CHAPTER 3

TECHNOLOGIES ENABLING ORGANIZATIONAL MEMORY

3.1 INTRODUCTION

In Chapter 2 we described the capabilities associated with BI—namely, organizational memory, information integration, business analytics, and presentation. We will present the details about these capabilities in the next four chapters. In this chapter we describe the topic of *organizational memory*, which refers to the storage of information in such a form that it can be later accessed and used for BI. Thus this chapter focuses on the technologies that enable organizational memory by storing structured information that is used for BI.

Organizational memory (which is related to aspects such as corporate memory, knowledge repository, and institutional memory) represents the aggregate intellectual assets of an organization. These include data, information, and explicit knowledge which may be either structured or unstructured. Structured data or information could reside, for example, in database records or transactional systems. Structured data is said to be organized according to a data model (also data structure), which is how a computer program can easily use that data. A data model describes how the data is represented and accessed, by providing the definition and format of the data. For example, a database model is a type of data model used to describe how a database is structured and used. Structured explicit knowledge could reside in knowledge repositories and answers to frequently asked questions (FAQs). Unstructured data and information, on the other hand, do not have an associated data model or metadata. Unstructured data, information, and knowledge reside in various forms, including e-mail messages stored over time, audio and video files, historical versions of Web sites, as well as prior presentations, memos, and other documents.

BI focuses on how to create value by discovering knowledge from explicit organizational memory. In Chapter 4 we discuss how BI can drive value from information integration capability, including the integration of structured as well as unstructured data, through text and Web mining, whereas our discussion in Chapter 5 focuses on how BI enables the creation of new insights through the use of analytics and data mining.

Two relevant technologies for the organizational memory capability—enterprise resource planning (ERP) systems and data warehouses—are discussed and illustrated in this chapter. ERP systems refer to transactional systems that capture organizational memory related to all the business processes that the organization engages in. One of these business processes is for example, order-to-cash, which captures all the transactions involved in the organization from the moment the customer places an order to the moment the organization receives payment for the product or service (cash). Transaction systems typically capture all the relevant information for one accounting period, which is a month, quarter, or year depending on the organization. Because organizations typically want to see how these figures evolve over time, data warehouses capture these data over time, providing the source of data and information for the BI analysis.

In addition, the chapter includes a discussion of the topic of enterprise architecture (EA). Finally, we present a case study that describes the ERP implementation at the IBM Personal Systems Group, together with a summary of some of the critical success factors in ERP implementations.

We begin with a description of ERP systems.

▶ 3.2 ENTERPRISE RESOURCE PLANNING SYSTEMS

This section describes **enterprise resource planning systems (ERP)** or, for short, enterprise systems. ERP systems are "software packages composed of several modules, such as human resources, sales, finance, and production, providing cross-organization integration of data through embedded business processes" (Esteves and Pastor, 2001, 2007). ERP systems originate from **manufacturing resource planning or material requirements planning (MRP) systems**. MRP systems were developed to control the aspects around manufacturing, including procurement of materials, production, delivery, and inventory control (Wailgum, 2007). ERP systems include software modules that support all aspects of the enterprise including finance, human resources, manufacturing, supply chain, and inventory. ERP systems are transactional, in the sense that they store data for the current accounting period, which is a month, quarter, or year depending on the organization. In addition, some ERP systems also offer customer relationship management, sales management, and an integrated data warehouse. In order to use the data in an ERP system to do trend analysis, the data will need to be stored in a data warehouse as described in the next section. A number of companies were propelled to implement ERP systems as a measure to bring their enterprise infrastructure to Y2K[1]

[1] Also known as the Year 2000 problem, Y2K problem, or the millennium bug.

compliance, as ERP systems would replace the legacy systems that faced the Y2K shortcoming.[2]

ERP systems offer organizations many benefits, including integrated business processes across the enterprise, a single database for the enterprise, access to real-time transaction data, and the elimination of costly legacy stand-alone systems plagued by maintenance complexities, including outdated spaghetti code used to artificially interconnect them. ERP systems provide the infrastructure to improve the way the organization manages the order fulfillment process, from the moment it takes a customer order, processes it, delivers it, and collects the associated revenue. The order fulfillment process effectively integrates the different departments involved in this process, from sales, to finance, to manufacturing, to the logistics department that delivers the product. The last few years have seen substantial consolidation on the vendors offering ERP solutions, and the top four vendors to date include: SAP,[3] Oracle,[4] Sage,[5] and Microsoft Dynamics.[6] There are also open source versions of ERP systems, including OpenERP,[7] GNU Enterprise,[8] and WebERP,[9] among others.

However, implementation of an ERP does not come without significant technical challenges, including large monetary investments and a great deal of organizational change. These technical challenges have resulted in a number of well-published failed ERP implementations. For example, operational problems at Hershey Foods, Whirlpool, FoxMeyer Drugs, and Hewlett Packard were all blamed on poorly implemented ERP solutions (Davenport, 1998; Songini, 2004; Wheatley, 2000).

ERP systems are typically configured to tailor the different software modules to a specific way an organization goes about its business. ERP systems are configured through the use of configuration tables, although the organization also has the option to customize the system by rewriting the software code or integrate the ERP system to other existing legacy systems. Typically users are encouraged to minimize the amount of code customization or integration to

[2] In the early days of computers, computer memory was scarce and expensive. Thus, saving the two digits associated with the date field was significant. Any calculations involving estimating time lapse would essentially work properly as long as the dates involved were before the year 2000 (e.g., 99 – 90 = 9 years). The problem essentially would arrive as the year 2000 rolled in, as time lapse would result in a negative number (00 – 90 = -90), which would crash the legacy systems involved. In order to prevent a system crash, legacy systems would need to be modified—for example, by expanding the date field to be four characters. But some organizations adopted a different approach, which involved replacing the legacy systems altogether—say, with ERP systems that would already be Y2K compliant.

[3] www.sap.com

[4] www.oracle.com

[5] http://www.sagenorthamerica.com

[6] http://www.microsoft.com/dynamics

[7] http://www.openerp.com/

[8] http://www.gnu.org/software/gnue/project/what.html

[9] http://www.weberp.org/

existing legacy systems because that increases the complexity of the ERP implementation.

Due to their scope and complexity, ERP implementations have been and will continue to be a challenge. Clearly, the well-published failures indicate that not all organizations have been up to this challenge. The factors that are critical for the success of ERP implementations have been broadly studied in a number of areas within IS research (Rockart, 1979) and specifically within the realm of ERP system implementation (Becerra-Fernandez, Murphy, and Elam, 2005; Markus, Axline, Petrie, and Tanis, 2000). A review of the literature on ERP projects identifies the number of critical factors as varying between 10 and 20 for any project. A careful analysis of this list reveals that many of these factors can be consolidated, leaving six unique, critical success factors for large-scale ERP project implementations:

1. *Top management commitment* includes: top management support, project champion, and business plan and vision.
2. *Strong project management* includes: project management, business process redesign, software development testing and troubleshooting, and monitoring and evaluation of performance.
3. *Team member skills* include: ERP teamwork and composition and appropriate business and IT legacy systems.
4. *Team member motivation and dedication*
5. *Effective communication with users* which refers to how the organization communicates with all those affected by the ERP system implementation, including the employees and customers.
6. *Effective change management* which refers to how the organization manages the changes associated with the new IS infrastructure, including how organizational business processes must change in order to adapt to the new methodologies imposed by the ERP system.

The case that we describe in Section 3.5 describes how all of these critical success factors were present at IBM's Personal Systems Group realization of one of the largest ERP implementations to date.

Some critics of ERP systems point out that in recent years, ERP project spending at organizations has absorbed the attention, budgets, and energy of information technology professionals worldwide. ERP projects represent the single largest investment in an IT project in the histories of these companies and, in many cases, the largest single investment in any corporate-wide project (Sumner, 2000). In addition, implementing ERP software, which aims to consolidate most of the disparate systems in an organization, requires a huge amount of business process and cultural change, resulting in a smooth implementation in only 10% to 15% of the cases (Rutherford, 2001). Actually, midmarket companies or those that wish to minimize their risks are left with no other alternative than to stick to "plain vanilla," meaning out-of-the-box, minimally customized and fast-track ERP implementations,

with their absolute lack of process customizability (Pender, 2001). Even in the best situations, ERP may not provide the level of integration expected, often requiring the integration of applications that were not, and will not be, replaced by the implemented systems (Pender, 2000). Furthermore, it is expected that critical knowledge-based systems—for example, actuarial repositories for insurance companies—will not be replaced by standard ERP systems, because the former represent organizational core competencies less likely to be substituted by standard ERP offerings.

ERP critics are also quick to point out that enterprise systems do not drive innovation. The more change in the organizational environment, the more the business requires business agility. The tie between ERP and BI is that a standardized integrated enterprise infrastructure environment creates better opportunities for the organization to be more agile and adopt innovation. ERP standardization focuses on commoditization of the business processes that are not the source of competitive advantage, so that the organization can focus its differentiation efforts on BI and innovation. By first building a standardized enterprise infrastructure, the organization can then focus its attention on using BI to respond with agility to environmental signals.

Another technology that can be used to create an enterprise system, without the use of ERP systems, is through **enterprise application integration (EAI)**. EAI is **middleware** (software that interconnects applications) that can parse, duplicate, or transform data from an application to present it in an acceptable format for each application that needs to receive the data (Slater, 2000). There are different types of systems integration, spanning from business architecture, to application architecture, to inter-organizational process integration (Hasselbring, 2000). EAI technology deals with data integration, sometimes from legacy systems, that facilitates repurposing of old systems into new ways of doing business. In contrast to ERP implementations, which require organizations to adapt their own business processes to ERP prescribed "best practices," EAI technology allows application integration without redefining business practices.

In Business Intelligence in Practice 3.1 we discuss the role of ERP systems in small and medium-size businesses (**SMBs**), some of the specific features they may provide for that market segment, and an example of an SMB success story.

We next discuss the role of data warehouses on preparing the enterprise data for BI.

▶ 3.3 DATA WAREHOUSE

The concept of the data warehouse (DW) was first described by IBM researchers Devlin and Murphy (Hayes, 2002). The practical application of the concept of DW is credited to Inmon (2005). A **data warehouse** can be defined as "a subject-oriented, integrated, nonvolatile, and time-variant collection of data in support of management's decisions" (Inmon, 2005) or "a copy of transaction

Business Intelligence in Practice 3.1: Is ERP Applicable for Small and Medium-Sized Businesses?

By now, most Fortune 500 companies have implemented some form of ERP system in order to support an integrated business strategy for their organization (Bingi et al., 1999, Davenport, 1998). But primarily due to the high costs and risks associated with their implementation, many small and medium-size businesses (SMBs) have avoided getting on the ERP bandwagon altogether. Since SMBs are a largely untapped marked segment for most ERP vendors, many of them have announced strategies specifically designed to boost sales to SMBs, including referral and incentive programs.

For example, as a strategy to become more attractive to the SMB sector, Microsoft has suited their Dynamics products with functionality specifically designed to appeal to this market segment. One such example is the Dynamics Environmental Sustainability Dashboard that will enable users to measure their energy consumption and greenhouse gas emissions at their organizations, in order to reduce their energy consumption or carbon footprint (Microsoft, 2009). The dashboard is intended to serve as a detailed guide to enable customers to collect the necessary data to create a greenhouse gas inventory, and setup the dashboard to depict accurate, auditable, and actionable environmental data.

An example of an SMB that has implemented SAP is Artisan Hardwood Floors. Artisan is an SMB that purchased SAP in order to support the operations of its 37 employees in the family-owned company. With no in-house IT staff, the implementation team at Artisan was just three people. Following a successful ERP implementation, Artisan is now able to have a much clearer financial and operational picture than it had prior to having SAP (Wailgum, 2008).

However, the jury is still out to determine whether ERP systems will be able to take a significant "byte" out of the SMB market.

data specifically structured for querying and reporting" (Kimball et al., 2008). A more comprehensive definition describes a data warehouse as "an environment—not a single technology—comprising a data store and multiple software products . . . includes tools for data extraction, loading, storage, access, query, and reporting . . . to support decision-oriented management queries" (Bashein and Markus, 2000, p. 3). The four characteristics of a DW environment are (Inmon, 2005):

1. Subject-oriented, which means that depending on the type of company, the major subject areas will be unique. For example, for an insurance company,

the subjects may be auto, life, health, and casualty; for a retailer the major subject areas may be SKU, sale, and vendor.

2. Integrated, so that data is fed into the DW from multiple, disparate sources, including the operational databases, data archives, legacy databases, and even external data. In the process of integrating the data into the DW, it gets converted, reformatted, resequenced, and summarized into a single physical corporate image. As the data is integrated into a single image, issues must be resolved in order to overcome inconsistencies, regarding, for example, naming conventions, data encoding rules, and physical and measurement attributes of the data.
3. Nonvolatile, which means that the DW must be loaded as a snapshot of the operational data, therefore it must be updated with a new snapshot in order to reflect the subsequent changes in the operational dataset.
4. Time-variant, so that every unit of data in the DW is stamped with a date that records the moment of time during which the record was accurate, therefore enabling the time-series analysis of the data. For example, operational data usually reflects a time period of 60 to 90 days, while DW typically contains more historical data and reflects a time horizon of 5 to 10 years. DW may be seen as a series of snapshots of data taken at a set period of time.

Data warehousing is considered to be a prerequisite for BI, since it helps the organization obtain value from its data sources by preparing and storing the enterprise data into a repository designed to support decision making. DW stores data and information in a way that it can be efficiently accessed later by the BI solutions. Data warehouse vendors include Oracle and NCR Teradata,[10] and open source versions like MySQL.[11]

A data warehouse is an architecture that describes the atomic level in the enterprise's data model, which consists of four levels (Inmon, 2005). The first level is the operational level, consisting of the enterprise resource planning systems, which we described in the previous section, and other legacy applications that serve the core transaction processes. The operational level holds the application-oriented raw data (also known as primitive data, as opposed to the derived data stored in the other levels) and serves the needs of all the organization's transaction data. The data warehouse level of data holds integrated historical primitive data as well as some derived data. The **data mart** is considered to be synonymous with the departmental level, and holds data in a specific format that serves the needs of a functional department—say, for example, the marketing or manufacturing department. Data marts are described in more detail in Chapter 6, when we explore the concept of online analytical processing (OLAP). Finally, the individual data is

[10] www.teradata.com

[11] http://www.mysql.com/

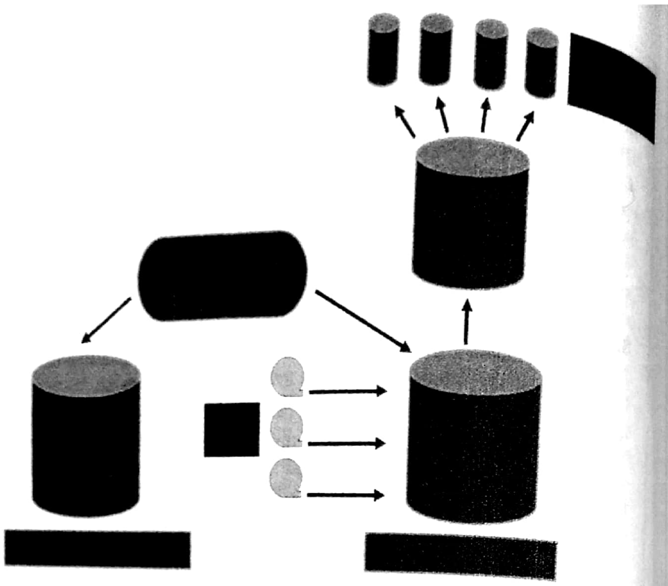

FIGURE 3.1 Four Levels of the Enterprise Data Model (Adapted from Inmon, 2005)

used by organization's individuals to serve specific needs—such as, for example, BI. Figure 3.1 illustrates the four levels of the enterprise data model.

In principle, there are two reasons why an enterprise needs to create a data warehouse (Jukic, 2006):

1. If operational queries running against the operational database (for example, the ERP system) had to compete with the analytical queries (for example, needed to run a BI process), the performance of the operational queries would degrade.
2. It is not possible to structure a database that can be queried for both operational and analytical purposes.

For these reasons, data warehouses are developed specifically to support analytical queries and are designed for the continuous retrieval of data from the operational data sources. A key step in the development of the data warehouse is to extract and integrate the operational data through the use of extract/transform (or transfer)/load (ETL) software tools. Typically, 50% to 70% of the data warehouse project development time is spent on ETL activities (Ponniah, 2001). The data warehouse implementation activities include data sourcing, data staging (or ETL), and the development of BI or decision support end-user applications.

Organizations must choose the appropriate modeling technique based on their analytical needs. Two data modeling techniques are the most popular for data warehousing, and both approaches offer a workable alternative for modeling and creating data warehouses (Sen and Sinha, 2005):

1. The **entity-relational (ER) modeling** technique, proposed by Inmon (2005), describes the data warehouse as an integrated database modeled using traditional ER database modeling techniques. This technique first requires the creation of a centralized DW. Then the central DW serves as the data source for data marts. Data marts are dimensionally modeled. The integration and consolidation of the operational data sources is effected through the ETL process. Once it is completed and populated with the data, the data is extracted into the various data marts that will support the OLAP queries. The process of integrating this process with other data extractions from all sorts of data sources—including, for example, legacy flat files—is known as the Corporate Information Factory (CIF) (Inmon et al., 2001). The Inmon approach requires creating a data warehouse ER model as a first step, which can be used as a basis for modeling dimensional and nondimensional extracts (Jukic, 2006). The Inmon approach could take longer to develop than Kimball's, but could provide a more powerful method by providing a data repository that could be used as a basis for extracts into other data stores structured in multiple ways.
2. The **dimensional modeling** technique, proposed by Kimball (2004), views the DW as a collection of dimensionally modeled data marts, and uses the operational data sources and ETL much like the Inmon approach does. The difference is that in this approach, common dimensions across data sources are modeled first, and then fact tables are added, which are connected to multiple dimensions that may be shared by more than one fact table. As a result, the resulting DW is nothing but a collection of intertwined dimensionally modeled data marts (Chenoweth et al., 2003; Jukic, 2006). In the Kimball approach, dimensionally modeled structures are created without creating an underlying ER model, therefore it could be a quicker and simpler way to create a data warehouse (Jukic, 2006). This methodology is often criticized as it is seen to lack an enterprise focus, but this may be an inaccurate perception. Figure 3.2 depicts the dimensional model process flow diagram.

The two above-mentioned approaches to building the data warehouse have also been described as the enterprise or centralized solution and the evolutionary or data mart approach, respectively. A third approach is the package approach, which employs a vendor's data warehousing solution, specifically designed to work with the corresponding ERP system and operational databases (Bashein and Markus, 2000). Deploying a data warehouse is not an insignificant endeavor by any

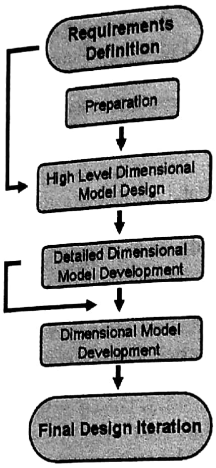

FIGURE 3.2 Dimensional Modeling Process Flow Diagram (Adapted from Kimball, 2004)

means, and presents a series of both technical and sourcing challenges (Bashein and Markus, 2000):

1. *Abundant technical options* which means that each stage (extract, storing, accessing, and analyzing data) offer a number of technical options, including for OLAP, modeling, and so on. These options add significant complexity to the DW project.
2. *Changes in technologies and vendors* which are due to ongoing consolidations in the software industry, that may result in expensive conversions from one vendor to another.
3. *Integration requirements* required, since data warehousing tools may not necessarily integrate seamlessly across vendors or even software versions from the same vendor.
4. *Knowledge transfer challenges* which are due to expertise gaps resulting from the lack of experience about how to successfully implement this type of project.

In addition there may be challenges around the data, caused by uncertain information requirements, large volumes of historical data, uncertain data quality particularly from legacy systems, lack of consistency in data names and definitions, and lack of access to all the data that the organization may require (Bashein and Markus, 2000). Finally, issues around organizational leadership, politics, and human factors may also interfere with the successful deployment of data warehouses. More recently, data warehouses have also been equipped with the ability

to extend the users' decision making from historical analysis via an active data warehouse, which supports interactive tactical decision support.

Because DW projects are ongoing, it has been said that data warehousing is more of a journey than a destination (Wixom et al., 2008). A series of characteristics distinguish a mature data warehouse (Watson, Ariyachandra, and Matyska, 2001; Wixom et al., 2008):

1. *Data*, which includes the number of subject areas, the data models used, and the quantity of data stored. In a mature DW, the data reflect the enterprise and are well integrated, over multiple time periods.
2. *Architecture*, which is the structure of the DW and data marts. For a mature DW, there is a clearly defined structure with dependent data marts.
3. *Stability of the production environment*, which refers to the existence of established processes for maintaining and growing the DW. For a mature DW, the procedures involved are routine and documented.
4. *Warehouse staff*, which includes the experience, skills, and specialization of the staff involved with the DW. For a mature DW, the staff includes experienced in-house personnel with well-defined roles and responsibilities.
5. *Users*, which includes the types, numbers, and locations of users of the DW. For a mature DW, users are organization-wide, including suppliers and customers.
6. *Impact on users' skills and jobs* refers to the ways users' jobs and required skills may need to change due to the DW. For a mature DW, users throughout the organization need improved computer skills to perform their jobs.
7. *Applications* refers to the kinds of applications that use the DW. For a mature DW, this includes reports, predefined queries, ad hoc queries, DSS, EIS, BI, and integration with operational systems.
8. *Costs and benefits*, which are those linked to the DW. For a mature DW, benefits include time saving, new and better information, improved decision making, redesigned business processes, and support for corporate objectives.
9. *Organizational Impact* refers to how much impact the DW has on the performance of the organization. For a mature DW, the impact is organization-wide and often strategic, tactical, and operational.

Even when a data warehouse reaches maturity, it continues to change to become the foundation for organization-wide BI and decision making, performance management (such as a balanced scorecard as described in Chapter 6), e-commerce, and customer relationship management (Watson et al., 2001). Business Intelligence in Practice 3.2 describes the development of a small data warehouse used to integrate funded research data across universities, to be used as the source for an interorganizational repository for an expertise locator system.

Business Intelligence in Practice 3.2: A Data Warehouse of University Experts—The Searchable Answer Generating Environment (SAGE)

The goal of SAGE was to enable researchers at the National Aeronautics and Space Administration (NASA) to partner with experts, via the creation of a searchable repository of university researchers in the State of Florida. Each university in Florida kept a database of funded research for internal use, but these databases were disparate and dissimilar. The SAGE Expert Finder created a single funded research data warehouse by incorporating a distributed database scheme, which could be searched by a variety of fields, including research topic, investigator name, funding agency, or university. Figure 3.3 represents the SAGE architecture. In this figure, the canisters in the Florida map represent the disparate databases at each of the Florida universities.

The content of each database was extracted, transformed, and loaded into the SAGE database. The extraction software used a file transfer protocol (FTP) client application that automatically obtained and transferred the database contents of each participating university. The file transfer took place according to a prescheduled transfer rule, to the SAGE database represented by the canister DATABASE. The FTP client was customized to each university, and it's marked by the abbreviations that represent each university (UWF, FAMU, UNF, FSU, UF, UCF, FAU, FIU,

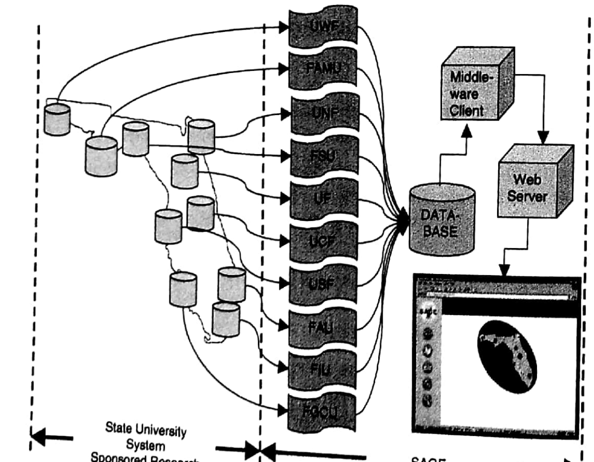

FIGURE 3.3 SAGE Architecture (Becerra-Fernandez, 2006)

(continued)

and FGCU). After the information was in the SAGE server, the next steps involved the migration of the data to the SQL Server format, followed by cleansing and transforming the data to a relational format. This methodology provided flexibility to users and the database administrator, regardless of the type of program used to collect the information at the source. The development of SAGE was marked by two design requirements: to minimize the impact on each of the universities' offices of sponsored research that collect most of the required data, and to validate the data used to identify experts.

One of the technical challenges faced during the design and implementation of this project was that the source databases of funded research from the various universities were dissimilar in design and file format. The manipulation of the source data was one of the most important issues, because the credibility of the system would ultimately depend on the consistency and accuracy of the information. Manipulating the data included the process of cleansing the data, followed by the data transformation into the relational model, and ultimately the databases' migration to a consistent format. One of the most important research contributions of SAGE was the merging of interorganizational database systems.

The next section describes how organizations go about designing the enterprise architecture.

▶ 3.4 DESIGNING THE ENTERPRISE ARCHITECTURE

The preceding sections have described the concepts of enterprise resource planning systems and the data warehouse. When companies face decisions about how to design their underlying IT infrastructure, they seek to align their IT with their business strategy. Most firms understand that it's necessary to do this in order to maximize the value they obtain from their IT investments. But how far they should pursue both their business process standardization as well as the integration of their business process across their business units may not necessarily be a straightforward decision. What complicates matters further when seeking this alignment between business strategy and IT is that the firm's business strategy is typically complex, encompasses many markets, requires different capabilities, and must respond to shifting priorities due to competitive pressures (Ross, 2005). This section describes how firms define their underlying enterprise architecture by making two important choices about their business operations units (Ross, Weill, and Robertson, 2006):

1. How standardized their business processes should be across operational units (meaning departments, regions, market segments, etc.)
2. How integrated their business processes should be across those units

None of the operating models should be considered as "the right one," as the dimensions of standardization and integration pose both benefits and challenges for the organization.

Standardization refers to "defining exactly how a process will be executed regardless of who is performing the process or where it is completed" (Ross et al., 2006, p. 27). In order to decide the level of standardization at your organization, you need to respond to the question: To what extent does the company benefit by having business units run their operations in the same way? (Ross et al., 2006, p.30). Standardization offers both efficiency and predictability across the organization, yet it limits local innovation and often requires the replacement of existing systems by standard, even perhaps inferior, systems. *Integration* "links the efforts of organizational units through shared data . . . between processes to enable end-to-end transaction processing, or across processes to allow the company to present a single face to customers (Ross et al., 2006, pp. 27–28). In order to decide the relevant level of integration for your firm, you should respond to the question: To what extent is the successful completion of one business unit's transactions dependent on the availability, accuracy, and timeliness of other business unit's data? (Ross et al., 2006, p. 30). Similarly, integration increases efficiency, coordination, transparency, and agility throughout the organization, but it poses challenges around the development of standard definitions and formats for the data shared across business units, which can be difficult and time consuming.

Based on these two choices of business process—standardization and integration—organizations will be seen to operate in one of the four possible **operating models** as depicted in Figure 3.4 (Ross, 2005; Ross et al., 2006):

1. The **diversification** model (low standardization, low integration) is used by organizations that follow a decentralized organizational design, pursuing different markets with different products and services, and that benefit from local autonomy. One example of a company that excels in the diversification quadrant is JM Family Enterprises (JMFE) consisting of four strongly interrelated businesses: a Toyota distributor serving 160 dealers, financial services, insurance products, and a Lexus dealership. These business units generate business for each other, and while JMFE may offer some centralized services to its businesses, the company has grown largely through the success of its individual units.
2. The **coordination** model (low standardization, high integration) is used by firms that have high levels of integration through the sharing of customers, products, suppliers, and partners. At the same time, they have little standardization of processes, and business unit leaders are given the

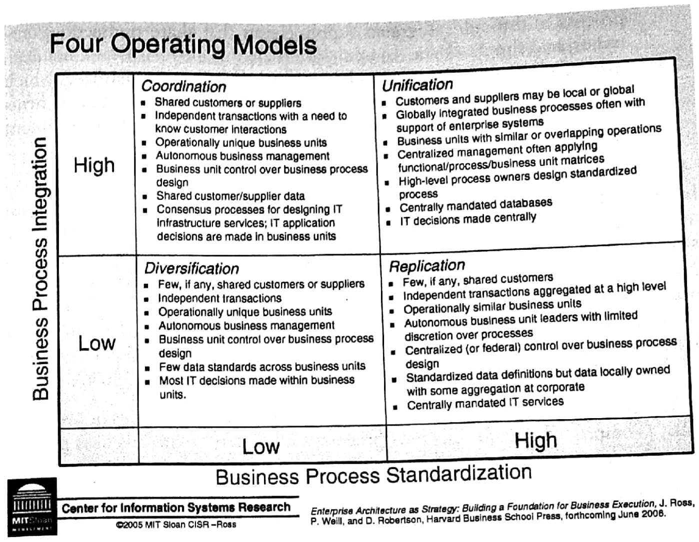

FIGURE 3.4 Characteristics of Four Operating Models (Ross et al., 2006)

autonomy to define their individualistic business processes in order to best serve their customers. One example of a firm in the coordination quadrant is Merrill Lynch,[12] composed of the global investment and banking group, the investment managers, and global private client (GPC) group. In order for GPC to meet the financial needs of its high-value customers, it must be able to deliver customized solutions (low standardization) to its target customers while providing access to integrated data across products and customers (high integration).
3. The **replication** model (high standardization, low integration) is used by firms that provide their units with the autonomy to run independently following highly standardized business processes. For firms in this model, their units do not depend on each other's transactions or data, and their success depends on their ability to innovate globally while implementing highly standardized and optimized business processes. Firms in this company include, for example, Mc Donald's and other similar franchises.
4. The **unification** model (high standardization, high integration) is used by organizations that are highly integrated around standardized business

[12] The company announced its acquisition by Bank of America in September 2008.

processes, through integrated supply chains that share transaction, customer, and supplier data. An example firm in this model is Dow Chemical. Around 60% of the firm's core chemical processes are standardized, which drives out inefficiencies and leverages economies of scale. Typically, firms in the unification model quadrant have highly centralized management environments, such as a single instantiation of an ERP system.

Once a firm has identified the appropriate operating model that reflects its business strategy, it is ready to define the *enterprise architecture*. Enterprise architecture refers to "the organizing logic for business processes and IT infrastructure reflecting the integration and standardization requirements of the company's operating model" (Ross et al., 2006, p. 47). The firm's enterprise architecture will serve as the blueprint that will lead its efforts for the development of an IT foundation that will support its future business initiatives, and refers to the high-level logic for business processes and IT capabilities. The IT unit typically defines four levels of architecture below the enterprise architecture (Ross et al., 2006):

1. The *business process architecture* which refers to the activities or tasks that comprise the major business processes identified by the business process owners.
2. The *data or information architecture* which includes the shared data definitions.
3. The *applications architecture* which refers to the individual application systems and their interfaces.
4. The *technology architecture* which includes the infrastructure services and the technology standards they are built on.

The enterprise architecture is typically depicted in a one-page picture called the *core diagram*, which synthesizes a high-level view of the processes, data, and technologies that comprise the organization's *digitized platform* for execution. The organization's core diagram is relevant to its operating model. Therefore, companies that select a specific operating model will have similar core diagrams. As an example, consider the core diagram for Delta Airlines (Ross, 2004), depicted in Figure 3.5. Delta Airlines follows a unification model. We use this example to describe the four elements of the enterprise architecture core diagram:

1. *Core business processes* specifies the set of company-wide capabilities the company requires to respond to its customers according to its operating model. These capabilities are those business processes that don't change for the organization. In the case of Delta, the core processes include the customer experience (Delta's customer touch points), the operational pipeline (loading, moving, unloading, and maintaining planes), the business reflexes (scheduling, pricing, and financial processes), and the employee relationship management (workforce scheduling, compensation, and development).

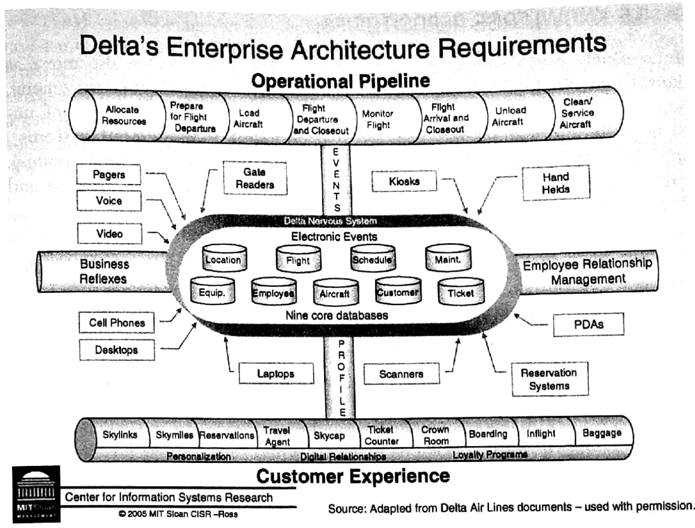

FIGURE 3.5 Delta's Operating Model (Ross et al., 2006)

2. *Sharing of data driving core processes* refers to the customer files shared across the product lines. In the case of Delta, the nine databases that are critical to process execution are depicted in the center of the core diagram: location, flight, schedule, maintenance, equipment, employee, aircraft, customer, and ticket.
3. *Key linking and automation technologies* refers to all the technologies that enable the integration of applications and access to the shared data, including ERP systems, software application packages, and middleware. It also includes those technologies, such as portals, that provide access to the systems, data, and interfaces. Referring to Figure 3.5, the Delta nervous system enables customers, employees, and the company's core processes to access the shared data.
4. *Key customers* include the major customer groups, including channels and segments that are supported by this foundation for execution.

For examples depicting the remaining three operational models, refer to Ross et al., 2006.

The next section describes the component of the organizational memory that deals with unstructured information—that is, the knowledge repository.

3.5 KNOWLEDGE REPOSITORIES

Increasingly, organizations are integrating knowledge repositories, also known as knowledge sharing systems, into their organizational memories. In general, knowledge repositories include technologies that support the capture of unstructured information and knowledge (**document management systems**), and the archiving of audio and video files (*digital content management systems*). Other related terms include **enterprise content management systems** and **Web content management systems**.

For knowledge repositories to be effective, they must attract a critical volume of knowledge seekers and knowledge owners (Dignum, 2002), so that knowledge owners:

1. Will seek to share their knowledge with a trusted and controllable group
2. Are able to decide when to share and the conditions for sharing
3. Feel reciprocity or receive a fair exchange or reward for sharing their knowledge

By the same token, knowledge seekers:

1. May not be aware of all the possibilities for sharing, thus the knowledge repository will typically help them explore these possibilities through searching and ranking
2. May require contextual information that will help them decide on the applicability of the explicit knowledge to the particular situation

A knowledge-sharing system is said to define a **learning organization**, which supports the sharing and reuse of individual, group, and organizational knowledge. As discussed above, the technologies that support knowledge repositories are document management systems and digital content management systems. Many vendors provide software offerings with these capabilities, including Microsoft (Sharepoint), OpenText (Hummingbird, Livelink), and EMC (Documentum). Furthermore, there are a number of open source offerings as well—as, for example, OpenKM, Alfresco, and KnowledgeTree, among others.

At the core of a document management system is a repository, an electronic storage medium with a primary storage location that affords multiple access points. The document management system essentially stores unstructured information. This repository can be centralized or it can be distributed. Document management builds upon the repository by adding support to the classification and organization of information, unifying the actions of storage and retrieval of unstructured data (such as documents, pictures, video, etc.) over a platform-independent system. A document management system aggregates relevant information through a common, typically Web-based interface. The document management collaborative application increases communication, thus allowing the sharing of organizational knowledge.

The document management application increases the sharing of documentation across the organization, thus assisting the sharing of organizational knowledge. Unstructured data are typically organized or *indexed* following a standard hierarchical structure or classification taxonomy, much like the index catalog is used to organize the books in a library. Frequently, *portal* technologies are used to build a common entry into multiple distributed knowledge repositories, using the analogy of a "door" as a common entry into the organization's knowledge resources. Portals provide a common user interface, which can often be customized to the user's preferences, such as local news, weather, and so on.

Knowledge repositories can be classified according to their attributes or the specific purpose they serve for the organization. Some examples of specific types of knowledge repositories include (Becerra-Fernandez and Sabherwal, 2009):

1. **Incident report databases** are used to disseminate information related to incidents or malfunctions—for example, of field equipment (like sensing equipment outages) or software (like bug reports). Incident reports typically describe the incident together with explanations of the incident, although they may not suggest any recommendations. Incident reports are typically used in the context of safety and accident investigations.
2. **Alert systems** were originally intended to disseminate information about a negative experience that has occurred or is expected to occur. However, recent applications also include increasing exposure to positive experiences. Alert systems could be used to report problems experienced with a technology, such as an alert system that issues recalls for consumer products. Alert systems could also be used to share more positive experiences. Alert systems may be applicable to a single organization or to a set of related organizations that share the same technology and suppliers.
3. **Best practices databases** describe successful efforts, typically from the reengineering of business processes, that could be applicable to organizational processes. Best practices differ from lessons learned in that they capture only successful events, which may not be derived from experience. Best practices are expected to represent business practices that are applicable to multiple organizations in the same sector, and are sometimes used to benchmark organizational processes.
4. **Lessons learned systems (LLS)** The goal of LLS is "to capture and provide lessons that can benefit employees who encounter situations that closely resemble a previous experience in a similar situation" (Weber, Aha, and Becerra-Fernandez, 2001). LLS could be pure repositories of lessons or sometimes could be intermixed with other sources of information (e.g., reports). In many instances, enhanced document management systems are supporting distributed project collaborations and their knowledge sources, while actively seeking to capture and reuse lessons from project report archives.

5. **Expertise locator systems** The goal of expertise locator systems (ELS) is to help locate intellectual capital—in other words, to find an expert who can be a source of information and who can perform a given organizational or social function (Becerra-Fernandez, 2006). The intent when developing these systems is to catalog knowledge competencies, including information not typically captured by human resources systems, in a way that could be queried later across the organization.

The differences among these types of knowledge repositories are based upon (Becerra-Fernandez and Sabherwal, 2009):

1. Content Origin: Does the content originate from experience like in lessons learned systems, or from industry standards and technical documentation as in best practice databases?
2. Application: Do they describe a complete process, or perhaps a task or a decision?
3. Results: Do they describe failures, like in incident report databases or alert systems, or successes, like in best practices databases?
4. Orientation: Do they support an organization or a whole industry?

Table 3.1 contrasts these knowledge repositories based on these attributes.

In summary, knowledge repositories are increasingly gaining the attention of IT leaders as an important source of intelligence that can be mined to improve decision making within the organization as the firm seeks to become a learning organization. In the next section we describe a case study that outlines the implementation of an ERP system at the IBM Personal Systems Group.

Table 3.1 Types of Knowledge Repositories (Adapted from Becerra-Fernandez, 2006, and Weber et al., 2001)

Knowledge Sharing System	Originates from experiences?	Describes a complete process?	Describes failures?	Describes successes?	Orientation
Incident Reports	Yes	No	Yes	No	Organization
Alerts	Yes	No	Yes	No	Industry
Lessons Learned System	Yes	No	Yes	Yes	Organization
Best Practices Databases	Possibly	Yes	No	Yes	Industry
Expertise Locator System	Yes	No	No	Yes	Organization and Industry

CASE STUDY 3.5

THE SAP IMPLEMENTATION AT IBM PERSONAL SYSTEMS GROUP[13]

In the spring of 1993, Lou Gerstner became the CEO of IBM, on a year that the company lost $8 billion dollars, capping a three-year loss of nearly $16 billion (Becerra-Fernandez et al., 2005). Big Blue was plagued with high expenses, too many employees, and redundancies in manufacturing and R&D. Whether IBM would be able to make the changes necessary to survive was very much in doubt. The strategic imperative for IBM in 1993 called for reducing costs, speeding up product development cycles, going to market as one IBM, becoming one integrated global organization, and making it easier for customers to do business with IBM. To accomplish this strategic imperative, IBM set out to transform the way it did business by adopting five major, common, end-to-end business processes: integrated product development, integrated supply chain (procurement, production, fulfillment), customer relationship management, human resources, and finance.

The need to transform the way IBM did business was evident in the Personal Systems Group (PSG). The PSG manufactured personal computers, including the ThinkPad™, Personal System/1™, Personal System/2™, Value Point™, and Ambra™. In 1993, in anticipation of strong PC sales, senior PSG management convinced the business to authorize a significant expenditure to buy more PC components, in order to avoid being out of stock of their most popular product. In the past, this had been one of the more frequent complaints from their dealers, and PSG management wanted to avoid this situation and provide high availability of its PC brands to meet market demand. Around the world, each manufacturing plant had its own application systems and technologies for supporting the various parts of the supply chain. This made it next to impossible to coordinate a PSG-wide response to this problem. By year's end, the PSG had missed their forecast "by several hundred million of dollars" and had over $3 billion remaining in inventory. The net loss was estimated to be $1 billion.

As a result, a new management team was brought to the PSG in 1994. A massive reengineering effort was launched to straighten out the manufacturing and supply chain by removing inefficiencies caused by the lack of PSG-wide coordination. The entire research and development operation was consolidated under one roof in Research Triangle Park (RTP), North Carolina. By 1999, the rest of IBM was vital and thriving again, and the stock had reached its all-time high, increasing its market capitalization more than fourfold to $169 billion. However, the Personal Systems Group continued to face a difficult competitive environment, and fierce price wars had sharply reduced its profits. Its competitors, most noticeably Dell, offered customers the ability to configure a PC to order, whereas the PSG could only offer standard predefined models. The PSG lost nearly $1 billion in 1998. In October 1999, IBM announced that up to 1,000 jobs were being cut in the PSG. The key factors for making the PSG profitable again would require getting in place a single, worldwide-integrated system at its manufacturing sites.

IBM, like many other Fortune 500 companies, had hundreds of duplicate, nonintegrated systems that had evolved in an uncoordinated manner for decades. One way to reduce costs was to reduce the number of production, logistics, and sales systems throughout the corporation. Corporate staff were asked to investigate the situation and make a recommendation. Their

(*continued*)

[13] This section was written in collaboration with Joyce Elam and Kenneth Murphy.

recommendation was that there was only one software solution big enough and broad enough to cover IBM across the board, and that was SAP[14]. IBM Corporate decided to adopt SAP as the corporate ERP solution in March 1995.

The business case was also based on PSG implementing a "group model." The concept of the group model for PSG included a common processing configuration, common architectures, education and process documentation, and overall requirements and release components. The group model embodied the following concepts (Becerra-Fernandez et al., 2005):

1. Common Processes: Business processes would be developed and configured in SAP to support common processes worldwide. Although there were legal, financial, and operating differences resulting in some sites, commonality was the cornerstone of the SAP implementation.
2. Common Development and Support: Configuration and code development would be supported by a group of programmers and analysts residing at RTP. The development groups would share one single development environment. End-user support would be located in each site to optimize responsiveness, but all fixes and enhancements would be managed centrally to maintain commonality control.
3. Common Architectures: Production would be run from one SAP production instance.
4. Common Education and Process Documentation: Training materials would be developed centrally and distributed to each manufacturing site. National languages would be accommodated.
5. Real-Time Access to Information: This would be available from all PSG production and distribution facilities, through real-time integration within SAP interfaces to PSG decision support tools and interfaces with complementary applications.

Implementing an integrated production system at the RTP site would include 300 new business processes and subprocesses and replace about twenty critical, technologically obsolete legacy systems. In addition, interfaces to many other legacy systems would need to be developed so that they could exchange data with the new ERP system, entitled Production Release 2 (PR2). Figure 3.6 shows the IT architecture map that would exist after the implementation of PR2.

The overall management structure for the PR2 project consisted of a matrix with process teams and plan teams. There were six process teams, each associated with one of the following modules in SAP: manufacturing, sales and distribution, production planning, finance and costing, procurement, and engineering change. The process teams had responsibility for designing the new business processes to be implemented in SAP. The process team leaders owned the human resources from the business side and were responsible for managing and allocating the human resources to the "plan owners." There were approximately 80 people in these teams.

Senior IT managers were the plan owners in each of these areas. Plan owners designed an overall project plan that detailed all the activities for their assigned area that had to occur during the nine-month life of the project. Plan owners were responsible for ensuring that milestones defined in their plans were met. In addition to the human resources from the process teams, these plans owners had access to the IT organization.

(continued)

[14] www.sap.com is one of the major ERP vendors.

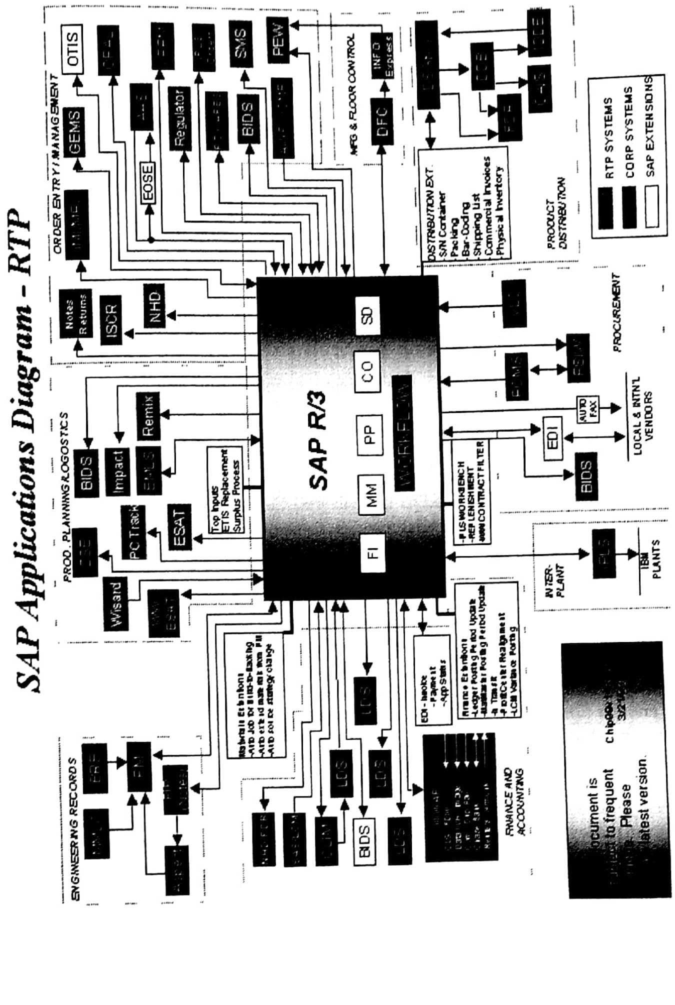

FIGURE 3.6 The IBM SAP (Source: Becerra-Fernandez et al., 2005)

As part of the process plan, IT employees configured in SAP the PSG business processes as defined by the process teams. The program plan required IT employees to develop the code to bridge SAP to the other "legacy" information systems supporting the business. In addition to building bridges to the remaining legacy systems, the IT team was responsible for extracting data from the systems being retired and placing data into SAP in support of the data migration plan. The IT team was also responsible for defining the systems architecture, which includes the hardware and software supporting the application, and the group model plan, which is the common architecture for a single system for all plants. Finally, there were three plans associated with deployment:

1. Organizational Change: Designing a plan for how jobs will change and making the changes
2. End-User Education: Designing a plan for educating the users of the system and offering training sessions
3. Deployment: Providing a liaison from the project team to the user community

An example of a high-level deployment plan is given in Figure 3.7.

Planning for PR2 deployment began in the third quarter of 1998 and continued until the go-live day. Weekly meetings were held to define responsibilities for the management of functional areas, data cleaning and migration, system testing, and user support. An extremely detailed "move to production plan" was designed that defined hour-by-hour the activities on the go-live date.

To successfully accomplish PR2 deployment, manufacturing commitments for the second quarter of 1999 had to be greatly reduced. RTP volume for the entire month of April was scheduled to be 20,000 boxes. This was equivalent to a typical week's production. The difference was off-loaded to the manufacturing plant in Guadalajara. The decision to off-load work to Guadalajara had to be made early in the planning cycle, since all components and other materials had to be delivered there. In addition, all suppliers to the RTP had to be informed of the shutdown. In some cases contracts had to be altered.

As the go-live date approached, there were still a number of serious unresolved issues. Four weeks before going live, a major problem with the implementation of the DB2 database being used by PR2 was encountered. There were also SAP software modules still being tested, and end-to-end system testing was not yet complete. Production ramp-down at RTP was completed on March 27, and at this point the team had the task of closing the books on the old legacy systems. This required working until midnight to ensure that every bit of revenue was captured. The legacy systems ran overnight batch bridges, so the team needed to allow enough time to fix any potential bridge errors.

During the shutdown period between March 27 and April 6, the focus was on data migration. The process of data conversion had begun months earlier with the data conversion managers working with the process team to define the data requirements for the SAP environment. After requirements were determined, the data conversion managers had to inform the legacy IT team of the requirements. The groups had to then agree to a data "translation" at the data level, which involved definition of field names, determination of field lengths, formatting of data, and so on. Once agreement was reached, the legacy team created programs to extract and clean the data. Once the data was moved into the test SAP system, it was tested with the assistance of a series of subroutines. This cycle of data extraction and clean-up was conducted nine times, using over 200 data conversion programs and handling more than 194,000 pieces of data.

(continued)

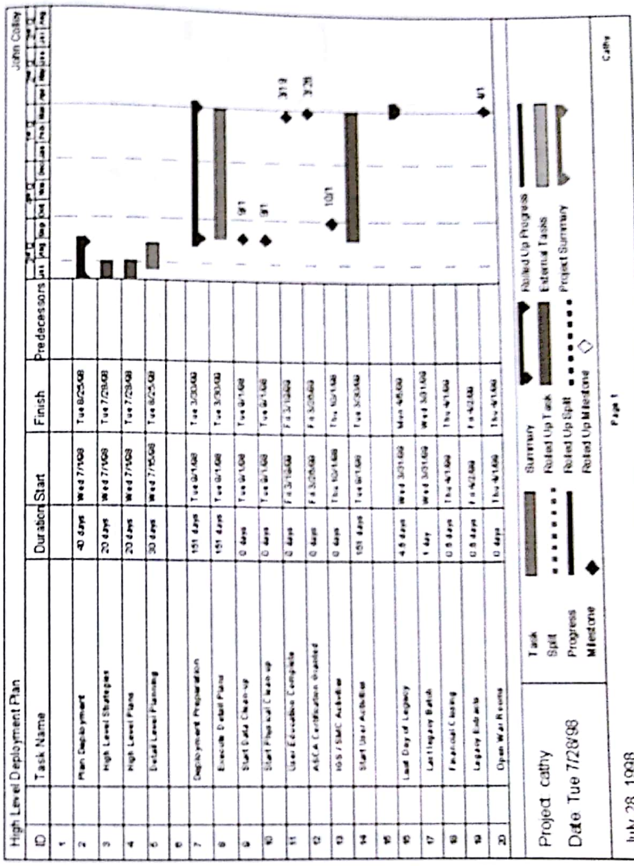

FIGURE 3.7 The SAP Implementation Project Plan

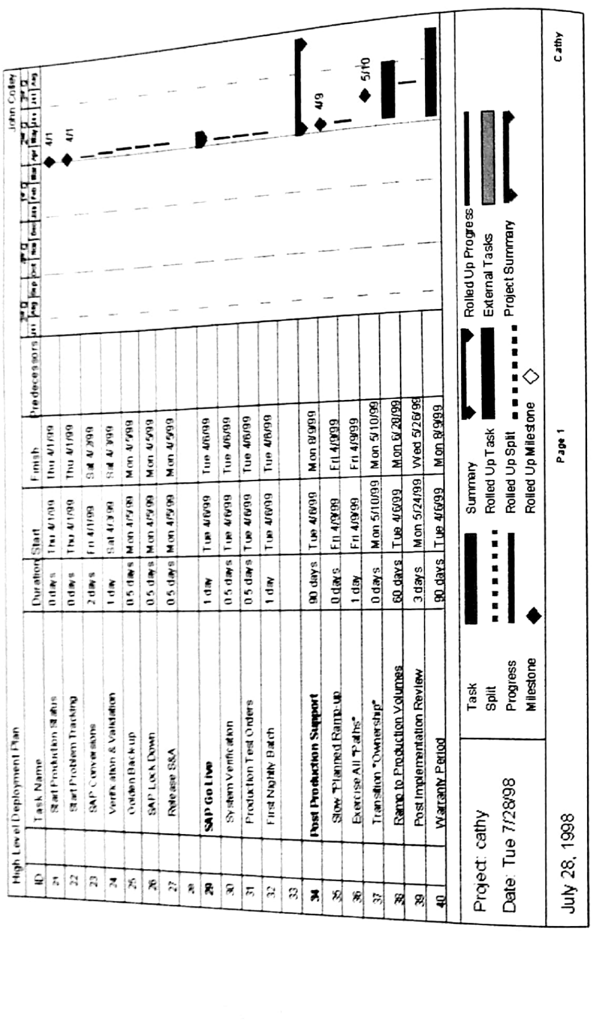

FIGURE 3.7 *(continued)*

The SAP deployment team instituted a "war room" for the days following ramp-up. The war room was open all the time and equipped with a direct phone line to the RTP production facility. During the first several weeks of deployment, there were three calls from the production facility per day and an executive status report at the end of the day. At the start of go-live, the team members focused all their energy on the war room and the events occurring in the RTP production facility. By May, it was clear that the system was stable. It had been a full month since PR2 had gone live at RTP. Everyone was ecstatic at the relative lack of major problems with the system rollout. It was a great relief, considering the incredible efforts and significant challenges in the months prior to the go-live date. The successful installation of PR2 in RTP allowed for the sunset several legacy systems and required 194,000 data items to be migrated from legacy systems to SAP. Only a few major problems with the system rollout had been encountered. The implementation of the ERP system proved to be a resounding success.

Due to their scope and complexity, ERP implementations have been and will continue to be a challenge. Clearly, the well-published failures indicate that not all organizations have been up to this challenge. The PSG at IBM was up to the challenge. The successful implementation demonstrates that a very large-scale ERP project can be completed successfully. Successful ERP implementations require that well-defined plans are executed in order to ensure that the project's critical success factors are addressed throughout the implementation. This case study demonstrates that the critical success factors for the implementation of ERP projects, previously described in Section 3.2, were significant to the project's success. In conclusion, the risks associated with these projects leave little room for poor project management.

Finally, this case study also confirms that ERP implementations require the unwavering commitment of upper management, who must be willing to provide the necessary resources and even intervene when required to do so. Furthermore, careful organizational and technical project planning and management must be exercised throughout the project, since the organization must be able to rely on significant project management expertise in order to be successful. In addition, project teams must be knowledgeable, diverse in business and technical expertise, and highly motivated, and they must understand the business strategy behind ERP implementations. Project managers must be cognizant of the entire landscape of IT applications and how ERP integrates with these, in addition to how the current business processes will need to change in the face of the new technology.

In this case study, IBM provides a very powerful model for how to succeed in implementing ERP solutions. Any organization beginning the process would benefit by assessing how it compares to IBM on each of the six factors. This assessment along with the practices and approaches taken by the Raleigh project team can provide very valuable guidance in successfully implementing ERP systems.

▶ 3.6 SUMMARY

In this chapter you learned the technologies that enable the creation of the organizational memory, or knowledge repository, the aggregate of all the corporate structured information that is used for BI. The chapter starts with a discussion of ERP systems and the critical success factors for their implementations. The chapter also describes data warehousing, the reasons why organizations implement them, the two most popular modeling methodologies used to implement them, and the challenges around deploying data warehouses. Finally, the chapter describes how to define the operating model for the enterprise and the corresponding enterprise architecture. The chapter concludes with a case study that documents the successful implementation of ERP at IBM Personal Systems Group, and how the project team successfully overcame the challenges posed by one of the largest ERP implementations to date.

▶ KEY TERMS

alert system
best practices database
business processes
critical success factors
content management system
coordination
data mart
data warehouse
dimensional modeling
diversification
document management system
enterprise application integration (EAI)
enterprise architecture (EA)
enterprise content management
enterprise resource planning (ERP) system
ERP implementation
entity-relational (ER) modeling
expertise locator system
extract/transform (or transfer)/load (ETL) software
incident report database
learning organization
lessons learned system
manufacturing resource planning or material requirements planning (MRP) system
middleware
operating model
replication
SMBs
unification
Web content management system

▶ REVIEW QUESTIONS

1. Describe the challenges around the deployment of ERP systems.
2. What are the six unique, critical success factors for large-scale ERP project implementations?
3. What are the two reasons why the enterprise needs to create a data warehouse?
4. Describe the two most popular data modeling techniques for data warehousing.
5. Describe the challenges around deploying a data warehouse.
6. What are the characteristics of mature data warehouses?

▶ APPLICATION EXERCISES

1. Review the ERP architecture of your organization or the one presented in the IBM case study in Figure 3.6. What modules are provided by the ERP system? What are the legacy systems integrated to the ERP system?

2. Review the architecture of the data warehouse of your organization or find an example in the literature. Is it a mature DW? Why or why not?

3. Describe the operating model of your organization. Given the characteristics of your organization, is that the appropriate operating model for the enterprise?

4. Go to www.teradata.com, and click on the *Enterprise Data Warehousing* option in the *Product and Services* tab. Download some of the demos on *Active Enterprise Intelligence*. Describe the components in active enterprise intelligence. What advantages do they present? Explain with some examples from the airlines, call centers, or the financial services industry.

5. Go to www.teradata.com, and click on the *Enterprise Data Warehousing* option in the *Product and Services* tab. Download some of the demos on health care solutions. Describe how health care organizations could improve quality of service while minimizing costs using data warehousing and BI.

6. Go to the Microsoft Dynamics Web page (http://www.microsoft.com/dynamics/environment.mspx) and download the Environmental Sustainability Dashboard demo. In your opinion, is this functionality something that SMBs will be interested in?

▶ REFERENCES

Bashein, B., and Markus, L. 2000. *Data Warehouses: More Than Just Mining*. Financial Executives Research Foundation.

Becerra-Fernandez, I. 2006. "Searching for Experts on the Web: A Review of Contemporary Expertise Locator Systems." *ACM Transactions on Internet Technology*, 6(4), 333–355.

Becerra-Fernandez, I., Murphy, K., and Elam, J. 2005. "Successfully Implementing ERP: The IBM Personal Systems Group Experience." *International Journal of Internet and Enterprise Management*, 3(1), 78–97.

Becerra-Fernandez, I., and Sabherwal, R. 2009. *Knowledge Management: Systems and Practices* (Armonk, N.Y.: M. E. Sharpe). Forthcoming.

Bingi, P., Sharma, M., and Godia, J. 1999. "Critical Issues Affecting an ERP Implementation." *Information Systems Management*, 16(3), 7–14.

Chenoweth, T., Schuff, D., and St. Louis, R. 2003. "A Method for Developing Dimensional Data Marts." *Communications of the ACM*, 46(12), 93–98.

Davenport, T. 1998. "Putting the Enterprise into the Enterprise System." *Harvard Business Review*, 76(4), 121–131.

Dignum, V. 2002. A Knowledge-Sharing Model for Peer Collaboration in the Non-Life Insurance Domain. In Proceedings of the 1st German Workshop on Experience Management, Berlin, Germany.

Esteves, J., and Pastor, J. 2001. "Enterprise Resource Planning Systems Research: An Annotated Bibliography." *Communications of the Association for Information Systems (CAIS)*, 7(8).

Esteves, J., and Pastor, J. 2007. "An Updated ERP Annotated Bibliography: 2001–2005."*Communications of the Association for Information Systems (CAIS)*, 19(18).

Hasselbring, W. 2000. Information System Integration: Integration. Communications of the ACM. 43, 6, 32–38.

Hayes, F. 2002. "The Story so Far." *Computerworld*. April 15.

Inmon, W., Imhoff, C., and Sousa, R. 2001. *Corporate Information Factory*, Second Edition (New York: John Wiley).