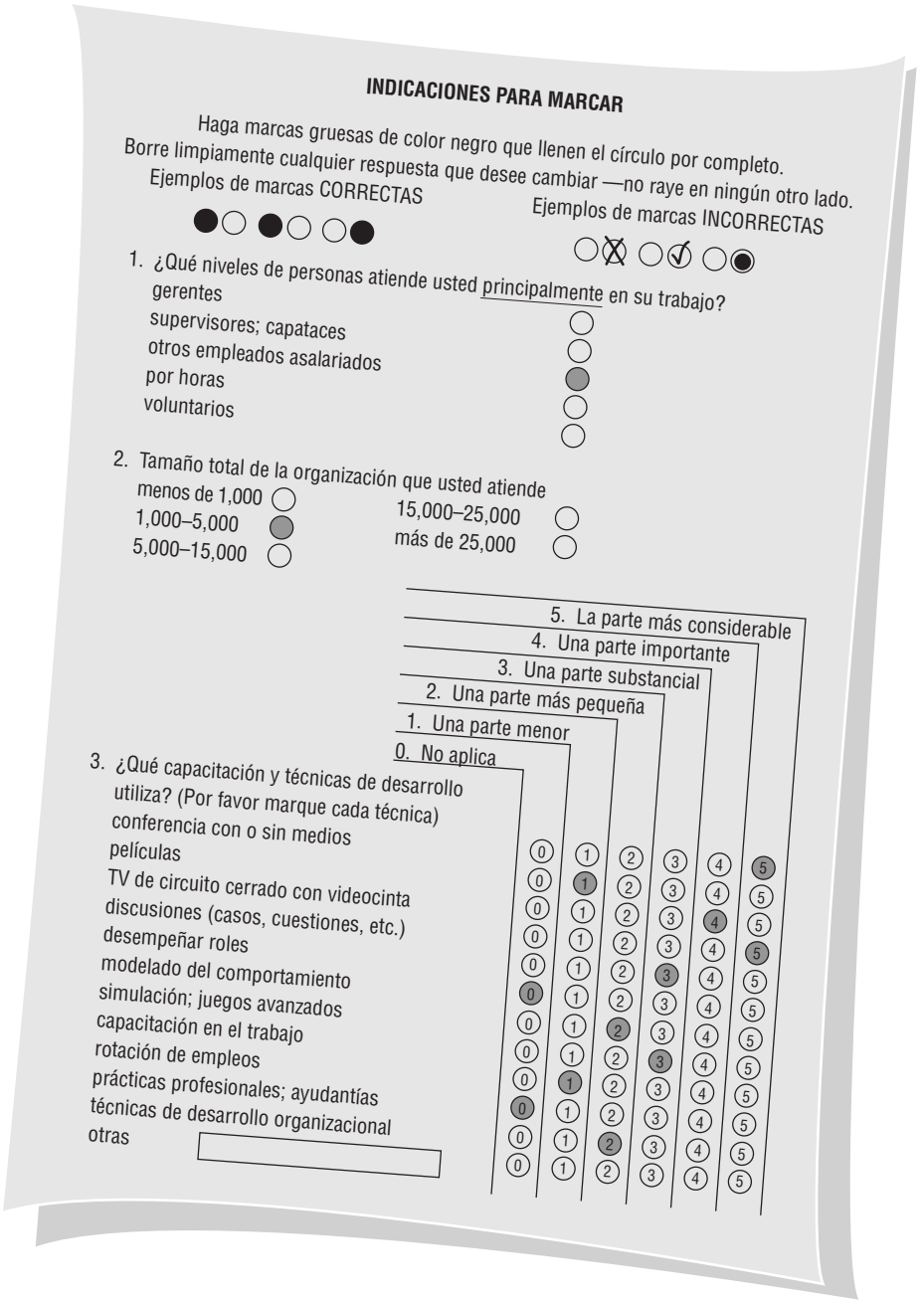


FIGURA 15.14

Un formulario de detección de marcas que se puede leer mediante un escáner agiliza el proceso de entrada de datos.

Una desventaja de los formularios de detección de marcas es que, aunque los lectores pueden determinar si se hizo una marca, no pueden interpretarla de la manera en que lo hacen los lectores ópticos de caracteres. Esto implica que se pueden introducir marcas fuera de lugar en los formularios como datos incorrectos. Además, las opciones se limitan a las respuestas proporcionadas en el formulario de detección de marcas, los formularios presentan dificultades al capturar datos alfanuméricos debido al espacio requerido para un conjunto completo de letras y números, y es fácil que los usuarios que llenan formularios de detección de marcas se confundan y pongan una marca en una posición incorrecta.

Códigos de barras Por lo general, los códigos de barras aparecen en las etiquetas de productos, pero también aparecen en brazaletes de identificación de pacientes en los hospitales y casi en cualquier contexto en el que una persona u objeto necesite reportarse como entrada o salida en cualquier tipo de sistema de inventario. Podemos considerar a los códigos de barras como metacódigos, o códigos que codifican otros códigos, debido a que aparecen como una serie de bandas estrechas y anchas en una etiqueta, las cuales codifican números o letras. A su vez, estos símbolos tienen acceso a los datos de los productos almacenados en la memoria de la computadora. Se proyecta un rayo de luz de un escáner o lápiz óptico a través de las bandas en la etiqueta para confirmar o registrar datos sobre el producto que se está escaneando.

Una etiqueta de código de barras como la que se muestra en la figura 15.15 incluye los siguientes elementos de codificación para un producto específico de abarrotes: el número de identificación del fabricante, el número de identificación del producto, un código para verificar la precisión del escaneo y códigos para marcar el principio y fin del escaneo.

Los códigos de barras ofrecen un grado de precisión extraordinariamente elevado para la entrada de datos. Ahorra costos de mano de obra a los vendedores, ya que no hay que marcar el precio de cada artículo por separado. Además, la codificación de barras permite capturar de manera automática datos que se pueden utilizar para reordenar, rastrear el inventario con mayor precisión y pronosticar las necesidades a futuro. Los precios de venta u otros cambios en el significado de los códigos de barras se introducen en el procesador central, con lo cual nos ahorramos el problema de rebajar muchos artículos.

Un nuevo uso de los códigos de barras es el de rastrear las compras de un individuo con su tarjeta de crédito, con el fin de crear un perfil de cliente que pueda a su vez utilizarse para refinar las estrategias de marketing para ese individuo o tipo de consumidor. Se desarrollan nuevos dispositivos de entrada en forma constante.

RFID La identificación por radiofrecuencia, conocida comúnmente como RFID, permite recolectar datos en forma automática mediante etiquetas RFID o transpondedores que contienen un chip y una antena. Una etiqueta RFID puede o no tener su propia fuente de energía. Si no cuenta con su propia fuente, la antena provee suficiente potencia de una señal entrante para energizar el chip y transmitir una respuesta. Las etiquetas RFID se pueden adjuntar a productos, paquetes, animales o incluso humanos, de manera que se pueda identificar el elemento o persona mediante una frecuencia de radio.

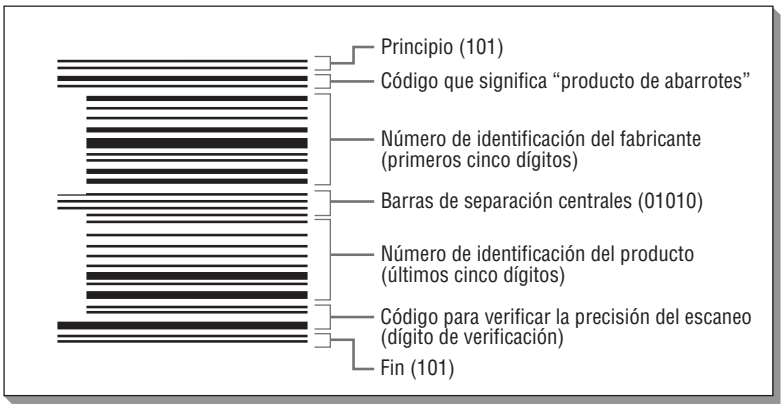
A las etiquetas RFID también se les conoce como tarjetas de proximidad debido a su rango limitado; pueden ser pasivas o activas. Las etiquetas RFID pasivas no tienen fuente de energía interna; las activas sí. Las etiquetas pasivas son económicas (menos de 5 centavos de dólar por etiqueta) y por lo general tienen el tamaño de una estampilla postal. Se utilizan en las grandes tiendas de menudeo, incluyendo Wal-Mart y Target. Wal-Mart ha estado trabajando mucho con la tecnología RFID para mejorar sus procesos de administración de inventario y cadena de suministro.

Las etiquetas activas son mucho más confiables, ya que tienen su propia fuente de energía. El Departamento de Defensa de los EE.UU. ha utilizado estas etiquetas para minimizar los costos relacionados con la logística e incrementar la visibilidad de la cadena de suministro. Las etiquetas activas cuestan sólo unos cuantos dólares cada una.

FIGURA 15.15

El código de barras, como se muestra en esta etiqueta para un producto de abarrotes, ofrece mucha precisión en la entrada de datos.

Utilizada con el permiso de Uniform Code Council, Dayton, Ohio.



OPORTUNIDAD DE CONSULTORÍA 15.3

Entrar o no entrar: esa es la cuestión

"Acabo de asumir la presidencia de Elsinore Industries", dice Rose N. Krantz. "Formamos parte de una industria de fabricación de casitas pequeñas, donde fabricamos aldeas de juguete para niños de siete años en adelante. Nuestras pequeñas aldeas consisten de varios kits para construir lo que quieran los niños a partir de cubos de plástico que se conectan entre sí, que pueden ser edificios básicos como la presidencia municipal, la estación de policía, la estación de gasolina y un carrito para hot-dogs. Cada kit tiene un número de pieza único de 200 a 800, pero no se utilizan todos los números. Los precios de mayoreo varían de $54.95 por la presidencia municipal a $1.79 para un carrito de hot-dogs.

"He estado melancólica en relación con lo que descubrí desde que entré a trabajar en Elsinore. 'Algo huele a podrido' aquí, como dice un famoso dramaturgo. De hecho, el sistema de facturación estaba tan descontrolado que he estado trabajando sin cesar con nuestra contadora, Gilda Stern", declara Krantz.

"Me gustaría que nos ayudara a ordenar todo", continúa Rose. "Enviamos a 12 almacenes de distribución por todo el país. Cada factura que escribimos incluye el número de almacén, del 1 al 12, su dirección física y el código postal estadounidense (zip code). También ponemos en cada factura la fecha en que surtiremos el pedido, los números de código para los kits de la aldea que ordenen, una descripción de cada kit, el precio por artículo y la cantidad de cada kit ordenado. Desde luego que también incluimos los subtotales de los costos de los kits, los costos de envío y el total que el almacén nos debe. No se agregan impuestos, ya que lo que nosotros les vendemos lo revenden a las jugueterías en los 50 estados. Quiero que usted nos ayude a diseñar un sistema computarizado de introducción de pedidos que forme parte del sistema de facturación de Elsinore Industries".

Para su diseño de un sistema de entrada de datos para Elsinore, tome en cuenta todos los objetivos para entrada de datos que vimos en el capítulo. Dibuje las pantallas que sean necesarias para ilustrar su diseño. ¿Cómo puede hacer que el sistema de introducción de pedidos sea eficiente? Responda en un párrafo. Especifique los datos que se pueden almacenar y recuperar, junto con los datos que se deben introducir de nuevo para cada pedido. ¿Cómo se puede evitar el trabajo innecesario? Escriba un párrafo para explicar por qué el sistema que usted propone es más eficiente que el anterior. ¿Cómo se puede asegurar la precisión de los datos? Mencione tres estrategias que funcionen con el tipo de datos que se van a introducir para Elsinore Industries.

Para capturar los datos en una etiqueta RFID se requiere un lector. Este lector activa la etiqueta para que se pueda leer. El lector decodifica los datos y el código único de producto en el chip dentro de la etiqueta; después lo pasa a una computadora anfitriona que procesa los datos.

Un ejemplo es el paso de peaje electrónico que se utiliza en los vehículos que viajan por las carreteras de cuota. Se puede colocar un transpondedor RFID en el parabrisas para leerlo cada vez que el vehículo pasa por una caseta de cuota. El lector RFID de la caseta de cuota también puede actuar como escritor, por lo que se puede almacenar un saldo en el chip RFID.

El Metro de Moscú fue el primer sistema de transporte en usar tarjetas inteligentes RFID en 1998. Otras aplicaciones incluyen el rastreo de ganado para identificar la manada de origen, lo cual permite rastrear mejor la enfermedad de las vacas locas, así como el rastreo RFID en librerías, servicios de equipaje de aerolíneas, farmacias e incluso pacientes o internos.

Las etiquetas RFID son de uso común en la mayoría de las aplicaciones relacionadas con envíos. La tecnología pronto se utilizará en las transacciones de efectivo electrónico en general. Incluso pueden llegar a reemplazar los códigos RPC, ya que sus ventajas incluyen la seguridad (al reducir el número de artículos robados) y no requerir escaneo (simplemente se pasan por la zona del lector).

Pero la tecnología RFID no escapa de la controversia. La privacidad es una cuestión importante. Un individuo que paga un artículo etiquetado con tarjeta de crédito o con una tarjeta de comprador corre el riesgo de ser identificado.

El analista de sistemas necesita pensar en los clientes involucrados y sus derechos al considerar si esta tecnología se adapta a la aplicación que está diseñando.

ASEGURAR LA CALIDAD DE LOS DATOS POR MEDIO DE LA VALIDACIÓN DE LA ENTRADA

Hasta ahora hemos visto cómo asegurar la captura efectiva de datos hacia documentos de origen y la entrada eficiente de datos al sistema por medio de diversos dispositivos de entrada. Aunque estas condiciones son necesarias para asegurar datos de calidad, por sí solas no son suficientes.

Los errores no se pueden descartar por completo; además no podemos sobre enfatizar la importancia crítica de atrapar errores durante la entrada, *antes* de procesarlos y almacenarlos. La maraña de problemas que surgen debido a una entrada incorrecta puede ser una pesadilla, en especial porque una gran cantidad de problemas tardan

FIGURA 15.16
Es importante validar la entrada para asegurar que se eliminen lo más pronto posible la mayoría de los problemas potenciales con los datos.

Este tipo de validación	Puede evitar estos problemas
Validar transacciones de entrada	Enviar los datos incorrectos Datos enviados por una persona no autorizada Pedir al sistema que realice una función inaceptable
Validar datos de entrada	Datos faltantes Longitud de campo incorrecta Datos con una composición inaceptable Datos fuera de rango Datos inválidos Datos que no coinciden con los datos almacenados

mucho tiempo en hacerse ver. El analista de sistemas debe suponer que *ocurrirán* errores en los datos y debe trabajar con los usuarios para diseñar pruebas de validación de la entrada, de manera que pueda evitar que se procesen y almacenen datos erróneos, ya que los errores iniciales que no se descubren durante largos periodos de tiempo son costosos y se requiere mucho tiempo para corregirlos.

No es posible imaginar todo lo que puede salir mal con la entrada, pero debemos cubrir los tipos de errores que provocan el mayor porcentaje de los problemas. En la figura 15.16 se proporciona un resumen de los problemas potenciales que debemos considerar al validar la entrada.

Validar las transacciones de entrada

El proceso de validar las transacciones de entrada se realiza principalmente por medio de software, lo cual es responsabilidad del programador, pero es importante que el analista de sistemas conozca los problemas comunes que pueden invalidar una transacción. Las empresas comprometidas con la calidad incluirán chequeos de validez como parte de su software de rutina.

Pueden ocurrir tres problemas principales con las transacciones de entrada: que se envíen los datos incorrectos al sistema, que una persona no autorizada envíe los datos o que se pida al sistema que realice una función inaceptable.

ENVIAR LOS DATOS INCORRECTOS Un ejemplo de enviar datos incorrectos al sistema sería tratar de introducir el número de seguro social de un paciente en el sistema de nóminas de un hospital. Por lo general este error es accidental, pero hay que marcarlo antes de procesar los datos.

DATOS ENVIADOS POR UNA PERSONA NO AUTORIZADA El sistema también debe ser capaz de descubrir si una persona no autorizada envía datos por lo demás correctos. Por ejemplo, sólo el farmacéutico supervisor podría introducir los totales del inventario para las sustancias controladas en la farmacia. La invalidación de las transacciones enviadas por un individuo no autorizado se aplica a las cuestiones de privacidad y seguridad relacionadas con los sistemas de nómina y los registros de evaluación de empleados que determinan los niveles de los pagos, las promociones o la disciplina; los archivos que contienen secretos comerciales; y los archivos que contienen información clasificada, como los datos de defensa nacional.

PEDIR AL SISTEMA QUE REALICE UNA FUNCIÓN INACEPTABLE El tercer error que invalida las transacciones de entrada es pedir al sistema que realice una función inaceptable. Por ejemplo, sería lógico que el gerente de recursos humanos actualizara el registro existente de un empleado actual, pero sería inválido pedir al sistema que creara un nuevo archivo en vez de sólo actualizar un registro existente.

Validar los datos de entrada

Es esencial que los datos de entrada (junto con las transacciones solicitadas) sean válidos. Se pueden incorporar varias pruebas al software para asegurar esta validez. Vamos a considerar ocho posibles formas de validar la entrada.

PRUEBA DE DATOS FALTANTES En el primer tipo de prueba de validez se examinan los datos para ver si hay elementos faltantes. En algunas situaciones deben estar presentes *todos* los datos. Por ejemplo, un archivo del seguro social para pagar los beneficios por retiro o discapacidad sería inválido si no incluyera el número de seguro social del beneficiario.

Además, el registro debe incluir los datos clave que distinguen a un registro de todos los demás, junto con el código de función que indica a la computadora lo que debe hacer con los datos. El analista de sistemas necesita interactuar con los usuarios para determinar qué elementos de datos son esenciales y averiguar si alguna vez ocurrieron casos excepcionales que permitieran considerar los datos válidos, incluso cuando faltaran algunos elementos de datos. Por ejemplo, una segunda línea de dirección que contenga un número de apartamento o la inicial del segundo nombre de una persona tal vez no sea una entrada requerida.

PRUEBA DE LONGITUD DE CAMPO CORRECTA En el segundo tipo de prueba de validez se comprueba la entrada para ver si tiene la longitud correcta para el campo. Por ejemplo, si la estación climatológica de Omaha, Nebraska se reporta con la computadora del servicio climatológico nacional y por error provee un código de ciudad de dos letras (OM) en vez del código nacional de ciudad de tres letras (OMA), los datos de entrada se podrían considerar inválidos y por ende no se procesarían.

PRUEBA DE CLASE O COMPOSICIÓN La prueba de la validez de la clase o composición verifica que todos los campos de datos que deben estar compuestos exclusivamente de números no contengan letras y viceversa. Al usar una prueba de composición, el programa no debe aceptar un número de cuenta de American Express que incluya tanto letras como números.

PRUEBA DE RANGO O SENSATEZ Las pruebas de validez de rango o sensatez son en realidad medidas de entrada de sentido común, las cuales responden a la pregunta de si los datos se encuentran dentro de un rango aceptable o si son razonables dentro de ciertos parámetros predeterminados. Por ejemplo, si un usuario estaba tratando de verificar una fecha de envío propuesta, la prueba de rango no permitiría una fecha de envío en el día 32 de octubre, ni aceptaría un envío en el mes 13, siendo los rangos respectivos de 1 a 31 días y de 1 a 12 meses.

Una prueba de sensatez averigua si el elemento tiene sentido para la transacción. Por ejemplo, al agregar un nuevo empleado a la nómina, no sería razonable introducir una edad de 120 años. Las pruebas de sensatez se utilizan para los datos que son continuos; es decir, datos que tienen un rango uniforme de valores. Estas pruebas pueden incluir un límite inferior, un límite superior o ambos.

PRUEBA DE VALORES INVÁLIDOS El proceso de comprobar que la entrada no tenga valores inválidos funciona si sólo hay unos cuantos valores válidos. Esta prueba no es viable para las situaciones en las que los valores no son restringidos ni predecibles. Este tipo de prueba es útil para verificar las respuestas donde los datos se dividen en un número limitado de clases. Por ejemplo, una empresa de correduría divide las cuentas en tres clases solamente: clase 1 = cuenta activa, clase 2 = cuenta inactiva y clase 3 = cuenta cerrada. Si los datos se asignan a cualquier otra clase por medio de un error, los valores son inválidos. Por lo general las verificaciones de valores se realizan para los datos discretos, que sólo tienen ciertos valores. Si hay muchos valores, por lo general se almacenan en un archivo de tabla de códigos. Al tener los valores en un archivo es más fácil agregar o modificar valores.

VERIFICACIONES DE REFERENCIAS CRUZADAS Estas verificaciones se utilizan cuando un elemento tiene una relación con otro. Para realizar una verificación de referencia cruzada, cada campo debe estar correcto por sí solo. Por ejemplo, el precio al que se vende un artículo debe ser mayor que el costo que se paga por él. Hay que introducir el precio, el cual debe ser numérico y mayor de cero. Se utiliza el mismo criterio para validar el costo. Cuando tanto el precio como el costo son válidos, se pueden comparar.

Una verificación geográfica es otro tipo de verificación de referencia cruzada. En los Estados Unidos se puede utilizar la abreviatura de estado para asegurar que un código de área telefónica sea válido para cierto estado y que los primeros dos dígitos del código postal sean válidos para el estado.

PRUEBA PARA COMPARAR CON DATOS ALMACENADOS La siguiente prueba de validez de los datos de entrada que veremos es en la que éstos se comparan con los datos que ya están almacenados en la computadora. Por ejemplo, un número de pieza recién introducido se puede comparar con el inventario de piezas completo para asegurar que el número exista y se esté introduciendo en forma correcta.

CONFIGURAR CÓDIGOS DE AUTOVALIDACIÓN (DÍGITOS DE VERIFICACIÓN) Otro método para asegurar la precisión de los datos, en especial los números de identificación, es utilizar un dígito de verificación en el mismo código. Este procedimiento implica empezar con un código numérico original, realizar algunas operaciones matemáticas para llegar a un dígito de verificación derivado y después agregar ese dígito de verificación al código original. El proceso matemático implica multiplicar cada uno de los dígitos del código original por ciertos pesos predeterminados, sumar estos resultados y después dividir esta suma por un número módulo. El número módulo es necesario debido a que la suma por lo general es un número grande y necesitamos reducir el resultado a un solo dígito. Por último, el residuo se resta del número módulo para obtener el dígito de verificación.

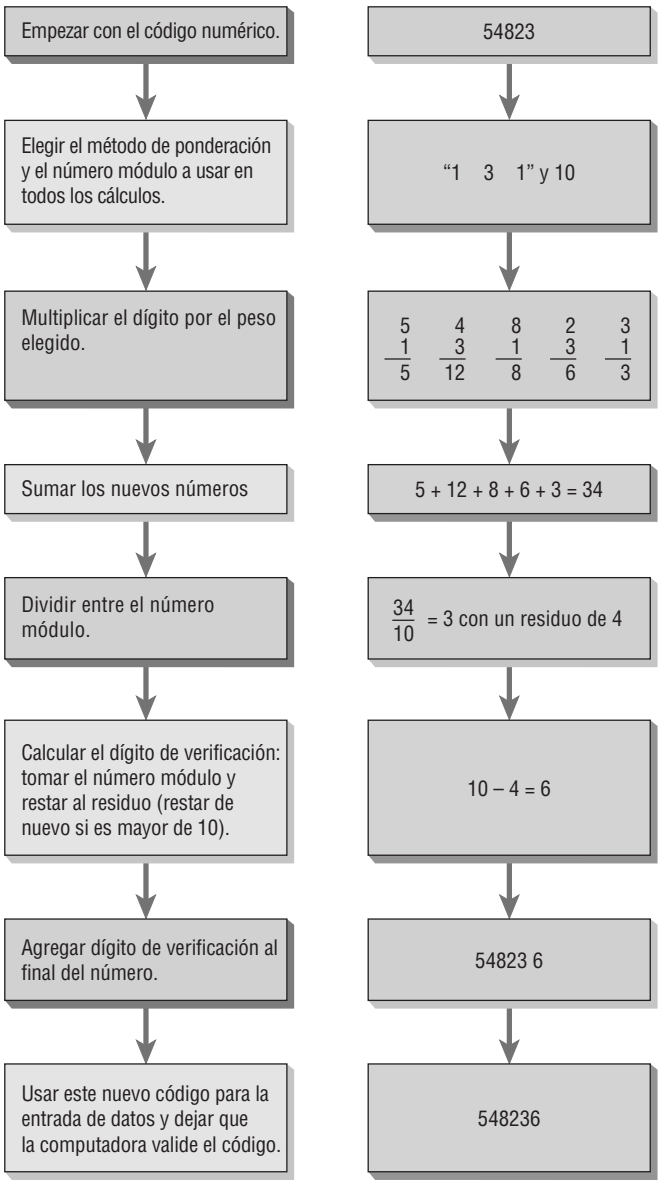


FIGURA 15.17
Los pasos para convertir un número de pieza de cinco dígitos en un número de seis dígitos que contiene un dígito de verificación.

La figura 15.17 muestra cómo se convierte el número de pieza de cinco dígitos de una manguera para radiador (54823) en un número de seis dígitos que contiene un dígito de verificación. En este ejemplo, los pesos elegidos fueron el sistema "1-3-1"; en otras palabras, los pesos alternan entre 1 y 3. Después de multiplicar los dígitos 5, 4, 8, 2 y 3 por 1, 3, 1, 3 y 1, se convirtieron en 5, 12, 8, 6 y 3. La suma de estos nuevos dígitos es 34. Después el 34 se divide entre el número módulo elegido, 10, para obtener el resultado de 3 y un residuo de 4. El residuo (4) se resta del número módulo (10) para obtener un dígito de verificación de 6. Ahora el dígito 6 se coloca al final del número original para producir el código del producto oficial para la manguera de radiador (548236).

Uso de los dígitos de verificación El sistema de dígitos de verificación funciona de la siguiente manera. Suponga que tenemos el número de pieza 53411. Hay que teclear este número en el sistema, y mientras hacemos eso pueden ocurrir distintos tipos de errores. Uno de los posibles errores es teclear mal un dígito; por ejemplo, el empleado teclea 54411 en vez de 53411. Sólo el dígito en la posición de los miles es incorrecto, pero este error puede provocar que se envíe la pieza incorrecta.

Un segundo tipo de error es el de los dígitos transpuestos. Es común que el número deseado 53411se teclee como el número 54311, sólo debido a que dos teclas se oprimieron en orden inverso. Los errores de transposición también son difíciles de detectar para los humanos.

Podemos evitar estos errores mediante el uso de un dígito de verificación, ya que cada uno de estos números (el correcto y el erróneo) tendrían un número de dígito de verificación distinto, como se muestra en la

Estado	Código original	Dígito de verificación	Nuevo código
Correcto	5 3 4 1 1	8	534118
Error al teclear un dígito	5 **4** 4 1 1	5	544115
Transposición	5 **4** **3** 1 1	6	543116

FIGURA 15.18
Evitar errores de entrada de datos comunes por medio del uso de un dígito de verificación.

figura 15.18. Si el número de pieza 53411 se modificara a 534118 (incluyendo el dígito de verificación 8) y ocurriera cualquiera de los dos errores antes descritos, se atraparía el error. Si el segundo dígito se tecleara en forma equivocada como 4, la computadora no aceptaría 544118 como un número válido, ya que el dígito de verificación para 54411 sería 5 y no 8. De manera similar, si se transpusieran el segundo y tercer dígitos, como en 543118, la computadora también rechazaría el número debido a que el dígito de verificación para 54311 sería 6 y no 8.

El analista de sistemas selecciona los pesos y el número módulo, pero una vez elegidos no deben cambiar. En la figura 15.19 encontrará algunos ejemplos de métodos de ponderación y números módulo.

VERIFICACIÓN DE TARJETAS DE CRÉDITO Al introducir tarjetas de crédito en un sitio Web o programa de computadora, la primera verificación es la longitud del número. Las compañías de tarjetas de crédito diseñaron sus tarjetas para incluir un número distinto de dígitos. Por ejemplo, las tarjetas Visa son de 16 dígitos mientras que los de las tarjetas American Express están compuestas por 15 dígitos.

Otra prueba es relacionar la compañía de la tarjeta de crédito y el banco para verificar que sea realmente una tarjeta emitida por esa compañía. Por lo general los primeros cuatro dígitos indican el tipo de tarjeta. Los dígitos de en medio comúnmente representan al banco y al cliente. El último dígito es de verificación.

Además de estos métodos de verificación, en el procesamiento de tarjetas de crédito se utiliza una fórmula de dígitos de verificación conocida como fórmula Luhn, la cual se creó en la década de 1960. Suponga que tene-

Método de dígito de verificación	Cálculos para agregar el dígito de verificación al número original 29645
Módulo 10 “2-1-2”	2 9 6 4 5 2 1 2 1 2 4 + 9 + 12 + 4 + 10 = 39/10 = 3 residuo Dígito de verificación igual a 10 − (9) = 1 El código con dígito de verificación es 296451.
Módulo 10 “3-1-3”	2 9 6 4 5 3 1 3 1 3 6 + 9 + 18 + 4 + 15 = 52/10 = 5 residuo Dígito de verificación igual a 10 − (2) = 8 El código con dígito de verificación es 296458.
Módulo 11 “Aritmético”	2 9 6 4 5 6 5 4 3 2 12 + 45 + 24 + 12 + 10 = 103/11 = 9 residuo Dígito de verificación igual a 11 − (4) = 7 El código con dígito de verificación es 296457.
Módulo 10 “Geométrico”	2 9 6 4 5 32 16 8 4 2 64 + 144 + 48 + 16 + 10 = 282/11 = 25 residuo Dígito de verificación igual a 11 − (7) = 4 El código con dígito de verificación es 296454.

FIGURA 15.19
Ejemplos de métodos de ponderación y números módulo.

OPORTUNIDAD DE CONSULTORÍA 15.4

¿Validan el estacionamiento?

"Qué vamos a hacer, Mercedes?", pregunta Edsel con cansancio. Mercedes y Edsel están revisando la impresión de facturación más reciente para su empresa, Denton and Denton Parking Garages. Han estado comprando servicios de facturación por lotes de una empresa local pequeña de servicios de computadora desde que adquirieron tres estacionamientos cerrados en un área metropolitana de tamaño medio. La empresa Denton and Denton Parking Garages renta lugares de estacionamiento por día, mes y año a las empresas e individuos.

Mercedes responde: "No estoy segura de cuál será nuestro siguiente movimiento, pero la facturación está toda mal. Tal vez deberíamos tratar de hablar con el personal de TI".

"Dijeron que podrían averiguar cómo calcular estos costos viendo lo que habían hecho a mano los dueños anteriores, y dijeron que no querían ejecutar el sistema antiguo y el nuevo en paralelo", comenta Edsel al tiempo que sacude la cabeza. "Pero eso no está bien. Por lo menos yo no puedo resolverlo. Tal vez usted pueda".

Mercedes acepta la noción de buscar la salida sospechosa y empieza a analizar el informe con detalle. "Bueno, al fin y al cabo no saben que recibimos automóviles de todas partes. Cada vez que recibimos un automóvil con placas que no son del estado, parece como si la computadora dejara de imaginar. Mira, nuestras placas empiezan con un número y después una letra, ¿verdad? Bueno, este de Nueva York empieza con tres letras. La computadora no puede manejarlo", dice.

Edsel se integra a la conversación y empieza a pensar sobre la empresa mientras analiza el documento impreso. "Sí, y mira aquí. Esta persona no tiene un número de cuenta anual, sólo mensual, por lo que no salió ninguna factura", dice. "Tenemos clientes mensuales también, y la computadora no lo sabe".

"Y mira esto. Todavía hace cobros diarios por los tres días de noviembre que les dijimos que no había vacantes para los clientes diarios. No es razonable", afirma Mercedes.

Edsel continúa avanzando página por página en el documento impreso, pero Mercedes lo detiene y dice: "Ya no sigas viendo. Voy a llamar al personal de TI para poder arreglar este desorden".

¿Cómo caracterizaría los problemas que se encontraron en el sistema de facturación actual del estacionamiento? Use un párrafo para formular una respuesta. ¿Cuáles son algunas de las pruebas de validez de datos que se podrían incluir en el software para producir un sistema de facturación revisado para los estacionamientos? Haga una lista. ¿Qué podrían haber hecho distinto el programador y los analistas de la empresa de servicios de cómputo para que el cliente no tuviera qué corregir la salida de mala calidad? Use tres párrafos para realizar un análisis crítico de lo que se hizo y lo que se *debió* hacer.

mos un número 7-7-7-8-8-8, donde los primeros cinco números representan un número de cuenta bancaria y el último dígito es de verificación. Vamos a aplicar la fórmula de Luhn para ver si es un número válido.

1. Duplique el penúltimo dígito y después los demás dígitos (es decir, omita un dígito, duplique el siguiente, omita un dígito, duplique el siguiente, etc.). Por ejemplo, el número 7-7-7-8-8-8 se convierte en 14-7-14-8-16-8.
2. Si al duplicar un dígito se produce un número mayor de 10, hay que reducir este número de dos dígitos a un solo dígito, para lo cual se suman los dos números. En nuestro ejemplo, el 14 se convierte en $1 + 4 = 5$ y el 16 se convierte en $1 + 6 = 7$. Después de esto, nuestro número original 7-7-7-8-8-8 se ha transformado en un nuevo número, 5-7-5-8-7-8.
3. Ahora sume todos los dígitos en el nuevo número. Así, $5 + 7 + 5 + 8 + 7 + 8 = 40$.
4. Vea el total. Si termina en cero, el número es válido de acuerdo con la fórmula de Luhn. Como 40 termina en cero, podemos decir que pasa la prueba de la fórmula de Luhn.

Podemos usar la fórmula de Luhn para identificar los errores al introducir una tarjeta de crédito incorrecta. Por ejemplo, el número de tarjeta de crédito 1334-1334-1334-1334 es válido, debido a que los dígitos del número transformado 2364-2364-2364-2364 se suman para un total de 60, un número que termina en cero. Si un usuario introduce un dígito incorrecto, el total no será un múltiplo de cero.

Sin embargo, la fórmula de Luhn no atrapa todos los errores. Si un usuario comete errores al introducir más de un dígito —por ejemplo, si introduce 1334-1334-1334-331— el total del número transformado (2364-2364-2364-6324) sigue siendo 60. Este error de transposición (invertir el tercer y cuarto dígitos de derecha a izquierda) no se atrapará.

Las compañías de tarjetas de crédito también usan la fecha de expiración y un código de verificación de tres o cuatro dígitos, a menudo escrito en el lado posterior de la tarjeta para una mayor seguridad.

Las siete pruebas para verificar la validez de la entrada pueden ayudar de manera considerable a proteger el sistema contra la entrada y el almacenamiento de datos erróneos. Siempre debemos suponer que es mucho más probable que ocurran errores humanos en la entrada a que no ocurran. Es su responsabilidad como analista comprender cuáles errores invalidarán los datos y cómo usar la computadora para protegerse contra esos errores humanos, para así limitar su intrusión en los datos del sistema.

El proceso de validación

Es importante validar cada campo hasta que sea válido o se detecte un error. El orden de prueba de los datos es primero verificar si hay datos faltantes. Después, una prueba de sintaxis puede verificar la longitud de los datos introducidos y que éstos tengan una clase y composición apropiadas. Sólo después de que la sintaxis sea correcta se validará la semántica o significado de los datos. Esto incluye una prueba de rango, razonable o de valor, seguida de una prueba de dígito de verificación.

Las pantallas de GUI ayudan a reducir el número de errores de entrada humanos cuando incorporan botones de opción, casillas de verificación y listas desplegables. Cuando se utilizan botones de opción, uno se debe establecer como el predeterminado y la única forma de deseleccionarlo es si el usuario hace clic en un botón de opción distinto. En caso de las listas desplegables, la primera elección siempre debe contener un mensaje para informar al usuario que debe cambiar la lista. Si la primera opción sigue aún seleccionada al momento de enviar el formulario, un mensaje debería informarle que debe seleccionar una opción distinta.

Por lo general, la validación de un solo campo se realiza mediante una serie de instrucciones IF…ELSE, pero también hay métodos de validación de patrones. Por lo general estos patrones se encuentran en el diseño de la base de datos (como en Microsoft Access), pero también se pueden incluir en lenguajes de programación como Perl, JavaScript y esquemas de XML. A los patrones se les conoce como expresiones regulares y contienen símbolos que representan el tipo de datos que deben estar presentes en un campo. La figura 15.20 muestra los caracteres que se utilizan en las expresiones regulares de JavaScript.

A continuación le mostramos un ejemplo de validación de patrón utilizado para evaluar una dirección de correo electrónico:

```
[A-Za-z0-9]\w{2,}@[A-Za-z0-9]{3,}\.[A-Za-z]{3}/
```

El significado de este patrón es el siguiente: la primera letra debe ser cualquier letra mayúscula, minúscula o número ([A-Za-z0-9]). Esto va seguido de dos o más caracteres que son cualquier letra, número o un guión bajo (\w{2,}). Después debe haber un símbolo @, seguido de por lo menos tres letras o números, un punto y exactamente tres caracteres después del punto.

Una verificación de referencias cruzadas supone que la validez de un campo puede depender del valor de otro campo. Un ejemplo de una verificación de referencia cruzada es verificar una fecha válida. En un caso muy especial, la validez del día del mes depende del año. Es decir, febrero 29 es sólo válido durante los años bisiestos. Una vez que se han verificado los campos individuales, podemos realizar verificaciones de referencias cruzadas. Obviamente, si uno de los campos es incorrecto, la verificación de referencia cruzada no tiene sentido y no se debe llevar a cabo.

Código de carácter	Significado utilizado en la validación con expresiones regulares
\d	Cualquier dígito del 0-9
\D	Cualquier carácter que no sea dígito
\w	Cualquier letra, número o guión bajo
\W	Cualquier carácter distinto de una letra, número o guión bajo
.	Coincide con cualquier carácter
[caracteres]	Coincide con los caracteres dentro de los corchetes
[car-car]	Coincide con el rango de caracteres
[a–z][A–Z][0–9]	Acepta cualquier letra o dígito
[^caracteres]	Coincide con cualquier cosa que no sean los caracteres
[^car-car]	Coincide con cualquier cosa fuera del rango de caracteres
[^a–z]	Aceptará todo excepto letras minúsculas
{n}	Coincide con exactamente n ocurrencias del carácter anterior
{n,}	Coincide con al menos n ocurrencias del carácter
\s	Cualquier carácter de formato de espacio en blanco (tabulador, nueva línea, retorno, etc.)
\S	Cualquier carácter que no sea de espacio en blanco

FIGURA 15.20
Estos caracteres se utilizan en la validación con expresiones regulares (patrones).

Para validar documentos XML debemos compararlos con una definición de tipo de documento (DTD) o con un esquema (consulte el capítulo 8). La DTD verificará si el formato del documento es válido, pero un esquema es mucho más potente y verificará el tipo de datos (por ejemplo, entero corto o largo, número decimal o fecha). Un esquema también verificará un rango de valores, el número de dígitos a la izquierda y a la derecha de un punto decimal, y los valores de los códigos. Hay herramientas gratuitas para validar una DTD o un esquema. IEXMLTLS es una extensión de Microsoft para Internet Explorer, la cual agrega nuevas opciones de menú cuando el usuario hace clic con el botón derecho en un documento de XML.

VENTAJAS DE LA PRECISIÓN EN LOS ENTORNOS DE COMERCIO ELECTRÓNICO

Uno de los muchos beneficios de las transacciones de comercio electrónico es una mayor precisión de los datos, debido a cuatro razones:

1. Por lo general los clientes teclean o introducen los datos ellos mismos.
2. Los datos que introducen los clientes se almacenan para su uso posterior.
3. Los datos que se introducen en el punto de ventas se reutilizan en todo el proceso de cumplimiento del pedido.
4. La información se utiliza como retroalimentación para los clientes.

Un analista debe estar consciente de las ventajas que se producen debido al comercio electrónico, la captura electrónica y el uso de la información.

Los clientes teclean sus propios datos

En primer lugar, los clientes conocen su información mejor que cualquier otra persona. Saben cómo deletrear su dirección física, saben si viven en una "Avenida" o "Calle" y conocen su propio código de área. Si transmiten esta información por teléfono, es más fácil cometer un error al deletrear la dirección; si se introduce mediante un formulario de papel vía fax, puede haber errores si la transmisión de fax es difícil de leer. Pero si los usuarios introducen su propia información, aumenta la precisión.

Almacenar datos para su uso posterior

Una vez que los clientes introducen la información, ésta se puede almacenar en sus propias computadoras personales. Si regresan a ese sitio de comercio electrónico y llenan el mismo formulario para completar una segunda transacción, experimentarán la ventaja de almacenar esta información. Cuando empiecen a teclear su nombre, una lista desplegable les mostrará su nombre completo incluso cuando sólo hayan introducido un par de caracteres. Si hacen clic en este indicador se introducirá su nombre completo y no tendrán que teclear más en ese campo. Esta característica de autocompletar puede sugerir coincidencias para la información de tarjetas de crédito y contraseñas también; además, esta información se cifra para que los sitios Web no puedan leer la información almacenada en la computadora del usuario.

Las empresas que desean almacenar información para permitir transacciones cada vez más rápidas lo hacen en pequeños archivos conocidos como cookies. Sólo la empresa que colocó la cookie en la computadora del usuario podrá acceder a la información personal.

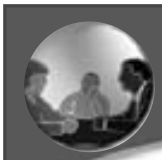
Utilizar datos por medio del proceso de cumplimiento de pedidos

Cuando las empresas capturan información sobre el pedido de un cliente, pueden usar y reutilizar esa información en todo el proceso de cumplimiento del pedido. Así, la información recopilada para completar un pedido también se puede usar para enviar una factura a un cliente, obtener el producto del almacén, enviar el producto, enviar retroalimentación al cliente y reabastecer el producto enviando una notificación al fabricante. También se puede volver a utilizar para enviar un catálogo impreso al cliente, o enviarle una oferta especial por correo electrónico.

Estas mejoras del comercio electrónico reemplazan a la metodología tradicional en la que se utilizaba un proceso de obtención basado en papel, donde las órdenes de compra se enviaban vía correo convencional o fax. Este proceso electrónico no sólo agiliza la entrega del producto sino que también incrementa la precisión para que el producto se entregue en la dirección correcta. En vez de leer un fax o un formulario enviado por correo convencional, un despachador utiliza la versión electrónica de los datos, que es más precisa. La información electrónica permite administrar mejor la cadena de suministro, incluyendo la verificación del producto y la disponibilidad de recursos en forma electrónica, además de automatizar los procesos de planificación, calendarización y proyección.

Proveer retroalimentación a los clientes

Las confirmaciones y actualizaciones del estado del pedido son formas de mejorar la retroalimentación para los clientes. Si un cliente recibe la confirmación de un error en un pedido que acaba de hacer, se puede corregir de



EXPERIENCIA DE HYPERCASE® 15

"Algunas veces creo que soy la persona más afortunada de la tierra. Incluso cuando he estado aquí por cinco años, alejado del Reino Unido todo ese tiempo, todavía disfruto el hecho de conocer personas y hacer lo que hago. Sí, ya sé que Snowden es exigente. Usted también lo ha experimentado, ¿verdad? Él, por su parte, ama los códigos. A mí me parecen una molestia. Siempre los olvido o trato de crear algunos, o algo por el estilo. Sin embargo, algunos de los médicos piensan que son grandiosos. Deben ser todas esas abreviaturas en latín que estudiaron en la escuela de medicina. Escuché que la asignatura más estresante que tiene usted esta semana tiene algo que ver con introducir la información en el sistema de informes de los proyectos. El grupo de capacitación quiere sus ideas, y las quiere rápido. Le deseo buena suerte con eso. Oh, y cuando Snowden regrese de Tailandia, estoy seguro de que querrá echar un vistazo a lo que su equipo ha estado haciendo".

Preguntas de HyperCase

1. Use una herramienta CASE, un paquete de software como Microsoft Access o un formulario de diseño impreso para diseñar un procedimiento de entrada de datos para el sistema de informes de proyectos propuesto para el grupo de capacitación. Suponga que estamos especialmente interesados en el personal de médicos de consulta, quienes no desean pasar mucho tiempo tecleando grandes cantidades de datos al usar el sistema.
2. Pruebe su procedimiento de entrada de datos con tres compañeros de su equipo. Pida retroalimentación para saber si el procedimiento es apropiado, dado el tipo de usuarios que tendrá el sistema.
3. Rediseñe el procedimiento de entrada de datos para incluir la retroalimentación que haya recibido. Explique en un párrafo cómo sus cambios reflejarán los comentarios que recibió.

inmediato. Por ejemplo, suponga que un cliente envía sin querer un pedido de dos ejemplares de un DVD en vez de una. Después de enviar el pedido, el cliente recibe un correo electrónico en el que se confirma el pedido. El cliente detecta el error, se pone en contacto de inmediato con la empresa y corrige el pedido, con lo cual se evita el tener que devolver la copia adicional del DVD. La precisión se mejora gracias a una mejor retroalimentación.

RESUMEN

Es imprescindible asegurar la calidad de la entrada de datos en el sistema de información para poder asegurar una salida de calidad. Podemos mejorar la calidad de los datos introducidos a través de una codificación efectiva, un proceso efectivo y eficiente de captura de datos y la validación de los datos.

Podemos agilizar la manera en que los humanos introducen los datos mediante el uso efectivo de la codificación, que se encarga de colocar los datos en secuencias cortas de dígitos y/o letras. Es posible usar códigos de secuencia simple y códigos de derivación alfabética para seguir el progreso de un elemento dado. Los códigos de clasificación y los códigos de secuencia en bloque son útiles para diferenciar unas clases de elementos de otras. Los códigos de cifrado también son útiles debido a que ocultan información sensible o restringida para los empleados.

Los códigos también son convenientes para revelar información a los usuarios, ya que pueden permitir a los empleados localizar artículos en existencia y hacer la entrada de datos más significativa. Los códigos de subconjuntos de dígitos significativos usan subgrupos de dígitos para describir un producto. Los códigos mnemónicos también revelan información al servir como ayudas para la memoria humana, de manera que un operador pueda introducir los datos correctamente, o también pueden ayudar al usuario. El conjunto de caracteres Unicode incluye todos los símbolos de los lenguajes estándar. Podemos mostrar páginas Web escritas en otros alfabetos si descargamos un editor de método de entrada de Microsoft. Los códigos de función son abreviaturas útiles para informar a las computadoras o las personas sobre las funciones a realizar o las acciones a tomar.

Para la entrada de datos efectiva también debemos considerar los dispositivos de entrada. Un formulario efectivo y bien diseñado que sirva como documento de origen es el primer paso. Los datos se pueden introducir mediante muchos métodos, cada uno de los cuales tiene distintas velocidades y grados de confiabilidad. Los teclados se rediseñaron para mejorar la eficiencia y la ergonomía. El reconocimiento óptico de caracteres (OCR), el reconocimiento de caracteres de tinta magnética (MICR) y los formularios de detección de marcas tienen cada uno capacidades especiales para mejorar la eficiencia. Los códigos de barras también agilizan la entrada de datos, mejoran la precisión de los datos y aumentan la confiabilidad. RFID permite la recolección automática de datos mediante el uso de etiquetas RFID en productos, personas o animales. Pueden mejorar los procesos de administración de inventario y la cadena de suministro.

La precisión en la entrada de datos también se puede mejorar mediante el uso de la validación de la entrada. Los analistas deben trabajar con los usuarios para diseñar pruebas de validación de la entrada, de manera que se evite el procesamiento y almacenamiento de datos erróneos, lo cual es costoso y potencialmente perjudicial.

Hay que verificar las transacciones de entrada para asegurar que la petición sea aceptable, esté autorizada y sea correcta. Los datos de entrada se pueden validar mediante software que utilice varios tipos de pruebas que verifiquen si hay datos faltantes, la longitud de los elementos de datos, el rango y la sensatez de los datos, y que no haya valores inválidos. Los datos de entrada también se pueden comparar con los datos almacenados para fines de validación. Una vez que se introducen datos numéricos, se pueden verificar y corregir en forma automática por medio de los dígitos de verificación y la fórmula de Luhn.

Hay un orden establecido en la prueba de los datos para validar cada campo. También hay métodos de validación mediante patrones que se encuentran en el diseño de la base de datos o se incluyen en los lenguajes de programación. A los patrones se les conoce como expresiones regulares y contienen símbolos que representan el tipo de datos que debe haber en un campo.

Los entornos de comercio electrónico nos ofrecen la oportunidad de mejorar la precisión de los datos. Con un énfasis apropiado en los elementos de diseño centrados en el usuario, los clientes pueden introducir sus propios datos, almacenarlos para su uso posterior, usar los mismos datos almacenados en todo el proceso de cumplimiento del pedido y recibir retroalimentación en relación con las confirmaciones y actualizaciones de los pedidos.

PALABRAS CLAVE Y FRASES

administración de la cadena de suministro
característica de autocompletar
codificación
código de autovalidación
código de barras
código de cifrado
código de clasificación
código de derivación alfabética
código de función
código de secuencia en bloque
código de secuencia simple
código de subconjunto de dígitos significativos
código nemónico
cookies
cuello de botella
diferenciados
dígito de verificación
expresión regular
fórmula de Luhn
formulario de detección de marcas
identificación por radiofrecuencia (RFID)
modificable
prueba de clase o composición
prueba de comparación con datos almacenados
prueba de datos faltantes
prueba de longitud de campo correcta
prueba de rango o sensatez
prueba de referencia cruzada
prueba de valores inválidos
reconocimiento de caracteres de tinta magnética (MICR)
reconocimiento óptico de caracteres (OCR)
redundancia en los datos de entrada
teclear
Unicode
validar la entrada

PREGUNTAS DE REPASO

1. ¿Cuáles son los cuatro principales objetivos de la entrada de datos?
2. Mencione los cinco propósitos generales de codificar los datos.
3. Defina el término *código de secuencia simple*.
4. ¿Cuándo puede ser útil un código de derivación alfabética?
5. Explique lo que se logra con un código de clasificación.
6. Defina el término *código de secuencia en bloque*.
7. ¿Cuál es el tipo más simple de código para ocultar información?
8. ¿Cuáles son los beneficios de usar un código de subconjunto de dígitos significativos?
9. ¿Cuál es el propósito de usar un código mnemónico para los datos?
10. Defina el término *código de función*.
11. Mencione los ocho lineamientos generales para una codificación apropiada.
12. ¿Qué son los datos modificables?
13. ¿Qué son los datos de diferenciación?
14. ¿Cuál es una manera específica de reducir la redundancia de los datos que se van a introducir?
15. Defina el término *cuello de botella* según su aplicación en la entrada de datos.
16. ¿Cuáles son las tres funciones repetitivas de la entrada de datos que una computadora puede realizar con más eficiencia que un operador de entrada de datos?
17. Mencione seis métodos de entrada de datos.
18. Mencione los tres principales problemas que pueden ocurrir con las transacciones de entrada.
19. Defina RFID. ¿Cuáles son las diferencias entre las etiquetas RFID activas y pasivas?
20. Mencione dos ejemplos del uso de etiquetas RFID en la administración de procesos o de inventario en entornos de ventas al menudeo o centros al cuidado de la salud.
21. ¿Cuáles son las ocho pruebas para validar los datos de entrada?
22. ¿Qué prueba verifica si los campos de datos se llenaron correctamente con números o letras?

23. ¿Qué error común no es detectado por la fórmula de Luhn?
24. ¿Qué prueba no permite que un usuario introduzca una fecha tal como octubre 32?
25. ¿Qué prueba asegura la precisión en los datos mediante la incorporación de un número en el mismo código?
26. Mencione cuatro mejoras en la precisión de los datos que pueden ofrecer las transacciones que se realizan a través de sitios Web de comercio electrónico.
27. ¿Qué es Unicode y cómo se utiliza?
28. ¿Cuál es el proceso para validar los datos que se introducen en los campos?
29. ¿Qué es una expresión regular?

PROBLEMAS

1. Una pequeña universidad privada especializada en programas de graduados necesita llevar el registro de la lista de estudiantes que a) hacen solicitud, b) son aceptados y c) se inscriben en la universidad. Por cuestiones de seguridad, la universidad también debe enviar un informe al gobierno con una lista de los estudiantes extranjeros que se inscriben pero no se registran. Sugiera un tipo de código para este fin y proporcione un ejemplo de su uso en la universidad, en donde se demuestre si es apropiado o no. ¿Cuáles son sus ventajas?

2. Las Ardillas de la Universidad Central Pacific han estado usando un código de secuencia simple para llevar el registro de los que poseen abonos de temporada y los fanáticos que no poseen abonos de temporada de todos sus programas deportivos. Se han producido varias confusiones desagradables.

Sugiera en un párrafo un esquema de codificación distinto que ayude a identificar en forma única a cada abonado y explique cómo podrá este esquema evitar las confusiones.

3. El código utilizado por una tienda de helados para ordenar sus productos es 12DRM215-220. Este código se descifra de la siguiente manera: 12 representa la cuenta de artículos en la caja, DRM representa a unas paletas de hielo llamadas "Dreamcicles" (un tipo específico de novedad en el ámbito de los helados) y 215-220 indica toda la clase completa de productos bajos en grasa con los que cuenta el distribuidor.
   a. ¿Qué tipo de código se utiliza? Describa el propósito detrás de cada parte (12, DRM, 215-220) del código.
   b. Construya una entrada codificada usando el mismo formato y la misma lógica para una novedad en los helados conocida como Pigeon Bars, que vienen en un paquete de seis y *no* son bajos en grasa.
   c. Construya una entrada codificada usando el mismo formato y la misma lógica para una novedad en los helados conocida como Airwhips, que viene en un paquete de 24 y es un producto bajo en grasa.

4. Los operadores de entrada de datos en la empresa Michael Mulheren Construction han estado cometiendo errores al introducir los códigos para los productos de recubrimientos residenciales, que son los siguientes: U = estUco, A = Aluminio, R = ladRillo, M = Masonite, EZ = masonite esmaltado tipo EZ color-lok, N = recubrimiento de madera Natural, PI = acabado con PIntura, SH = SHake SHingles. Sólo se permite un código por dirección.
   a. Haga una lista de los posibles problemas con el sistema de codificación que podrían estar provocando entradas con error. (*Sugerencia*: ¿Son las clases mutuamente excluyentes?).
   b. Desarrolle un código mnemónico que ayude a los operadores a comprender lo que están introduciendo para mejorar su precisión.
   c. ¿Cómo rediseñaría las clases para los materiales de recubrimiento? Responda en un párrafo.

5. A continuación le mostramos un código para un producto en una extensa línea de cosméticos: L02002Z621289. La L significa que es un lápiz labial, 0 significa que se introdujo sin barniz de uñas complementario, 2002 es un código de secuencia que indica el orden en el que se produjo, Z es un código de clasificación que indica que el producto es hipoalergénico y 621289 es el número de la planta (hay 15 plantas) donde se produce el producto.
   a. Critique el código mediante una lista de las características que podrían provocar una entrada de datos imprecisa.
   b. El diseñador Brian d'Arcy James es propietario de la empresa de cosméticos que utiliza este esquema de codificación. Con un interés continuo en el nuevo diseño, Brian está dispuesto a considerar un código más elegante que codifique la *misma* información de una mejor manera. Rediseñe el esquema de codificación y proporcione una clave para su trabajo.
   c. Escriba un enunciado para cada cambio que haya sugerido, indicando cuál problema de entrada de datos (del problema 5a) eliminará ese cambio.
   d. El Sr. d'Arcy James está encantado con su trabajo. Dice que a la empresa le gustaría contratarlo para que les ayudara a expandir su actividad comercial para vender maquillaje para teatro (las obras como *Wicked* y *Shrek* con ocho funciones por semana usan mucho maquillaje color verde). Agregue los nuevos códigos que sean necesarios para el esquema de codificación que sugirió en el inciso b y proporcione una clave para su trabajo.

6. La empresa de cosméticos de d'Arcy James requiere que sus vendedores usen computadoras notebook para introducir los pedidos de las tiendas departamentales (sus mayores clientes). Después esta información se transmite a los almacenes y los pedidos se envían según el orden en el que hayan llegado. Por desgracia las tiendas están conscientes de esta política y son muy competitivas en cuanto a cuál de ellas ofrecerá primero un nuevo producto de d'Arcy James. Muchos vendedores han tomado el camino oscuro y han persuadido a los vendedores para que falsifiquen las fechas de sus pedidos en los formularios de ventas, de manera que parezcan haberse hecho antes.
   a. Este problema está creando una gran confusión en el almacén. No es viable disciplinar al personal involucrado. ¿Cómo se puede usar la computadora del almacén para certificar la fecha en que se realizan los pedidos? Explique en un párrafo.

b. Los vendedores se quejan de que tienen que ignorar su verdadero trabajo de vender para poder teclear los datos. Mencione los elementos de datos relacionados con las ventas de cosméticos a las tiendas que se deban almacenar y recuperar de la computadora central, en vez de tener que teclearlos en cada pedido.
c. Describa en uno o dos párrafos cómo los códigos de barras podrían ayudar a resolver el problema del inciso 6b.

7. Mencione el mejor método de entrada de datos y la razón por la que lo eligió para cada una de las siguientes seis situaciones:
a. El documento de respuesta para una empresa de servicios públicos que desea la notificación de un cambio en la dirección del cliente.
b. Sólo se pueden recuperar datos si hay una identificación de máquina positiva de la parte que solicita los datos.
c. No hay suficiente personal capacitado disponible para interpretar respuestas largas por escrito; se enviaron muchos formularios que capturan respuestas a exámenes de opción múltiple; es necesario un alto nivel de confiabilidad; no se requiere un tiempo de respuesta rápido.
d. Se estableció un almacén para una operación de venta de discos compactos con descuento; los botes están identificados con la información sobre el precio, pero los discos individuales no; hay pocos operadores experimentados disponibles para introducir los datos de los precios.
e. Centro de control de venenos que mantiene una base de datos extensa de venenos y antídotos; necesita una manera de introducir datos sobre el veneno ingerido; también hay que introducir el peso, edad y condición física general de la víctima cuando una persona llama al número de atención gratuito del centro para obtener consejos de emergencia.
f. Un consumidor compró en línea un DVD con su tarjeta de crédito.

8. Ben Coleman, uno de los miembros de su equipo de análisis de sistemas, lo sorprende al afirmar que cuando un sistema utiliza una prueba para la longitud de campo correcta, es redundante incluir también una prueba de rango o sensatez. Describa en un párrafo un ejemplo que demuestre que Ben está equivocado en este caso.

9. Varios vendedores se reunieron y empezaron a emitir una tarjeta de crédito "estatal" que sólo es válida en las tiendas en su estado. Como cortesía, se permite a los vendedores transcribir el número de cuenta de 15 dígitos a mano (después de obtenerlo de la oficina de contabilidad) si el cliente no lleva consigo la tarjeta. El único problema con las cuentas que los vendedores han detectado hasta ahora es que algunas veces se aceptan números de cuenta erróneos en el sistema de cómputo, lo cual provoca que se envíe una factura a una cuenta inexistente.
a. ¿Qué tipo de prueba de validez resolvería el problema? ¿Cómo? Responda en un párrafo.
b. Sugiera un método de entrada de datos alternativo que podría solucionar este problema por completo.

10. Los siguientes son números de piezas: 238902, 238933, 239402, 235693, 235405, 239204, 240965. Desarrolle un dígito de verificación para estos números mediante el uso de multiplicadores 1-3-1-3-1 y módulo 11. Use el método que presentamos en este capítulo. ¿Por qué algunos números tienen el mismo dígito de verificación?

11. Desarrolle un sistema de dígitos de verificación para los números de piezas del problema 10; use multiplicadores 5-4-3-2-1 y módulo 11.

12. Desarrolle un sistema de dígitos de verificación para los números de piezas del problema 10; use la fórmula de Luhn.

13. ¿Por qué un sistema de dígitos de verificación tal como 1-1-1-1-1 no funciona tan bien como otros métodos? ¿Qué errores no detectaría?

14. Defina una expresión regular para validar cada uno de los siguientes casos:
a. Un código postal de los EE.UU., que debe tener cinco dígitos seguidos de un guión corto opcional y de otros cuatro dígitos.
b. Un número telefónico en el formato (aaa) nnn-nnnn, en donde aaa representa el código de área y las n representan dígitos.
c. Una fecha en la forma de día-mes-año, en donde el mes es un código de tres letras y el año es de cuatro dígitos. Debe haber un guión corto para separar el día del mes y otro para separar el mes del año.
d. El código de derivación alfabética que se muestra en este capítulo para el suscriptor de una revista. El formato es 99999XXX9999XXX, en donde X representa a una letra y 9 representa a un número.

15. Para los siguientes códigos, defina los criterios de validación (puede haber varias verificaciones para cada campo) y el orden en el que evaluaría cada una de las condiciones.
a. **Un número de tarjeta de crédito introducido en un formulario Web:** El cliente seleccionó el tipo de tarjeta de crédito de una lista desplegable.
b. **Un número de pieza en una tienda de ferretería:** El número de pieza es un código complejo, en donde el primer dígito representa el departamento (como artículos del hogar, automotriz, etcétera) y el número debería ser de autoverificación. Hay siete departamentos distintos.
c. **La fecha en que se colocó el matasellos en un libro al momento de devolverlo a una biblioteca en línea:** Se debe incluir una copia del recibo del cliente con el libro. Los libros devueltos se deben sellar en un plazo no mayor a 30 días después de la fecha de compra.
d. **El código hablado de un lenguaje utilizado en un sitio Web:** (*Sugerencia*: Busque en la Web los códigos de lenguajes estándar.).
e. **El número de licencia de un conductor, compuesto de varias partes:** El mes de nacimiento de la persona, el cumpleaños y el año de nacimiento, no necesariamente juntos; un código que representa el color de ojos y un número de secuencia. La licencia de conductor contiene la fecha de nacimiento, el color de ojos y de pelo, así como el nombre y dirección de la persona.
f. **El código postal canadiense:** El formato es X9X 9X9 (X es cualquier letra, 9 es cualquier número).
g. **Códigos de equipaje de aerolíneas:** Como LAX para Los Ángeles o DUB para Dublín.

h. **Una clave de producto utilizada para desbloquear el software comprado:** La clave consiste en cuatro grupos de cinco caracteres cada uno. El primer grupo debe tener dos letras seguidas de tres números; el segundo grupo debe contener dos números seguidos de tres letras; el tercer grupo debe contener dos letras, cada una de la A a la G, seguidas de tres números del uno al cuatro; y el último grupo debe contener una letra, E, G o C, dos dígitos con valores de cuatro a siete y dos letras, ya sean A, B o C. (*Sugerencia*: Un patrón puede ser la mejor forma de validar la clave del producto.).

PROYECTOS EN GRUPO

1. Lea junto con los miembros de su grupo la Oportunidad de consultoría 15.3, "Entrar o no entrar: esa es la cuestión" que presentamos anteriormente en este capítulo. Diseñe un sistema de entrada de datos apropiado para Elsinore Industries. El diseño de su grupo debe hacer énfasis en la eficiencia y precisión. Además, debe haber distinción entre los datos modificables y los datos que indican la diferencia entre un elemento introducido y todos los demás. Dibuje los prototipos de las pantallas necesarias para explicar lo que recomienda.
2. Divida su grupo en analistas y empleados de Elsinore Industries para un juego de roles. Los analistas deben presentar el nuevo sistema de entrada de datos, completo con pantallas de prototipos. Pida a los empleados de Elsinore una retroalimentación sobre el diseño.
3. Escriba un breve párrafo para describir cómo mejorar el diseño original de entrada de datos con base en los comentarios recibidos.

BIBLIOGRAFÍA SELECCIONADA

Davis, G. B. y M. H. Olson, *Management Information Systems, Conceptual Foundations, Structure, and Development*, 2ª. Edición. Nueva York: McGraw-Hill, 1985.

Lamming, M. G., P. Brown, K. Carter, M. Eldridge, M. Flynn, G. Louie, P. Robinson y A. Sellan. "The Design of a Human Memory Prosthesis". *Computer Journal*, Vol. 37, 1994, pp. 153-163.

Lee, Y. M., F. Cheng y Y. T. Leung. "Exploring the Impact of RFID on Supply Chain Dynamics". En *Proceedings of the 2004 Winter Simulation Conference*, pp. 1145-1152. Editado por R. G. Ingalls, M. D. Rossetti, J. S. Smith y B. A. Peters.

MacKay, D. J. *Information Theory, Inference and Learning Algorithms*. Cambridge, Reino Unido: Cambridge University Press, 2004.

Miller, G. A. "The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capability for Processing Information". *Psychological Review*, Vol. 63, Núm. 2, marzo de 1956, pp. 81-97.

Newman, W. N., M. G. Lamming y M. Lamming. *Interactive System Design*. Reading, MA: Addison-Wesley Longman Publishing Co., 1995.

Niederman, F., R. G. Mathieu, R. Morley e I. Kwon. "Examining RFID Applications in Supply Chain Management", *Communications of the ACM*, Vol. 50, Núm. 7, julio de 2007, pp. 92-101.

Owsowitz, S. y A. Sweetland. "Factors Affecting Coding Errors", *Rand Memorandum RM-4346-PR*. Santa Monica, CA: Rand Corporatioin, 1965.

Robey, D. y W. Taggart. "Human Processing in Information and Decision Support Systems". *MIS Quarterly*, Vol. 6, Núm. 2, junio 1982, pp. 61-73.

Ryder, J. "Credit Card Validation Using LUHN Formula". www.freevbcode.com. Último acceso en 26 de agosto de 2009.

EPISODIO 15

CASO DE LA CPU

ALLEN SCHMIDT, JULIE E. KENDALL Y KENNETH E. KENDALL

Introducir en forma natural

El martes al mediodía Anna y Chip tienen su sesión semanal de revisión de análisis y diseño. Chip agita sus manos hacia una gran pila de documentos que están organizados en forma ordenada en una mesa grande. "No puedo creer que casi terminemos con el diseño de este sistema", comenta. "Ha sido un largo proceso, pero apuesto a que obtuvimos la retroalimentación suficiente como para asegurar un sistema de alta calidad. Todo lo que queda es el diseño de los procedimientos de entrada de datos y estaremos listos para empezar a empaquetar las especificaciones para los programadores".

"Así es", le responde Anna, "se acerca el fin. Empecemos por examinar el diseño de la porción del sistema correspondiente a la entrada".

"El programa AGREGAR SOFTWARE (ADD SOFTWARE) está en línea", observa Chip. "El operador tendrá que verificar visualmente cada transacción. Después de editar todos los campos de datos para asegurar la precisión, aparecerá un mensaje en la parte inferior de la pantalla para pedir a los operadores que verifiquen la precisión de los datos en la pantalla, los comparen con el formulario y hagan clic en el botón **Guardar registro** (**Save Record**) si todo está correcto. Los operadores tendrán la oportunidad de realizar cambios si los datos se teclean en forma incorrecta".

Hay que editar cada campo de datos para mantener la precisión. Chip comenta: "A la larga es mejor editar por completo los programas para asegurar que sean precisos, en vez de descubrir que se almacenaron los datos incorrectos en los archivos maestros y se imprimieron en los informes".

La estrategia para editar los campos es verificar los datos en el siguiente orden:

1. Sintaxis (si los datos son numéricos o alfabéticos) y la longitud de los datos. Un ejemplo es el NUMERO INVENTARIO HARDWARE (HARDWARE INVENTORY NUMBER), que debe tener ocho caracteres de longitud y ser numérico.
2. El contenido del campo, incluyendo rango, límite y valores para los datos. Al validar la FECHA COMPRA (DATE PURCHASED), el mes debe ser de 1 a 12. Esta verificación debe ocurrir sólo después de haber verificado que el mes sea numérico.
3. Verificaciones de referencias cruzadas entre dos o más elementos de datos. Para verificar la porción de FECHA COMPRA correspondiente al día, se utilizará una tabla del número de días posibles en cada mes para un límite superior. Esta tabla no se podría utilizar si el mes no estuviera entre 1 y 12. Los dígitos de verificación son otro ejemplo de una edición de referencias cruzadas.
4. Las ediciones externas, como leer un archivo para verificar si el registro que se va a agregar no existe ya en el archivo. Leer registros es más lento que editar, lo cual se lleva a cabo en memoria principal y debe ocurrir sólo después de que los datos pasen con éxito todas las demás pruebas de edición.

Se introdujeron los criterios de edición en la pantalla de visualización **Repositorio de elementos** (**Element Repository**) de Visible Analyst a medida que se agregaban elementos al diseño. Estos elementos incluyen criterios de edición simples y verificación de tablas. Se puede usar el área **Notas** (**Notes**) para introducir criterios de edición. La entrada NUMERO INVENTARIO HARDWARE (HARDWARE INVENTORY NUMBER) incluye una referencia para usar el método módulo 11 de verificar la porción del número correspondiente al dígito de verificación. Además, al agregar una nueva computadora, hay que leer el archivo MAESTRO DE COMPUTADORAS para asegurar que no exista ya un registro con el mismo NUMERO INVENTARIO HARDWARE.

Varios elementos tienen áreas de **Notas** referentes a las tablas, así como entradas para los códigos en el área **Valores y significados** (**Values and Meanings**). Un ejemplo es el elemento SISTEMA OPERATIVO (OPERATING SYSTEM). "Produciré una lista de las tablas que vamos a requerir", ofrece Anna. Para producir la lista hay que buscar en la página Web Repositorio (Repository) o usar la herramienta **Consulta de informes** (**Report Query**) de Visible Analyst. Se imprimirá una lista de todos los elementos que contienen notas empezando con "Tabla de códigos (Table of codes)". En la lista se incluyen **Imagen** (**Picture**) y **Longitud** (**Length**) para mostrar la sintaxis del código. Con esta lista se crean tablas.

Cada tabla de códigos se define mediante tablas de Microsoft Access. Chip y Anna pasan tiempo trabajando en las tablas. Se selecciona un código mnemónico para SISTEMA OPERATIVO y para representar la CATEGORIA SOFTWARE (SOFTWARE CATEGORY) debido a que facilita el trabajo a los usuarios. "Creo que un esquema de codificación de subconjunto de dígitos significativos sería el mejor para codificar los edificios del campus", comenta Chip. Anna está de acuerdo. "Eso es bueno, Chip. El primer dígito para la ubicación del campus y los otros dos dígitos para representar edificios individuales en el campus".

Chip diseña los códigos utilizados para la tabla SISTEMA OPERATIVO. La pantalla de Microsoft Access se muestra en la figura E15.1. Se utilizan dos columnas para definir los códigos. La columna izquierda contiene el código y la columna derecha contiene su significado. Esas entradas se pueden modificar y es posible agregar nuevas entradas, con lo cual se obtiene flexibilidad en el sistema final.

"He aquí la tabla CATEGORIA SOFTWARE que creé", dice Anna. "Esta tabla se puede actualizar con facilidad, a medida que la universidad desarrolle y adquiera nuevo software. Es sorprendente la rapidez con la que puede cambiar el software".

"Ése es un valioso componente del sistema", comenta Chip. "Provee consistencia para todos los códigos y sus significados".

Chip y Anna terminan su trabajo la siguiente mañana aproximadamente a las 11:30. Miran felizmente alrededor de la sala y vuelven a examinar con frecuencia el diseño final. Finalmente terminan los meses de análisis, trabajo de diseño, consultas con los usuarios y una cuidadosa adherencia a los estándares.

"Me siento realmente bien con este proyecto", dice Anna.

Chip asiente, "Estoy orgulloso de la calidad que aportamos".

FIGURA E15.1
TABLA DE CODIGOS SISTEMA OPERATIVO (OPERATING SYSTEM TABLE OF CODES) definida mediante Microsoft Access.

EJERCICIOS

E-1. Modifique e imprima los siguientes elementos con criterios de edición en el área **Notas** (o **Valores y significados** para códigos específicos). Puede encontrarlos en la página Web Repositorio o en el repositorio de Visible Analyst.

Elemento	*Criterio de edición*
a. CATEGORIA SOFTWARE	Tabla de códigos: Código Categoría Software (Software Category Code)
b. CODIGO NIVEL CAPACITACION CURSO (COURSE TRAINING LEVEL CODE)	B - Principiante; I - Intermedio; A - Avanzado
c. SISTEMA OPERATIVO (OPERATING SYSTEM)	W - Windows Vista Ultimate; B - Windows Vista Business; U - Unix

E-2. Modifique e imprima los siguientes elementos con los criterios de edición colocados en el área **Notas**:

a. Elemento: NUMERO INVENTARIO SOFTWARE (SOFTWARE INVENTORY NUMBER)
   Notas: Hay que verificar un dígito de verificación módulo 11 al introducir el número. El programa AGREGAR SOFTWARE (ADD SOFTWARE) crea el dígito de verificación.
   El programa AGREGAR SOFTWARE también debe verificar el archivo MAESTRO DE SOFTWARE (SOFTWARE MASTER) para asegurar que no exista ya un registro con el mismo número de inventario.

b. Elemento: FECHA COMPRA (DATE PURCHASED)
   Notas: Verificar que FECHA COMPRA sea menor o igual que la fecha actual.

c. Elemento: CANTIDAD RECIBIDA (QUANTITY RECEIVED)
   Notas: Verificar que CANTIDAD RECIBIDA sea menor o igual que CANTIDAD ORDENADA (QUANTITY ORDERED).

d. Elemento: VERSION ACTUALIZACION SOFTWARE (SOFTWARE UPGRADE VERSION)
   Notas: Asegurar que VERSION ACTUALIZACION (UPGRADE VERSION) sea mayor que la versión actual.

e. Elemento: SEGUNDO DISCO DURO (SECOND HARD DRIVE)
   Notas: SEGUNDO DISCO DURO sólo puede existir si hay una entrada para DISCO DURO (HARD DRIVE).

E-3. Después de hablar con Dot Matricks y Mike Crowe, es aparente que los códigos del campus deben tener capacidad de ordenamiento para instalar hardware y software, así como para crear hojas de inventario. Use Microsoft Access para modificar e imprimir la tabla CODIGOS UBICACION CAMPUS (CAMPUS LOCATION CODES). El primer dígito representa la ubicación del campus. Los valores son los siguientes:

1	Campus Central
2	Campus Waterford
3	Campus Hillside

Los siguientes tres dígitos representan los edificios en el campus, con los siguientes códigos de edificios:

001	Administración	010	Estudios Ambientales
002	Admisiones	011	Geología
003	Agricultura	012	Leyes
004	Astronomía	013	Biblioteca
005	Negocios	014	Matemáticas
006	Ingeniería Química	015	Medicina
007	Ciencias Computacionales	016	Física
008	Educación	017	Psicología
009	Ingeniería	018	Zoología

Use una combinación (de su elección) de códigos y edificios del campus para construir la tabla final de códigos. Incluya el significado del código.

Los ejercicios en los que se antepone un icono www indican material de valor agregado disponible en el sitio Web www.pearsonhighered.com/kendall. Los estudiantes pueden descargar un archivo de ejemplo de Microsoft Visio, Visible Analyst, Microsoft Project o Microsoft Access que pueden usar para completar los ejercicios.