Database Processing

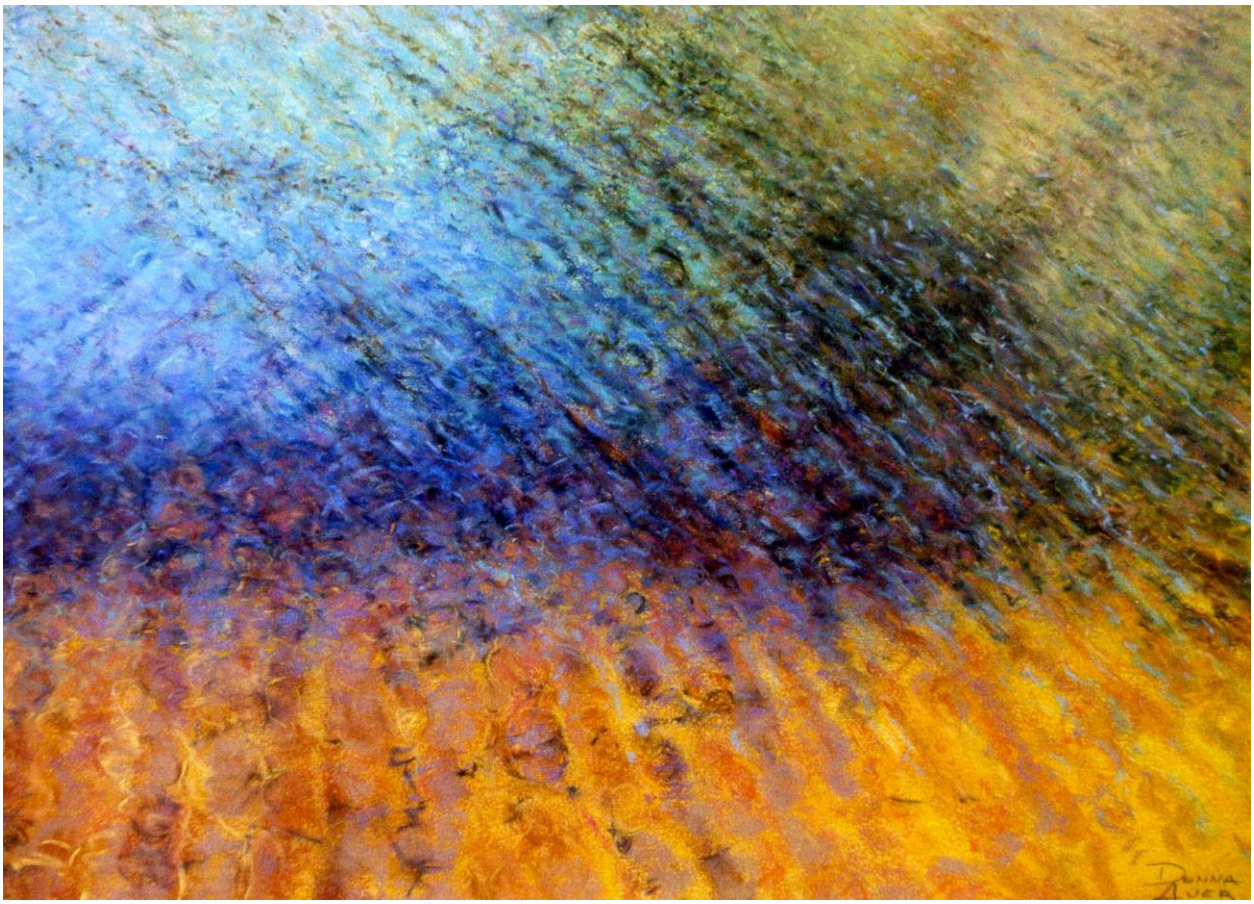
Fundamentals, Design, and Implementation

14th Edition

David M. Kroenke • David J. Auer

Online Appendix J

Business Intelligence Systems



Vice President, Business Publishing: Donna Battista
Editor in Chief: Stephanie Wall
Acquisitions Editor: Nicole Sam
Program Manager Team Lead: Ashley Santora
Program Manager: Denise Weiss
Editorial Assistant: Olivia Vignone
Vice President, Product Marketing: Maggie Moylan
Director of Marketing, Digital Services and Products: Jeanette Koskinas
Executive Product Marketing Manager: Anne Fahlgren
Field Marketing Manager: Lenny Ann Raper
Senior Strategic Marketing Manager: Erin Gardner
Product Marketing Assistant: Jessica Quazza
Project Manager Team Lead: Jeff Holcomb
Project Manager: Ilene Kahn
Operations Specialist: Diane Peirano
Senior Art Director: Janet Slowik
Text Designer: Integra Software Services Pvt. Ltd.
Cover Designer: Integra Software Services Pvt. Ltd.
Cover Art: Donna Auer
Vice President, Director of Digital Strategy & Assessment: Paul Gentile
Manager of Learning Applications: Paul Deluca
Digital Editor: Brian Surette
Digital Studio Manager: Diane Lombardo
Digital Studio Project Manager: Robin Lazrus
Digital Studio Project Manager: Alana Coles
Digital Studio Project Manager: Monique Lawrence
Digital Studio Project Manager: Regina DaSilva
Full-Service Project Management and Composition: Integra Software Services Pvt. Ltd.
Printer/Binder: RRD Willard
Cover Printer: Phoenix Color/Hagerstown
Text Font: 10/12 Mentor Std Light

Library of Congress Cataloging-in-Publication Data

Kroenke, David M.
Database processing: fundamentals, design, and implementation/David M. Kroenke, David J. Auer.–Fourteenth edition.
pages cm
Includes bibliographical references and index.
ISBN 978-0-13-387670-3 (student edition)–ISBN 978-0-13-387676-5
(instructor's review copy)
1. Database management. I. Auer, David J. II. Title.
QA76.9.D3K76 2016
005.74–dc23

2015005632

10 9 8 7 6 5 4 3 2 1

PEARSON

ISBN 10: 0-13-387670-5
ISBN 13: 978-0-13-387670-3

Appendix J – 10 9 8 7 6 5 4 3 2 1

Chapter Objectives

- To learn the basic concepts of business intelligence (BI) systems
- To learn the basic concepts of data warehouses and data marts
- To learn the basic concepts of reporting systems
- To learn the basic concepts of data mining
- To learn how to create RFM reports
- To learn the basic concepts of market basket analysis

What Is the Purpose of This Appendix?

In Chapter 12, we discussed Big Data, dimensional databases, and data warehouses in depth. We introduced business intelligence (BI) systems, and learned that they can be categorized as reporting systems and data mining systems. We then explored Online Analytical Processing (OLAP) systems, which are a type of BI reporting system. This appendix takes a more thorough look at BI systems.

Business Intelligence Systems

As discussed in Chapter 12, **Business intelligence (BI) systems** are information systems that assist managers and other professionals in the analysis of current and past activities and in the prediction of future events. Unlike transaction processing systems, they do not support operational activities, such as the recording and processing of orders. Instead, BI systems are used to support management assessment, analysis, planning, control, and, ultimately, decision making.

Reporting Systems and Data Mining Applications

BI systems fall into two broad categories: reporting systems and data mining applications. **Reporting systems** sort, filter, group, and make elementary calculations on operational data. **Data mining applications**, in contrast, perform sophisticated analyses on data, analyses that usually involve complex statistical and mathematical processing. The characteristics of BI applications are summarized in Figure J-1.

Reporting Systems

Reporting systems filter, sort, group, and make simple calculations. All reporting analyses can be performed using standard SQL, although extensions to SQL, such as those used for **Online Analytical Processing (OLAP)**, are sometimes used to ease the task of report production.

Reporting systems summarize the current status of business activities and compare that status with past or predicted future activities. Report delivery is crucial. Reports must be delivered to the proper users on a timely basis, in the appropriate format. For example, reports may be delivered on paper, via a Web browser, or in some other format.

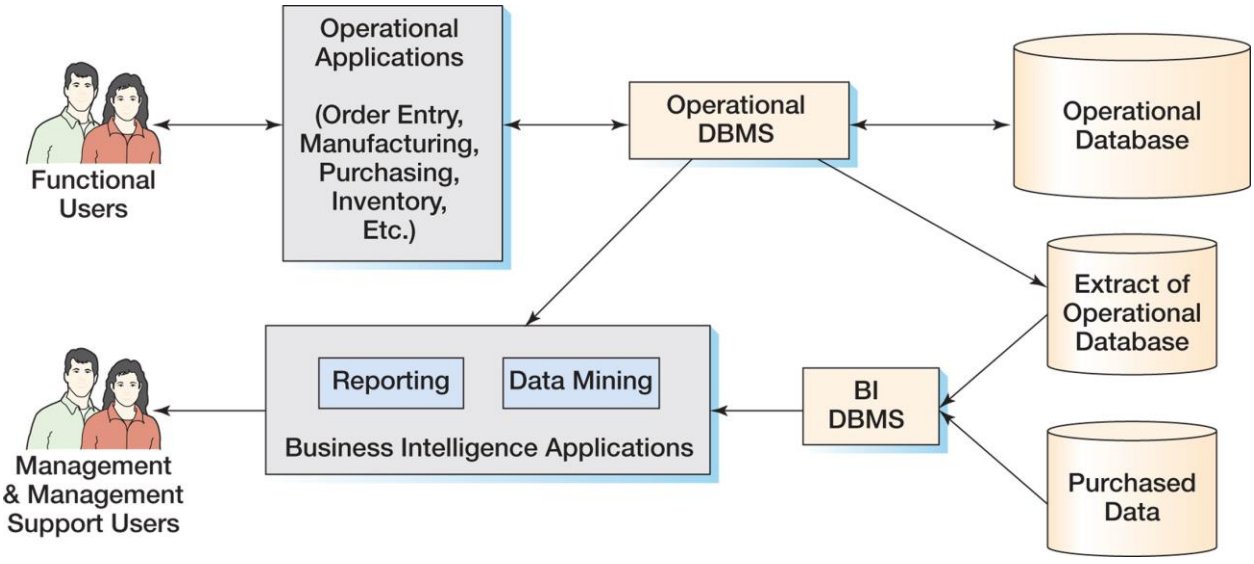


Figure J-1 — Characteristics of Business Intelligence Systems

Data Mining Applications

Data mining applications use sophisticated statistical and mathematical techniques to perform what-if analyses, to make predictions, and to facilitate decision making. For example, data mining techniques can analyze past cell phone usage and predict which customers are likely to switch to a competing phone company. Or, data mining can be used to analyze past loan behavior to determine which customers are most (or least) likely to default on a loan.

Report delivery is not as important for data mining systems as it is for reporting systems. First, most data mining applications have only a few users, and those users have sophisticated computer skills. Second, the results of a data mining analysis are usually incorporated into some other report, analysis, or information system. In the case of cell phone usage, the characteristics of customers who are in danger of switching to another company may be given to the sales department for action. Or, the parameters of an equation for determining the likelihood of a loan default may be incorporated into a loan approval application.

The Components of a Data Warehouse

A **data warehouse** is a database system that has data, programs, and personnel that specialize in the preparation of data for BI processing. Figure J-2 shows the components of the basic data warehouse architecture. Data are read from operational databases by the **Extract, Transform, and Load (ETL) system**. The ETL system then cleans and prepares the data for BI processing. This can be a complex process.

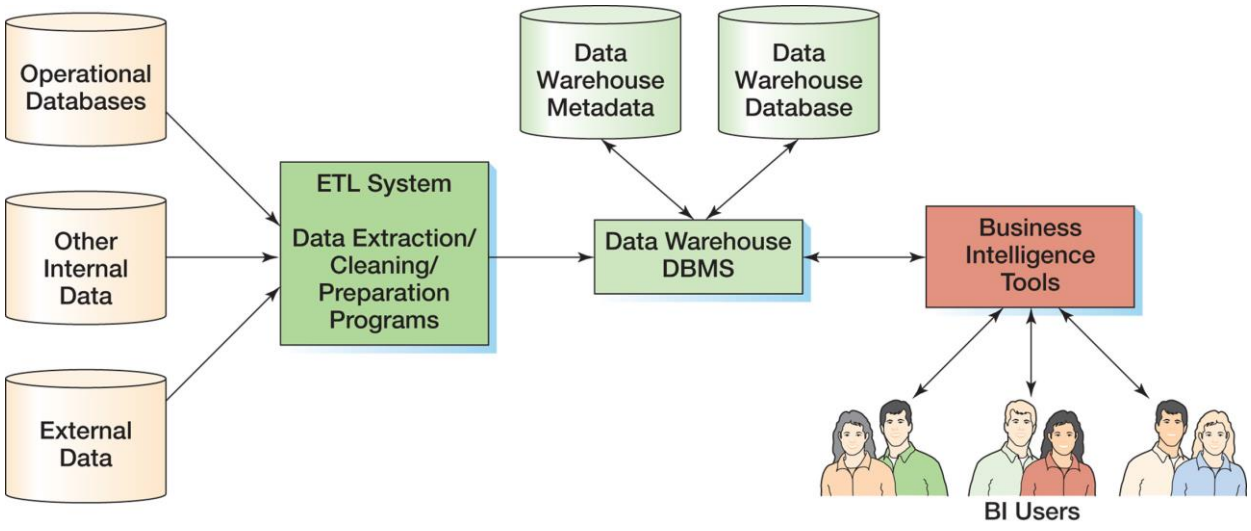


Figure J-2 — Components of a Data Warehouse

First, operational data often cannot be directly loaded into BI applications—the data may be problematic, which we will discuss in the next section. Some of the problems of using operational data for BI processing include:

- "Dirty data" (for example, problematic data such as value of "G" for customer gender, a value of "213" for customer age, a value of "999-999-9999" for a U.S. phone number, or a part color of "gren").
- Missing values.
- Inconsistent data (for example, data that have changed, such as a customer's phone number or address).
- Nonintegrated data (for example, data from two or more sources that need to be combined for BI use).
- Incorrect format (for example, data that are gathered such that there are either too many data or not enough data, such as time measures in either seconds or hours when they are needed in minutes for BI use).
- Too much data (for example, an excess of columns [attributes], rows [records], or both).

Second, data may need to be changed or transformed for use in a data warehouse. For example, the operational systems may store data about countries using standard two-letter country codes, such as US (United States) and CA (Canada). However, applications using the data warehouse may need to use the country names in full. Thus, the data transformation **{CountryCode → CountryName}** will be needed before the data can be loaded into the data warehouse.

When the data are prepared for use, the ETL system loads the data into the data warehouse database. The extracted data are stored in a data warehouse database, using a data warehouse DBMS, which may be from a different vendor than the organization's operational DBMS. For example, an organization might use Oracle for its operational processing but use SQL Server for its data warehouse.

Problematic operational data that have been cleaned in the ETL system can also be used to update the operational system to fix the original data problems.

Metadata concerning the data's source, format, assumptions and constraints, and other facts is kept in a **data warehouse metadata database**. The data warehouse DBMS provides extracts of its data to BI tools, such as data mining programs.

Data Warehouses and Data Marts

As shown in Figures J-1 and J-2, some BI applications read and process operational data directly from the operational database. Although this is possible for simple reporting systems and small databases, such direct reading of operational data is not feasible for more complex applications or larger databases. Operational data are difficult to use for several reasons:

- Querying data for BI applications can place a substantial burden on the DBMS and unacceptably slow the performance of operational applications.
- The creation and maintenance of BI systems requires application programs, facilities, and expertise that are not normally available from operations.
- Operational data have problems that limit their use for BI applications.

Therefore, larger organizations usually process a separate **data warehouse** database constructed from an extract of the operational database.

You can think of a data warehouse as a distributor in a supply chain. The data warehouse takes data from the data manufacturers (operational systems and purchased data), cleans and processes them, and locates the data on the shelves, so to speak, of the data warehouse. The people who work in a data warehouse are experts at data management, data cleaning, data transformation, and the like. However, they are not usually experts in a given business function.

A **data mart** is a collection of data that is smaller than that in the data warehouse and that addresses a particular component or functional area of the business. A data mart is like a retail store in a supply chain. Users in the data mart obtain data from the data warehouse that pertain to a particular business function. Such users do not have the data management expertise that data warehouse employees have, but they are knowledgeable analysts for a given business function.

Figure J-3 illustrates these relationships. In this example, the data warehouse takes data from the data producers and distributes the data to three data marts. One data mart analyzes **click-stream data** for the purpose of designing Web pages. A second data mart analyzes store sales data and determines which products tend to be purchased together for the purpose of training sales staff. A third data mart analyzes customer order data for the purpose of reducing labor when picking up items at the warehouse. (Companies such as Amazon.com go to great lengths to organize their warehouses to reduce picking expenses.)

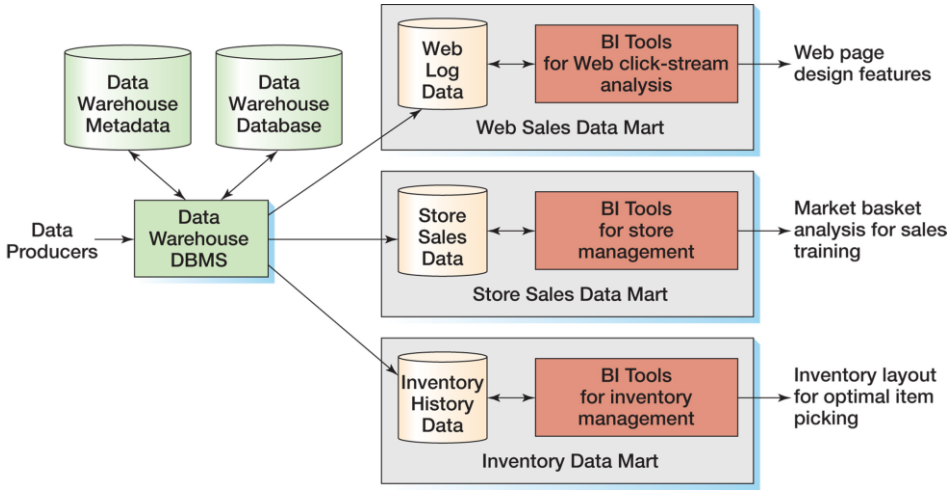


Figure J-3 — Data Warehouses and Data Marts

When the data mart structure shown in Figure J-3 is combined with the data warehouse architecture shown in Figure J-2, the combined system is known as an **enterprise data warehouse (EDW) architecture**. In this configuration, the data warehouse maintains all enterprise BI data and acts as the authoritative source for data extracts provided to the data marts. The data marts receive all their data from the data warehouse—they do not add or maintain any additional data.

Of course, it is expensive to create, staff, and operate data warehouses and data marts, and only large organizations with deep pockets can afford to operate a system such as an EDW. Smaller organizations operate subsets of such systems. For example, they may have just a single data mart for analyzing marketing and promotion data.

Data Warehouses and Dimensional Databases

As discussed in Chapter 12, data warehouse databases are built using a dimensional database design. This design typically uses a star schema, as shown in Figure J-4. In a dimensional database, fact tables (PRODUCT_SALES and SALES_FOR_RFM in the diagram) are linked to dimension tables (TIMELINE and CUSTOMER in the diagram).

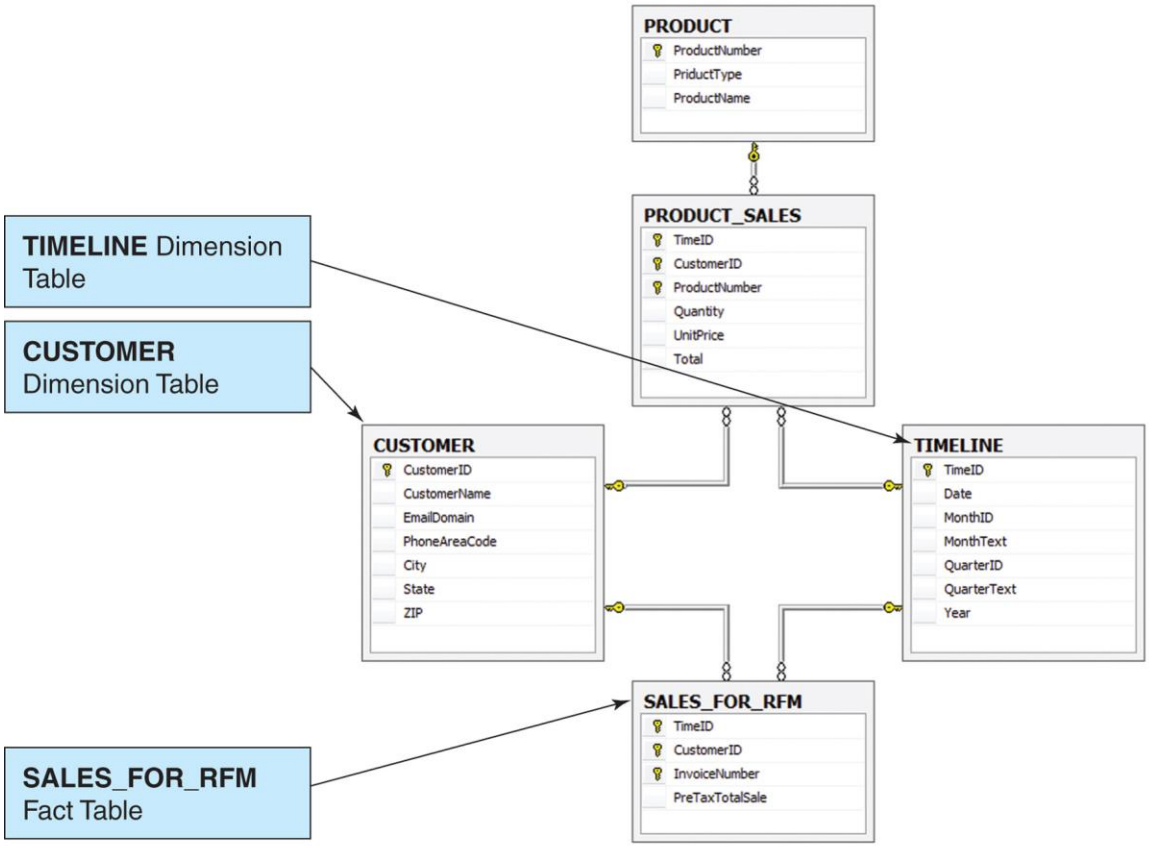


Figure J-4 — Dimensional Databases and the Star Schema

Reporting Systems

The purpose of a reporting system is to create meaningful information from disparate data sources and to deliver that information to the proper users on a timely basis. Unlike data mining, which uses sophisticated statistical techniques, reporting systems create information by using the simple operations of sorting, filtering, grouping, and making simple calculations.

It is easier to understand reporting systems if you are familiar with a typical report, so let us take a look at two typical reporting problems: OLAP and RFM analysis.

OLAP

Online Analytical Processing (OLAP), which is discussed in detail in Chapter 12, provides the ability to sum, count, average, and perform other simple arithmetic operations on groups of data. OLAP systems produce **OLAP reports**. An OLAP report is also called an **OLAP cube**. This is a reference to the dimensional data model discussed in Chapter 12, and some OLAP products show OLAP displays using three axes, like a geometric cube. The remarkable characteristic of an OLAP report is that it is dynamic: The format of an OLAP report can be changed by the viewer, hence the term *online* in the name Online Analytical Processing.

RFM Analysis

RFM analysis analyzes and ranks customers according to their purchasing patterns. It is a simple customer classification technique that considers how *recently* (**R**) a customer orders, how *frequently* (**F**) a customer orders, and *how much money* (**M**) the customer spends per order. RFM is summarized in Figure J-5.

To produce an RFM score, we need only two things: customer data and sales data for each purchase (the date of the sale and the total amount of the sale) made by each customer. If you look at the SALES_FOR_RFM table and its associated CUSTOMER and TIMELINE dimension tables in Figure J-4, you see that we have exactly those data: The SALES_FOR_RFM table is the starting point for RFM analysis in the HSD-DW BI system, as developed in Chapter 12. Although we will not do it here, RFM analysis can be done using SQL statements and a table such as SALES_FOR_RFM.

To calculate an **R score**, you first sort the customer purchase records by the date of the most recent (R) purchase. (Note that only the most recent purchase for each customer is used in this calculation.) In a common form of this analysis, the customers are then divided into five groups, and a score of 1 to 5 is given to customers in each group. The 20 percent of the customers having the most recent orders are given an R score of 1, the 20 percent of the customers having the next most recent orders are given an R score of 2, and so forth, down to the last 20 percent, who are given an R score of 5.

To calculate an **F score**, you re-sort the customers on the basis of how frequently they order. As before, the customers are again divided into five groups. The 20 percent of the customers who order most frequently are given an F score of 1, the next 20 percent most frequently ordering customers are given a score of 2, and so forth, down to the least frequently ordering customers, who are given an F score of 5.

- Simple report-based customer classification scheme
- Score customers on recentness, frequency, and monetary size of orders
- Typically, divide each criterion into 5 groups and score from 1 to 5

Figure J-5 — RFM Analysis

Each customer is ranked for **R** (recent), **F** (frequent), and **M** (money) characteristics—1 is highest (best) and 5 is lowest (worst) score

	RFM Score		
Customer	R	F	M
Able, Ralph	1	1	2
Baker, Susan	2	2	3
George, Sally	3	3	3
Tyler, Jenny	5	1	1
Jacobs, Chantel	5	5	5

Figure J-6 — The RFM Score Report

To calculate an **M score**, you re-sort the customers according to the average amount of their orders. The 20 percent who have placed the biggest orders are given an M score of 1, the next 20 percent are given an M score of 2, and so forth, down to the 20 percent who spend the least, who are given an M score of 5.

Figure J-6 shows sample RFM data for Heather Sweeney Designs. (Note that this data have *not* been calculated and are for illustrative purposes only.) The first customer, Ralph Able, has a score of {1 1 2}, which means that he has ordered recently and orders frequently. His M score of 2 indicates, however, that he does not order the most expensive goods. From these scores, the salespeople can surmise that Ralph is a good customer but that they should attempt to up-sell Ralph to more expensive goods.

Susan Baker is above average in terms of how frequently she shops and how recently she shops, but her purchases are average in value. Sally George is truly in the middle. Jenny Tyler could be a problem. Jenny has not ordered in some time, but in the past, when she did order, she ordered frequently, and her orders were of the highest monetary value. These data suggest that Jenny may be going to another vendor. Someone from the sales team should contact her immediately. However, no one on the sales team should be talking to Chantel Jacobs. She has not ordered for some time, she does not order frequently, and, when she does order, she only buys inexpensive items and not many of them.

Producing the RFM Report

Like most reports, an RFM report can be created using a series of SQL expressions. This section presents two SQL Server stored procedures that produce RFM scores. Figure J-7 shows the SQL scripts to create the five tables that are used.

```
CREATE TABLE CUSTOMER_SALES (
    TransactionID           Int             NOT NULL,
    CustomerID              Int             NOT NULL,
    TransactionDate         Date            NOT NULL,
    OrderAmount             Money           NOT NULL,
    CONSTRAINT  Customer_Sales_PK   PRIMARY KEY(TransactionID)
    );

CREATE TABLE CUSTOMER_RFM (
    CustomerID              Int             NOT NULL,
    R                       SmallInt        NULL,
    F                       SmallInt        NULL,
    M                       SmallInt        NULL,
    CONSTRAINT  Customer_RFM_PK     PRIMARY KEY(CustomerID)
    );

CREATE TABLE CUSTOMER_R (
    CustomerID              Int             NOT NULL,
    MostRecentOrderDate     Date            NULL,
    R_Score                 SmallInt        NULL,
    CONSTRAINT  Customer_R_PK       PRIMARY KEY(CustomerID)
    );

CREATE TABLE CUSTOMER_F (
    CustomerID              Int             NOT NULL,
    OrderCount              Int             NULL,
    F_Score                 SmallInt        NULL,
    CONSTRAINT  Customer_F_PK       PRIMARY KEY(CustomerID)
    );

CREATE TABLE CUSTOMER_M (
    CustomerID              Int             NOT NULL,
    AverageOrderAmount      Money           NULL,
    M_Score                 SmallInt        NULL,
    CONSTRAINT  Customer_M_PK       PRIMARY KEY(CustomerID)
    );
```

Figure J-7 — Microsoft SQL Server 2014 Tables for the RFM Analysis

The CUSTOMER_SALES table contains the raw data that are used in the RFM calculations. CUSTOMER_RFM contains CustomerID and the final R, F, and M scores. The remaining three tables—CUSTOMER_R, CUSTOMER_F, and CUSTOMER_M—are used to store intermediate results. Note that all CustomerID columns are NOT NULL.

```
CREATE PROCEDURE RFM_Analysis

AS

/* Delete any existing RFM data  *************************/

DELETE FROM CUSTOMER_RFM;
DELETE FROM CUSTOMER_R;
DELETE FROM CUSTOMER_F;
DELETE FROM CUSTOMER_M;

/* *** Compute R, F, M Scores ****************************/
Exec Calculate_R;
Exec Calculate_F;
Exec Calculate_M;

/* *** Dipslay Results ***********************************/
SELECT      R_Score, Count(*) AS R_Count
FROM        CUSTOMER_R
GROUP BY    R_Score;

SELECT      F_Score, Count(*) AS F_Count
FROM        CUSTOMER_F
GROUP BY    F_Score;

SELECT      M_Score, Count(*) AS M_Count
FROM        CUSTOMER_M
GROUP BY    M_Score;

/* *** Store Results *************************************/
INSERT INTO CUSTOMER_RFM (CustomerID)
    (SELECT CustomerID
    FROM    CUSTOMER_SALES);

UPDATE CUSTOMER_RFM
SET R =
    (SELECT R_Score
     FROM CUSTOMER_R
     WHERE CUSTOMER_RFM.CustomerID = CUSTOMER_R.CustomerID);

UPDATE CUSTOMER_RFM
SET F =
    (SELECT F_Score
     FROM CUSTOMER_F
     WHERE CUSTOMER_RFM.CustomerID = CUSTOMER_F.CustomerID);

UPDATE CUSTOMER_RFM
SET M =
    (SELECT M_Score
     FROM CUSTOMER_M
     WHERE CUSTOMER_RFM.CustomerID = CUSTOMER_M.CustomerID);

/* *** End of Procedure RFM Analysis *********************/
```

Figure J-8 — The Microsoft SQL Server 2014 RFM_Analysis Stored Procedure

```
CREATE PROCEDURE Calculate_R

AS

/* *** Compute R_Score  ***********************************/

INSERT INTO CUSTOMER_R (CustomerID, MostRecentOrderDate)
    (SELECT CustomerID, MAX (TransactionDate)
     FROM   CUSTOMER_SALES
     GROUP BY CustomerID);

UPDATE  CUSTOMER_R
    SET R_Score = 1
    WHERE CustomerID IN
        (SELECT TOP 20 PERCENT CustomerID
         FROM CUSTOMER_R
         ORDER BY MostRecentOrderDate DESC);

UPDATE  CUSTOMER_R
    SET R_Score = 2
    WHERE CustomerID IN
        (SELECT TOP 25 PERCENT CustomerID
         FROM CUSTOMER_R
         WHERE R_Score IS NULL
         ORDER BY MostRecentOrderDate DESC);

UPDATE  CUSTOMER_R
    SET R_Score = 3
    WHERE CustomerID IN
        (SELECT TOP 33 PERCENT CustomerID
         FROM CUSTOMER_R
         WHERE R_Score IS NULL
         ORDER BY MostRecentOrderDate DESC);

UPDATE  CUSTOMER_R
    SET R_Score = 4
    WHERE CustomerID IN
        (SELECT TOP 50 PERCENT CustomerID
         FROM CUSTOMER_R
         WHERE R_Score IS NULL
         ORDER BY MostRecentOrderDate DESC);

UPDATE  CUSTOMER_R
    SET R_Score = 5
    WHERE CustomerID IN
        (Select CustomerID
         FROM CUSTOMER_R
         WHERE R_Score IS NULL);
```

Figure J-9 — The Microsoft SQL Server 2014 Calculate_R Stored Procedure

The stored procedure shown in Figure J-8 is used to calculate and store the R, F, and M scores. It begins by deleting the results from any prior analysis from the CUSTOMER_R, CUSTOMER_F, and CUSTOMER_M tables. It then calls three procedures for computing the R, F, and M scores. Next, it displays the numbers of customers with each R, F, and M score. Finally, it stores the R, F, and M scores in the CUSTOMER_RFM table. Only this table is needed for reporting purposes.

The Calculate_R stored procedure shown in Figure J-9 illustrates how the R score is calculated. This procedure first places the date of each customer's most recent order into the MostRecentOrderDate column. Then, it uses the **SQL TOP . . . PERCENT expression** in a series of SQL SELECT statements to set the R_Score values. The first UPDATE statement sets the value of R_Score to 1 for the top 20 percent of customers (after they have been sorted in descending order according to MostRecentOrderDate). Then, it sets the R_Score to 2 for the top 25 percent of customers who have a null value for R_Score in descending order of MostRecentOrderDate. The procedure continues to set the R values for all customers. The Calculate_F and Calculate_M procedures are similar and will be left to you as exercise J.40.

Figure J-10 shows how the CUSTOMER_RFM table is used for reporting purposes. Figure J-9 shows a SELECT on CUSTOMER_RFM that was prepared using a data extract of 5061 records from a database of more than 5,000 customers and over 1 million transactions. The records were imported from an Excel worksheet into a table named RFM_DATA_2014$, and that data was used to populate the CUSTOMER_SALES table, which is the basis for the data in the CUSTOMER_RFM table. The preparation of the CUSTOMER_RFM table required less than 30 seconds on a moderately powered personal computer.

The results in Figure J-10 are interesting, but unless this information is delivered to the correct users it will be of no ultimate value to the organization. For example, the 21st row in this figure shows

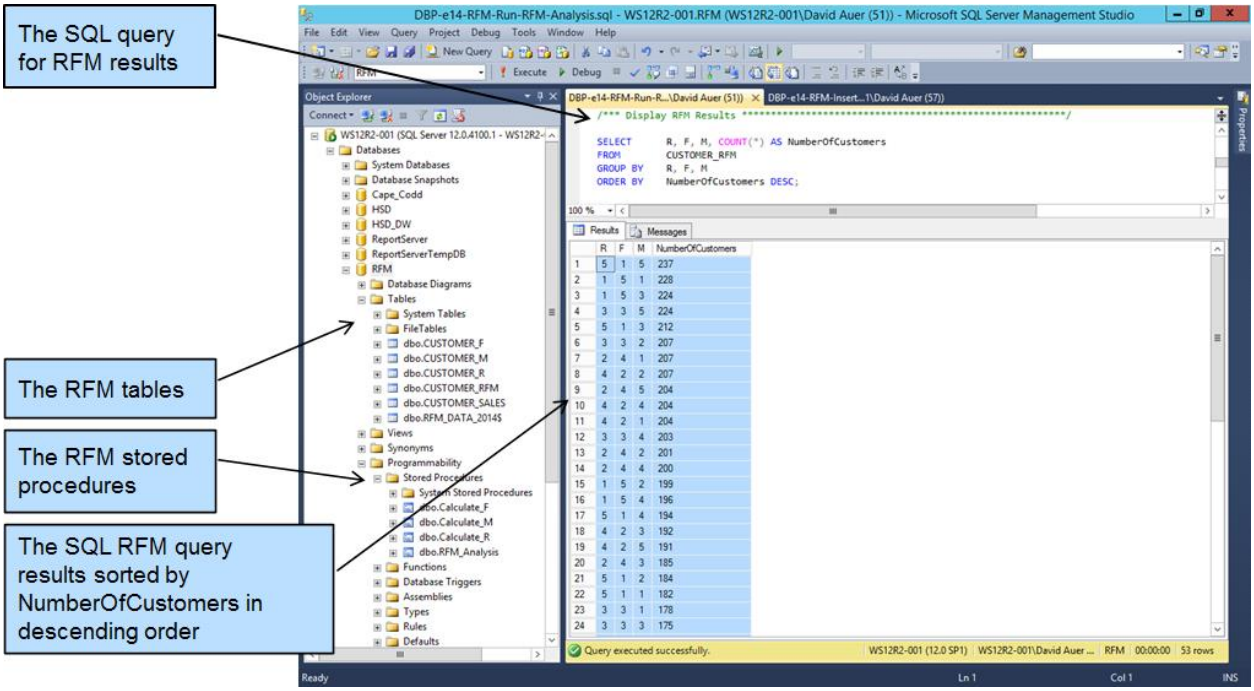


Figure J-10 — Example RFM Results

that 186 customers have an RFM score of {5 1 2}. These customers order frequently, they order items of relatively high monetary value, but they have not ordered recently. The company may be in danger of losing them. Somehow this report and the customers who have these scores (see Review Question J.21) need to be made available to the appropriate sales personnel. To understand the modern means for accomplishing this, we will consider the components of a reporting system.

Reporting System Components

Figure J-11 shows the major components of a reporting system. Data from disparate data sources are read and processed. As shown, reporting systems can obtain data from operational databases, data warehouses, and data marts.

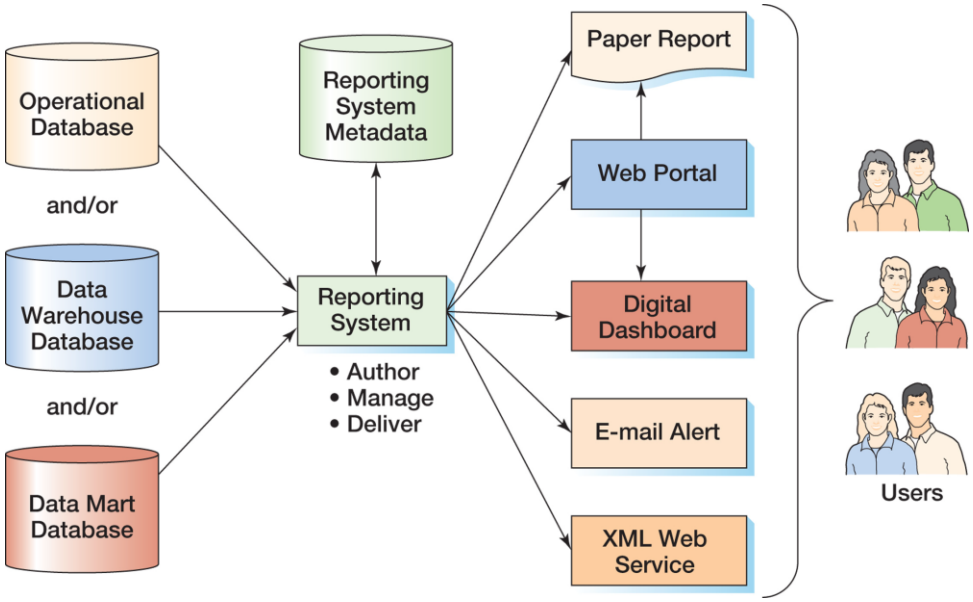


Figure J-11 — Components of a Reporting System

Type	Media	Mode
Static	Paper	Push
Dynamic	Web portal	Pull
Query	Digital dashboard	
OnLine Analytical Processing (OLAP)	E-mail/alert	
	XML Web service and application specific	

Figure J-12 — Report Characteristics

A reporting system maintains a database of reporting metadata. The metadata describes reports, users, groups, roles, events, and other entities involved in the reporting activity. The reporting system uses the metadata to prepare and deliver appropriate reports to the proper users in the correct format on a timely basis. As shown in Figure J-11, reports can be prepared in a variety of media or formats. Figure J-12 lists report characteristics.

Report Types

Some reports are **static reports**. They are prepared once from the underlying data, and they do not change. A report of the past year's sales, for example, is a static report. Other reports are **dynamic reports**—at the time of their creation, the reporting system reads the latest, most current data and generates the report using those fresh data. Reports on today's sales and on current stock prices are dynamic reports.

Query reports are prepared in response to information entered by users. Google is an example of a reporting system that uses query reports: You enter the keywords you want to search for, and Google's reporting system searches its database and generates a response that is particular to your query. Within a specific organization, such as Heather Sweeney Designs, a query report could be generated to show current inventory levels. The user would enter item numbers, and the reporting system would respond with inventory levels of those items.

In terms of the reporting system, **OLAP reports** enable the user to dynamically change the report grouping structures. We discuss OLAP in detail in Chapter 12.

Report Media

As illustrated in Figure J-11 and summarized in Figure J-12, reports are delivered via many different channels. Some reports are **printed** on paper, or its electronic equivalents, such as in PDF format. Other reports are delivered via **Web portals**. An organization might place a sales report on the sales department's Web portal and a report on customers serviced on the customer service department's Web portal.

A **digital dashboard** is an electronic display that is customized for a particular user. Companies such as Google, MSN, and Yahoo! offer digital dashboard services that you might have seen or used. Users can define the content they want to see—say, a local weather forecast, a list of stock prices, and a list of news sources—and the vendor constructs a customized display for each user. Such pages are called, for example, myhomemsn.com, and My Yahoo!. Other dashboards are designed specifically for organizations. Executives at a manufacturing organization, for example, might have a dashboard that shows up-to-the-minute production and sales activities.

Reports can also be delivered via **alerts**. Users can indicate that they want to be notified of news and events by email or cell phone. (Many cell phones are capable of displaying Web pages and can use digital dashboards.)

Finally, reports can be delivered to other information systems. The modern way to do this is to publish reports via **XML Web Services**, as discussed in Chapter 11. This style of reporting is particularly useful for inter-organizational information systems, such as supply chain management.

Report Modes

The final report characteristic summarized in Figure J-12 is report mode. A **push report** is sent to users based on a predetermined schedule. Users receive the report without any activity on their part. In contrast, users must request a **pull report**. To obtain a pull report, a user goes to a Web portal or digital dashboard and clicks a link or button to cause the reporting system to produce and deliver the report.

Report System Functions

As shown in Figure J-11, report systems serve three functions: report authoring, report management, and report delivery.

Report Authoring

Report authoring involves connecting to the required data sources, creating the report structure, and formatting the report. Reports created using a report authoring system are then assigned to groups and users. The report assignment metadata not only includes the user or group and the reports assigned but also indicates the format of the report that should be sent to the user, the channel by which the report will be delivered, and whether the report is to be pushed or pulled. If it is to be pushed, the administrator declares whether the report is to be generated on a regular schedule or as an alert.

Figures J-13 and J-14 show the use of **SQL Server Data Tools Business Intelligence (SSDT-BI)** to author a report that publishes the results of an RFM analysis done in SQL Server.

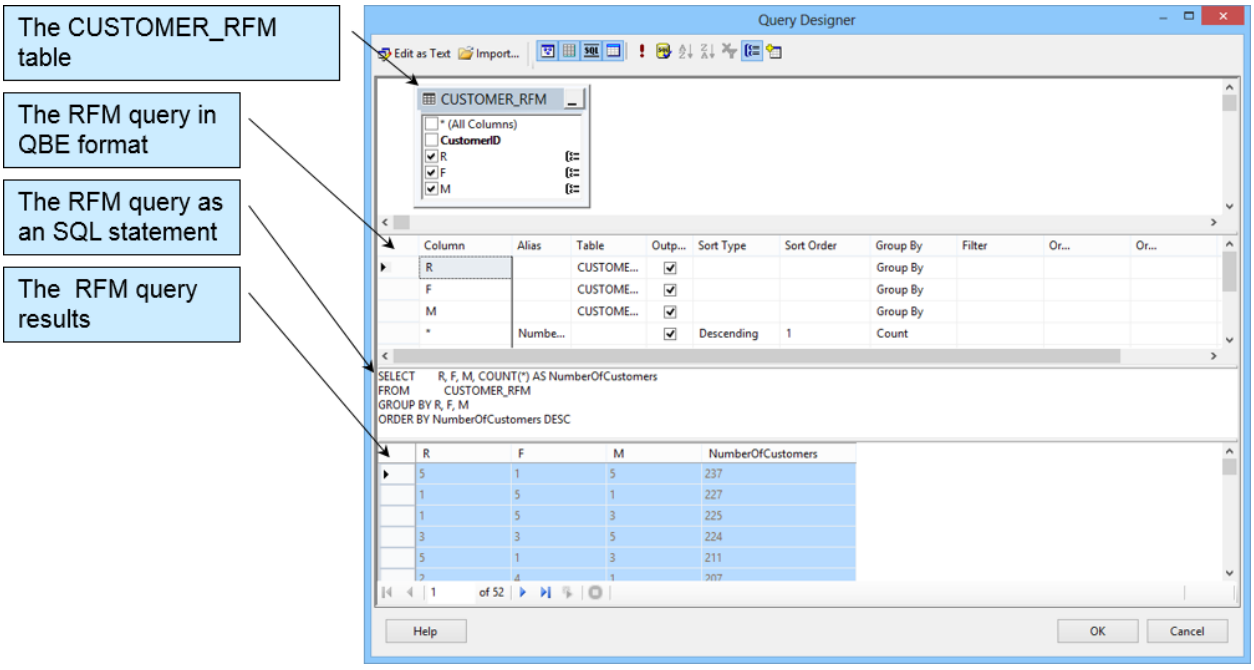


Figure J-13 — Setting Up a Report Data Source using SSDT-BI

SQL Server Data Tools Business Intelligence (SSDT-BI) is a tool available for Microsoft Visual Studio 2013 that is downloadable from https://msdn.microsoft.com/en-us/data/hh297027 .

In Figure J-13, the developer has specified a database that contains the CUSTOMER_RFM table and has just entered the SQL statement shown in Figure J-10. You can see the SQL statement in the lower-center portion of this display.

In Figure J-14, the report author creates the format of the report by specifying the headings and selecting the format for the data items. In a more complicated report, the author would specify the sorting and grouping of data items, as well as page headers and footers. The developer uses the property list in the right-hand side of the display in Figure J-14 to set the values for item properties. The final report, as it appears in a browser window, is shown in Figure J-15. To learn more about this application, search for "reporting services" at *www.microsoft.com*.

Report Management

Report management consists of defining who receives what reports, when, and by what means. Most report management systems enable the report system administrator to define user accounts and user groups and to assign particular users to particular groups. For example, all the salespeople would be assigned to the Sales group, all upper-level management would be assigned to the Executive group, and so forth. All these objects and assignments are stored in the reporting system metadata shown in Figure J-11.

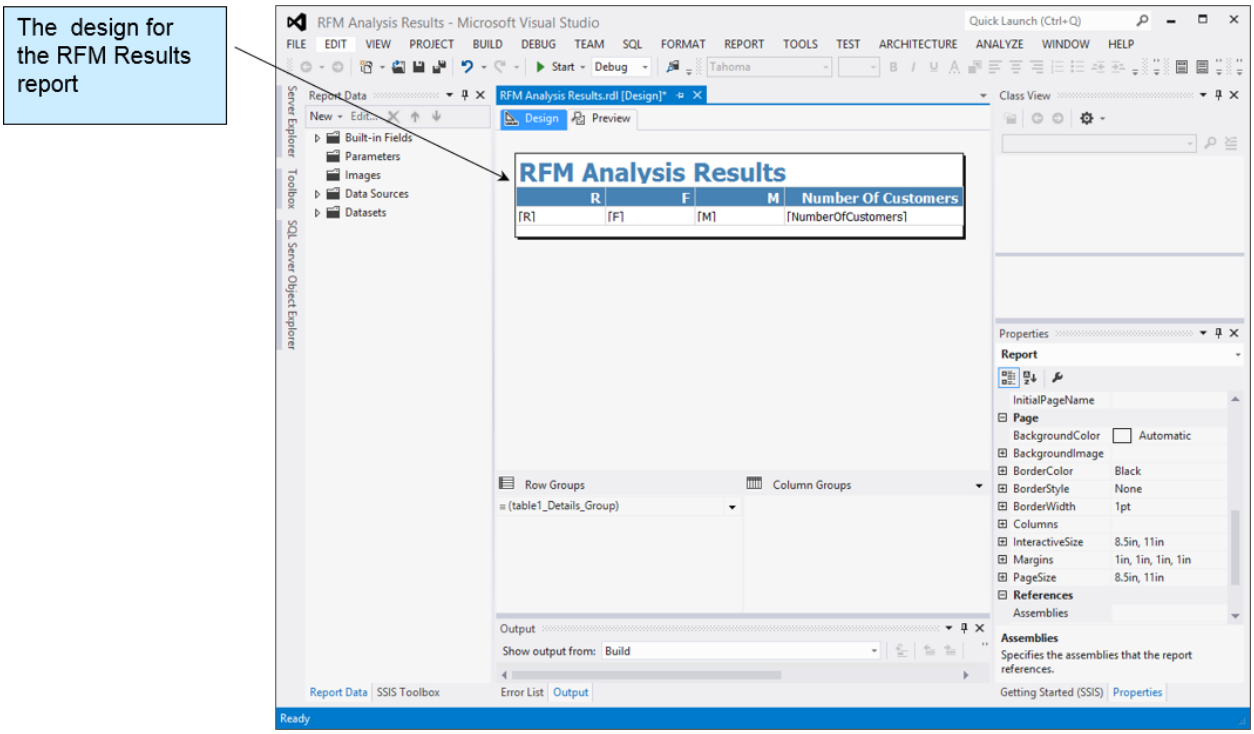
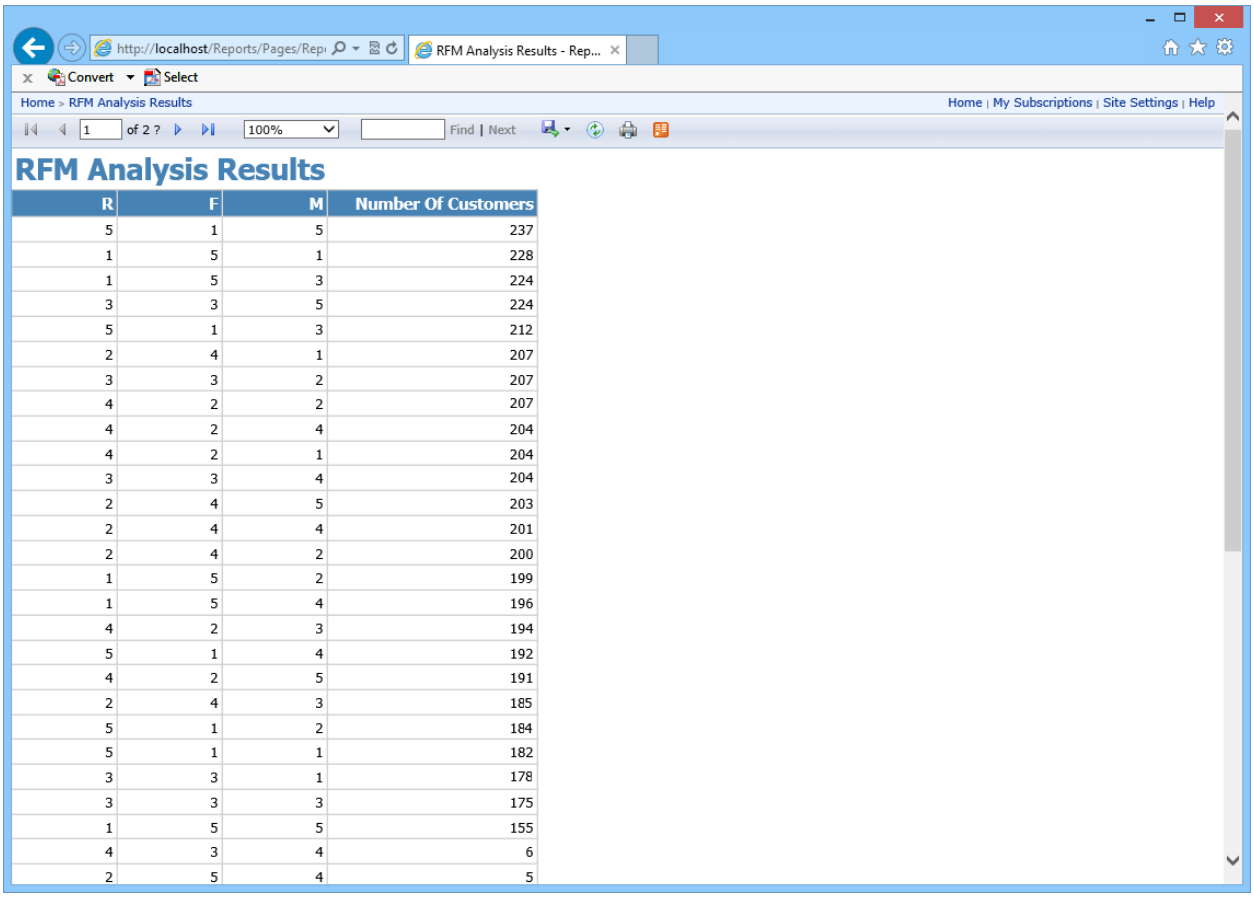


Figure J-14 — Formatting a Report Using SSDT-BI



RFM Analysis Results

R	F	M	Number Of Customers
5	1	5	237
1	5	1	228
1	5	3	224
3	3	5	224
5	1	3	212
2	4	1	207
3	3	2	207
4	2	2	207
4	2	4	204
4	2	1	204
3	3	4	204
2	4	5	203
2	4	4	201
2	4	2	200
1	5	2	199
1	5	4	196
4	2	3	194
5	1	4	192
4	2	5	191
2	4	3	185
5	1	2	184
5	1	1	182
3	3	1	178
3	3	3	175
1	5	5	155
4	3	4	6
2	5	4	5

Figure J-15 — The RFM Report in a Web Browser

Reports created using the report authoring system are assigned to groups and users. Assigning reports to groups saves the administrator work; when a report is created, changed, or removed, the administrator need only change the report assignments to the group. All of the users in the group will inherit the changes. The report assignment metadata includes not only the user or group and the reports assigned, but also indicates the format of the report that should be sent to that user and the channel by which the report will be delivered. For example, Figure J-16 shows part of the RFM report materialized in XML. The XML file can then be input into any program that consumes XML and manipulated via XSL, as described in Chapter 11.

As stated earlier, the report management metadata indicates which format of the report should be sent to which user. It also indicates what channel is to be used and whether the report is to be pushed or pulled. If pushed, the administrator declares whether the report is to be generated on a regular schedule or as an alert based on some event in the database.

```
<?xml version="1.0" encoding="UTF-8"?>
<MyData xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:noNamespaceSchemaLocation="C:\inetpub\wwwroot\DBP\RFM\DBP-e14-Figure-J-16.xsd">
    <CUSTOMER_RFM>
        <R>5</R>
        <F>1</F>
        <M>5</M>
        <NumberOfCustomers>237</NumberOfCustomers>
    </CUSTOMER_RFM>
    <CUSTOMER_RFM>
        <R>1</R>
        <F>5</F>
        <M>1</M>
        <NumberOfCustomers>228</NumberOfCustomers>
    </CUSTOMER_RFM>
    <CUSTOMER_RFM>
        <R>1</R>
        <F>5</F>
        <M>3</M>
        <NumberOfCustomers>224</NumberOfCustomers>
    </CUSTOMER_RFM>
    <CUSTOMER_RFM>
        <R>3</R>
        <F>3</F>
        <M>5</M>
        <NumberOfCustomers>224</NumberOfCustomers>
    </CUSTOMER_RFM>
    <CUSTOMER_RFM>
        <R>5</R>
        <F>1</F>
        <M>3</M>
        <NumberOfCustomers>212</NumberOfCustomers>
    </CUSTOMER_RFM>
    <CUSTOMER_RFM>
        <R>2</R>
        <F>4</F>
        <M>1</M>
        <NumberOfCustomers>207</NumberOfCustomers>
    </CUSTOMER_RFM>
    <CUSTOMER_RFM>
        <R>3</R>
        <F>3</F>
        <M>2</M>
        <NumberOfCustomers>207</NumberOfCustomers>
    </CUSTOMER_RFM>
</MyData>
```

Figure J-16 — A Portion of the RFM Report in XML Format

Report Delivery

The **report delivery** function of a reporting system pushes reports or allows them to be pulled based on the report management metadata. Reports can be delivered by hand, via an email server, a Web portal, XML Web Services, or by other program-specific means. The report delivery system uses the operating system and other program security components to ensure that only authorized users receive authorized reports, and it also ensures that push reports are produced at appropriate times.

For query reports, the report delivery system serves as an intermediary between the user and the report generator. It receives a user query request, such as the item numbers in an inventory query, passes the query request to the report generator, receives the resulting report, and delivers the report to the user.

Data Mining

Instead of the basic calculations, filtering, sorting, and grouping used in reporting applications, data mining involves the application of sophisticated mathematical and statistical techniques to find patterns and relationships that can be used to classify data and predict future outcomes. As shown in Figure J-17, data mining represents the convergence of several phenomena. Data mining techniques have emerged from the statistical and mathematics disciplines and from the artificial intelligence and machine-learning communities. In fact, data mining terminology is an odd combination of terms used by these different disciplines.

Data mining techniques take advantage of developments for processing enormous databases that have emerged in the past dozen or so years. Of course, all these data would not have been generated were it not for fast and inexpensive computers, and, without such computers, the new techniques would be impossible to compute.

Most data mining techniques are sophisticated and difficult to use. However, such techniques are valuable to organizations, and some business professionals, especially those in finance and marketing, have developed expertise in their use. Almost all data mining techniques require specialized software. Popular data mining products are Enterprise Miner from SAS Corporation, Clementine from SPSS, and Insightful Miner from Insightful Corporation. However, there is a movement to make data mining

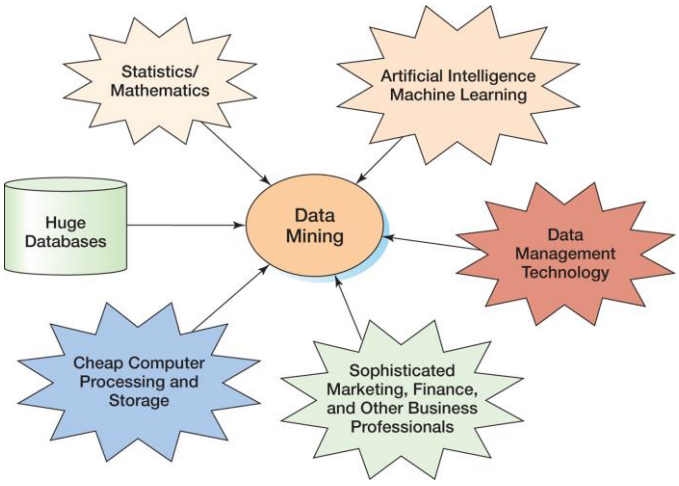


Figure J-17 — Convergence of Disciplines for Data Mining

available to more users. For example, Microsoft has created the Microsoft SQL Server 2012 SP1 Data Mining Add-ins for Microsoft Office.[1] Figure J-18 shows Microsoft Excel 2013 with the Data Mining command tab and command groups. With this add-in, data stored in Microsoft Excel are sent to SQL Server Analysis Services for processing, and the results are returned to Microsoft Excel for display. Oracle also offers data mining functionality via the "Oracle Advanced Analytics" option, with a GUI interface as part of SQL Developer. Data mining techniques fall into two broad categories: unsupervised and supervised.

Unsupervised Data Mining

When using **unsupervised data mining** techniques, analysts do not create a model or hypothesis prior to beginning the analysis. Instead, the data mining technique is applied to the data, and results are observed. After the analysis, explanations and hypotheses are created to explain the patterns found.

One commonly used unsupervised technique is **cluster analysis**. With cluster analysis, statistical techniques are used to identify groups of entities that have similar characteristics. A common use for cluster analysis is to find customer groups in order and customer demographic data. For example, Heather Sweeney Designs could use cluster analysis to determine which groups of customers are associated with the purchase of specific products.

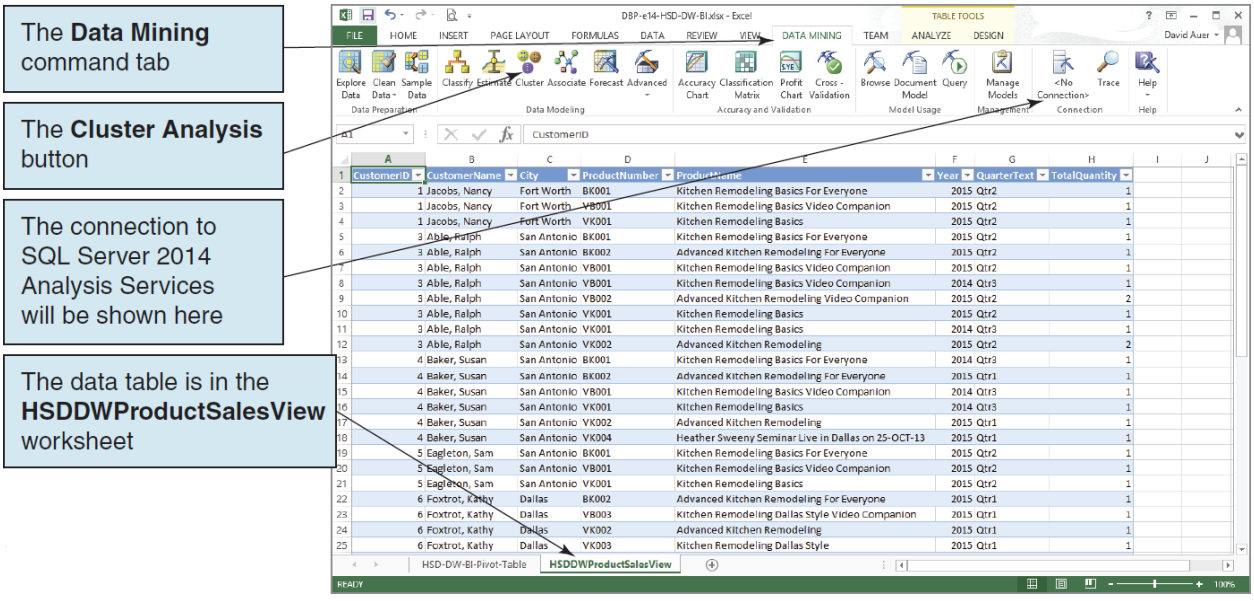


Figure J-18 — Microsoft Excel 2013 with the Microsoft SQL Server 2012 SP1 Data Mining Add-ins

[1] The Microsoft SQL Server 2012 SP1 Data Mining Add-ins for Office are available at http://www.microsoft.com/en-us/download/details.aspx?id=35578 . They work with both Microsoft Office 2010 and Microsoft Office 2013. Note, however, that these add-ins will not work with SQL Server Express Edition. You have to have a version of SQL Server with SQL Server Analysis Services.

Supervised Data Mining

When using **supervised data mining** techniques, data miners develop a model prior to the analysis and then apply statistical techniques to the data to estimate parameters of the model. For example, suppose that marketing experts at a communications company believe that the use of cell phone weekend minutes is determined by the age of the customer and the number of months the customer has had the cell phone account. A data mining analyst would then run a statistical analysis technique known as **regression analysis** to determine the coefficients of the equation of that model. A possible result is:

CellPhoneWeekendMinutes = 12 + (17.5 * CustomerAge) + (23.7 * NumberMonthsOfAccount)

As you will learn in your statistics classes, considerable skill is required to interpret the quality of such a model. The regression tool will create an equation; whether the equation is a good predictor of future cell phone usage depends on *t* values, confidence intervals, and related statistical techniques.

Three Popular Data Mining Techniques

Three popular data mining techniques are decision tree analysis, logistic regression, and neural networks. **Decision tree analysis** classifies customers or other entities of interest into two or more groups, according to history. **Logistic regression** produces equations that offer probabilities that particular events will occur. Common applications of logistic regression are using donor characteristics to predict the likelihood of a donation in a given period and using customer characteristics to predict the likelihood that customers will switch to another vendor. **Neural networks** are complex statistical prediction techniques. The name is actually a misnomer—although there is some loose similarity between the structure of a neural network and a network of biological neurons, the similarity is only superficial. In data mining, neural networks are just a technique for creating very complex mathematical functions for making predictions.

These three techniques, like almost all data mining techniques, require specialized software. Popular data mining products are Enterprise Miner from the SAS Corporation, Clementine from SPSS, and Insightful Miner from the Insightful Corporation. All of these products have facilities for importing data from relational databases, and, as a database professional, you may be asked to prepare data for input to a data mining product. Typically, this work involves joining relations together into a large flat file and then filtering the data for particular data cases. Simple SQL is used to create such files. In addition, as mentioned previously, many enterprise-class DBMS products, such as Microsoft SQL Server and Oracle Database, now include this functionality within the DBMS.

Market Basket Analysis

Data mining techniques are usually complex. However, **market basket analysis** is a data mining technique that can be readily implemented with pure SQL. All the major data mining products have features and functions to perform market basket analysis. Market basket analysis is also known as **association rules**.

Suppose that you run a diving shop, and one day you realize that one of your salespeople is much better than others at up-selling your customers. Any of your sales associates can fill a customer's order, but

this particular salesperson is especially able to sell customers items in addition to those for which they ask. One day you ask him how he does it.

"It's simple," he says. "I just ask myself 'What is the next product they'll want to buy?' If someone buys a dive computer, I don't try to sell her fins. If she's buying a dive computer, she's already a diver, and she already has fins. But, look, these dive computer displays are hard to read. A better mask makes it easier to read the display and get the full benefits from the dive computer."

Market basket analysis is a data mining technique for determining such patterns. A market basket analysis shows the products that customers tend to purchase at the same time. Several different statistical techniques can be used to generate a market basket analysis. Here we discuss a technique that involves conditional probabilities.

Figure J-19 shows hypothetical data from 1,000 transactions at a dive shop. The first row of numbers under each column is the total number of transactions that include the product in that column. For example, the 270 in the first row of Mask means that 270 of the 1,000 transactions include the purchase of a mask. The 120 under Dive Computer means that 120 of the 1,000 purchase transactions included a dive computer.

Note that in this example, every transaction involves 1 or 2 items; those transactions with 2 items will be counted in two columns of the first row. Also note that some of the 1000 transactions do not contain any of the five products listed in the table (e.g., somebody purchases a wet suit and nothing else).

You can use the numbers in the first row to estimate the probability that a customer will purchase an item. Because 270 out of 1,000 transactions included a mask, you can estimate the likelihood that a customer will buy a mask to be 270/1,000, or .27. Similarly, the likelihood of a tank purchase is 200/1,000, or .2, and the likelihood of a fins purchase is 280/1,000, or .28.The remaining rows in this table show the occurrences of transactions that involve two items. For example, the last column indicates that 50 transactions included both a dive computer and a mask, 30 transactions included a dive computer and a tank, 20 included a dive computer and fins, 10 included a dive computer and weights, 5 included a dive computer and another dive computer (meaning the customer bought two dive computers), and 5 transactions had a dive computer and no other product.

These data are interesting, but you can refine the analysis by computing additional factors. Marketing professionals define **support** as the probability that two items will be purchased together. From these data, the support for fins and mask is 150 out of 1,000, or .15.

Confidence is defined as the probability of a customer buying one product, given that he or she purchased another product. The confidence of fins, given that the customer has already purchased a mask, is the number of purchases of fins and masks out of the number of purchases of masks. Thus, in this example, the confidence is 150 out of 270, or .55556. The confidence that a customer purchases a tank, given that the customer has purchased fins, is 40 out of 280, or .14286.

Lift is defined as the ratio of confidence divided by the base probability of an item purchase. The lift for fins, given a mask, is the probability that a customer buys fins, given that the customer has purchased a mask, divided by the overall probability that the customer buys fins. If the lift is greater than 1, then the

1,000 Transactions	Mask	Tank	Fins	Weights	Dive Computer
	270	200	280	130	120
Mask	20	20	150	20	50
Tank	20	80	40	30	30
Fins	150	40	10	60	20
Weights	20	30	60	10	10
Dive Computer	50	30	20	10	5
No Additional Product	10	–	–	–	5

Support = P (A & B) Example: P (Fins & Mask) = 150 / 1000 = .15
Confidence = P (A | B) Example: P (Fins | Mask) = 150 / 270 = .55556
Lift = P (A | B)/ P (A) Example: P (Fins | Mask) / P (Fins) = .55556 / .28 = 1.98
Note: P (Mask | Fins) / P (Mask) = 150 / 280 / .27 = 1.98

Figure J-19 — A Market Basket Analysis Example

probability of buying fins goes up when a customer buys a mask; if the lift is less than 1, the probability of buying fins goes down when a customer buys a mask.

For the data in Figure J-19, the lift for fins, given a mask purchase, is .55556/.28 or 1.98. This means that when someone purchases a mask, the likelihood he or she will also purchase fins almost doubles. The lift for fins, given a dive computer purchase, is 20/120 (the confidence of fins, given a dive computer) divided by .28, the probability that someone buys fins (280 of the 1,000 transactions involved fins). Therefore, 20/120 is .16667, and .16667/.28 is .59525. So the lift for fins, given purchase of a dive computer, is just under .6, meaning that when a customer buys a dive computer the likelihood that he or she will buy fins decreases.

Note that, as shown in the last line of Figure J-19, lift is symmetrical. If the lift of fins, given purchase of a mask, is 1.98, then the lift of mask, given purchase of fins, is also 1.98.

Summary

Business intelligence (BI) systems assist managers and other professionals in the analysis of current and past activities and in the prediction of future events. BI applications are of two major types: reporting applications and data mining applications. Reporting applications make elementary calculations on data; data mining applications use sophisticated mathematical and statistical techniques.

BI applications obtain data from three sources: operational databases, extracts of operational databases, and purchased data. A BI system sometimes has its own DBMS, which may or not be the operational DBMS.

Direct reading of operational databases is not feasible for any but the smallest and simplest BI applications and databases—for several reasons. Querying operational data can unacceptably slow the performance of operational systems, operational data have problems that limit their usefulness for BI applications, and BI system creation and maintenance require programs, facilities, and expertise that are normally not available for an operational database.

Operational data may have problems. Because of the problems with operational data, many organizations have chosen to create and staff data warehouses and data marts. Extract, Transform, and Load (ETL) systems are used to extract data from operational systems; transform the data and load them into data warehouses; and maintain metadata that describes the source, format, assumptions, and constraints about the data. A data mart is a collection of data that is smaller than that held in a data warehouse and that addresses a particular component or functional area of the business. In Figure J-3, the enterprise data warehouse distributes data to three smaller data marts, each of which services the needs of a different aspect of the business.

The purpose of a reporting system is to create meaningful information from disparate data sources and to deliver that information to the proper users on a timely basis. Reports are produced by sorting, filtering, grouping, and making simple calculations on the data. Online Analytical Processing (OLAP) and RFM analysis are typical reporting applications.

Online Analytical Processing (OLAP) reporting applications, discussed in detail in Chapter 12, enable users to dynamically restructure reports.

In RFM analysis, customers are grouped and classified according to how recently they have placed an order (R), how frequently they order (F), and how much money (M) they spend on orders. The result of an RFM analysis is three scores. In a typical analysis, the scores range from 1 to 5. An RFM score of 1 1 4 indicates that the customer has purchased recently, purchases frequently, and does not purchase expensive items. An RFM report can be produced using SQL statements.

For RFM data to add value to an organization, an RFM report must be prepared and delivered to the appropriate users. The components of a modern reporting system are shown in Figure J-11. Reporting systems maintain metadata that supports the three basic report functions: authoring, managing, and delivering reports. The metadata includes information about users, user groups, and reports and data about which users are to receive which reports, in what medium, and when. As shown in Figure J-12, reports vary by type, media, and mode.

Data mining is the application of mathematical and statistical techniques to find patterns and relationships and to classify and predict. Data mining has arisen in recent years because of the confluence of factors shown in Figure J-156

With unsupervised data mining, analysts do not create models or hypotheses prior to the analysis. Rather, results are explained after the analysis is performed. With supervised techniques, hypotheses

are formed and tested before the analysis. Five popular data mining techniques are cluster analysis, regression analysis, decision tree analysis, logistic regression, and neural networks.

Market basket analysis is an unsupervised data mining technique used to determine which sets of products are likely to be sold at the same time. According to market basket analysis terminology, the support for two products is the frequency with which they appear together in transactions. Confidence is the conditional probability that one item will be purchased, given that another item has already been purchased. Lift is confidence divided by the base probability that an item will be purchased.

Key Terms

alert	**association rules**
business intelligence (BI) system	**click-stream data**
cluster analysis	**confidence**
data mart	**data mining application**
data warehouse	**data warehouse metadata database**
decision tree analysis	**digital dashboard**
dimension table	**dimensional database**
dirty data	**dynamic report**
enterprise data warehouse (EDW) architecture	**Extract, Transform, and Load (ETL) System**
F score	**fact table**
lift	**logistic regression**
M score	**market basket analysis**
neural network	**OLAP cube**
OLAP report	**Online Analytical Processing (OLAP)**
pull report	**push report**
query report	**R score**
regression analysis	**report authoring**

report delivery	**report management**
reporting system	**RFM analysis**
SQL TOP . . . PERCENT property	**static report**
supervised data mining	**support**
unsupervised data mining	**Web portal**
XML Web Services	

Review Questions

J.1 What are BI systems?

J.2 How do BI systems differ from transaction processing systems?

J.3 Name and describe the two main categories of BI systems.

J.4 What are the three sources of data for BI systems?

J.5 Summarize the problems with operational databases that limit their usefulness for BI applications.

J.6 What is an ETL system, and what functions does it perform?

J.7 What problems in operational data create the need to clean data before loading the data into a data warehouse?

J.8 What does it mean to transform data? Give an example other than the ones used in this book.

J.9 Why are data warehouses necessary?

J.10 Give examples of data warehouse metadata.

J.11 Explain the difference between a data warehouse and a data mart. Give an example other than the ones used in this book.

J.12 What is the enterprise data warehouse (EDW) architecture?

J.13 State the purpose of a reporting system.

J.14 In RFM analysis, what do the letters *RFM* stand for?

J.15 Describe, in general terms, how to perform an RFM analysis.

J.16 Explain the characteristics of customers that have the following RFM scores:

{1 1 5}, {1 5 1}, {5 5 5}, {2 5 5}, {5 1 2}, {1 1 3}

J.17 In the RFM analysis in Figures J-7 through J-10, what role does the CUSTOMER_RFM table serve? What role does the CUSTOMER_R table serve?

J.18 Explain the purpose of the following SQL statement from Figure J-9:

```
INSERT INTO CUSTOMER_R (CustomerID, MostRecentOrderDate)
      (SELECT     CustomerID, MAX (TransactionDate)
       FROM       CUSTOMER_SALES
       GROUP BY   CustomerID);
```

J.19 Explain the purpose and operation of the following SQL statement from Figure J-9:

```
UPDATE   CUSTOMER_R
      SET      R_Score = 1
      WHERE    CustomerID IN
                  (SELECT     TOP 20 PERCENT CustomerID
                   FROM       CUSTOMER_R
                   ORDER BY   MostRecentOrderDate DESC);
```

J.20 Explain the purpose and operation of the following SQL statement from Figure J-9:

```
UPDATE   CUSTOMER_R
      SET      R_Score = 2
      WHERE    CustomerID IN
                  (SELECT     TOP 25 PERCENT CustomerID
                   FROM       CUSTOMER_R
                   WHERE      R_Score IS NULL
                   ORDER BY   MostRecentOrderDate DESC);
```

J.21 Write an SQL statement to query the CUSTOMER_RFM table and display the CustomerID values for all customers having an RFM score of {5 1 1} or {4 1 1}. Why are these customers important?

J.22 Name and describe the purpose of the major components of a reporting system.

J.23 What are the major functions of a reporting system?

J.24 Summarize the types of reports described in this chapter.

J.25 Describe the various media used to deliver reports.

J.26 Summarize the modes of reports described in this chapter.

J.27 Describe the major tasks in report management. Explain the role of report metadata in report management.

J.28 Name three tasks of report authoring.

J.29 Describe the major tasks in report delivery.

J.30 What does OLAP stand for?

J.31 Define *data mining*.

J.32 Explain the difference between unsupervised and supervised data mining.

J.33 Name five popular data mining techniques.

Use the data in Figure J-19 to answer questions J.34 through J.39.

J.34 What is the probability that someone will buy a tank?

J.35 What is the support for buying a tank and fins? What is the support for buying two tanks?

J.36 What is the confidence for fins, given that a tank has been purchased?

J.37 What is the confidence for a second tank, given that a tank has been purchased?

J.38 What is the lift for fins, given that a tank has been purchased?

J.39 What is the lift for a second tank, given that a tank has been purchased?

Project Questions

J.40 Using the code in Figure J-9 as an example, write the procedures Calculate_F and Calculate_M that are called from the Calculate_RFM stored procedure in Figure J-8.

Case Questions

Marcia's Dry Cleaning

Assume that Marcia uses a database that includes the following tables:

CUSTOMER (CustomerID, FirstName, LastName, Phone, Email)

INVOICE (InvoiceNumber, *CustomerID*, DateIn, DateOut, Subtotal, Tax, TotalAmount)

INVOICE_ITEM (*InvoiceNumber*, ItemNumber, *ServiceID*, Quantity, UnitPrice, ExtendedPrice)

SERVICE (ServiceID, ServiceDescription, UnitPrice)

(The SERVICE table, included above for completeness, is not needed for these exercises.)

A. Describe how an RFM analysis could be useful in Marcia's business.

B. Using five tables based on the tables in Figure J-7, write a set of stored procedures to compute an RFM analysis on Marcia's data.

C. Show SQL to process the table generated in your answer to B to display the names and e-mail data for all customers having an RFM score of {5 1 1} or {4 1 1}.

D. Describe, in general terms, how a market basket analysis can be used on the items in a dry cleaning order.

The Queen Anne Curiosity Shop

Suppose that you have designed a database for The Queen Anne Curiosity Shop that has the following tables:

CUSTOMER (CustomerID, LastName, FirstName, Address, City, State, ZIP, Phone, Email)

EMPLOYEE (EmployeeID, LastName, FirstName, Phone, Email)

VENDOR (VendorID, CompanyName, ContactLastName, ContactFirstName, Address, City, State, ZIP, Phone, Fax, Email)

ITEM (ItemID, ItemDescription, PurchaseDate, ItemCost, ItemPrice, *VendorID*)

SALE (SaleID, *CustomerID*, *EmployeeID*, SaleDate, SubTotal, Tax, Total)

SALE_ITEM (*SaleID*, SaleItemID, *ItemID*, ItemPrice)

A. Describe how an RFM analysis could be useful to The Queen Anne Curiosity Shop.

B. Using five tables based on the tables in Figure J-7, write a set of stored procedures to compute an RFM analysis on data from The Queen Anne Curiosity Shop.

C. Show SQL to process the table generated in your answer to B to display the names and e-mail data for all customers having an RFM score of {5 1 1} or {4 1 1}.

D. Describe, in general terms, how a market basket analysis can be useful to The Queen Anne Curiosity Shop.

Morgan Importing

Note that since the Morgan Importing database that we have created is intended to track purchases and shipping rather than customer purchases, neither RFM nor market basket analysis can be applied to customer data.

A. Describe how an RFM analysis could be useful to Morgan Importing. What part of the business would you apply RFM analysis to?

B. Describe in general terms how a market basket analysis could be useful to Morgan Importing. What part of the business would you apply market basket analysis to?