



BUILDING E-COMMERCE APPLICATIONS & INFRASTRUCTURE¹

Content

Tracking United Parcel Service Shipments

- 12.1 The Development Process
- 12.2 The Major EC Applications and Their Functionalities
- 12.3 Development Options for EC Applications
- 12.4 Criteria for Selecting a Development Approach
- 12.5 Third-Party EC Components and Suites
- 12.6 Connecting to Databases and Other Enterprise Systems
- 12.7 Vendor and Software Selection
- 12.8 Usage Analysis and Site Management

Managerial Issues

Summary

Real-World Case:

Whirlpool's Trading Portal

Learning objectives

Upon completion of this chapter, you will be able to:

- Discuss the major steps in developing an EC application.
- Describe the major EC applications and list their major functionalities.
- List the major EC application development options along with their benefits and limitations.
- Describe various E application outsourcing options.
- Discuss the major components of an electronic catalog and EC application suite.
- Describe various methods for connecting an EC application to back-end systems and databases.
- Describe the criteria used in selecting an outsourcing vendor and package.
- Understand the value and uses of EC application log files.
- Discuss the importance and difficulties of EC application maintenance.

¹Contributions to this chapter were made by Mohamed Khalifa of City University of Hong Kong and by Joseph Walls of the University of Michigan, Ann Arbor.

TRACKING UNITED PARCEL SERVICE SHIPMENTS

The Problem

United Parcel Service (UPS) has been in the package distribution business since 1907. It is the world's largest package distribution company, transporting over 13 billion parcels and documents each year. For some time, UPS has provided the means for customers to track their shipments to determine the status and whereabouts of a particular package. In the past, this was done primarily over the telephone. Customers would call UPS with the tracking number of their shipment. An operator would look up the status of the shipment in the UPS database and relay the information to the customer. Servicing these calls over the phone was an expensive proposition (estimated at \$2 per call).

The Solution

In an effort to reduce transaction costs and to efficiently service more customers, UPS created a Web site (*ups.com*) that enabled customers to track their shipments online, to determine the cost and transit time for delivery of a package, to schedule a package for pickup, and to locate the nearest drop-off facility. These online facilities are accessed from the UPS Web site homepage. For example, if a customer wants to track a shipment, they click the "Track" icon at the top of the homepage. This takes the customer to an online form where the customer simply enters the tracking number, hits the "Track" button, and receives precise information about the location of the designated shipment.

Although the front end of the UPS Web site is simple enough, the back-end processing used to handle a tracking request is a little more complicated. When a

request first reaches the UPS site, it is handed off to one of a handful of Web servers. The particular server that is selected depends on a variety of factors such as the current load on the various machines. Next, the selected server passes the request to the appropriate application server. In this case, the application is a tracking. From there, the application server passes the request to an IBM AS/400 computer, which is attached to the UPS tracking database. This database is one of the largest transaction databases in the world, containing over 20 terabytes of data. The mainframe actually performs the database search for the status information associated with the tracking number. Once the information is found, it is passed back up the line through the various servers to the customer's browser. Exhibit 12.1 provides a schematic of the whole operation.

The Results

Today, the UPS site services over 4 million online tracking requests per day. It has kept UPS competitive with other shipping companies (notably, FedEx), that also offer online tracking services and customer information. Recently, the UPS site has begun to offer customers the option of tracking their packages through wireless devices (cell phones, PDAs, and Web-enabled pagers). In these cases, the Web pages have been modified to support the particular wireless device being used. In the same vein, specialized servers

are used to deliver the pages over the wireless communication networks.

In addition to their own site, UPS offers a set of e-commerce solutions and a technology infrastructure that enables other companies to incorporate UPS' online order entry, shipping, and tracking capabilities (*ec.ups.com*). For instance, Amazon.com utilizes UPS' online tools to provide customers with the online means to track packages shipped by UPS. UPS also offers e-commerce tools and services for managing an enterprises' overall supply chain.

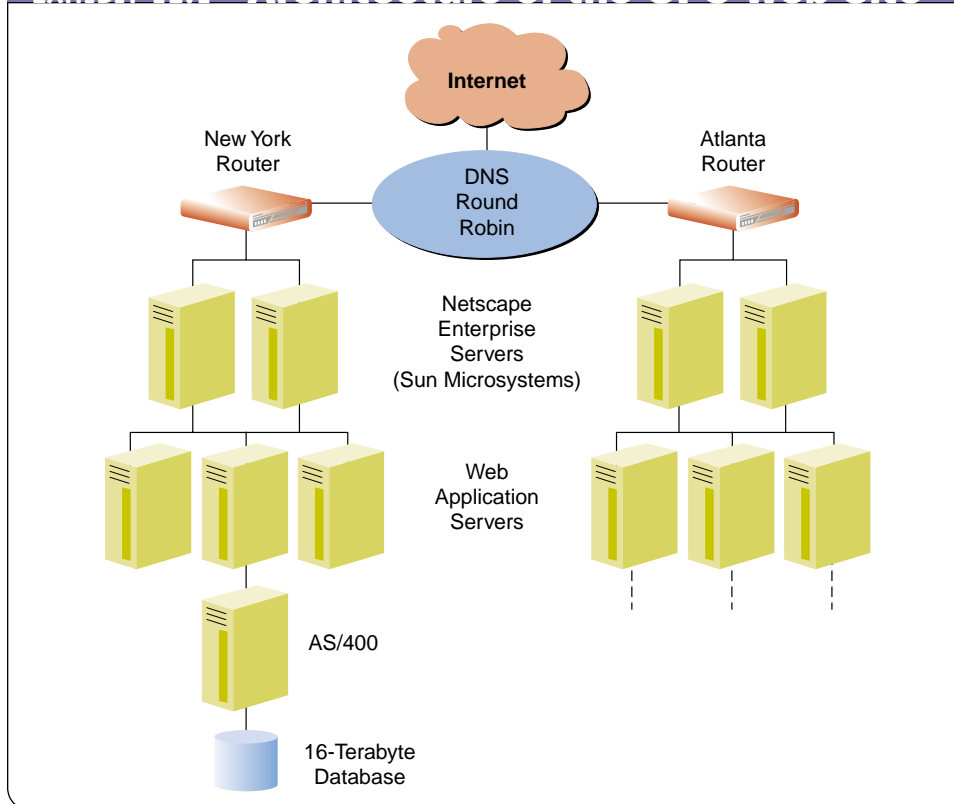
Source: *ups.com*, 2001, 2002.

What We Can Learn . . .

There is more to an EC Web site than meets the eye. For example, on the surface, the UPS Web site appears to be relatively simple. However, behind the scenes a number of hardware and software components are supporting these applications. The same is true of virtually every e-commerce site.

This chapter describes the process for developing EC applications. The discussion covers the general steps in the process and considers a number of different alternatives for implementing e-commerce applications.

EXHIBIT 12.1 Architecture of the UPS Web Site



Source: UPS Communication Group, Atlanta, GA.

12.1 THE DEVELOPMENT PROCESS

Several options are available for developing an EC site. In some cases, the components making up a Web site come from a small number of vendors. In other instances, they come from several different vendors. Small storefronts with a few key components can be developed with HTML, Java, or another programming language. They can also be quickly implemented with commercial packages or leased from an **application service provider (ASP)**, a company that provides business applications to users for a small monthly fee. Some packages are available for a free trial period ranging from 30 to 90 days. Building medium to large applications requires extensive integration with existing information systems such as corporate databases, intranets, enterprise resource planning (ERP), and other application programs. Larger applications can be outsourced or developed in-house. They can also be implemented with a commercial package or customized by hand.

Although the cost of developing an EC site of any size has dropped considerably over the past few years (Carmichael 2001), the cost can still run in the thousands of dollars, even for a small site. To ensure at least a modicum of success, a systematic development process is needed to create, implement, and maintain a site. A project team is also needed to manage the development process, as well as

application service provider (ASP)

A company that provides business applications to users for a small monthly fee.

the relationships with outside vendors and business partners. The development process should include the steps detailed in the following sections.

STEP 1: CREATING AN EC ARCHITECTURE

EC architecture

A plan for organizing the underlying infrastructure and applications of a site.

An **EC architecture** is a plan for organizing the underlying infrastructure and applications of a site. The plan includes the business goals and vision for the site; the information and data required to fulfill the goals and vision; the application modules that will deliver and manage the information and data; the specific hardware and software on which the application modules will run, as well as the security, scalability, and reliability required by the applications; and the human resources and procedures for implementing the architecture.

Various IT tools and methodologies can be used to support the creation of an application architecture (e.g., see Kendall and Kendall 1999). As the creation of an architecture is an iterative process, collaborative methodologies such as Joint Application Development (JAD) are especially useful in identifying and modifying system requirements.

STEP 2: SELECTING A DEVELOPMENT OPTION

EC applications can either be developed in-house, outsourced to another party, or some combination of both. If the application is built in-house, it can be built entirely from scratch or commercial products can be used and modified to meet specific needs. If the application is outsourced, the application can be run entirely by an ASP, run on a third-party marketplace (e.g., an auction site, exchange, or online mall), or run on another company's Web site. Even if the application is outsourced, a company will still need to participate in the architectural design. At the end of this step, an application is built (or purchased) and ready to be installed.

STEP 3: INSTALLING, TESTING, AND DEPLOYING EC APPLICATIONS

Once an option has been selected, the next step involves getting the application up and running on the selected hardware and network environment. As the UPS tracking application at the beginning of the chapter demonstrated, one of the steps in installing an application is connecting it to back-end databases, to other applications, and often to other Web sites. For example, if a prospective customer orders a product from a site, it would be helpful if the site could determine if the product was in stock. In order to do this, the ordering system would need to be connected to the inventory system.

During this step, the modules that have been installed need to be tested. A series of tests are required: **unit testing** (testing the modules one at a time), **integration testing** (testing the combination of modules acting in concert), **usability testing** (testing the quality of the user's experience when interacting with the site), and **acceptance testing** (determining whether the site meets the original business objectives and vision). Once the applications making up an EC site pass all of the tests, they can be deployed to the end users.

unit testing

Testing application software modules one at a time.

integration testing

Testing the combination of application modules acting in concert.

usability testing

Testing the quality of the user's experience when interacting with a Web site.

acceptance testing

Determining whether a Web site meets the original business objectives and vision.

STEP 4: OPERATION AND MAINTENANCE

It usually takes as much time, effort, and money to operate and maintain a site as it does to build and install it in the first place. To enjoy continued usage, a site needs to be continually updated. For example, at a B2C site, new products need to be added to the catalog, prices need to be changed, and new promotions need to be run. These changes and updates need to undergo the same testing procedures used during the installation process. Additionally, usage patterns and performance need to be studied to determine which parts of the underlying applications need to be modified or eliminated from the site.



- ▶ List the major steps in developing an EC application.
- ▶ Define the various types of testing used during the EC development process.

12.2 THE MAJOR EC APPLICATIONS AND THEIR FUNCTIONALITIES

When the Web was first getting started, there were few guidelines for developing an EC site. Today, there are numerous examples of all types of sites, including B2C, B2B, exchanges, and the like. Sites of a particular type generally have the same underlying applications and provide similar sorts of functionality. This simplifies the task of creating the underlying application architecture for a particular type of site. The discussion that follows describes the major characteristics and functionalities of some of the better-known EC applications.

B2C STOREFRONTS

An electronic storefront must support the same steps and tasks that a physical store must support. In particular, an *electronic storefront* (a seller's Web site where purchases can be made) needs to offer buyers the ability to:

- ▶ Discover, search for, evaluate, and compare products for purchase using e-catalogs
- ▶ Select products to purchase and negotiate or determine their total price
- ▶ Place an order for desired products using a shopping cart
- ▶ Pay for the ordered products, usually through some form of credit
- ▶ Confirm an order, ensuring that the desired product is available
- ▶ Track orders once they are shipped

On the other hand, the merchant needs to:

- ▶ Provide access to a current catalog of product offerings, allowing prospective buyers to analyze and evaluate the offerings
- ▶ Provide an electronic shopping cart where buyers can assemble their purchases

- ▶ Verify a customer's credit and approve the customer's purchase
- ▶ Process orders (back-end services)
- ▶ Arrange for product delivery
- ▶ Track shipments to ensure that they are delivered
- ▶ Provide the means for buyers and visitors to register at the site, to make comments, or to request additional information
- ▶ Answer customers' questions or pass queries and requests to a Web-based call center
- ▶ Analyze purchases in order to customize buyers' experiences
- ▶ Provide Web-based post-sale support
- ▶ Create the capability for cross-sell and up-sell
- ▶ Provide language translation if needed
- ▶ Measure and analyze the traffic at the site in order to modify and maintain the various applications

In order to provide these capabilities, an electronic storefront must contain at least three interrelated subsystems (DeWire 1998):

1. A merchant system or storefront that provides the merchant's catalog with products, prices, and promotions. A shopping cart is usually included.
2. A transaction system for processing orders, payments, and other aspects of the transaction.
3. A payment gateway that routes payments through existing financial systems primarily for the purpose of credit card authorization and settlement.

SUPPLIER SELL-SIDE B2B SITES

A sell-side B2B site is similar to a B2C storefront, enabling one business to purchase goods and services from another. However, a B2B site also has additional features, including:

- ▶ Personalized catalogs and Web pages for all major buyers
- ▶ A B2B payment gate
- ▶ Electronic contract negotiation features
- ▶ Product configuration by customers (e.g., Cisco or Dell)
- ▶ Affiliate program capabilities
- ▶ Business alerts (e.g., special sales)

E-PROCUREMENT

An e-procurement site is an online intermediary that offers businesses access to hundreds of parts and services provided by suppliers. There are several variations of e-procurement systems, each with their own specialized capabilities.

Aggregating Catalogs

In large organizations, multiple buyers are involved in making purchases from a large number of suppliers. One way to reduce costs and other inefficiencies in the purchase process is to aggregate the items from approved suppliers into a single online catalog. Some of the specialized requirements for this type of site include:

- ▶ Search engine for locating items with particular characteristics
- ▶ Comparison engine for alternative vendors
- ▶ Ordering mechanism
- ▶ Budget and authorization feature
- ▶ Usage comparisons (among various departments)
- ▶ Payment mechanism (e.g., use of a purchasing card)

Reverse Auctions

In a reverse auction, buyers list the items they wish to purchase and sellers bid to provide those items at the lowest price. Sites of this sort provide:

- ▶ Catalog of items to be tendered and their content management
- ▶ Search engine (if there are many items)
- ▶ Personalized pages for potential large bidders
- ▶ Reverse auction mechanisms, sometimes in real time
- ▶ Facility to help prepare, issue, manage, and respond to a buyer's request for quotes (RFQs)
- ▶ Ability to bid dynamically
- ▶ Automatic vendor approval and workflow (e.g., SmartMatch's supplier identification technology)
- ▶ Electronic collaboration with trading partners
- ▶ Standardization of RFQ writing
- ▶ Site map
- ▶ Mechanism for selecting suppliers
- ▶ Automatic matching of suppliers with RFQs
- ▶ Automatic business process workflow
- ▶ Ability for bidders to use m-commerce for bidding
- ▶ Automated language translation

EXCHANGES

An exchange is an e-marketplace tying many buyers to many suppliers. In addition to combining the functionalities of buy-side, e-procurement, and auction sites, they also have a number of other capabilities:

- ▶ Collaboration services (including multichannel)
- ▶ Community services
- ▶ Web-automated workflow

- ▶ Integrated business process solutions
- ▶ Central coordination of global logistics for members, including warehousing and shipping services
- ▶ Integration services (systems/process integration into e-marketplace, trading partners, and service providers)
- ▶ Data mining, customized analysis and reporting, real-time transactions, trend and customer behavior tracking
- ▶ Transaction flow managers
- ▶ Negotiation mechanisms
- ▶ Language translation
- ▶ Comprehensive links to related resources

These lists of major characteristics and functionalities can be used by application developers as outlines or checklists from which to develop plans for specific EC applications.



- ▶ List the major subsystems of an electronic storefront.
- ▶ Describe some of the major functions required by an aggregating catalog.
- ▶ Describe some of the major functions needed to build a reverse auction.
- ▶ List some of the functional requirements of an online exchange.

12.3 DEVELOPMENT OPTIONS FOR EC APPLICATIONS

There are three basic options for developing an EC Web site: (1) develop the site in-house either from scratch or with off-the-shelf components, (2) use a packaged application designed for a particular type of EC site, or (3) lease the application from a third party. Each of these approaches has its benefits and limitations.

IN-HOUSE DEVELOPMENT: INSOURCING

Although in-house development—**insourcing**—can be time-consuming and costly, it may lead to EC applications that better fit an organization's strategy and vision and differentiate it from the competition. Rarely is a site built from scratch. Instead, the required applications are often constructed from standard components (e.g., Web servers such as Apache or Microsoft's IIS) using Web scripting languages such as PHP, Microsoft's Active Server Pages (ASP), JavaServer Pages (JSP), or ColdFusion. These scripting languages make it easier to integrate application functionality with back-end databases and other back-office systems (e.g., order entry).

Insourcing is a challenging task that requires specialized IT resources. For this reason, most organizations usually rely on packaged applications or completely outsource the development and maintenance of their EC sites.

insourcing

In-house development of applications.

BUY THE APPLICATIONS: TURNKEY APPROACH

A number of commercial packages provide the standard features required by EC applications. A *turnkey approach* involves buying a commercial package, installing it, and starting it up. Buying a commercial package requires much less time and money than in-house development. When selecting a particular package, the package should not only satisfy current needs, it must also be flexible enough to handle future ones. Otherwise the package may quickly become obsolete. Additionally, because one package can rarely meet all of an organization's requirements, it is sometimes necessary to acquire multiple packages. In this case, the packages need to be integrated with each other and with other software and data.

This option has several major advantages:

- ▶ Many different types of off-the-shelf software are available.
- ▶ It saves time and money (compared to in-house development).
- ▶ It requires fewer personnel.
- ▶ The company knows what it is getting before it invests in the product.
- ▶ The company is not the first and only user.

This option also has some major disadvantages:

- ▶ Software may not exactly meet the company's needs.
- ▶ Software may be difficult or impossible to modify or it may require huge process changes.
- ▶ The company may experience loss of control over improvements and new versions.
- ▶ Off-the-shelf applications can be difficult to integrate with existing systems.
- ▶ Vendors may drop a product or go out of business.

The buy option is especially attractive if the software vendor allows for modifications. However, the option may not be attractive in cases of high obsolescence rates or high software cost. In such cases, one should consider leasing.

LEASING

Leasing an application package can result in substantial cost and time savings. In those cases where extensive maintenance is required or where the cost of buying is very high, leasing is very advantageous, especially for small to medium enterprises (SMEs). Leasing is also advantageous when a company wants to experiment with a package before making a heavy up-front investment, protect its own internal networks, quickly establish a presence in the market, or rely on experts to establish a site over which they can later assume control.

Leasing can be done in one of two ways. The first way is to lease the application from an outsourcer and install it on the company's premises. The vendor can help with the installation and frequently will offer to contract the operation and maintenance of the system. Many conventional applications are leased this way.

The second way is to lease the application from an ASP that hosts the application at its data center (Kern and Kreijger 2001). The applications are then accessed via the Internet through a standard Web browser interface. In such an arrangement, applications can be scaled, upgrades and maintenance can be centralized, physical security over the applications and servers can be guaranteed, and the necessary critical mass of human resources can be efficiently utilized. Leasing from ASPs not only saves various expenses (e.g., labor costs) in the initial development stage, it also helps reduce the software maintenance, upgrading, and user training costs in the long run.

Leasing from ASPs does have its disadvantages. Many companies are concerned with the adequacy of protection offered by the ASP against hackers, theft of confidential information, and virus attacks. More importantly, with the demise of many dot-com companies, a large number of ASPs have gone out of business (Koch 2000). Before an ASP is used, it is imperative that a prospective customers investigate the ASP's financial viability.

A detailed list of the benefits and risks associated with ASPs is provided in Exhibit 12.2. Information about the general state of the ASP marketplace can be obtained from the ASP Industry Consortium (aspconsortium.org).

EXHIBIT 12.2 Benefits and Risks of Using an ASP		
Type	Benefits	Potential Risks
Business	<ul style="list-style-type: none"> Reduces the need to attract and retain skilled IT professionals Enables company to concentrate on strategic use of IT Enables small and medium-sized companies to use Tier 1 applications (e.g., ERP, SCM, and CRM) Application scalability enables rapid growth of companies 	<ul style="list-style-type: none"> Loss of control and high level of dependence on ASP Inability of ASP to deliver quality of service; lack of skills and experience
Technical	<ul style="list-style-type: none"> Fast and easy application deployment Higher degree of application standardization Access to wide range of applications Application maintenance simplified and performed by ASP Simplified user support and training 	<ul style="list-style-type: none"> Level of customization and legacy application integration offered by ASP is insufficient Low reliability and speed of delivery due to bandwidth limitations Low capability of ASP to deal with security and confidentiality issues
Economic	<ul style="list-style-type: none"> Low total cost of ownership Low up-front investments in hardware and software Improved cost control as result of predictable subscription costs 	<ul style="list-style-type: none"> Pricing changes by ASP unpredictable for application updates and services

Source: Kern and Kreijger, "An Exploration of the ASP Outsourcing Option," *Proceedings*, HICSS 31 Hawaii, 1997 © 2001 IEEE.

OTHER DEVELOPMENT OPTIONS

In addition to ASPs, several other leasing and hosting options are available for developing and maintaining an EC site. These include:

- ▶ **E-marketplaces, exchanges, auctions, or reverse auctions.** With this option, a company plugs itself into an existing EC Web site. For example, a company can place its catalogs in Yahoo!'s marketplace and rely on Yahoo!'s store to attract buyers. (See the Online Tutorial "Building an Application with Yahoo! Store.") In this instance, a company pays Yahoo! a monthly fee for the catalog space. In other cases, fees are charged for each transaction or for a percentage of the sales revenues. In most of these cases, templates are provided for plugging into the site.
- ▶ **Joint ventures and consortia.** Several different partnership arrangements may facilitate EC application development. In some cases, companies can team up with a company or a consortium that already has an application in place or they can join with a series of partners to create a new application and site (e.g., Covisint).
- ▶ **Internet malls.** The Web has several thousand Internet malls. Like a real-world mall, an Internet mall consists of a single entry displaying a collection of electronic storefronts. In contrast to earlier cybermalls, today's malls have a common look and feel. A well-run mall offers cross-selling from one store to another and provides a common payment structure by which buyers can use a single credit card purchase to buy products from multiple stores. Theoretically, a mall has wider marketing reach than a stand-alone site and, as a consequence, generates more traffic. The downside is that income must be shared with the mall owner.
- ▶ **ISPs.** In addition to providing Internet access to companies and individual users, a large number of ISPs offer hosting services for EC. For the most part, ISPs are focused on operating a secure transaction environment and not on store content. This means that merchants using the services of an ISP must still design their own pages. Of course, this task can be outsourced to a third party.
- ▶ **Telecommunication companies.** Increasingly, the large telecommunications companies are expanding their hosting services to include the full range of EC solutions. MCI, for example, offers Web Commerce for a fee of \$500 per month. Web Commerce runs on Open Market and Microsoft Commerce Server technologies. Similarly, AT&T provides a number of EC services, including the AT&T eCommerce Suite for \$695 per month.
- ▶ **Software houses.** Many software companies, such as IBM and Ariba, offer a range of outsourcing services for developing, operating, and maintaining EC applications.



- ▶ Define insourcing.
- ▶ List some of the pros and cons of using packaged EC applications.
- ▶ Describe the major forms of application leasing.
- ▶ List some of the alternative leasing and hosting options.

12.4 CRITERIA FOR SELECTING A DEVELOPMENT APPROACH

If a company decides to buy or lease an EC application, the following selection criteria need to be considered.

- ▶ **Flexibility.** Commercial packages need to be modified or adapted to the specific requirements of an application. Therefore, it is important to evaluate the extent to which a package can be adapted and the willingness of the vendor to perform or support the adaptation.
- ▶ **Information requirements.** The selected package should satisfy the information requirements of the EC application. Information collection, storage, and retrieval capabilities and the database structure should be carefully examined.
- ▶ **User friendliness.** User friendliness is especially important for B2C, G2C, and some B2B sites. In these cases, if an application is hard for the average visitor or customer to use, then it will have an immediate impact on the bottom line.
- ▶ **Hardware and software resources.** The computer type and the operating system required by the package must be compatible with the existing platform. The CPU and storage requirements are also important considerations.
- ▶ **Installation.** The installation effort required to implement the package should also be taken into account. Some packages are complex, and their installation requires extensive consultation. The installation process can also take a considerable amount of time.
- ▶ **Maintenance services.** Because EC application requirements are constantly changing, continuous maintenance is required. It is important to consider how often the package needs to be upgraded and whether the vendor provides assistance for its maintenance.
- ▶ **Vendor quality and track record.** It is less risky to acquire an EC package from a vendor that has a good reputation and track record than from one with a less-than-stellar or unknown reputation. The quality of the vendor can be indicated by their related experience in the particular application, their sales and financial records, as well as their responsiveness to clients' requests. Vendor support may include online help, customer relationship management (CRM) programs, as well as partner relationship management (PRM) tools. To minimize risk, minor applications should be leased first.
- ▶ **Estimating costs.** The costs of EC projects are usually difficult to assess and often underestimated. In addition to the obvious costs associated with EC development, it is also important to factor in the costs of installation, integration, customization, and maintenance.
- ▶ **Personnel.** Staffing requirements should be planned for in advance to ensure that the organization has the appropriate human resources for systems development (in the case of in-house development), implementation, operation, and maintenance. Currently, it is difficult to recruit and retain IT personnel with appropriate knowledge and experience in EC application development. Special expertise can be acquired from external consultants, but usually at a very high cost.

- ▶ **Technological evolution.** Planning ahead for technological evolution facilitates the upgrade of EC applications and enables the organization to adopt innovations more quickly than the competition. It is therefore very important to allow for flexibility in the application design so that the chosen options do not impose major limitations on future choices. Given the rapid pace of IT evolution, it is sometimes preferable to develop EC applications incrementally in order to take advantage of the latest developments in the technology.
- ▶ **Scaling.** System **scalability** refers to how big a system can grow in various dimensions to provide more service. Scalability can be measured in several ways, including the total number of users, the number of simultaneous users, and the transaction volume. These dimensions are not independent, as scaling up the size of the system in one dimension can affect the other dimensions. The growth of scale is facilitated or constrained by the system architecture.
- ▶ **Sizing.** The required size and performance of an application are also difficult to predict, as the growth of the user population of certain EC applications is hard to anticipate. Overloading the application decreases performance. For regular IT applications, deterioration in performance may affect productivity and user satisfaction; for EC applications, it could result in a major loss of business.
- ▶ **Performance.** System performance is a critical factor for business success, particularly if the system is used for EC. In addition to convenience, good performance also brings customers and competitive advantages. Performance is measured by two main metrics: latency and throughput. **Latency** measures the time required to complete an operation such as downloading a Web page. It is an indicator of the users' experience with the system. **Throughput** measures the number of operations completed in a given period of time. It indicates the capacity or number of users that a system can handle. Throughput and latency are interrelated. An increase in either measure will lead to an increase in the other.
- ▶ **Reliability.** Reliability is an essential requirement for a successful system. System failures and downtime may lead to public embarrassment. When an EC application fails, business is interrupted and the company loses customers. System reliability can be enhanced through redundancy (i.e., back-up systems).
- ▶ **Security.** Security is one of the most important factors for the adoption and diffusion of EC. Data and information flow in EC, as well as stored data, may include private and/or proprietary information. Thus, a selected package must meet strict security requirements. Systems, communication, and data security must be addressed early in the design of EC applications and not after their implementation. In addition to technological solutions such as firewalls and encryption, physical and procedural security measures must also be enforced.

scalability

How big a system can grow in various dimensions to provide more service; measured by total number of users, number of simultaneous users, or transaction volume.

latency

The time required to complete an operation such as downloading a Web page.

throughput

The number of operations completed in a given period of time; indicates the number of users that a system can handle.



- ▶ List some of the major criteria to consider when deciding whether to buy or lease an EC application.
- ▶ Define latency.
- ▶ Define throughput.

12.5 THIRD-PARTY EC COMPONENTS AND SUITES

If a company opts to purchase its EC applications from a third-party vendor, two basic categories of software are available—electronic catalogs and EC suites.

ELECTRONIC CATALOGS

electronic catalog

The virtual-world equivalent of a traditional product catalog; contains product descriptions and photos, along with information about various promotions, discounts, payment methods, and methods of delivery.

merchant server software

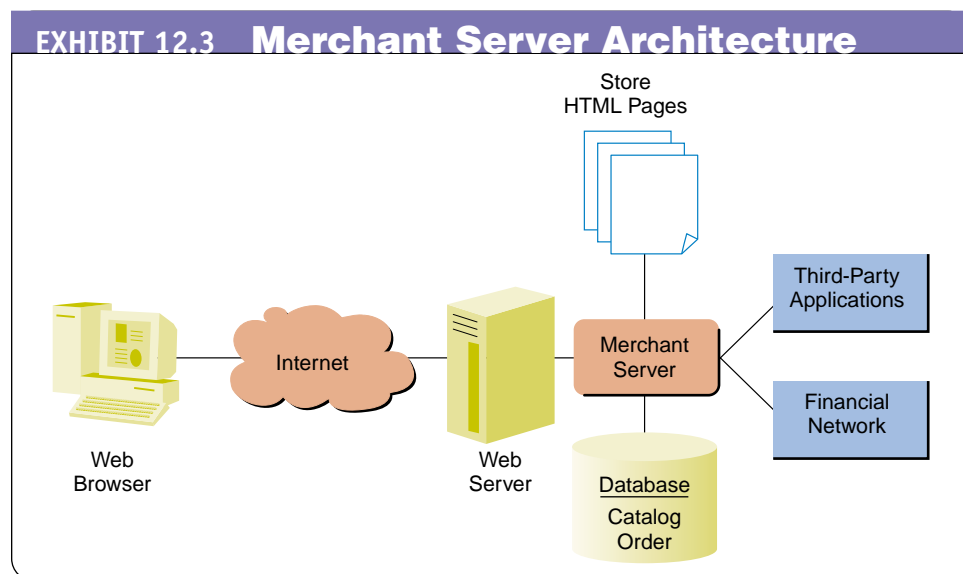
Electronic catalogs.

Electronic catalogs are the virtual version of traditional product catalogs. Like its paper counterpart, an electronic catalog contains written descriptions and photos of products, along with information about various promotions, discounts, payment methods, and methods of delivery. Electronic catalogs, also known as **merchant server software**, include features that make it simple and relatively inexpensive (usually less than \$10,000) to set up a catalog operation that has a straightforward pricing and product configuration.

This type of software commonly includes the following features:

- ▶ Templates or wizards for creating a storefront and catalog pages with pictures describing products for sale
- ▶ Electronic shopping carts that enable consumers to gather items of interest until they are ready for checkout
- ▶ Web-based order forms for making secure purchases (either through SSL encryption or the SET protocol)
- ▶ A database for maintaining product descriptions, pricing, and customer orders
- ▶ Integration with third-party software for calculating taxes and shipping costs and for handling distribution and fulfillment

Exhibit 12.3 outlines the major components in an electronic catalog or merchant server system. As shown in the figure, a single server is used to handle product pre-

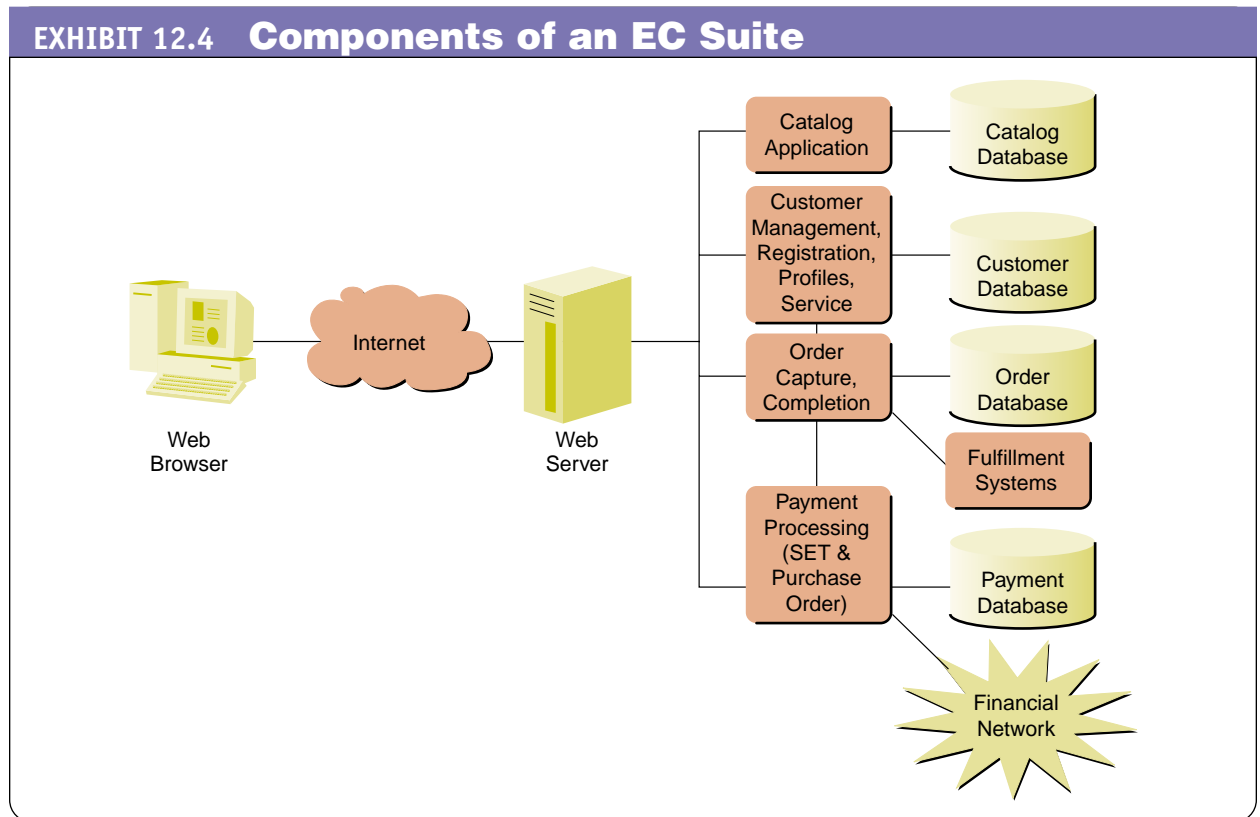


sentation, order processing, and payment processing (Treese and Stewart 1998). Likewise, in such systems a single database is used to store the catalog (i.e., product descriptions) and handle the details of customer orders. The pages of the electronic catalog are created dynamically from the product descriptions contained in the catalog database. For those merchants with only a few products for sale, there is no need to store the product descriptions in a database. Instead, the pages of the Web catalog can be created ahead of time.

EC SUITES

EC suites offer builders and users greater flexibility, specialization, customization, and integration in supporting complete front- and back-office functionality. In an EC suite, the functionality is distributed across a number of servers and databases instead of relying on a single server and database, as is done in electronic catalog and merchant server systems. The elements displayed in Exhibit 12.4 are indicative of the components contained in an EC suite, the processes supported by an EC suite, and the back-end databases and operational systems utilized by the processes.

Over the past few years, the EC suite marketplace has experienced a substantial amount of consolidation. Among the products that remain are InterWorld's Commerce Suite (interworld.com) and IBM's WebSphere Commerce Suite (ibm.com).



Source: Interworld.com.

InterWorld's Commerce Suite

InterWorld's Commerce Suite is in its fifth release. The suite can be used to develop, deploy, and maintain a variety of EC applications, including B2C, B2B, and exchanges of all sorts. The suite has a "process-oriented" architecture. The suite supports four major business processes:

- ▶ **Channel marketing.** Interactive catalog modules support extensive personalization facilities, including personalized product presentations; dynamic product pricing and personalized discounts and coupons; up-selling and cross-selling, pointing customers to alternative, complementary, or substitute products; product comparisons, alternatives, and recommendations based on buyer characteristics or past purchases; and buyer assistance for making product selections.
- ▶ **Order management.** Order management modules support the capture of the information required to place an order (order entry). They also finalize the details of the order, including the payment, shipping, inventory, and taxation (order processing). The modules also support billing and account management, including definition of preferences such as billing addresses, ship-to addresses, credit card information, credit limits, and credit card verification (payment).
- ▶ **Account management.** Account management modules enable online customer self-service. Buyers can use the modules to check an order or quote status, update their address book, and manager their profile information. These modules can also be used to manage backorders, cancellations, and returns in a complete, personalized self-service environment.
- ▶ **Customer service.** Customer service functions offer customers the ability to verify, edit, and change their profiles; review their orders; and review their payment history. They also enable customer service representatives to track customer comments and buying history in order to respond to customer requests.

IBM's WebSphere Commerce Suite

IBM's WebSphere Commerce suite is a comprehensive EC development platform designed to support B2C, B2B, or private exchange business models. The suite provides the following functions:

- ▶ Order management that optimizes movement of products through the supply chain
- ▶ Collaborative filters that enable an enterprise to better understand customers' buying patterns and preferences
- ▶ Portal capabilities that provide customers with personalized access to multiple commerce and noncommerce site applications
- ▶ Localization support that enables customized price, tax, and shipping calculations in the currency format and language dictated by the shopper's locale
- ▶ E-coupons that can be used by customers during online shopping
- ▶ Additional bundled products including WebSphere Catalog Manager and WebSphere Payment Manager

IBM's suite is built on open industry standards such as Java, JavaServer Pages, Enterprise JavaBeans (EJB), and XML. These standards make it easier to integrate with existing back-office transaction systems and databases.



- ▶ List the major features of an electronic catalog.
- ▶ Describe the basic business processes supported by InterWorld's Commerce suite.
- ▶ Describe the functions supported by IBM's WebSphere Commerce suite.

12.6 CONNECTING TO DATABASES AND OTHER ENTERPRISE SYSTEMS

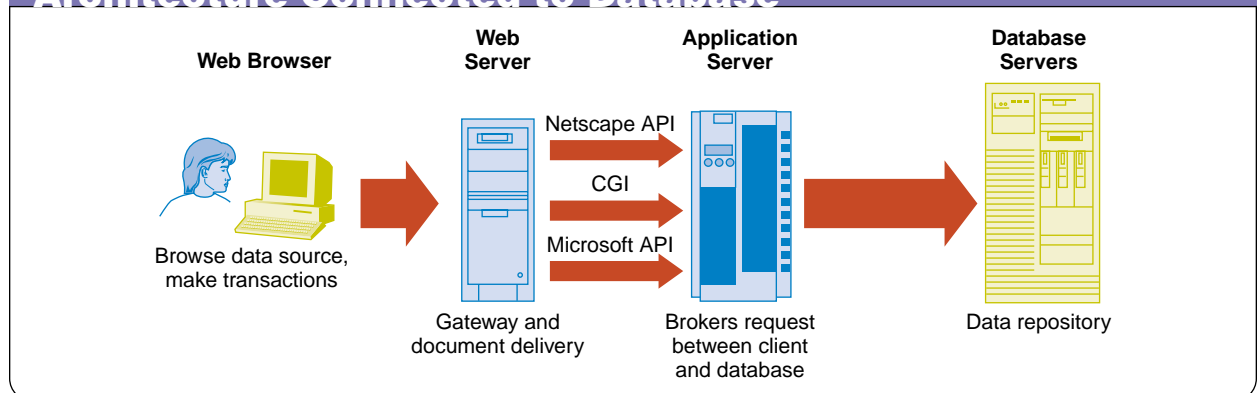
Like the UPS system described at the beginning of the chapter, virtually every EC application requires database access. For example, when you order a product online, the product description, inventory count, and order information are likely to be retrieved from and stored in one or more databases (see Exhibit 12.5). There are a variety of ways to connect an EC application to a back-end database. Today, most of these connections are accomplished via a **multitiered application architecture** like the one depicted in Exhibit 12.5. In this architecture there are four tiers:

1. A Web browser where data and information are presented to and data are collected from the end user.
2. A Web server that delivers Web pages, collects the data sent by the end user, and passes data to and from the application server.
3. An application server that executes business rules (e.g., user authorization), formulates database queries based on the data passed by the Web server, sends the queries to the back-end database, manipulates and formats the data resulting from the database query, and sends the formatted response to the Web server.
4. A database server where the data are stored and managed and database requests are processed.

multitiered application architecture

EC architecture consisting of four tiers: Web browsers, Web servers, application servers, and database servers.

EXHIBIT 12.5 Example of Multitiered Application Architecture Connected to Database



This separation of functions makes it easier to change any tier (or layer) without impacting the other layers. Thus, an application server can be designed to interface or communicate with a wide variety of databases and database management systems (e.g., Oracle, MS SQL Server, DB2).

In some cases, the data being accessed are stored in an existing (legacy) database (e.g., inventory or order databases). In these cases, it is better to tie the application server directly to the legacy database rather than duplicating the data in a database established solely for the EC application. This approach ensures that the data are up-to-date, that they are consistent across the applications accessing the data, that a minimum of storage space is used, and that there is only one database to create and maintain rather than two.

Several technologies can be used to integrate an EC application with a back-end database. All of the commercial electronic catalogs and EC suites have built-in integration capabilities. If a company wants to build its own database interface, a couple of options are available. First, all of the Web scripting languages (e.g., PHP, JSP, and ASP) have commands that simplify the process. More specifically, these scripting languages enable a programmer to build Web pages that can issue queries to a back-end (relational) database and process the database's response to the query. Second, a number of specialized application servers are available that simplify the task of integrating an EC application with one or more back-end databases. Among these specialized servers, BEA Inc. WebLogic Server is the market leader.

In addition to connecting to back-end databases, most EC applications also require integration with a variety of other systems—ERP, CRM, SCM, EDI, data warehouses, and other important internal systems—both inside and outside the company. Again, electronic catalogs and EC suites usually have built-in modules for integration with these systems. The integration can also be handled with a class of software called **enterprise application integration (EAI)**. These products focus on the integration of large systems. Vitria Technology (vitria.com), TIBCO (tibco.com), and webMethods (webmethods.com) are examples of companies that have offerings in the EAI arena.

enterprise application integration (EAI)

Class of software that integrates large systems.



- ▶ Describe the basic elements of a multitiered application architecture.
- ▶ List the ways in which an EC application can be connected to back-end databases and other transaction processing systems.

12.7 VENDOR AND SOFTWARE SELECTION

Few organizations, especially SMEs, have the time, financial resources, or technical expertise required to develop today's complex e-business systems. This means that most EC applications are built with hardware, software, hosting services, and development expertise provided by outside vendors. Thus, a major aspect of developing an EC application revolves around the selection and management of these vendors and their software offerings. Martin et al. (2000) identified six steps in selecting a software vendor and a package.

STEP 1: IDENTIFY POTENTIAL VENDORS

Potential application vendors can be identified from software catalogs, lists provided by hardware vendors, technical and trade journals, consultants experienced in the application area, peers in other companies, and Web searches.

These sources often yield so many vendors and packages that one must use some preliminary evaluation criteria to eliminate all but a few of the most promising ones from further consideration. For example, one can eliminate vendors that are too small or that have no track record or a questionable reputation. Also, packages may be eliminated if they do not have the required features or will not work with available hardware, operating system, communications network, or database management software.

STEP 2: DETERMINE THE EVALUATION CRITERIA

The most difficult and crucial task in evaluating a vendor and a packaged system is to determine a weighted set of detailed criteria for choosing the best vendor and package. Some areas in which detailed criteria should be developed are characteristics of the vendor, functional requirements of the system, technical requirements the software must satisfy, amount and quality of documentation provided, and vendor support of the package.

These criteria should be documented in a **request for proposal (RFP)**, which is sent to potential vendors inviting them to submit a proposal describing their software package and how it would meet the company's needs. The RFP provides the vendors with information about the objectives and requirements of the system, the environment in which the system will be used, the general criteria that will be used to evaluate the proposals, and the conditions for submitting proposals. It may also request a list of current users of the package who may be contacted, describe in detail the form of response that is desired, and require that the package be demonstrated at the company's facilities using specified inputs and data files.

request for proposal (RFP)

Notice sent to potential vendors inviting them to submit a proposal describing their software package and how it would meet the company's needs.

STEP 3: EVALUATE VENDORS AND PACKAGES

The collective responses to an RFP generate massive volumes of information that must be evaluated to determine the gaps between the company's needs (as specified by the requirements) and the capabilities of the vendors and their application packages. Often, the vendors and packages are given an overall score by assigning an importance weight to each of the criteria, ranking the vendors on each of the weighted criteria (say 1 to 10), and then multiplying the ranks by the associated weights. A short list of potential suppliers can be chosen from those vendors and packages with the highest overall scores.

STEP 4: CHOOSE THE VENDOR AND PACKAGE

Once a short list has been prepared, negotiations can begin with vendors to determine how their packages might be modified to remove any discrepancies with the company's desired EC application. Thus, one of the most important factors in the decision is the additional development effort that may be required to tailor the system to the company's needs or to integrate it into the company's environment.

Additionally, the opinions of the users who will work with the system and the IT personnel who will have to support the system have to be considered.

STEP 5: NEGOTIATE A CONTRACT

The contract with the software vendor is very important. Not only does it specify the price of the software, but it also determines the type and amount of support to be provided by the vendor. The contract will be the only recourse if the system or the vendor does not perform as specified. Furthermore, if the vendor is modifying the software to tailor it to the company's needs, the contract must include detailed specifications (essentially the requirements) of the modifications. Also, the contract should describe in detail the acceptance tests the software package must pass.

Contracts are legal documents, and they can be quite tricky. Experienced contract negotiators and legal assistance may be needed. Many organizations have software purchasing specialists who assist in negotiations and write or approve the contract. They should be involved in the selection process from the start. If an RFP is used, these purchasing specialists may be very helpful in determining its form and in providing boilerplate sections of the RFP.

STEP 6: ESTABLISH A SERVICE LEVEL AGREEMENT

Service level agreements (SLAs) are formal agreements regarding the division of work between a company and its vendors. Such divisions are based on a set of agreed-upon milestones, quality checks, “what-if” situations, how checks will be made, and what is to be done in case of disputes. If the vendor is to meet its objectives of installing EC applications, it must develop and deliver support services to meet these objectives. An effective approach to managing SLAs must achieve both facilitation and coordination. SLAs do this by (1) defining the partners' responsibilities, (2) providing a framework for designing support services, and (3) allowing the company to retain as much control as possible over their own systems.

service level agreement (SLA)

A formal agreement regarding the division of work between a company and its vendors.



- ▶ List the major steps in selecting an EC application vendor and package.
- ▶ Describe a request for proposal (RFP).
- ▶ Describe a service level agreement (SLA).

access log

A record kept by a Web server that shows when a user accesses the server; kept in a common log file format, each line of this text file details an individual access.

12.8 USAGE ANALYSIS AND SITE MANAGEMENT

To improve EC Web sites, it is advisable to monitor what customers are doing (usage analysis). Both B2C and B2B Web sites require a thorough understanding of the usage patterns of their sites—the who, what, where, when, and how.

LOG FILES

Every time a user accesses a Web server, the server logs the transaction in a special access log file. **Access logs** are text files. Each line of the file details an individual access. Regardless of the type of Web server, access logs use a common log file for-

mat. This makes them easy to analyze and compare. Because log files can become quite voluminous, it is hard to analyze the accesses by hand. For this reason, most Web server vendors provide “free” software for analyzing access log files. Commercial products that provide more sophisticated log analyses are also available (e.g., WebTrends).

Access logs provide a variety of statistics that can be used for analyzing and improving marketing and advertising strategies. Among the more valuable statistics are:

- ▶ **Pageviews by time slot.** Pageviews allow frequent review of the number of site accesses. Group pageviews by “time bucket” (time slot) also enables the company to ascertain the time slots, such as morning, afternoon, or evening, during which customers visit the site.
- ▶ **Pageviews by customers’ logging-in status.** This information helps determine whether requiring customers to log in is worthwhile or not. For instance, if the number of pageviews of customers who log in is substantially greater than those who do not, the company may find the login requirement effective and worthwhile.
- ▶ **Pageviews by referrers.** Some customers are drawn or referred to the site by clicking on banners or links on other Web sites. Knowing the source of such referrers is useful for assessing the effectiveness of the location of banners, and customers’ interest can also be determined from the nature of the Web site with those banners.
- ▶ **Pageviews by visitor’s hardware platform, operating system, browser, and/or browser version.** These types of pageviews allow the company to obtain information on the hardware platform (e.g., Macs or PCs) and browser type (e.g., Internet Explorer or Netscape) used by the viewer.
- ▶ **Pageviews by visitor’s host.** This type of pageview provides information on the customers’ host site. Knowing where customers are coming from can enable the company to target potential customers via popular hosts, such as AOL.

Some of the marketing and business questions to which these statistics can be applied are listed in Exhibit 12.6 (on p. 12-22).

E-COMMERCE MANAGEMENT TOOLS

Managing the performance of a Web site is a time-consuming and tedious administrative task. Several vendors offer suites of products or individual packages that can assist with the management process. A detailed list of vendors is provided by Hower (2002).

One of the more prominent vendors of IT and Web management tools is BMC Corp. (bmc.com). BMC offers the following products:

- ▶ **Patrol for e-business management.** This package includes Patrol for Internet Services for measuring Web response time, Patrol for Firewalls for firewall administration, and Patrol for Microsoft or Netscape application servers.
- ▶ **MainView for e-business management.** This package includes MainView for WebSphere for managing mainframe-based EC applications, MainView

EXHIBIT 12.6 Areas of Usage Analysis and Sample Business Questions for Online Stores

Area of Analysis

Overall Store Performance

Advertising

External Referrals (from others to your site)

Shopper Segmentation

Product Grouping

Promotions and Recommendations

Shopping Metaphor

Design Features

Product Assortment

Business Questions

- ▶ What is the sales value for a specific period of time, say, 1 week?
- ▶ What is the number of customer visits for the day?
- ▶ What is the store conversion rate of the week?
- ▶ What is the sales value index for the week?

- ▶ Which banner ads are pulling in the most traffic?
- ▶ How many sales are driven by each banner ad?
- ▶ What products do shoppers select from a particular banner?
- ▶ What is the conversion rate for each banner ad?

- ▶ Which portal sites are pulling in the most traffic?
- ▶ Which are generating the most sales?
- ▶ How many sales are generated by each referral site/search engine?
- ▶ What products do shoppers from a particular portal site purchase?

- ▶ How many visitors are from a specific domain?
- ▶ What is the distribution of first-time vs. repeat shoppers?
- ▶ What characterizes shoppers of a particular set of products?
- ▶ What characterizes shoppers who abandon shopping baskets?

- ▶ How much do cross-sells/up-sells contribute to gross revenue?
- ▶ What are the best performing cross-sell pairs? Worst?
- ▶ What is the overall conversion rate for cross-sells/up-sells?

- ▶ How much do promotions contribute to gross revenue?
- ▶ Which promotions are generating the most sales?
- ▶ What is the overall conversion rate for promotions?
- ▶ What is the overall conversion rate for recommendations?
- ▶ At what levels in site hierarchy are the best promotions located?

- ▶ What generates the most sales value: searching or browsing?
- ▶ How much does searching contribute to gross revenue?
- ▶ What is the conversion rate for searching?

- ▶ What are the features of links customers most frequently click?
- ▶ What are the features of links customers most frequently buy from?
- ▶ What parts of pages do customers most frequently buy from?
- ▶ Do products sell better in the upper-left corner?

- ▶ What are the top sellers for the week?
- ▶ What is the conversion rate for a particular department?
- ▶ How is a product purchased: purchase frequency and quantity?
- ▶ What characterizes the products that end up being abandoned?
- ▶ How much of the sales of each product are driven by searching?

for Network Management for monitoring mainframe network connections, and MainView for Systems Management for systems administration.

- ▶ **Service assurance center for e-business.** This package includes a combination of methodology, products, and services designed to optimize the performance and availability of business applications.



- ▶ List some of the statistics provided by an access log.
- ▶ Describe some of the uses of an access log.

MANAGERIAL ISSUES

Some managerial issues related to this chapter are as follows.

1. **What is our business perspective?** When one thinks of the Web, one immediately thinks of the technology. But some of the most successful sites on the Web rely on basic technologies—freeware Web servers, simple Web page design, and few bells and whistles. What makes these sites successful is not the technology, but their owners' understanding of how to meet the needs of their online customers.
2. **Do we have a systematic development plan?** The cost of developing and maintaining even a small EC site can be substantial. To ensure success, development and maintenance issues need to be approached systematically, just like any other IT development project. Within this plan, the specification of the EC architecture is crucial. If the architecture is wrong, the entire project is at risk.
3. **Insource or outsource?** Many large-scale enterprises are capable of running their own EC Web sites. However, EC Web sites may involve complex integration, security, and performance issues. For those companies venturing into the EC arena, a key issue is whether the site should be built in-house (insourced), thus providing more direct control, or outsourced to a more experienced provider. Outsourcing services, which allow companies to start small and evolve to full-featured functions, are available through many ASPs, ISPs, telecommunication companies, Internet malls, and software vendors that offer merchant server and EC applications.
4. **How should we choose a vendor/software?** Because most EC applications are built from either packaged applications and components or outsourced to a third party, the success of the EC applications rests on choosing the best vendor and package. Like any other part of the development process, a detailed list of selection criteria are needed for the selection process.
5. **Have we analyzed the data?** All EC sites provide the means to gather data about system usage. These data should be analyzed frequently to modify and redesign an existing site to better meet the needs of current and prospective customers and users. This analysis can also be used to personalize the experience of these same users.

SUMMARY

In this chapter, you learned about the following EC issues as they relate to the learning objectives.

- 1. The major steps in developing an EC application.** Because of their cost and complexity, EC sites need to be developed in a systematic fashion. The development of an EC site should proceed in steps. First, the EC architecture is defined. Next, a decision is made whether to build, buy, or outsource the development. Third, the system is installed, tested, and deployed. Finally, the system goes into maintenance mode, with continual changes being made to ensure the system's continuing success.
- 2. The major EC applications and their major functionalities.** Every type of EC application has a long list of functional requirements. Fortunately, most of these requirements can be met by packaged applications. Online storefronts can be developed with the aid of electronic catalog or merchant server software. Similarly, B2C, B2B, and exchange applications of all sorts can be constructed from more advanced EC suites.
- 3. The major EC application development options along with their benefits and limitations.** EC sites and applications are rarely built from scratch. Instead, enterprises either buy a packaged EC suite and customize it to suit their needs or they outsource the development to a third party. The selection of one option over another should be based on a systematic comparison of a detailed list of requirements that examines flexibility, information needs, user friendliness, hardware and software resources, and so on.
- 4. EC application outsourcing options.** If an enterprise elects to outsource the development and maintenance of its EC site and applications, a number of alternatives are available. EC applications can be hosted by ASPs, ISPs, or by a telecommunication company. An enterprise can rely on an existing e-marketplace or exchange. An online storefront can be hosted by an Internet mall. Or, an enterprise could enter into a joint development agreement with a venture partner or a consortium. Again, the choice depends on the functional requirements of the EC site or application.
- 5. The major components of an electronic catalog and EC application suite.** An online storefront has the same requirements as a brick-and-mortar storefront. Simple sites can be built from packaged electronic catalog or merchant server software. More complex online storefronts and other types EC sites (e.g., B2B, exchanges, etc.) can be built from comprehensive EC suites such as InterWorld's Commerce Suite or IBM's WebSphere Commerce Suite.
- 6. Methods for connecting an EC application to back-end systems and databases.** Virtually every EC application requires access to back-end relational databases and other transaction systems (e.g., ERP, SCM, CRM, etc.). Integration can be accomplished in a variety of ways, including using integration modules supplied with electronic catalog or EC suite packages, customizing the integration with a Web scripting language (e.g. PHP, ASP, or JSP), employing specialized application servers, or employing a full-blown EAI tool.
- 7. Criteria used in selecting an outsourcing vendor and package.** A systematic process should be used in selecting a third-party tool or outsourcing service. Among the key steps in making the selection are: (1) identifying potential vendors and packages, (2) detailing the evaluation criteria, (3) using the criteria to produce a short list of possible vendors, (4) choosing a candidate from the short list, (5) negotiating the deal and modifications needed to meet overall application needs, and (6) establishing a SLA to define who is responsible for specific aspects of the development and maintenance and the quality metrics for the services to be rendered.
- 8. The value and uses of EC application log files.** Most EC applications produce log files of detailed system usage. The data in these files

can be analyzed with an eye toward modifying the application's content and flow. In this way, the application can be better aligned with the enterprise's marketing and advertising strategies. In the same vein, the application can be adjusted to meet users' needs.

9. The importance and difficulties of EC application maintenance. Maintenance of an ongoing application requires as much, if not more, time and effort than the original development and installation. Several vendors offer tools to assist with the process.

KEY TERMS

Acceptance testing,	p. 12-4	Insourcing,	p. 12-8	Scalability,	p. 12-13
Access log,	p. 12-20	Integration testing,	p. 12-4	Service level	
Application service		Latency,	p. 12-13	agreement	
provider (ASP),	p. 12-3	Merchant server		(SLA),	p. 12-20
EC architecture,	p. 12-4	software,	p. 12-14	Throughput,	p. 12-13
Electronic catalog,	p. 12-14	Multitiered application		Unit testing,	p. 12-4
Enterprise application		architecture,	p. 12-17	Usability testing,	p. 12-4
integration		Request for proposal			
(EAI),	p. 12-18	(RFP),	p. 12-19		

DISCUSSION QUESTIONS

1. Discuss the advantages of leasing an application over purchasing one.
2. A large company with a number of products wants to start selling on the Web. Should it use a merchant server or an EC application suite? Assuming it elects to use an EC application suite, how would you determine whether the company should outsource the site or run it themselves?
3. A large chemical manufacturing company is interested in starting an online exchange. What are some of the ways it could achieve this goal?
4. A firm decides to make its EC Web site more dynamic by tying its application to a back-end database. What are some of the ways in which the firm could accomplish this task?
5. An enterprise wants to modify its EC site so that it conforms more closely with its overall business strategies. What sorts of online data are available for this purpose? What types of business strategy questions can be addressed by these data?
6. In what ways do you think a Web site's log files violate your privacy?
7. You have decided to use a third-party application to develop and deploy a sell-side B2B site. Create a checklist for determining which third-party EC application products will best meet your application requirements.

INTERNET EXERCISES



1. Access the Choice Mall Web site (choicemail.com). Visit some of the online stores in the mall. What are the functionalities of the mall? What are some of the benefits of the online mall to the participating vendors? To shoppers? Do you think a shopper is better off using an online mall or using a search engine such as AltaVista to locate a store providing a product of interest? In what ways could Choice Mall improve the chances that buyers will make return visits?
2. Visit a large online storefront of your choice. What functions does it provide to shoppers? In what ways does it make shopping easy? In what ways does it make shopping more enjoyable? What support services does it provide?
3. Enter hotwired.lycos.com/webmonkey and find the tracking tutorials. What is the difference between a “hit” and a “pageview”? Write a summary of the three tutorials—gathering data, using databases, and using pageviews.
4. Go to the WebTrends site (webtrends.com). What types of information does its Analysis Suite provide? How can this information be used to improve a Web site? What types of tracking information are not provided by this suite? (See Discussion Question 3.)
5. Visit IBM’s site (ibm.com). Find its WebSphere product. Read some of the recent customer success stories. What makes this software so popular?
6. Visit the Microsoft Web site (microsoft.com). Find its Biztalk product. What kind of software is this? What role could it play in an EC application?
7. Go to covisint.com and also to EC Application Case 6.1 in this book (page 260). What is Covisint? Who are the partners involved in Covisint? What types of B2B functionality does it provide? Based on the press releases at Covisint’s Web site, has Covisint been successful? Why or why not?

TEAM ASSIGNMENTS AND ROLE PLAYING

1. Select a series of Web sites that cater to the same type of buyer (e.g., several Web sites that offer CDs or computer hardware). Divide the sites among the teams and ask each team to prepare an analysis of the different sorts of functions provided by the sites, along with a comparison of the strong and weak points of each site from the buyer’s perspective.
2. Several vendors offer products for creating online stores. The Web sites of these vendors usually list those online stores using their software (customer success stories). Assign each team a number of vendors. Each team should prepare reports comparing the similarities and differences among the vendors’ sites and evaluating the customers’ success stories. Do the customers take advantage of the functionality provided by the various products?
3. As a team, explore the desired capabilities of various EC applications (B2B, B2C, auctions, portals G2C, etc.). Look at the capabilities of these applications and at their functionalities, and then compare the two. (See Section 12.3 for a list of functionalities.) If the functionalities of the applications are not sufficient, explain what additional functionalities are needed.

REFERENCES

- Carmichael, M. "2001 Median Prices for Full-Site Development." btobonline.com/webPriceIndex (May 14, 2001).
- DeWire, D. *Thin Clients: Delivery Information over the Web*. New York: McGraw-Hill, 1998.
- Hower, R. "Web Site Test Tools and Site Management Tools." softwareqatest.com/qatweb1.html (April 20, 2002).
- Kendall, K. E., and J. E. Kendall. *Systems Analysis and Design*, 4th ed. Upper Saddle River, NJ: Prentice Hall, 1999.
- Kern, T., and J. Kreijger. "An Exploration of the ASP Outsourcing Option." *Proceedings, 34th HICSS*, Maui, HI (January 2001).
- Koch, C. "Boy, That Was Fast." *CIO Magazine* (November 15, 2000).
- Martin, E. W., et al. *Managing Information Technology*. Upper Saddle River, NJ: Prentice Hall, 2000.
- Treese, G. W., and L. C. Stewart. *Designing Systems for Internet Commerce*. Reading, MA: Addison-Wesley, 1998.

REAL-WORLD CASE

WHIRLPOOL'S TRADING PORTAL

Whirlpool (whirlpool.com) is a world leader in the manufacture and marketing of major home appliances. Competing in a \$75 billion annual, global market, the company considers its distributors and partners to be critical players in its continual quest to maintain industry leadership. As a consequence, Whirlpool is constantly trying to improve the efficiency of its operations while still providing top-notch service to members of its sales chain.

Until recently, one area of concern was its relationships with its middle-tier trade partners. These partners represent 25 percent of Whirlpool's trading partners but only 10 percent of its revenues. Because of their size, Whirlpool was unable to dedicate system-to-system connections with these partners. Instead, partners had to submit their orders by phone or fax.

Wanting to infuse greater efficiency into this process, Whirlpool turned to EC, developing a B2B trading-partner portal that enables its middle-tier sellers to order online. To make the portal work, the company needed to integrate it with its SAP R/3 inventory system and Tivoli systems management tools. Whirlpool conducted a vendor and product analysis and decided to use IBM integration tools.

Following the guidelines of the IBM Application Framework for e-business, Whirlpool built its portal with the IBM WebSphere Application Server, Advanced Edition, IBM Net.Commerce (new part of the IBM WebSphere Commerce Suite family), IBM HTTP Server, IBM VisualAge for Java, and IBM Commerce Integrator with IBM MQSeries.

Through the portal, called Whirlpool Web World, several thousand middle-tier trade partners select the goods they want to order by checking off the appropriate SKUs and indicating quantities. Aside from appliance ordering, they can also log on to the password-protected site to track the status of their orders. The portal has cut the cost per order to under \$5—a savings of at least 80 percent. Whirlpool has also gained an unexpected benefit—an extendable EC platform that it has leveraged for other applications.

Whirlpool's B2B portal is now in its second generation (as of summer 2001). The first-generation portal was developed with low-level products, giving the company a chance to test the Web waters. However, the portal took off faster than the company expected. In its first 3 months, the amount of revenue that flowed through the portal was what Whirlpool had planned for the first 12 months. The investment paid for itself in 8 months.

With the success of its first-generation trading-partner portal, Whirlpool was ready to migrate the solution to a bigger, more scalable, and easier-to-manage platform. At the same time, the company was also implementing SAP R/3 for order entry. Thus, it was important for its second-generation portal to integrate with SAP R/3.

For its second-generation portal, Whirlpool wanted to partner with a vendor that would be around a while. It again checked out IBM, plus a few others. IBM was selected because IBM has worked

with Whirlpool on joint product development as well as with Whirlpool's ERP system design and architecture. In addition, when Whirlpool talked to financial analysts, it found that an overwhelming number of *Fortune* 100 companies use IBM e-business solutions. Finally, Whirlpool saw that IBM is on the cutting edge of industry Web standards such as Java and XML. After committing to IBM, Whirlpool also decided to develop its e-business platform following the Application Framework for e-business, taking advantage of its rapid development cycles and associated cost reductions. In developing the first-generation portal, Whirlpool had to build certain functionalities on its own, because the desired functionalities were not included in the tools supplied by IBM. A year and a half later, the suite of IBM tools included those functionalities, which enabled Whirlpool to get its applications to market much faster.

Source: *IBM.com* (2002).

Questions

1. From Whirlpool's point of view, what kind of a B2B application is this: e-procurement, sell-side, collaborative commerce, or other? Justify your answer.
2. Why did Whirlpool decide to utilize third-party applications with its second-generation portal? Explain in detail.
3. Imagine you were in charge of selecting the third-party applications to be used with the second-generation portal. What sorts of criteria would you use in making your choice? How did IBM's WebSphere meet these criteria?
4. How can Whirlpool leverage this application with other B2B processes?