

Capítulo 6

Fundamentos de inteligencia de negocios: bases de datos y administración de la información

OBJETIVOS DE APRENDIZAJE

Después de leer este capítulo, usted podrá responder a las siguientes preguntas:

1. ¿Cuáles son los problemas de administrar los recursos de datos en un entorno de archivos tradicional y cómo se resuelven mediante un sistema de administración de bases de datos?
2. ¿Cuáles son las principales capacidades de los sistemas de administración de bases de datos (DBMS) y por qué es tan poderoso un DBMS?
3. ¿Cuáles son algunos principios importantes del diseño de bases de datos?
4. ¿Cuáles son las principales herramientas y tecnologías para acceder a la información de las bases de datos y mejorar tanto el desempeño de negocios como la toma de decisiones?
5. ¿Por qué son la política de información, la administración de datos y el aseguramiento de la calidad de los datos esenciales para administrar los recursos de datos de la empresa?

RESUMEN DEL CAPÍTULO

- 6.1 **ORGANIZACIÓN DE LOS DATOS EN UN ENTORNO DE ARCHIVOS TRADICIONAL**
Términos y conceptos de organización de archivos
Problemas con el entorno de archivos tradicional
- 6.2 **LA METODOLOGÍA DE LAS BASES DE DATOS PARA LA ADMINISTRACIÓN DE DATOS**
Sistemas de administración de bases de datos
Capacidades de los sistemas de administración de bases de datos
Diseño de bases de datos
- 6.3 **USO DE BASES DE DATOS PARA MEJORAR EL DESEMPEÑO DE NEGOCIOS Y LA TOMA DE DECISIONES**
Almacenes de datos
Herramientas para la inteligencia de negocios: análisis de datos multidimensional y minería de datos
Las bases de datos y Web
- 6.4 **ADMINISTRACIÓN DE LOS RECURSOS DE DATOS**
Establecimiento de una política de información
Aseguramiento de la calidad de los datos
- 6.5 **PROYECTOS PRÁCTICOS SOBRE MIS**
Problemas de decisión gerencial
Obtención de la excelencia operacional: creación de una base de datos relacional para la administración del inventario
Mejora de la toma de decisiones: uso de las bases de datos en línea para buscar recursos de negocios en el extranjero

MÓDULO DE TRAYECTORIAS DE APRENDIZAJE

- Diseño de bases de datos, normalización y diagramas de entidad-relación
Introducción a SQL
Modelos de datos jerárquico y de red

Sesiones interactivas:

¿Qué pueden aprender las empresas de la minería de texto?

Errores del buró de crédito: grandes problemas de la gente

RR DONNELLEY TRATA DE DOMINAR SUS DATOS

Es probable que en estos momentos usted esté utilizando un producto de RR Donnelley. Esta empresa con base en Chicago es una gigantesca compañía de impresión y servicio comercial que provee servicios de impresión, formularios, etiquetas, correo directo y otros servicios. Quizás este libro de texto provenga de esas imprentas. La reciente expansión de la empresa se ha visto impulsada por una serie de adquisiciones, entre las que están, la imprenta comercial Moore Wallace en 2005 y la compañía de administración de la cadena de suministro e impresión llamada Banta, en enero de 2007. Los ingresos de RR Donnelley dieron un salto considerable, de \$2.4 mil millones en 2003 a más de \$9.8 mil millones en la actualidad.

Sin embargo, todo ese crecimiento generó desafíos en cuanto a la administración de la información. Cada compañía adquirida tenía sus propios sistemas, conjunto de datos de clientes, distribuidores y productos. Al provenir de tantas fuentes distintas, con frecuencia los datos eran inconsistentes, duplicados o incompletos. Por ejemplo, cada una de las diferentes unidades de la empresa podría tener un significado distinto para la entidad "cliente". Una podría definir "cliente" como una ubicación específica de facturación, mientras que otra lo podría definir como la entidad matriz legal de una compañía. Donnelley tuvo que utilizar procesos manuales que consumían mucho tiempo para reconciliar los datos almacenados en varios sistemas, para poder obtener una imagen clara a nivel empresarial de cada uno de sus clientes, ya que podrían estar haciendo negocios con varias unidades de la compañía. Estas condiciones aumentaron las ineficiencias y los costos.

RR Donnelley había crecido tanto que ya no era práctico almacenar la información de todas sus unidades en un solo sistema. No obstante, Donnelley aún necesitaba un solo conjunto claro de datos que fuera preciso y consistente para toda la empresa. Para resolver este problema, RR Donnelley recurrió a la administración de datos maestros (MDM). El objetivo de la MDM es asegurar que una organización no utilice varias versiones de la misma pieza de datos en distintas partes de sus operaciones, para lo cual fusiona los registros dispares en un solo archivo maestro autenticado. Una vez implementado el archivo maestro, los empleados y las aplicaciones acceden a una sola vista consolidada de los datos de la compañía. En especial, es útil para las compañías como Donnelley que tienen problemas de integración de sus datos como resultado de las fusiones y adquisiciones.

La implementación de la MDM es un proceso de varios pasos que incluye el análisis de los procesos de negocios, la limpieza de los datos, la consolidación, reconciliación de los datos y la migración de datos hacia un archivo maestro de toda la información de la compañía. Las compañías deben identificar qué grupo es "propietario" de cada pieza de datos y responsable de resolver las definiciones inconsistentes de datos, además de otras discrepancias. Donnelley lanzó su programa MDM a finales de 2005 y empezó a crear un solo conjunto de identificadores para los datos de sus clientes y distribuidores. La compañía optó por un modelo de registro mediante el concentrador de datos (Data Hub) de Purisma, en el cual los datos del cliente aún residen en el sistema en donde se originan, pero se registran en un "concentrador" o "hub" maestro y se hace referencia cruzada a ellos, de modo que las aplicaciones puedan encontrarlos, y los que están en el sistema de origen no se tocan.

Casi un año después, Donnelley lanzó su almacén de datos maestro de clientes (Customer Master Data Store), el cual integra los datos provenientes de numerosos sistemas debido a las adquisiciones de Donnelley. Los datos obsoletos, incompletos o que tienen un formato incorrecto se corrigen o eliminan. Al tener un solo conjunto consistente a nivel empresarial de datos con definiciones y estándares comunes, la gerencia puede averiguar con facilidad qué tipo de negocios y qué tanta actividad comercial tiene con un cliente específico para identificar los mejores clientes y las oportunidades de ventas. Y cuando Donnelley adquiera una compañía, podrá ver con rapidez una lista de los clientes que se traslanen.

Fuentes: John McCormick, "Mastering Data at R.R. Donnelley", *Information Management Magazine*, marzo de 2009; www.rdonnelley.com, visitado el 10 de junio de 2010, y www.purisma.com, visitado el 10 de junio de 2010.

La experiencia de RR Donnelley ilustra la importancia de la administración de datos para las empresas. Donnelley ha experimentado un crecimiento fenomenal, en su mayor parte debido a las adquisiciones, aunque su desempeño de negocios depende de lo que pueda o no hacer con sus datos. La forma en que las empresas almacenan, organizan y administran sus datos tiene un tremendo impacto sobre la efectividad organizacional.

El diagrama de apertura del capítulo dirige la atención a los puntos importantes generados por este caso y por este capítulo. La gerencia decidió que la compañía necesitaba centralizar la administración de sus datos. La información sobre los clientes, distribuidores, productos y demás entidades importantes se había almacenado en varios sistemas y archivos distintos, de donde no era fácil recuperarlos y analizarlos. A menudo eran redundantes e inconsistentes, lo cual limitaba su utilidad. La gerencia no podía obtener una vista a nivel empresarial de todos sus clientes en todas sus adquisiciones para comercializar sus productos y servicios, además de proveer un mejor servicio y soporte.

En el pasado, RR Donnelley había utilizado mucho los procesos manuales en papel para reconciliar sus datos inconsistentes y redundantes, y para administrar su información desde una perspectiva a nivel empresarial. Esta solución ya no era viable a medida que la organización crecía cada vez más. Una solución más apropiada era identificar, solidificar, limpiar y estandarizar los datos de los clientes junto con los demás tipos de datos en un solo registro de administración de datos maestro. Además de usar la tecnología apropiada, Donnelley tuvo que corregir y reorganizar los datos en un formato estándar, así como establecer reglas, responsabilidades y procedimientos para actualizar y utilizar los datos.

Un sistema de administración de datos maestro ayuda a RR Donnelley a impulsar la rentabilidad, al facilitar la identificación de clientes y las oportunidades de ventas. También mejora la eficiencia organizacional y la toma de decisiones, al tener disponibles datos más precisos y completos sobre los clientes, y al reducir el tiempo requerido para reconciliar los datos redundantes e inconsistentes.



6.1

ORGANIZACIÓN DE LOS DATOS EN UN ENTORNO DE ARCHIVOS TRADICIONAL

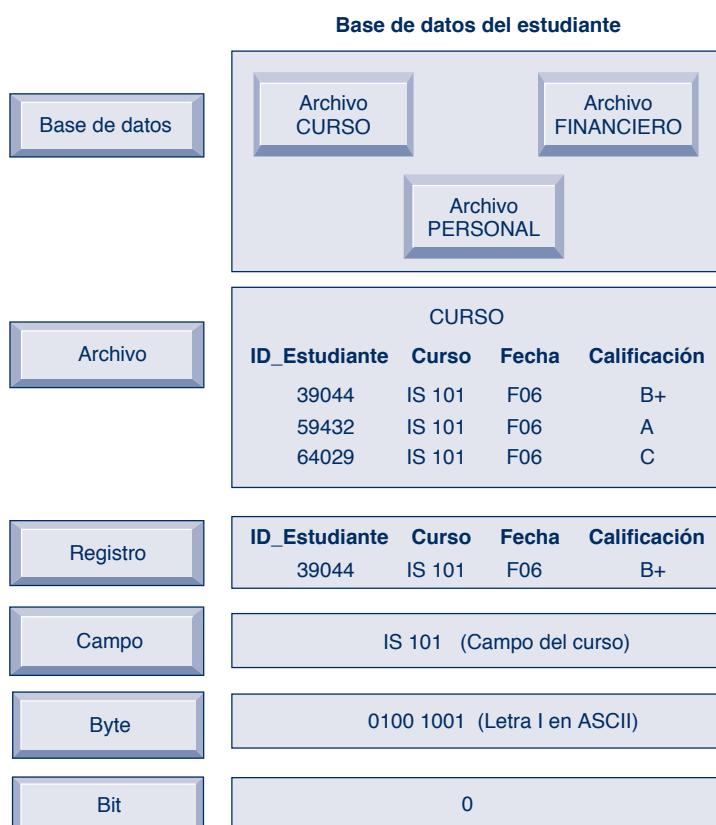
Un sistema de información efectivo provee a los usuarios información precisa, oportuna y relevante. La información precisa está libre de errores. La información es oportuna cuando está disponible para los encargados de tomar decisiones en el momento en que la necesitan. Así mismo, es relevante cuando es útil y apropiada tanto para los tipos de trabajos como para las decisiones que la requieren.

Tal vez le sorprenda saber que muchas empresas no tienen información oportuna, precisa o relevante debido a que los datos en sus sistemas de información han estado mal organizados y se les ha dado un mantenimiento inapropiado. Ésta es la razón por la que la administración de los datos es tan esencial. Para comprender el problema, veamos cómo es que los sistemas de información organizan los datos en archivos de computadora, junto con los métodos tradicionales de administración de archivos.

TÉRMINOS Y CONCEPTOS DE ORGANIZACIÓN DE ARCHIVOS

Un sistema computacional organiza los datos en una jerarquía que empieza con bits y bytes, y progresá hasta llegar a los campos, registros, archivos y bases de datos (vea la figura 6-1). Un bit representa la unidad más pequeña de datos que una computadora pue-

FIGURA 6-1 LA JERARQUÍA DE DATOS



Un sistema computacional organiza los datos en una jerarquía, la cual empieza con el bit, que representa 0 o 1. Los bits se pueden agrupar para formar un byte que representa un carácter, número o símbolo. Los bytes se pueden agrupar para formar un campo, y los campos relacionados para constituir un registro. Los registros relacionados se pueden reunir para crear un archivo, y los archivos relacionados se pueden organizar en una base de datos.

de manejar. Un grupo de bits, denominado byte, representa a un solo carácter, que puede ser una letra, un número u otro símbolo. Un agrupamiento de caracteres en una palabra, un conjunto de palabras o un número completo (como el nombre o la edad de una persona) se denomina **campo**. Un grupo de campos relacionados, como el nombre del estudiante, el curso que va a tomar, la fecha y la calificación, representan un **registro**; un grupo de registros del mismo tipo se denomina **archivo**.

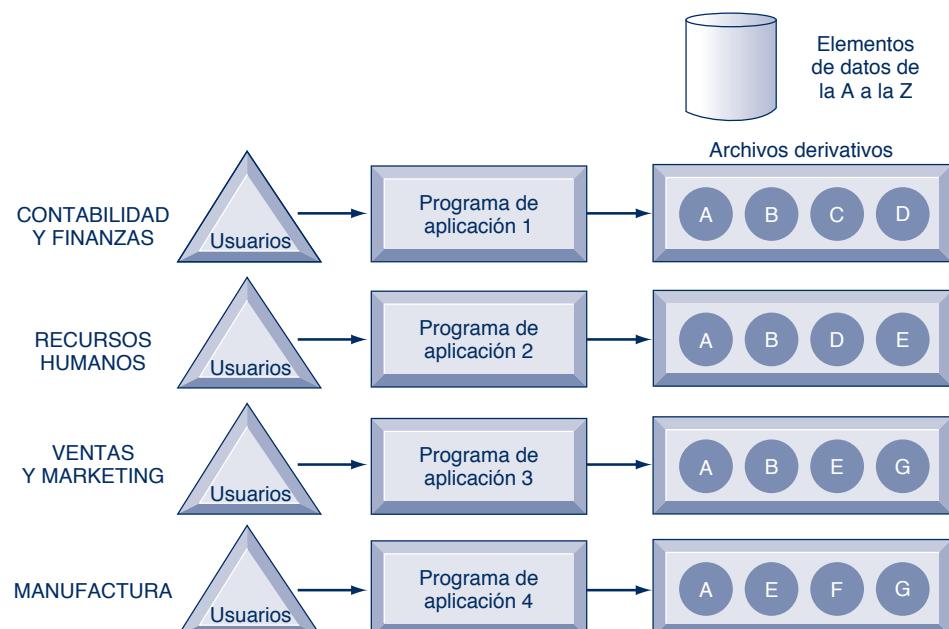
Por ejemplo, los registros en la figura 6-1 podrían constituir un archivo de cursos de estudiantes. Un grupo de archivos relacionados constituye una **base de datos**. El archivo de cursos de estudiantes que se ilustra en la figura 6-1 se podría agrupar con los archivos en los historiales personales de los estudiantes y sus antecedentes financieros, para crear una base de datos de estudiantes.

Un registro describe a una entidad. Una **entidad** es una persona, lugar, cosa o evento sobre el cual almacenamos y mantenemos información. Cada característica o cualidad que describe a una entidad específica se denomina **atributo**. Por ejemplo, ID_Estudiante, Curso, Fecha y Calificaciones son atributos de la entidad CURSO. Los valores específicos que pueden tener estos atributos se encuentran en los campos del registro que describe a la entidad CURSO.

PROBLEMAS CON EL ENTORNO DE ARCHIVOS TRADICIONAL

En la mayoría de las organizaciones, los sistemas tendían a crecer de manera independiente sin un plan a nivel de toda la compañía. Contabilidad, finanzas, manufactura, recursos humanos, ventas y marketing han desarrollado sus propios sistemas y archivos de datos. La figura 6-2 ilustra la metodología normal para el procesamiento de la información.

FIGURA 6-2 PROCESAMIENTO DE ARCHIVOS TRADICIONAL



El uso de una metodología tradicional para el procesamiento de archivos impulsa a cada área funcional en una corporación a desarrollar aplicaciones especializadas. Cada aplicación requiere un archivo de datos único que probablemente sea un subconjunto del archivo maestro. Estos subconjuntos producen redundancia e inconsistencia en los datos, inflexibilidad en el procesamiento y desperdicio de los recursos de almacenamiento.

Desde luego que cada aplicación requería sus propios archivos y programa para operar. Por ejemplo, el área funcional de recursos humanos podría tener un archivo maestro de personal, uno de nómina, uno de seguros médicos, uno de pensiones, uno de listas de correos y así en lo sucesivo, hasta que existieran decenas, tal vez cientos, de archivos y programas. En la compañía en general, este proceso condujo a varios archivos maestros creados, mantenidos y operados por divisiones o departamentos separados. A medida que continúa este proceso durante cinco o 10 años, la organización se ve atestada de cientos de programas y aplicaciones que son muy difíciles de mantener y administrar. Los problemas resultantes son la redundancia e inconsistencia de los datos, la dependencia programa-datos, la inflexibilidad, la seguridad defectuosa de los datos y la incapacidad de compartir datos entre aplicaciones.

Redundancia e inconsistencia de los datos

La **redundancia de los datos** es la presencia de datos duplicados en varios archivos, de modo que se almacenen los mismos datos en más de un lugar o ubicación. La redundancia ocurre cuando distintos grupos en una organización recolectan por separado la misma pieza de datos y la almacenan de manera independiente unos de otros. Desperdicia recursos de almacenamiento y también conduce a la **inconsistencia de los datos**, en donde el mismo atributo puede tener distintos valores. Por ejemplo, en las instancias de la entidad CURSO que se ilustran en la figura 6-1, la Fecha puede estar actualizada en algunos sistemas pero no en otros. El mismo atributo, ID_Estudiante, también puede tener distintos nombres en los distintos sistemas en toda la organización. Por ejemplo, algunos sistemas podrían usar ID_Estudiante y otros ID.

Asimismo se podría generar una confusión adicional al utilizar distintos sistemas de codificación para representar los valores de un atributo. Por ejemplo, los sistemas de ventas, inventario y manufactura de un vendedor minorista de ropa podrían usar distintos códigos para representar el tamaño de las prendas. Un sistema podría representar el tamaño como "extra grande", mientras que otro podría usar el código "XL" para el mismo fin. La confusión resultante dificultaría a las compañías el proceso de crear sistemas de administración de relaciones con el cliente, de administración de la cadena de suministro o sistemas empresariales que integren datos provenientes de distintas fuentes.

Dependencia programa-datos

La **dependencia programa-datos** se refiere al acoplamiento de los datos almacenados en archivos y los programas específicos requeridos para actualizar y dar mantenimiento a esos archivos, de tal forma que los cambios en los programas requieran cambios en los datos. Todo programa de computadora tradicional tiene que describir la ubicación y naturaleza de los datos con los que trabaja. En un entorno de archivos tradicional, cualquier cambio en un programa de software podría requerir un cambio en los datos a los que accede ese programa. Tal vez un programa se modifique de un código postal de cinco dígitos a nueve. Si el archivo de datos original se cambiara para usar códigos postales de nueve dígitos en vez de cinco, entonces otros programas que requirieran el código postal de cinco dígitos ya no funcionarían en forma apropiada. La implementación apropiada de dichos cambios podría costar millones de dólares.

Falta de flexibilidad

Un sistema de archivos tradicional puede entregar informes programados de rutina después de cierto esfuerzo extenso de programación, pero no puede entregar informes ad hoc ni responder de manera oportuna a los requerimientos de información no anticipados. La información requerida por las solicitudes ad hoc está en alguna parte del sistema, pero tal vez sea demasiado costoso recuperarla. Tal vez varios programadores tengan que trabajar durante semanas para reunir los elementos de datos requeridos en nuevo archivo.

Seguridad defectuosa

Como hay poco control o poca administración de los datos, el acceso a la información, así como su diseminación, pueden estar fuera de control. La gerencia tal vez no tenga forma de saber quién está accediendo a los datos de la organización, o incluso modificándolos.

Falta de compartición y disponibilidad de los datos

Como las piezas de información en los distintos archivos y las diferentes partes de la organización no se pueden relacionar entre sí, es casi imposible compartir o acceder a la información de una manera oportuna. La información no puede fluir con libertad entre áreas funcionales o partes de la organización distintas. Si los usuarios encuentran valores desiguales de la misma pieza de información en dos sistemas diferentes, tal vez no quieran usar estos sistemas debido a que no pueden confiar en la precisión de sus datos.

6.2

LA METODOLOGÍA DE LAS BASES DE DATOS PARA LA ADMINISTRACIÓN DE DATOS

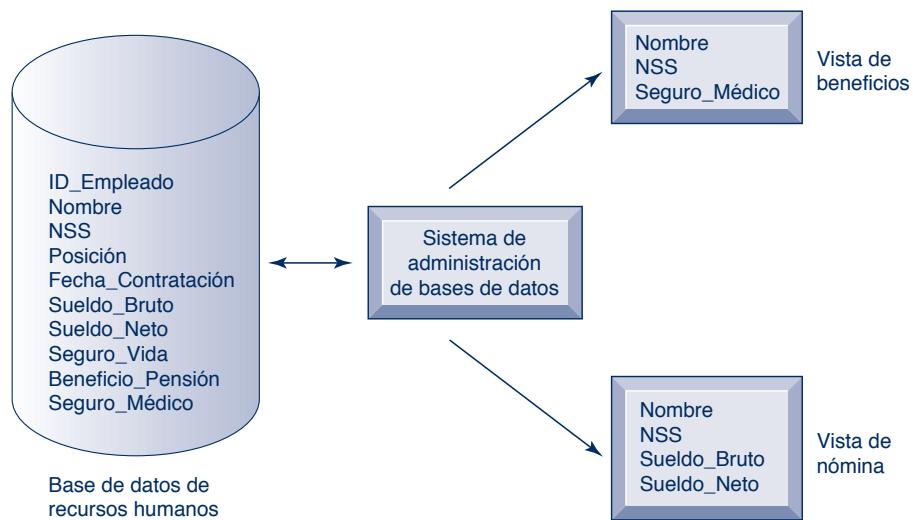
La tecnología de las bases de datos resuelve muchos de los problemas de la organización de los archivos tradicionales. Una definición más rigurosa de una **base de datos** es la de una colección de datos organizados para dar servicio a muchas aplicaciones de manera eficiente, al centralizar los datos y controlar los que son redundantes. En vez de guardar los datos en archivos separados para cada aplicación, se almacenan de modo que los usuarios crean que están en una sola ubicación. Una sola base de datos da servicio a varias aplicaciones. Por ejemplo, en vez de que una corporación almacene los datos de los empleados en sistemas de información y archivos separados para personal, nómina y beneficios, podría crear una sola base de datos común de recursos humanos.

SISTEMAS DE ADMINISTRACIÓN DE BASES DE DATOS

Un **Sistema de Administración de Bases de Datos (DBMS)** es software que permite a una organización centralizar los datos, administrarlos en forma eficiente y proveer acceso a los datos almacenados mediante programas de aplicación. El DBMS actúa como una interfaz entre los programas de aplicación y los archivos de datos físicos. Cuando el programa de aplicación solicita un elemento de datos, como el sueldo bruto, el DBMS lo busca en la base de datos y lo presenta al programa de aplicación. Si utilizará archivos de datos tradicionales, el programador tendría que especificar el tamaño y formato de cada elemento de datos utilizado en el programa y después decir a la computadora en dónde están ubicados.

El DBMS libera al programador o al usuario final de la tarea de comprender en dónde y cómo están almacenados los datos en realidad, al separar las vistas lógica y física de los datos. La *vista lógica* presenta los datos según la manera en que los perciben los usuarios finales o los especialistas de negocios, mientras que la *vista física* muestra la verdadera forma en que están organizados y estructurados los datos en los medios de almacenamiento físicos.

El software de administración de bases de datos se encarga de que la base de datos física esté disponible para las diferentes vistas lógicas requeridas por los usuarios. Por ejemplo, para la base de datos de recursos humanos que se ilustra en la figura 6-3, un especialista de negocios podría requerir una vista que consista en el nombre del empleado, número de seguro social y cobertura del seguro médico. El miembro de un departamento de nómina podría necesitar datos tales como el nombre del empleado, el número de seguro social, el sueldo bruto y neto. Los datos para todas estas vistas se

FIGURA 6-3 BASE DE DATOS DE RECURSOS HUMANOS CON VARIAS VISTAS

Una sola base de datos de recursos humanos provee muchas vistas distintas de los datos, dependiendo de los requerimientos de información del usuario. Aquí se ilustran dos posibles vistas, una de interés para un especialista de beneficios y otra de interés para un miembro del departamento de nómina de la compañía.

almacenar en una sola base de datos, en donde la organización puede administrarlos con más facilidad.

Cómo resuelve un DBMS los problemas del entorno de archivos tradicionales

Un DBMS reduce la redundancia e inconsistencia de los datos al minimizar los archivos aislados en los que se repiten los mismos datos. Tal vez el DBMS no logre que la organización elimine la redundancia de datos en su totalidad, pero puede ayudar a controlarla. Incluso si la organización mantiene ciertos datos redundantes, el uso de un DBMS elimina la inconsistencia de los datos debido a que puede ayudar a la organización a asegurar que cada ocurrencia de datos redundantes tenga los mismos valores. El DBMS desacopla los programas y los datos, con lo cual estos últimos se pueden independizar. El acceso y la disponibilidad de la información serán mayores, a la vez que se reducirán los costos de desarrollo y mantenimiento de los programas debido a que los usuarios y programadores pueden realizar consultas ad hoc de la información en la base de datos. El DBMS permite a la organización administrar los datos, su uso y su seguridad en forma central.

DBMS relacional

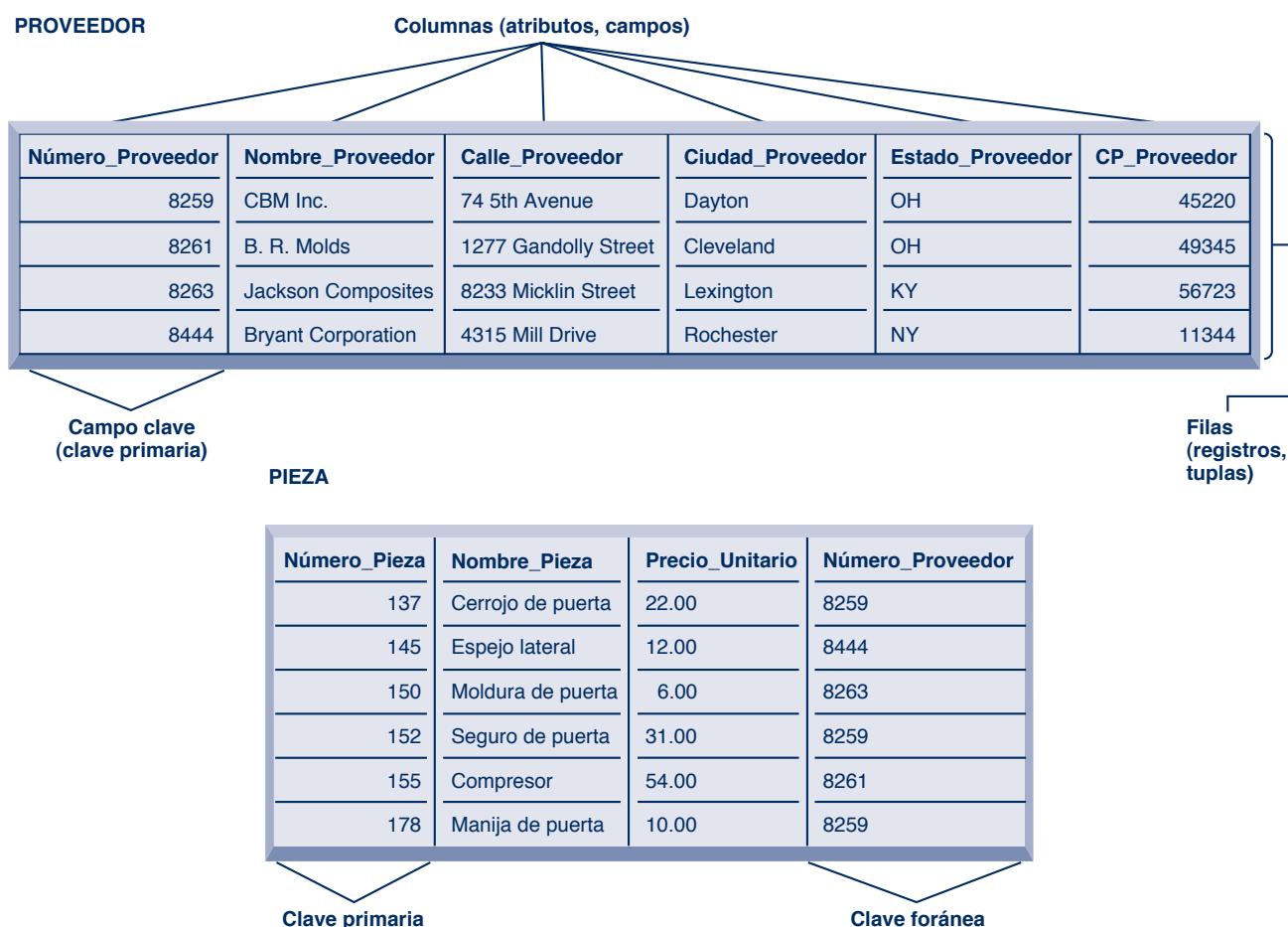
Los DBMS contemporáneos utilizan distintos modelos de bases de datos para llevar el registro de las entidades, atributos y relaciones. El tipo más popular de sistemas DBMS en la actualidad para las PCs, así como para computadoras más grandes y mainframes, es el **DBMS relacional**. Las bases de datos relacionales representan los datos como tablas bidimensionales (llamadas relaciones), a las cuales se puede hacer referencia como si fueran archivos. Cada tabla contiene datos sobre una entidad y sus atributos. Microsoft Access es un DBMS relacional para sistemas de escritorio, mientras que DB2, Oracle Database y Microsoft SQL Server son DBMS relacionales para las grandes mainframes y las computadoras de rango medio. MySQL es un popular DBMS de código fuente abierto; Oracle Database Lite es un DBMS para pequeños dispositivos de cómputo portátiles.

Veamos ahora cómo organiza una base de datos relacional la información sobre proveedores y piezas (vea la figura 6-4). La base de datos tiene una tabla separada para la entidad PROVEEDOR y una para la entidad PIEZA. Cada tabla consiste en una cuadrícula de columnas y filas de datos. Cada elemento individual de datos para cada entidad se almacena como un campo separado, y cada campo representa un atributo para esa entidad. Los campos en una base de datos relacionales también se llaman columnas. Para la entidad PROVEEDOR, el número de identificación de proveedor, nombre, calle, ciudad, estado y código postal se almacenan como campos separados dentro de la tabla PROVEEDOR y cada campo representa un atributo para la entidad PROVEEDOR.

La información real sobre un solo proveedor que reside en una tabla se denomina fila. Por lo general las filas se conocen como registros, o en términos muy técnicos, como **tuplas**. Los datos para la entidad PIEZA tienen su propia tabla separada.

El campo para Nombre_Proveedor en la tabla PROVEEDOR identifica a cada registro de forma única, de modo que ese registro se pueda recuperar, actualizar u ordenar, y se denomina **campo clave**. Cada tabla en una base de datos relacional tiene un campo que se designa como su **clave primaria**. Este campo clave es el identificador único para toda la información en cualquier fila de la tabla y su clave primaria no puede estar duplicada. Numero_Proveedor es la clave primaria para la tabla PROVEEDOR y

FIGURA 6-4 TABLAS DE BASES DE DATOS RELACIONALES



Una base de datos relacional organiza los datos en forma de tablas bidimensionales. Aquí se ilustran las tablas para las entidades PROVEEDOR y PIEZA, las cuales muestran cómo representan a cada entidad y sus atributos. Numero_Proveedor es una clave primaria para la tabla PROVEEDOR y una clave foránea para la tabla PIEZA.

Numero_Pieza es la clave primaria para la tabla PIEZA. Observe que Numero_Proveedor aparece tanto en la tabla PROVEEDOR como en PIEZA. En la tabla PROVEEDOR, Numero_Proveedor es la clave primaria. Cuando el campo Numero_Proveedor aparece en la tabla PIEZA se denomina **clave foránea**, la cual es en esencia un campo de búsqueda para averiguar datos sobre el proveedor de una pieza específica.

Operaciones de un DBMS relacional

Las tablas de bases de datos relacionales se pueden combinar con facilidad para ofrecer los datos requeridos por los usuarios, siempre y cuando dos tablas cualesquiera comparten un elemento de datos común. Suponga que queremos encontrar en esta base de datos los nombres de los proveedores que nos puedan suministrar el número de pieza 137 o el 150. Necesitaríamos información de dos tablas: la tabla PROVEEDOR y la tabla PIEZA. Observe que estos dos archivos tienen un elemento de datos compartido: Numero_Proveedor.

En una base de datos relacional se utilizan tres operaciones básicas, como se muestra en la figura 6-5, para desarrollar conjuntos útiles de datos: seleccionar, unir y proyectar. La operación *seleccionar* crea un subconjunto que consiste en todos los registros del archivo que cumplan con criterios establecidos. En otras palabras, la selección crea un subconjunto de filas que cumplen con ciertos criterios. En nuestro ejemplo, queremos seleccionar registros (filas) de la tabla PIEZA en donde el Numero_Pieza sea igual a 137 o 150. La operación *unir* combina tablas relacionales para proveer al usuario más información de la que está disponible en las tablas individuales. En nuestro ejemplo, queremos unir la tabla PIEZA, que ya está recortada (sólo se presentarán las piezas 137 o 150), con la tabla PROVEEDOR en una sola tabla nueva.

La operación *proyectar* crea un subconjunto que consiste de columnas en una tabla, con lo cual el usuario puede crear nuevas tablas que contengan sólo la información requerida. En nuestro ejemplo queremos extraer de la nueva tabla sólo las siguientes columnas: Numero_Pieza, Nombre_Pieza, Numero_Proveedor y Nombre_Proveedor.

DBMS orientado a objetos

En la actualidad y en el futuro, muchas aplicaciones requerirán bases de datos que puedan almacenar y recuperar no sólo números y caracteres estructurados, sino también dibujos, imágenes, fotografías, voz y video en movimiento completo. Los DBMS diseñados para organizar datos estructurados en filas y columnas no se adaptan bien al manejo de aplicaciones basadas en gráficos o multimedia. Las bases de datos orientadas a objetos son más adecuadas para este propósito.

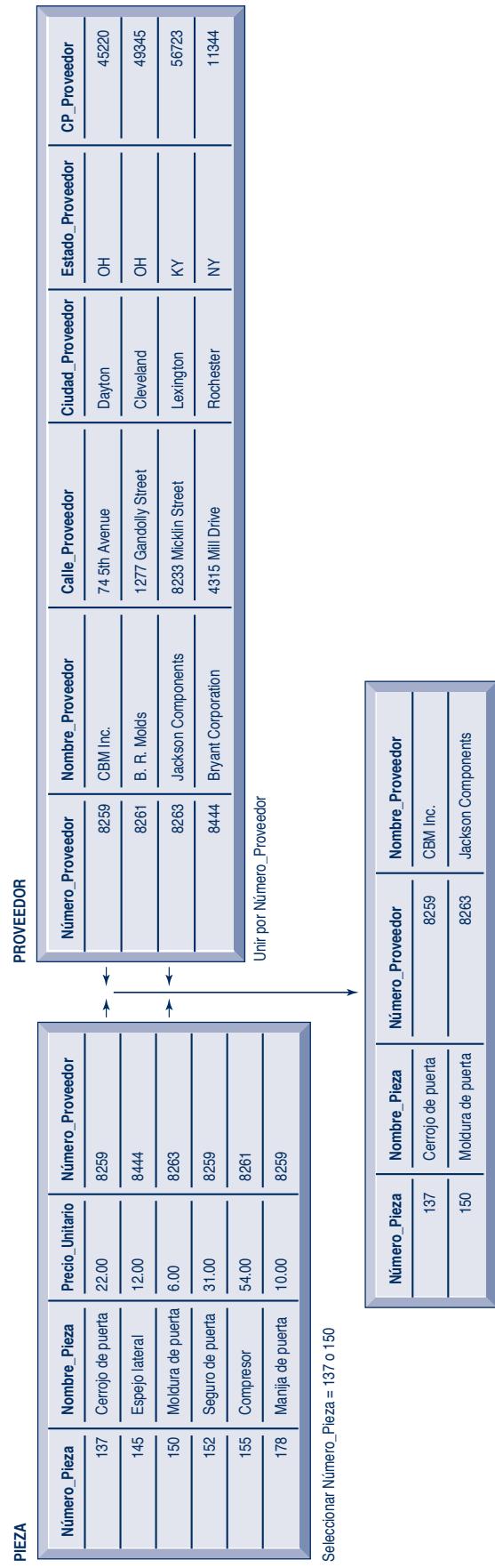
Un **DBMS orientado a objetos** almacena los datos y los procedimientos que actúan sobre esos datos como objetos que se pueden recuperar y compartir de manera automática. Los Sistemas de Administración de Bases de Datos Orientados a Objetos (OODBMS) están ganando popularidad debido a que se pueden utilizar para manejar los diversos componentes multimedia o los applets de Java que se utilizan en las aplicaciones Web, que por lo general integran piezas de información provenientes de una variedad de orígenes.

Aunque las bases de datos orientadas a objetos pueden almacenar tipos más complejos de información que los DBMS relacionales, son lentos en comparación con los DBMS relacionales para procesar grandes números de transacciones. Ahora hay sistemas **DBMS objeto-relacional** híbridos, que ofrecen las capacidades de los sistemas DBMS tanto orientados a objetos como relacionales.

Bases de datos en la nube

Suponga que su compañía desea utilizar los servicios de computación en la nube. ¿Hay alguna forma de administrar los datos en la nube? La respuesta es un "sí" condicional. Los proveedores de computación en la nube ofrecen servicios de administración de bases de datos, pero por lo general estos servicios tienen menos funcionalidad que sus contrapartes dentro de las premisas de la empresa. Por el momento, la base de clientes primordial para la administración de datos basados en la nube consiste en

FIGURA 6-5 LAS TRES OPERACIONES BÁSICAS DE UN DBMS RELACIONAL



empresas iniciales enfocadas en Web o negocios desde pequeños hasta medianos que buscan capacidades de bases de datos a un menor precio que el de un DBMS relacional estándar.

Amazon Web Services cuenta con una base de datos no relacional simple llamada SimpleDB y también con un servicio de bases de datos relacionales, el cual se basa en una implementación en línea de MySQL, el DBMS de código fuente abierto. Amazon Relational Database Service (Amazon RDS) ofrece el rango completo de capacidades de MySQL. El precio se basa en el uso (los costos varían desde 11 centavos por hora para una pequeña base de datos que utilice 1.7 GB de memoria del servidor, hasta \$3.10 por hora para una base de datos extensa que utilice 68 GB de memoria del servidor). También hay cargos por el volumen de datos almacenado, el número de solicitudes de entrada-salida, la cantidad de datos que se escriben en la base de datos y la cantidad que se leen de ella.

Además, Amazon Web Services ofrece a los clientes de Oracle la opción de obtener una licencia de Oracle Database 11g, Oracle Enterprise Manager y Oracle Fusion Middleware para ejecutarlos en la plataforma Amazon EC2 (nube de cómputo elástica).

Microsoft SQL Azure Database es un servicio de bases de datos relacionales basado en la nube y en el DBMS SQL Server de Microsoft. Ofrece un servicio de bases de datos con alta disponibilidad y escalable, hospedado por Microsoft en la nube. SQL Azure Database ayuda a reducir los costos al integrarse con las herramientas de software existentes y proveer simetría con las bases de datos tanto en las premisas de la empresa como en la nube.

TicketDirect, que vende boletos para conciertos, eventos deportivos, obras de teatro y películas en Australia y Nueva Zelanda, adoptó la plataforma de nube SQL Azure Database para poder mejorar la administración de las cargas pico del sistema durante los períodos con muchas ventas de boletos. Migró sus datos a la base de datos SQL Azure. Al cambiar a una solución en la nube, TicketDirect pudo escalar sus recursos de cómputo en respuesta a la demanda en tiempo real, al tiempo que mantuvo sus costos bajos.

CAPACIDADES DE LOS SISTEMAS DE ADMINISTRACIÓN DE BASES DE DATOS

Un DBMS incluye capacidades y herramientas para organizar, administrar y acceder a los datos en la base de datos. Las más importantes son: su lenguaje de definición de datos, el diccionario de datos y el lenguaje de manipulación de datos.

Los DBMS tienen una capacidad de **definición de datos** para especificar la estructura del contenido de la base de datos. Podría usarse para crear tablas de bases de datos y definir las características de los campos en cada tabla. Esta información sobre la base de datos se puede documentar en un **diccionario de datos**, el cual es un archivo automatizado o manual que almacena las definiciones de los elementos de datos y sus características.

Microsoft Access cuenta con una herramienta rudimentaria de diccionario de datos, la cual muestra información sobre el nombre, la descripción, el tamaño, tipo, formato y otras propiedades de cada campo en una tabla (vea la figura 6-6). Los diccionarios de datos para las grandes bases de datos corporativas pueden capturar información adicional, como el uso, la propiedad (quién en la organización es responsable de dar mantenimiento a la información), autorización, seguridad y los individuos, funciones de negocios, programas e informes que utilizan cada elemento de datos.

Consultas e informes

Un DBMS contiene herramientas para acceder a la información en las bases de datos y manipularla. La mayoría de los DBMS tienen un lenguaje especializado conocido como **lenguaje de manipulación de datos** el cual se utiliza para agregar, modificar, eliminar y recuperar los datos en la base. Este lenguaje contiene comandos que permiten a los usuarios finales y a los especialistas de programación extraer los datos de la base para satisfacer las solicitudes de información y desarrollar aplicaciones. El lenguaje de manipulación de datos más prominente en la actualidad es el **lenguaje de consulta estructurado**, o **SQL**. La figura 6-7 ilustra la consulta de SQL que produciría la nueva tabla

**FIGURA 6-6 CARACTERÍSTICAS DEL DICCIONARIO DE DATOS
DE MICROSOFT ACCESS**

FIGURA 6-8 UNA CONSULTA EN ACCESS

se construiría la misma consulta que la SQL para seleccionar piezas y proveedores, pero ahora mediante las herramientas para crear consultas de Microsoft.

Microsoft Access y otros sistemas DBMS tienen herramientas para generación de informes, de modo que se puedan mostrar los datos de interés en un formato más estructurado y elegante que el de las consultas. Crystal Reports es un popular generador de informes para los DBMS corporativos extensos, aunque también se puede utilizar con Access. Este último también cuenta con herramientas para desarrollar aplicaciones de sistemas de escritorio. Se incluyen herramientas para crear pantallas de captura de datos, para generar informes y desarrollar la lógica de procesamiento de transacciones.

DISEÑO DE BASES DE DATOS

Para crear una base de datos hay que comprender las relaciones entre la información, el tipo de datos que se mantendrán en la base, cómo se utilizarán y la forma en que tendrá que cambiar la organización para administrarlos desde una perspectiva a nivel de toda la compañía. La base de datos requiere tanto un diseño conceptual como uno físico. El diseño conceptual o lógico de la base de datos es un modelo abstracto de ésta desde una perspectiva de negocios, mientras que el diseño físico muestra la verdadera disposición de la base de datos en los dispositivos de almacenamiento de acceso directo.

Diagramas de normalización y de entidad-relación

El diseño de bases de datos conceptual describe la forma en que se deben agrupar los elementos de datos en la base. El proceso de diseño identifica las relaciones entre los elementos de datos y la manera más eficiente de agruparlos en conjunto para satisfacer los requerimientos de información de la empresa. Este proceso también identifica a los elementos de datos redundantes y las agrupaciones de elementos de datos requeridas

FIGURA 6-9 UNA RELACIÓN SIN NORMALIZAR PARA PEDIDO

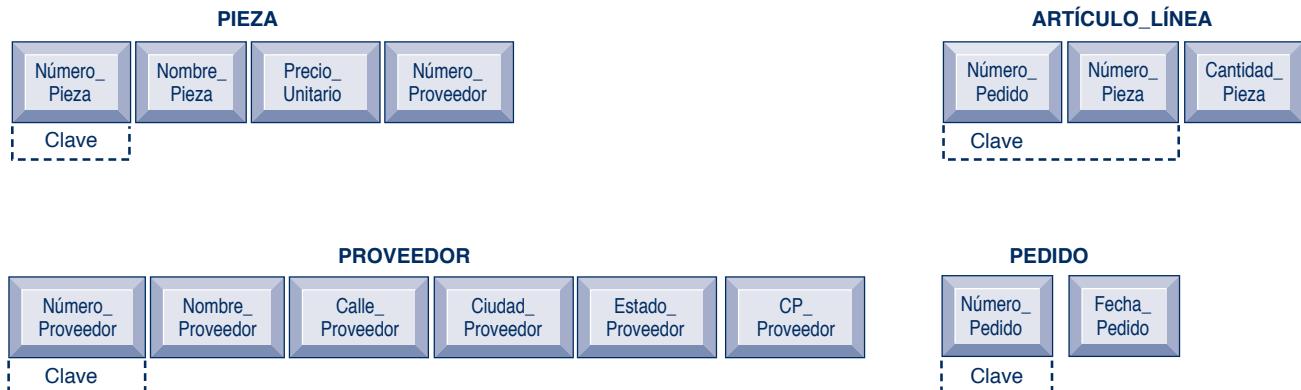
Una relación sin normalizar contiene grupos repetitivos. Por ejemplo, puede haber muchas piezas y proveedores para cada pedido. Sólo hay una correspondencia de uno a uno entre Número_Pedido y Fecha_Pedido.

para ciertos programas de aplicaciones específicos. Los grupos de datos se organizan, refinan y optimizan hasta que emerge una vista lógica general de las relaciones entre todos los datos en la base.

Para usar un modelo de base de datos relacional en forma efectiva, hay que optimizar los agrupamientos complejos de datos para minimizar los elementos de datos redundantes y las incómodas relaciones de varios a varios. El proceso de crear estructuras de datos pequeñas y estables pero a la vez flexibles y adaptivas a partir de grupos complejos de datos se denomina **normalización**. Las figuras 6-9 y 6-10 ilustran este proceso.

En la empresa específica que se modela aquí, un pedido puede tener más de una pieza, pero cada una sólo es proporcionada por un proveedor. Si creamos una relación llamada PEDIDO con todos los campos que se incluyen aquí, tendríamos que repetir el nombre y la dirección del proveedor para cada pieza del pedido, aun y cuando éste sea de piezas de un solo proveedor. Esta relación contiene lo que se denomina grupos de datos repetitivos, ya que puede haber muchas piezas en un solo pedido para un proveedor dado. Una manera más eficiente de ordenar los datos es dividir PEDIDO en relaciones más pequeñas, cada una de las cuales describe a una sola entidad. Si avanzamos paso a paso y normalizamos la relación PEDIDO, obtendremos las relaciones que se ilustran en la figura 6-10. Para averiguar más sobre la normalización, los diagramas entidad-relación y el diseño de bases de datos, consulte las Trayectorias de aprendizaje de este capítulo.

Los sistemas de bases de datos relacionales tratan de cumplir reglas de **integridad referencial** para asegurar que las relaciones entre las tablas acopladas permanezcan consistentes. Cuando una tabla tiene una clave foránea que apunta a otra, no es posible agregar un registro a la tabla con la clave foránea a menos que haya uno correspondiente en la tabla vinculada. En la base de datos que examinamos antes en el

FIGURA 6-10 TABLAS NORMALIZADAS CREADAS A PARTIR DE PEDIDO

Después de la normalización, la relación original PEDIDO se divide en cuatro relaciones más pequeñas. La relación PEDIDO se queda con sólo dos atributos y la relación ARTICULO_LINEA tiene una clave combinada, o concatenada, que consiste en Número_Pedido y Número_Pieza.

capítulo, la clave foránea Numero_Proveedor vincula la tabla PIEZA con la tabla PROVEEDOR. No podemos agregar un nuevo registro a la tabla PIEZA para una pieza con el Numero_Proveedor 8266 a menos que haya un registro correspondiente en la tabla PROVEEDOR para el Numero_Proveedor 8266. También debemos eliminar el registro correspondiente en la tabla PIEZA si quitamos el registro en la tabla PROVEEDOR para el Numero_Proveedor 8266. En otras palabras, ¡no debemos tener piezas de proveedores que no existen!

Los diseñadores de bases de datos documentan su modelo de datos con un **diagrama entidad-relación**, el cual se ilustra en la figura 6-11. Este diagrama muestra la relación entre las entidades PROVEEDOR, PIEZA, ARTICULO_LINEA y PEDIDO. Los cuadros representan las entidades, y las líneas que conectan los cuadros, las relaciones. Una línea que conecta dos entidades que termina en dos marcas cortas designa una relación de uno a uno. Una línea que conecta dos entidades y termina con una pata de cuervo y una marca corta encima de ella indica una relación de uno a varios. La figura 6-11 muestra que un PEDIDO puede contener varios ARTICULO_LINEA (es posible ordenar una PIEZA muchas veces y aparecer otras tantas como artículo de línea en un solo pedido). Cada PIEZA sólo puede tener un PROVEEDOR, pero muchos elementos PIEZA pueden ser proporcionados por el mismo PROVEEDOR.

No podemos enfatizarlo lo suficiente: si el modelo de datos de la empresa no es el correcto, el sistema no podrá dar buen servicio a la empresa. Los sistemas de la compañía no serán tan efectivos como podrían serlo debido a que tendrán que trabajar con datos que tal vez sean imprecisos, incompletos o difíciles de recuperar. Comprender los datos de la organización y la forma en que se deben representar en una base de datos es tal vez la lección más importante que puede usted aprender de este curso.

Por ejemplo, Famous Footwear, una cadena de zapaterías con más de 800 sucursales en 49 estados, no pudo lograr su objetivo de tener “el estilo correcto de zapato en la tienda apropiada para venderse al precio adecuado”, ya que su base de datos no estaba diseñada en forma correcta para ajustar con rapidez el inventario de las tiendas. La compañía tenía una base de datos relacional Oracle operando en una computadora IBM AS/400 de medio rango, pero el objetivo primordial para el que se diseñó la base de datos era producir informes estándar para la gerencia, en vez de reaccionar a los cambios en el mercado. La gerencia no pudo obtener datos precisos sobre artículos específicos en el inventario en cada una de sus tiendas. Para solucionar este problema, la compañía tuvo que crear una nueva base de datos en donde se pudieran organizar mejor los datos de las ventas y del inventario para realizar análisis y administrar el inventario.

6.3

USO DE BASES DE DATOS PARA MEJORAR EL DESEMPEÑO DE NEGOCIOS Y LA TOMA DE DECISIONES

Las empresas utilizan sus bases de datos para llevar el registro de las transacciones básicas, como pagar a los proveedores, procesar pedidos, llevar el registro de los clientes y pagar a los empleados. Pero también se necesitan bases de datos para proveer

FIGURA 6-11 UN DIAGRAMA ENTIDAD-RELACIÓN



El diagrama muestra las relaciones entre las entidades PROVEEDOR, PIEZA, ARTICULO_LINEA y PEDIDO que se podrían usar para modelar la base de datos de la figura 6-10.