Content
Opening Case: Developing a Web 2.0 Platform to Enable Innovative Market Research at Del Monte
18.1 A Five-Step Approach to Developing an E-Commerce System
18.2 Development Strategies for E-Commerce Major Applications
18.3 Selecting a Development Method
18.4 Technology Support: From Blogger.com to Infrastructure Services
18.5 Vendor and Software Selection
18.6 Connecting to Databases and Other Enterprise Systems
18.7 Usage Analysis and Site Management
Managerial Issues
Closing Case: TomTom Develops a New E-Commerce Infrastructure

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

1. Discuss the major steps in developing an EC system.
2. Describe the major EC development strategies and list their major advantages and disadvantages.
3. List the various EC application development methods along with their benefits and limitations.
4. Discuss various EC application outsourcing options, including application service providers (ASPs), software as a service (SaaS), and utility computing.
5. Discuss the major EC software packages and EC application suites.
6. Discuss the value and technical foundation of Web Services and Web 2.0.
7. Understand service-oriented architecture (SOA) and virtualization and their relationship to EC application development.
8. Describe the criteria used in selecting software vendors and packages.
9. Describe various methods for connecting an EC application to back-end systems and databases.
10. Understand the value and uses of EC application log files.
11. Discuss the importance of usage analysis and site management.
The Problem
Utilizing the latest innovations in technology can often give companies a competitive advantage, particularly if they are the first to use that technology in the marketplace. Even if a company is not the first to market, keeping up with technology trends and capabilities often becomes a necessity to thrive in the business world. One key to using technology to create a competitive advantage is to observe what strides are being made in various industries or sectors, and then develop that idea or tool to be applicable to the business at hand.

One major trend in today’s society is social networking. Sites such as Facebook and MySpace have caused a phenomenon—especially among teenagers and young adults. Social networks are powerful tools that allow people to build or maintain relationships with others around the world. Now the idea has evolved to incorporate social networking into businesses. One such company to introduce social networking into its business environment is Del Monte.

Del Monte is known mostly for manufacturing canned fruits and vegetables. In addition to these products, Del Monte also produces 9Lives, Gravy Train, Meow Mix, Milk-Bone, College Inn, Contadina, and Starkist. Once Del Monte made the decision to experiment with social networking, it had to decide how to best implement it to support its diverse product line.

The Solution
Even though Del Monte executives could see the value of a social networking system, the IT department was not equipped to handle the implementation of a social network. So, Del Monte recruited MarketTools, a firm that specializes in market collection and analysis and also has experience with Web 2.0 capabilities.

With the help of MarketTools, Del Monte’s Web site now offers a platform for customers to chat and blog about products, or share tips and recipes. MarketTools can search millions of blogs in order to recognize key ideas and trends among consumers. The information that customers share with each other is collected and analyzed by MarketTools. They then team with another company, such as Umbria (a division of J.D. Power and Associates and a pioneer in drawing market intelligence from the online community), to assist them in further analyzing and profiling the information collected and stored in data warehouses. By utilizing social networks, Del Monte can conduct market research much more efficiently. Focus groups have become a thing of the past. All that is required now is to sift through the vast amount of customer information that is collected in cyberspace.

The Result
Del Monte used this method of market research when developing a dog treat, Snausages Breakfast Bites. By paying attention to customer blogs and by posting questions to customers, Del Monte concluded that owners of small dogs would be the major purchasers of Snausages Breakfast Bites. Del Monte depended on the dog lovers group for guidance in the development of this product. By doing so, a smaller treat was produced, packaging decisions were revised, product cycle time was reduced to 6 months, and Del Monte was able to cut costs.

Sources: Steel (2008) and Greengard (2008).

WHAT WE CAN LEARN . . .
Market research is imperative to the success of a business. The more a company knows and understands about its customer base, the better it can serve them. Del Monte has proven that market research no longer needs to be conducted in a sterile room where sample groups are formed and asked a series of questions. Now the data are available through social networks, and in most cases, customers are more candid with one another than they would be in the traditional research setting. Since Del Monte does not necessarily disclose which sites and blogs are being used to conduct research, it is important for consumers to recognize that some companies are using this type of approach and to be aware that information shared on blogs may be used for marketing purposes. Clearly, Del Monte has made some pioneering efforts in using social networks. This case demonstrates one way in which a new information system can be developed by combining new technologies, such as Web 2.0 and social networking, to create an innovative technique that enables companies to gain a competitive advantage. These, and other related issues, are the focus of this chapter.
### 18.1 A FIVE-STEP APPROACH TO DEVELOPING AN E-COMMERCE SYSTEM

Once it has been determined that a business can benefit from an online presence, the business type, the product line, the business's organization, and the budget dictate what functionality the Webstore should have and how the Web site should be developed. Companies can choose from a number of different types of Web sites, including B2C, B2B, exchanges, and the like. Sites of a particular type (e.g., retailer, provider of business services, manufacturer, distributor/wholesaler, media, travel/entertainment) usually use the same underlying applications and provide similar sorts of functionality. Although this simplifies the task of creating the underlying application architecture, the site requirements must still be considered carefully. Before discussing the best approach to developing a site, it would be useful to review previous chapters to consider the major characteristics, functionalities, and requirements of the EC system being developed. Some typical capabilities of Webstores are shown in Exhibit 18.1.

A well-developed Web site not only adds to the value of products or services being offered, but it also enhances the worth of the company. Therefore, once you have a clear understanding of the e-commerce system requirements and consider all of the elements of the e-commerce system (see Exhibit 18.2), it is important that a firm choose the correct development strategy to obtain the greatest return on its investment. The diversity of e-business models and applications, which vary in size from small stores to global exchanges, requires a variety of development methodologies and approaches.

| EXHIBIT 18.1 Capabilities Needed by Webstore Users |
|--------------------------------------------------|--------------------------------------------------|
| **Buyers need the ability to:**                  | **Sellers need the ability to:**                 |
| • Discover, search for, evaluate, and compare   | • Provide access to a current catalog of product  |
|   products for purchase using e-catalogs.       |   offerings, allowing prospective buyers to     |
| • Select products to purchase and negotiate     |   analyze and evaluate the offering.            |
|   or determine their total price.               | • Provide an electronic shopping cart in which   |
| • Place an order for desired products using a    |   buyers can assemble their purchases.          |
|   shopping cart.                               | • Verify a customer’s credit and approve the    |
| • Pay for the ordered products, usually         |   customer’s purchase.                          |
|   through some form of credit.                  | • Process orders (back-end services).           |
| • Confirm an order, ensuring that the desired   | • Arrange for product delivery.                 |
|   product is available.                         | • Track shipments to ensure that they are       |
| • Track orders once they are shipped.          |   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 postsale support.          |
|                                                 | • Create the capability for cross-sell and     |
|                                                 |   up-sell.                                    |
|                                                 | • Provide language translation if needed.       |
|                                                 | • Measure and analyze the traffic at the site   |
|                                                 |   to modify and maintain the various applications. |
For example, small Webstores with a few key components can be developed with HTML, Java, Web 2.0 or another programming language. They also can be implemented with commercial packages, leased from an application service provider (ASP), or purchased from a site builder. Larger or special EC applications can be developed in-house or outsourced (see the opening case). 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. Therefore, although the process of building EC systems can vary, in many cases, it tends to follow a fairly standard format.

The traditional *systems development life cycle* (SDLC) systematically leads developers through six analysis and design stages: problem identification, analysis, logical design, physical design, implementation, and maintenance. The SDLC is the basis for development of the majority of traditional business systems (see Whitten and Bentley 2007 for more details on this approach). However, innovative new software and hardware are enabling a move to a more streamlined approach to e-commerce development, as discussed in Case 18.1. Exhibits 18.3 and 18.4 show the five major steps needed to develop a typical e-commerce application.
STEP 1: IDENTIFYING, JUSTIFYING, AND PLANNING EC SYSTEMS

EC applications, like all other information systems, are usually built to enable one or more business processes. Consequently, their planning must be aligned with that of the organization’s overall business plan and the specific processes involved. Always remember that existing processes may need to be restructured to take full advantage of the benefits of the supporting IT. Furthermore, each application must be carefully analyzed, using different methods, such as the methodology discussed in the opening case, to ensure that it will have the needed functionality to meet the requirements of the business processes and the users and that its benefits will justify its cost (see Chapter 5). Both of these activities may be complex, but they are necessary, especially for systems that require high investment to acquire, operate, and maintain. The output of this step is a decision to go with a specific application, with a timetable, budget, and assigned responsibility. This first step is typically performed in-house (with consultants if needed). All other steps can be completed either in-house or outsourced.

STEP 2: CREATING AN EC ARCHITECTURE

An EC architecture is a plan for organizing the underlying infrastructure and applications of a site. The plan specifies the following:
- Information and data required to fulfill the business goals and vision
- Application modules that will deliver and manage the information and data

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


**STEP 3: SELECTING A DEVELOPMENT OPTION**

EC applications can be developed through several alternative approaches that will be discussed in detail in Section 18.3. The major options are:

- Build the system in-house.
- Have a vendor build a customized system.
- Buy an existing application and install it, with or without modifications, by yourself or through a vendor.
- Lease standard software from an application service provider (ASP), lease as a service (SaaS), or lease via utility computing.
- Enter into a partnership or alliance that will enable the company to use someone else’s application.
- Join a third-party e-marketplace, such as an auction site, a bidding (reverse auction) site, or an exchange, that provides needed capabilities to participants (e.g., Yahoo! Store).
- Use a combination of approaches.

The criteria for selecting from among the various options are presented in Section 18.3. Once an option is chosen, the system can be developed. At the end of this step, an application is ready to be installed and made available. No matter what option is chosen, it is important to keep in mind that all the different applications that support the various capabilities of the Web site must be coordinated and considerable collaboration between developers and users is necessary. In addition, there is a strong possibility that the firm will work with vendor(s) and/or software provider(s). In this case, the firm will need to manage its vendor relationships (see Section 18.7).

**STEP 4: INSTALLING, TESTING, INTEGRATING, AND DEPLOYING EC APPLICATIONS**

Once a system has been developed, the next step involves getting the application up and running in the selected hardware and network environment. 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 is in stock. To do this,
the ordering system would need to be connected to the inventory system. Details of the connection process are supplied in Section 18.6. This step can be done in-house or outsourced.

At this point, the modules that have been installed need to be tested using a series of different tests:

- **Unit testing.** Test each module one at a time.
- **Integration testing.** Test the combination of modules acting in concert.
- **Usability testing.** Test the quality of the user’s experience when interacting with the site.
- **Acceptance testing.** Determine whether the site meets the firm’s original business objectives and vision.

Once all of the Web site applications pass all of the tests, they can be made available to the end users. At this stage, developers may need to address issues such as conversion strategies, training, and resistance to change.

**STEP 5: OPERATIONS, MAINTENANCE, AND UPDATES**

It typically takes more time, effort, and money to operate and maintain a site than it does to build and install it in the first place. To enjoy continual usage, a site needs to be updated frequently. 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 should be modified or eliminated from the site.

**MANAGING THE DEVELOPMENT PROCESS**

The development process can be fairly complex and must be managed properly. For medium-to-large applications, a project team is usually created to manage the process and the vendors. Collaboration with business partners also is critical. As shown in various chapters of this book, some e-business failures are the result of a lack of cooperation by business partners. For example, a firm can install a superb e-procurement system, but if their vendors do not use it properly, the system will collapse. Projects can be managed with project management software (see examples of various project management software at en.wikipedia.org/wiki/List_of_project_management_software). Best practice management also includes periodic evaluations of system performance. Standard project management techniques and tools are useful for this task. Finally, do not rule out the possibility that implementing an EC project may require restructuring one or more business processes.

**Section 18.1 REVIEW QUESTIONS**

1. Examine 10 different Web sites and choose your 5 favorites.
2. Go to the Web site of each of the developers/Webmasters of your 5 favorite Web sites. What expertise do they profess to have? What projects have they completed? Would you feel comfortable hiring their services?
3. List the major steps in developing an EC application.
4. Define the various types of testing used during the EC development process.

**18.2 DEVELOPMENT STRATEGIES FOR E-COMMERCE MAJOR APPLICATIONS**

If the desired Web site is relatively simple, a firm may decide to build the Web site itself. However, the firm must ask a few questions: Is the firm capable of developing the site? Does the firm have access to the proper tools to create the pages? If the firm does not have these capabilities, it is usually best to turn over the task to a professional developer.
The ideal developer is one who can design a site with the correct look and feel, who has an in-depth knowledge of search engine optimization, and who is able to correctly handle any complex coding that may be required. A useful site for finding an experienced Web site designer is WebDesigners-Directory (webdesigners-directory.com). Other resources for developing a Web site are available at Sell IT! (sellontheweb.com). The Microsoft Small Business Center (microsoft.com/smallbusiness/resources/technology/ecommerce/5_common_e_commerce_site_mistakes.mspx) offers some tips for avoiding five common e-commerce mistakes associated with overall site design and infrastructure. Regardless of the complexity of the site, three basic options for developing an EC Web site are available:

1. **Develop the site in-house** either from scratch or with off-the-shelf components.
2. **Buy a packaged application** designed for a particular type of EC site.
3. **Lease the application** from a third party.

Each of these approaches has its benefits and limitations, and it is important to remember that the development options are not mutually exclusive. A combination of hard and soft project and change management methodologies can guide successful in-house development of enterprisewide information systems. Each of these development options is discussed in detail next.

**IN-HOUSE DEVELOPMENT: INSOURCING**

The first generation of EC development was accomplished largely through proprietary programming and in-house development that is widely referred to as **insourcing**. Using this approach, the Internet browser serves as the development platform. The programmers write EC systems using a combination of HTML and script languages such as HTX, CGI, IDC, and JavaScript. Databases developed on top of a database management system (DBMS) usually serve as the information repository to store EC data. Although this first generation of EC development has built up valuable experience and achieved industrial momentum, the lack of **reusability** (i.e., the likelihood a segment of source code can be used again to add new functionalities with slight or no modification) in current EC applications and the lack of **interoperability** (i.e., the ability to connect people, data, and diverse systems, standards) created a great barrier to widespread application of EC.

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. Companies that have the resources to develop their e-business application in-house may follow this approach in order to differentiate themselves from the competition, which may be using standard applications that can be bought or leased. The in-house development of EC applications, however, is a challenging task, because most applications are novel, have users from outside the organization, and involve multiple organizations.

**Insourcing Options**

Developers have three major options for building an application in-house:

- **Build from scratch.** This option is used rarely. It should be considered only for specialized applications for which components are not available. It is expensive and slow, but it may provide the best fit.
- **Build from components.** 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).
Enterprise application integration. The enterprise application integration (EAI) option is similar to the build from components option, but instead of using components, an entire application is employed. This is an especially attractive option when applications from several business partners need to be integrated.

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.

BUY THE APPLICATIONS (OFF-THE-SHELF APPROACH)

A number of commercial packages provide standard features required by EC applications. These packages are ready to turn on and operate. This option is also known as a turnkey approach; the package is ready to use without further assembly or testing.

The turnkey approach involves buying a commercial package, installing it as is, 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, but 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 packages are available.
- It saves time and money (compared to in-house development).
- The company need not hire programmers specifically dedicated to an EC project.
- The company knows what it is getting before it invests in the product.
- The company is neither the first nor the only user.
- The price is usually much lower than the in-house option.
- The vendor updates the software frequently.

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.

For a directory of vendors of EC turnkey systems, see softwaresearch.us/search.aspx?keywords=E+commerce+turnkey. The buy option is especially attractive if the software vendor allows for modifications. However, the option may not be as attractive in cases of high obsolescence rates or high software cost. In such cases, leasing may be a more appealing option.

OUTSOURCING/LEASING EC APPLICATIONS

The use of outside contractors or external organizations (often software vendors) to acquire EC applications is called outsourcing. It is a method of transferring the management and/or day-to-day execution of an entire business function to a third-party service provider. Outsourcing is a valuable option that more and more companies are using. In many cases, systems need to be built quickly, and the special expertise of outside contractors and software vendors is necessary.

Large companies may choose outsourcing when they want to experiment with new EC technologies without a great deal of up-front investment. Outsourcing also allows large firms to take advantage of the economies of scale that large companies enjoy.
to protect their internal networks and to gain expert advice. Small firms with limited IT expertise and tight budgets also find outsourcing advantageous.

Outsourcers can perform any or all tasks in EC applications development. For example, they can plan, program, and build applications and integrate, operate, and maintain them. It is useful for firms to develop good relationships with outsourcers.

**Outsourcing Options**

Several types of vendors offer services for creating and operating EC applications:

- **Software houses.** Many software companies, from IBM to Oracle, offer a range of outsourcing services for developing, operating, and maintaining EC applications.

- **Outsourcers and others.** IT outsourcers, such as EDS (eds.com), offer a variety of services. Also, the large CPA companies and management consultants (e.g., Accenture) offer some outsourcing services.

- **Telecommunications companies.** Increasingly, the large telecommunications companies are expanding their hosting services to include the full range of IT and EC solutions. MCI, for example, offers Web Commerce services for a monthly fee.

Although the trend to outsource is rising, so is the trend to conduct outsourcing offshore—mainly in India and China. This approach is not without risks. For example, although outsourcing offshore may lead to substantial dollar savings, offshore labor skills may be inferior to those found onshore, and the resultant quality of the Web site development may be unacceptable.

**OTHER DEVELOPMENT OPTIONS**

Besides the three major options for developing EC applications (buy, develop in-house, and outsource/lease), several other options are currently available and are appropriate under certain circumstances:

- **Join an e-marketplace.** With this option, the company “plugs” itself into an e-marketplace. For example, a company can place its catalogs in Yahoo!’s marketplace. Visitors to Yahoo!’s store will find the company’s products and will be able to make purchases. The company pays Yahoo! monthly space-rental fees. In such a case, Yahoo! is a hosting service for the company as well. As for development, the company will use templates to build its store, and it can start to sell after only a few hours of preparation work.

- **Join an auction or reverse auction third-party site.** Joining a third-party site is another alternative. Again the plug-in can be done quickly. Many companies use this option for certain e-procurement activities.

- **Form joint ventures.** Several different joint-venture partnerships may facilitate e-business application development. For example, four banks in Hong Kong developed an e-banking system. In some cases, a company can team up with another company that already has an application in place.

- **Join a consortium.** This option is similar to the previous one, except that the company will be one of the e-market owners. Thus, the company may have more control over the market architecture.

- **Use a hybrid approach.** A hybrid approach combines the best of what the company does internally with an outsourced strategy to develop contracted partnerships. Hybrid models work best when the outsourced partner offers a higher level of security, faster time-to-market, and service-level agreements.

**Section 18.2 • REVIEW QUESTIONS**

1. List the major e-commerce development strategies.
2. Define insourcing.
3. List some of the pros and cons of using packaged EC applications.
4. Compare the buy option against the lease option. What are the benefits and risks associated with each option?

5. Compare the other development options. If you were the owner of a small company trying to establish a new Webstore, which would you choose?

18.3 SELECTING A DEVELOPMENT METHOD

When choosing one or more development methods, a number of questions must be carefully answered. For example, will the chosen method(s) facilitate the completion and implementation of the e-commerce system within the time prescribed? Can the project be brought in on budget? Does the Web site adequately provide all the specified functionality? In doing so, it is important to also prioritize the different criteria in relation to current and future business needs. To help in selecting a development approach, software comparisons are available from several independent organizations and magazines. The majority of vendors offer software demos for limited testing times and many will offer proof of concept demonstrations. The various development methods are described in the following sections. Selecting a software vendor will be discussed in greater detail in Section 18.5.

SOFTWARE ON DEMAND

To accommodate the increasingly popular move to EC development by vendors, a less risky outsourcing option known as software on demand has emerged as a development method. Initially, this leasing option was provided through utility computing, then through ASPs, and more recently, through firms that provide software as a service (SaaS). Large and small firms alike will often choose the lease option to experiment with new software before making a large up-front investment or as an ongoing method of acquiring and supporting EC software. Another benefit is that it enables firms to protect their internal networks.

Through leasing, new entrants into e-business (usually smaller firms) are able to establish a market presence in a much shorter period of time. Each of these alternative sources of software acquisition is discussed in the following sections.

UTILITY COMPUTING

Utility (on-demand) computing is a business model whereby computer resources are provided on an on-demand and pay-per-use basis. This contrasts sharply with the traditional SDLC model of purchasing physical systems, configuring them, and devoting them to one application for their useful life. With utility computing, customers do not own the expensive computer resources but are billed only for their actual use of the resources. Because the utility computing provider can spread customers’ variance in resource needs, resource utilization can be optimized. Because the utility computing service is based on usage, computing resources are metered and the user charged on that basis. This is comparable to the use of electricity, gas, and most other utilities; hence, the name utility computing. Utility computing is sometimes also called on-demand computing.

Utility computing has a long history. Utility computing was first described by John McCarthy in 1961 at the MIT Centennial (en.wikipedia.org/wiki/Utility_computing) as follows:

*If computers of the kind I have advocated become the computers of the future, then computing may someday be organized as a public utility just as the telephone system is a public utility. . . . The computer utility could become the basis of a new and important industry.*

HP was the original leader of the utility computing concept, recognizing that computing power and resources can be delivered as a service, flowing on demand as and where needed. Subsequently, IBM offered this type of flexible delivery of computing power and database storage to big banks from its worldwide data centers. In 2000, Sun offered utility computing to consumers through its Sun Grid service. HP introduced the Utility Data Center in 2001. Since 2000, many companies have entered the utility computing market. Some of these organizations use utility computing to help offset hardware costs; others use...
Alexa launched Alexa Web Search Platform, a Web search building tool, for which the underlying power is utility computing. Alexa charges users for storage, utilization, and so on. SoftLayer Technologies is continuing to develop utility services aimed at meeting the needs of the emerging Web 2.0 market.

As shown in Exhibit 18.5, the utility-computing value proposition consists of three layers of tools and two types of value-added services. Each tool must be seamlessly integrated to create a comprehensive solution but will usually be implemented separately. These three tools are:

1. **Policy-based service-level management tools**, which coordinate, monitor, and report on the ways in which multiple infrastructure components come together to deliver a business service.

2. **Policy-based resource management tools**, which coordinate, monitor, and report on the ways in which multiple infrastructure components come together to deliver a business service. They automate and standardize all types of IT management best practices, from initial configuration to ongoing fault management and asset tracking.

3. **Virtualization tools**, which enable server, storage, and network resources to be deployed and managed as giant pools and seamlessly changed as needs change.

These tools share multisourcing delivery and financing services (left side of Exhibit 18.5) and provide for customer access and management services (right side of Exhibit 18.5).

**APPLICATION SERVICE PROVIDERS (ASPs)**

An **application service provider (ASP)** manages application servers in a centrally controlled location rather than on a customer’s site. Applications are then accessed via the Internet or value-added networks through a standard Web browser interface. Such an arrangement provides a full range of services for the company using the ASP.

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.

The end-user businesses pay a licensing fee. Monthly fees are separate and are paid to the maker of the software and to the ASP “host” of the software. In general, these fees include payment for the application software, hardware, service and support, maintenance, and upgrades. The fee can be fixed or may be based on utilization.

---

**EXHIBIT 18.5 The Five Elements of a Successful Utility-Computing Value Proposition**

<table>
<thead>
<tr>
<th>Multisourcing Delivery and Financing Services</th>
<th>Policy-Based Service-Level Management Tools</th>
</tr>
</thead>
<tbody>
<tr>
<td>Business-based and eventually, ROI-based management</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Policy-Based Resource Management Tools</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fault, performance, operations management, etc.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Virtualized Infrastructures</th>
</tr>
</thead>
<tbody>
<tr>
<td>Virtualized servers, storage and networks, and dynamic provisioning</td>
</tr>
</tbody>
</table>

Advantages and Disadvantages of ASPs

Leasing from an ASP is a particularly desirable option for SMEs, for which in-house development and operation of IT applications can be time-consuming and expensive. Leasing from ASPs saves various expenses (e.g., labor costs) in the initial development stage. It also helps reduce software maintenance, upgrading, and user training costs in the long run. A company can select other software products from the same ASP to meet its changing needs and does not have to invest further in upgrading the existing one. Thus, overall business competitiveness can be strengthened through reducing time-to-market and enhancing the firm’s ability to adapt to changing market conditions. ASPs are particularly effective for IT applications for which timing, flexibility, and agility are crucial.

Leasing from ASPs does have its disadvantages. ASP applications are typically hosted by third parties, who ordinarily do not have application expertise but merely manage the servers; they serve up traditional client-server applications with front-ends added as an afterthought. ASPs generally do not build the application themselves but instead take an off-the-shelf application (such as a messaging platform, an enterprise requirement planning tool, or a sales force automation package) and run it for customers. For example, an ASP might make the latest version of Microsoft Office available across the Web to customers who pay a fee per month for access to the software.

In addition, many companies are concerned with the adequacy of protection offered by ASPs against hackers, theft of confidential information, and virus attacks. Leased software often does not provide the perfect fit for the desired application. It is also important to ensure that the speed of the Internet connection is compatible with that of the application to avoid distortions in its performance. For example, it is not advisable to run heavy-duty applications on a modem link below a T1 line or high-speed DSL.

From the ASP vendor’s point of view, the benefits presented by the ASP model are many. For one, in the long-distance carrier and Internet service providers (ISP) markets, revenues are squeezed due to heavy competition. These companies are looking to generate revenues from sources other than connectivity and transport, and ASP services offer a new outlet. The ASP Industry Consortium has emerged to support the ASP concept; its founding members include AT&T, Cisco, Citrix Systems, Ernst & Young, Verizon, IBM, Marimba, Sharp Electronic, Sun Microsystems, UUNET, and Verio.

A detailed list of the benefits and risks associated with ASPs is provided in Exhibit 18.6. Major ASPs for enterprise EC systems include SAP, Oracle, and IBM. A comprehensive list of ASPs for EC can be found at Business.com (business.com/directory/internet_and_online/ecommerce/application_service_providers).

SOFTWARE AS A SERVICE (SAAS)

Because most applications typically provided by ASPs are not written as Internet native applications, performance can be poor and application updates are often no better than self-managed applications. Consequently, a need to supplement ASP offerings emerged. in the form of an on-demand computing software known as software as a service (SaaS).

Software as a service (SaaS) is a form of software delivery in which software is specifically designed for delivery in an online environment. It is essentially “leased” from a software company that provides maintenance, daily technical operation, and support for the software provided to the client. “Leading the charge” into this software delivery paradigm shift is Google. Exhibit 18.7 lists SaaS software that developers are now offering and the diverse applications they address.

To engage in SaaS, the user pays the software provider for the user’s actual usage or a fixed monthly/quarterly/annual fee instead of the one-time large fee plus support, as with the traditional software licensing model. SaaS is particularly advantageous when a company wants to experiment with a package before making a heavy up-front investment. This solution also enables a firm to protect its internal networks and quickly establish a presence in the market. With SaaS, software can be delivered to any market segment, including home office users, small businesses, and medium and large businesses. It can result in substantial cost and time savings. Although SaaS is useful to any size company,
it is particularly appealing for SMEs, who often have limited IT resources and tight budgets.

SaaS allows organizations to access business functionality from a central location over the Web at a cost typically less than that for licensed applications because SaaS pricing is based on a fee. Also, because the software is hosted remotely, users do not need to invest in additional hardware or software support. SaaS removes the need for organizations to handle installation, setup, and daily upkeep and maintenance. SaaS may also be referred to as hosted applications.

Example: Verio

Verio (verio.com) hosts corporate Web sites and Webstores at various service levels. At each level, Verio offers state-of-the-art services, a tier-1 network, and a team of experts to assist in selecting applications or designing a system. It also promises availability (get started quickly), reliability (99.9% uptime), confidentiality (secure files), data integrity (daily backups are

---

### EXHIBIT 18.6 Benefits and Risks of Using an ASP

<table>
<thead>
<tr>
<th>Type</th>
<th>Benefits</th>
<th>Potential Risks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Business</td>
<td>Reduces the need to attract and retain skilled IT professionals</td>
<td>Loss of control and high level of dependence on ASP</td>
</tr>
<tr>
<td></td>
<td>Enables companies to concentrate on lack of strategic use of IT</td>
<td>Inability of ASP to deliver quality of service; skills and experience</td>
</tr>
<tr>
<td></td>
<td>Enables SMEs to use tier-1 applications (e.g., ERP, SCM, and CRM)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Application scalability enables rapid growth of companies</td>
<td></td>
</tr>
<tr>
<td>Technical</td>
<td>Fast and easy application deployment</td>
<td>Level of customization and legacy application integration offered by ASP is insufficient</td>
</tr>
<tr>
<td></td>
<td>Higher degree of application standardization</td>
<td>Low reliability and speed of delivery due to bandwidth limitations</td>
</tr>
<tr>
<td></td>
<td>Access to wide range of applications</td>
<td>Low capability of ASP to deal with security and confidentiality issues</td>
</tr>
<tr>
<td></td>
<td>Application maintenance simplified and performed by ASP</td>
<td>Pricing changes by ASP unpredictable for application updates and services</td>
</tr>
<tr>
<td>Economic</td>
<td>Simplified user support and training</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Low total cost of ownership</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Low up-front investments in hardware and software</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Improved cost control as a result of predictable subscription costs</td>
<td></td>
</tr>
</tbody>
</table>


---

### EXHIBIT 18.7 List of SaaS Vendors and Applications

- Business Objects (Crystal Reports) (business intelligence)
- CollabNet (collaboration)
- Google (Google Maps, Google Docs, Google Spreadsheets, JotSpot)
- NetSuite (for small firms)
- Oracle (PeopleSoft on Demand; E-Business Suite on Demand)
- Replicon (Timesheets)
- RightNow Technologies (CRM and Customer Experience)
- Salesforce.com (CRM)
- SPS Commerce (EDI for retailers, distributors, manufacturers)
- WebEx (Web conferencing, video conferencing)
- YouSendIt (Web-based file hosting; large e-mail attachments)
performed to protect files from loss or corruption), throughput (pages come up quickly), scalability (an account can be upgraded as the client grows), support (24/7 access), control (easily changed Web design), and information (Web site traffic statistics, etc.). Verio clients can acquire self-service accounts or simple transactional functionality, or they can work closely with Verio to design and build robust systems to be hosted by Verio.

DY Home Décor, an online retail store that sells home décor merchandise, selected Verio Web hosting with the ShopSite tool, which integrated seamlessly with its site. Through the ShopSite tool, DY Home Décor is able to provide its customers with a high level of security and sophisticated features, such as FedEx real-time quotes, which complement the company's global shipping capabilities.

ShopSite 8.1 is certified by VISA as a Payment Application Best Practices applicant and will automatically pass purchasing information by VeriSign's Fraud Control system. Additionally, a new auction-integration function allows customers with eBay seller accounts to offer products for auction on eBay directly from the ShopSite back-office interface. The software helps its Web-hosting customers create and manage successful online stores without complex installation or high cost.

ShopSite 8.1 enables small and medium firms to manage inventory, payments, and sales records on their Web sites. It is compatible with multiple payment processing services, including PayPal, Authorize.Net, and WorldPay.

Support for the new Google Checkout allows merchants to accept credit card payments without having to obtain a merchant account, and Verio customers who configure ShopSite in accordance with Payment Card Industry Security Practices guidelines can be confident that their credit card and order information are safe and secure. ShopSite 8.1 Starter comes pre-installed with Verio's Hosting 2000 and Windows eStorefront hosting plans; it can be added to other Verio hosting plans for a reasonable monthly fee. Customers can upgrade ShopSite to the Manager or Pro level for an additional monthly fee. This example illustrates that the shift to SaaS requires the ability to integrate or combine software components. SaaS at its most sophisticated must be delivered using a service-oriented architecture (SOA) with Web Services.

INTEGRATING WITH WEB SERVICES AND SERVICE-ORIENTED ARCHITECTURE

Except in the simplest of cases, EC sites require the integration of software applications written in different programming languages and residing on different computer hardware distributed across the Internet. For example, on many B2C sites, order entry is handled by one software application or module, payment authorization by another application or module, and shipping by yet another application or module. In these cases, there is a good chance that the order entry, payment authorization, and shipping software modules all reside on separate application servers linked through a Web server. Even when packaged applications are used, a substantial amount of the implementation effort revolves around the task of tying together these disparate applications or modules in such a way that the underlying connections are transparent to the end users.

In the past, technologies made integration a difficult task for a number of reasons:

- **Platform-specific objects.** Existing EC software applications consist of a series of software objects. Software objects have properties (attributes) and methods (actions that can be performed on or by the object). For example, an order-entry application might have an “order” object that has a property specifying the “quantity” being ordered and a method called “set” that allows the quantity to be updated. In a distributed application, such as an EC storefront, the application objects residing on different computers must have a way to communicate with one another across the network.

  Two main technologies are available for accomplishing this task: Microsoft’s Distributed Component Object Model (DCOM) for Microsoft’s Windows operating systems and the Object Management Group’s (OMG) Common Object Request Broker Architecture (CORBA) for Unix-based systems. The problem is that there is limited interoperability between these two technologies. If one component or application is based on DCOM and another component or application is based on CORBA, then they
cannot communicate easily with one another. Special software called a DCOM/CORBA bridge must be used to accomplish the task.

- **Dynamic environment.** In today’s rapidly changing business environment, business partners come and go, and so do software vendors and their applications. If a software component or application is no longer available because the vendor is no longer in business or has dropped a particular product line, then an existing EC application has to be flexible enough to substitute a new component or application for the old one. If a new business partner requires additional functionality, then an existing EC application has to be flexible enough to incorporate new features, functions, or applications. Again, existing EC application architectures make it difficult to accommodate these types of changes.

- **Security barriers.** Companies use firewalls to protect their networks against security risks. Firewalls are designed to limit the types of communications and requests that can be made from one computer to another. In most cases, only the simplest sorts of Web requests using standard communication protocols (such as HTTP) are allowed. This makes it very difficult for one component or application residing on one computer to communicate with another component or application residing on another computer. For these reasons, a universal standard has emerged facilitated by the ongoing evolution of Web Services, XML, and service-oriented architecture that have combined to assist the process of integration.

**Web Services**

Web Services are self-contained, self-describing business and consumer modular applications, delivered over the Internet, that users can select and combine through almost any device, ranging from personal computers to mobile phones. By using a set of shared protocols and standards, these applications permit disparate systems to “talk” with one another—that is, to share data and services across different hardware and operating systems, including new and legacy systems—without requiring people to translate the conversation and minimizing network security risks. The result promises to provide on-the-fly and in-real-time links among the online processes of different systems and companies. These links could shrink corporate IT departments, foster new interactions among businesses, and create a more user-friendly Web for consumers. Web Services provide for inexpensive and rapid solutions for application integration, access to information, and application development. Web Services can be extremely useful for EC, especially B2B structures.

People generally view information systems, including the Web, as relating to information (data) processing. Web Services enable the Web to become a platform for applying business services as components in IT applications. For example, user authentication, currency conversion, and shipping arrangement are components of broad business processes or applications, such as e-commerce ordering or e-procurement systems.

The idea of taking elementary services and joining them together to create new applications is not new. As described earlier, this is the approach of component-based development. The problem is that earlier approaches were cumbersome and expensive. Early component-integration technologies exhibited problems with data format, data transmission, interoperability, flexibility (they are platform specific), and security. Web Services offers a fresh approach to integration. Furthermore, business processes that are composed of Web Services are much easier to adapt to changing customer needs and business climates than are “home-grown” or purchased applications.

As a simple example of how Web Services operate, consider an airline Web site that provides consumers the opportunity to purchase tickets online. The airline recognizes that customers also might want to rent a car and reserve a hotel room as part of their travel plans. The consumer would like the convenience of logging onto only one system rather than three, saving time and effort. Also, the same consumer would like to input personal information only once.

The airline does not have a car rental or hotel reservation system in place. Instead, the airline relies on car rental and hotel partners to provide Web Service access to their systems. The specific services the partners provide are defined by a series of WSDL documents. When a customer makes a reservation for a car or hotel on the airline’s Web site, SOAP messages are
sent back and forth in the background between the airline’s and the partners’ servers. In setting up their systems, there is no need for the partners to worry about the hardware or operating systems each is running. Web Services overcome the barriers imposed by these differences. An additional advantage for the hotel and car reservation systems is that their Web Services can be published in a UDDI so that other businesses can take advantage of their services.

**Web 2.0** enables less programming and more assembly to be transferred from the power of the programmer into the hands of the user. A number of innovative products that provide a drag-and-drop interface to weave Web Services together to build small applications are already available. Essentially, all that is needed is an Internet connection, a browser, and a good imagination. A representative sample of user-friendly software tools that take the programmer out of building custom business applications is shown in Exhibit 18.8. A number of techniques supporting this move to empower the user-developer are discussed in the following sections.

Developers of many sites on the Web have turned to Web Services as a way of providing access to their clients’ internal systems. In this way, the content and operations supported by these systems can be transparently integrated with applications developed by other companies or individuals. Web services offered by Google and Amazon.com have received the most publicity.

Google’s Web Services API (application programming interface) enables programmers and application developers to issue SOAP-based search requests to Google’s index of more than 3 billion Web pages and to receive results as XML data. Google’s Web Services provide a range of possibilities, including issuing regularly scheduled search requests to monitor the Web for new information on a subject; performing market research by analyzing differences in the amount of information available on different subjects over time; and searching via non-HTML interfaces, such as the command line, pagers, or visualization applications.

Amazon.com also offers an extensive set of Web Services that can be used by its “Associates” and other product sellers and vendors. The Amazon.com Associates program enables Web sites to link to Amazon.com’s site and to earn money for sales generated through the link. For Associates, Web Services provide a way to make their Web sites fresher and more dynamic. For example, Associates can use Amazon.com’s Web Services to dynamically retrieve prices, generate lists of products, display search results, produce recommendation lists, and even add items to the Amazon.com shopping cart directly on their Web sites. Other companies use Amazon.com’s development platform to sell their products on Amazon.com’s Web site. In this case, Amazon.com’s new Web Services enable these companies to do things such as manage their inventory, generate orders, and produce competitive pricing information.

### Advantages and Disadvantage of Implementing Web Services

Web Services offer a number of distinct advantages over previous programming initiatives that have attempted to solve the problem of interoperability (i.e., getting software and

---

**EXHIBIT 18.8 Representative Software Tools to Create User-Generated Web Applications**

<table>
<thead>
<tr>
<th>Product Name</th>
<th>Company</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ning</td>
<td>Ning</td>
<td>ning.com</td>
</tr>
<tr>
<td>Dabble DB</td>
<td>Smallthought Systems</td>
<td>dabbledb.com</td>
</tr>
<tr>
<td>Project Runner</td>
<td>BEA Systems</td>
<td>bea.com/framework.jsp?CNT=index.htm&amp;FP=/content/products&amp;WT.ac=topnav_products</td>
</tr>
<tr>
<td>Project Builder</td>
<td>BEA Systems</td>
<td>bea.com/framework.jsp?CNT=index.htm&amp;FP=/content/products&amp;WT.ac=topnav_products</td>
</tr>
<tr>
<td>QEDWiki project</td>
<td>IBM</td>
<td>ibm.com</td>
</tr>
</tbody>
</table>
applications from different vendors running on different hardware and operating systems to communicate with one another in a transparent fashion).

Some of the advantages of Web services include the following:

- Web Services rely on universal, open, text-based standards that greatly simplify the problems posed by interoperability and that lower the IT costs of collaborating with external partners, vendors, and clients.
- Web Services enable software running on different platforms to communicate, reducing the cost and headaches of multiple platforms running on everything from mainframes to servers to desktops to PDAs.
- Web Services promote modular programming, which enables reuse by multiple organizations.
- Web Services are easy and inexpensive to implement because they operate on the existing Internet infrastructure. They also offer a way to maintain and integrate legacy IT systems at a lower cost than typical EAI efforts.
- Web Services can be implemented incrementally rather than all at once.

However, Web Services are not a panacea. Among the current barriers inhibiting widespread adoption of the technology are the following:

- The standards underlying Web Services are still being defined; thus, interoperability is not automatic. Even in those instances where the standards are relatively stable, it still requires programming skill and effort to implement and deploy Web Services.
- One area where the Web Services standards are not well defined is security. In terms of interoperability, the good news is that Web Services enable distributed applications to communicate with ease. The bad news is that Web Services also enable applications to bypass security barriers (such as firewalls) with ease. Standards such as XML, SOAP, WSDL, and UDDI say nothing about privacy, authentication, integrity, or nonrepudiation. In an effort to bridge the gap between these standards and existing security standards (such as public key encryption), several vendors and organizations have proposed and are currently debating a number of Web Service security standards. One of these is WS-Security, which is being proposed by Microsoft, IBM, and VeriSign.
- Although Web Services rely on XML as the mechanism for encoding data, higher-level standards are still required, especially in B2B applications. For example, if two banks want to communicate, they still need standards to define the specific content that is being communicated. This is where standards such as OFX (Open Financial Exchange), which defines the content of transactions between financial institutions, come into play. The lack of coordination among all interested parties for high-level standards is why Web Services will be adopted first within organizations and later across organizations.

If you were to compare the major Web sites, applications, and Web traffic that existed before and after the dot-com era (around 2001), you would notice some substantial functional and technical differences. Many of today’s Web sites, applications, and technologies fall under the rubric of Web 2.0.

Web 2.0 applications are those that make the most of the intrinsic advantages of the network as a platform: They deliver software as a continually updated service that improves as people use it. Web 2.0 applications consume and remix data from multiple sources, including individual users, while providing their own data and services in a form that allows remixing by others, creating network effects through an “architecture of participation,” and going beyond the page analogy of Web 1.0 to deliver more sophisticated user experiences. Web 2.0 allows a level of customer engagement that previously wasn’t possible or expected.

The most popular Web 2.0 applications include Wikipedia, Flickr, Del.icio.us, YouTube, and Technorati. All of these new applications require the support of an integrative architecture, known as service-oriented architecture (SOA), which is described next. An excellent example of the use of Web 2.0 as building blocks can be seen at MasterCard and is detailed in Case 18.1.
MasterCard is the critical link among many financial institutions, businesses, and cardholders. MasterCard’s streamlined and intelligent approach to processing enables efficient commerce based on an agile network, and one of the largest virtual private networks (VPNs) in the world, which provides unparalleled speed, integration, and reliability. As consumers and businesses continue to seek out faster and more secure, smarter methods of payment for an increasing assortment of transactions, MasterCard continues to provide a sophisticated set of transaction processing and consulting services. MasterCard helps banks and merchants grow by enabling rapid adoption of innovative ways to pay and offering customized solutions that deliver value through technology. MasterCard provides industry-leading insight and solutions, therefore advancing commerce on a global scale, by using sophisticated processing and data mining capabilities. Most importantly, MasterCard is advancing commerce through innovations that benefit everyone, everywhere, as evidenced by their innovative use of Web 2.0.

The problem is that companies are continually faced with budgetary and time limitations, so they are using Web 2.0 technologies to speed, simplify, and reduce the cost of software development. By making individual software components reusable and interoperable, elements can be rearranged and combined without limiting the ability to meet demand for larger, more complex applications. According to Faeth and Busateri (2009), companies can’t take advantage of all Web 2.0 has to offer unless the business fully understands the new principles of development.

To address this dilemma, MasterCard recently developed a plan to raise awareness and understanding of these principles when teams worked on two public-facing applications based on Web services. The first application, Find-A-Card, was developed by multiple teams working together to deliver a sophisticated, influential online product that facilitates MasterCard-affiliated banks to easily market their card products. The second application, ATM Locator, harnesses the power and popularity of Google Maps to create MasterCard Location Services. User searches yield the familiar Google Map format, with icons indicating each service location. These Location Services can be customized to meet specific branding needs. MasterCard has developed the following four guiding principles to support application development involving Web services:

1. Application developers must fully understand business objectives and requirements, based on how the business operates, rather than creating applications to which the business must adapt. This means that developers must view the system from the perspective of the customer to help drive the organization forward by anticipating future business needs, while making sure the technology can be leveraged to help create practical solutions. Individuals who are successful at implementing these principles will be known internally as “business technologists.” Application developers must also keep a clear destination in mind.

2. Managers must realign their organization to foster Web Services. From the top down, it must be clearly understood that utilizing Web Services is totally different from building other customized software. Successful implementation of Web Services relies on all members of an organization to share information across silos, which may require changes in corporate governance and compensation policies. In order to successfully implement this principle, a few senior-level technologists must be selected to make sure that the appropriate application development guidelines and procedures are followed.

3. Both developers and business owners must agree on the compromises they are willing to accept regarding the application’s functionality, features, and stability, as weighed against achieving greater simplicity and speed to the market. This principle is particularly important because online users are now accustomed to applications that exist in perpetual development. The organization must also be open to learning from mistakes in order to improve solutions.

4. Management must not underestimate the significant changes that implementing Web Services entails, or the challenges it poses to employees at all levels. All team members must be made aware of why their organization is implementing Web Services and what benefits it will bring. The final results are well worth the time and effort involved in implementation.

These innovative uses of Web 2.0 and the principles associated with their development and implementation are expected to form building blocks to better align business and IT at MasterCard.

Sources: Compiled from Faeth and Busateri (2009) and MasterCard (2009).

Service-Oriented Architecture (SOA)

Service-oriented architecture (SOA) is a term that has emerged to describe executable components—such as Web Services—that can be invoked by other programs that act as clients or consumers of those services. The basic premise of SOA is to reuse and reconnect existing IT assets (also called services) rather than take the time and expense to invent new service-oriented architecture (SOA)

An application architecture in which executable
components, such as Web Services, can be invoked and executed by client programs based on business rules.

A growing number of firms are addressing the constraints of their current IT architecture and infrastructure by moving to SOA.

The three basic SOA roles are the service provider, the service consumer, and the service broker. A service provider makes the service available and publishes the contract that describes its interface. It then registers the service with a service broker. A service consumer queries the service broker and finds a compatible service. The service broker gives the service consumer directions on where to find the service and its service contract. The service consumer uses the contract to bind the client to the service. For the three conceptual roles to accomplish the three conceptual operations, an SOA system must supply four core functional architecture components:

- **Transport.** The transport component represents the formats and protocols used to communicate with a service.
- **Description.** The description component represents the language used to describe a service. The description provides the information needed to tie it to a service.
- **Discovery.** The discovery component represents the mechanism used to register or advertise a service and to find a service and its description.
- **Registration.** The registration component registers the services of a service provider with a service broker.

With SOA and Web Services, functions within existing programs and suites, as well as functions within ERP (enterprise resource planning), CRM (customer relationship management), SCM (supply chain management), and other packages, can be invoked automatically and executed anywhere in the world based on business rules. A developer does not need to know how the programs work—only the input that they require, the output they provide, and how to invoke them for execution. Developers can also swap out one service and replace it with another service that is designed to achieve the same or an enhanced result without having to worry about the inner workings of the services. SOA and Web Services are expected to totally transform the way EC systems are built, moving from slow, error-prone manual coding to automation.

The essence of an SOA is reusable services and the ability to link disparate applications. EC service components (such as credit card processing or bills payment) can be created once and then reused to drive better cost efficiency from EC and IT investments. What an SOA aims to achieve is to align IT with real business requirements. So, if a business user has a requirement, it can be translated into services in the SOA environment. A number of software vendors are currently competing in the SOA environment, including IBM, with its award-winning WebSphere Portal V6, which offers improved efficiency and productivity through a significantly enhanced user experience and additional personalization capabilities.

An up-and-coming subset of SOA that provides business partners with a way to connect, integrate, and mash up is Web-oriented architecture. The term Web-oriented architecture (WOA) was coined by Nick Gall of the Gartner Group and is used to describe a set of Web protocols (e.g., HTTP and plain XML) as the most dynamic, scalable, and interoperable Web Services approach. The main difference between a traditional SOA and a WOA is that a WOA advocates REST, an increasingly popular, powerful, and simple method of leveraging HTTP as a Web Service. Representational State Transfer (REST) refers to a collection of architectural principles. The term also is used in a loose sense to describe any simple interface that transmits domain-specific data over HTTP without an additional messaging layer. It is important to note, however, that certain applications, particularly in the high end of the enterprise, require the more sophisticated component of SOA. However, for the vast majority of uses, it would appear that WOA is the most interoperable, easiest to implement, and most highly scalable technique for building open Web Services that anyone can use.

An example of the growing trend of integrating Web 2.0 and SOA can be found at H&R Block, a large income tax preparation firm and the largest seasonal employer in the United States. To streamline staffing operations, H&R Block delivered SOA-connected Ajax portlets to more than 12,000 branch offices to support temporary work spaces. Moving away from PC-based software, their IT department was able to offer a faster, cheaper, and more flexible solution to the company’s seasonal staffing needs.
A development option that highly complements Web 2.0 and SOA is the concept of mashups.

**MASHUPS**

A **mashup** is essentially the combination of two or more Web sites into a single Web site that provides the content of both sites (whole or partial) to deliver a novel product to consumers. Mashups reach into the application program interface (API) for a given application and extract information, including Web page elements, and use them to launch a new application that adds value. The most popular type of mashup relates to maps (e.g., Google Maps, Yahoo! Maps), which can be combined with other data sources to produce interesting results. Examples of such value-adding mashups include chicagocrime.org, which combines local crime stats with Google Maps to show crimes committed by geographical area, and 1001Seafoods.com, which mashes up a database of fishing holes with Google Maps to create a U.S. map that shows fishing sites and offers directions and information on the location and sources for bait and tackle. Combining Google Maps and real estate information, Zillow (zillow.com) enables users to zoom in on a neighborhood of interest to see the environment and examine home values, home sales' history, and so on. For those who prefer rental, HousingMaps (housingmaps.com) offers a combination of Craigslist rentals with Google Maps to show rental opportunities in nearly 40 regions across the United States. Other mashups combine travel, shopping, sports, news, video, and photo sites. An interesting site that tracks the mashup phenomenon is ProgrammableWeb (programmableweb.com), which offers a mashup dashboard showing all the latest mashup sites. One of its most powerful tools is its “mashup matrix” (programmableweb.com/matrix), which presents approximately 200 Web sites in a grid with dots that highlight the intersections where two sites have been brought together in a mashup. At each intersection between sites, pop-up boxes list all mashups that have been created using the sites; in many cases, there are multiple sites at each intersection. For a more detailed discussion of mashups, refer to en.wikipedia.org/wiki/Mashup_(web_application_hybrid).

To assist in user development of mashups, support tools have evolved, including Dapper (dapper.net), an add-on to the Firefox browser, and Yahoo! Widget Engine 3.1 (widgets.yahoo.com), a JavaScript runtime engine for Windows and Mac OS X that lets users run small files called **widgets** (e.g., alarm clocks, calculators, weather reports, stock reports, and hundreds of other applications) to create a mashup Web site or desktop. A **widget** is a Web page feature or a small Web toolbox that is often designed on top of Web applications. Combining a set of powerful widgets enables users to create the equivalent of a customized business portal page.

If a company decides to buy or lease an EC application, the following representative 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 under development. 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 B2B sites. In these cases, if an application is difficult for the average visitor or customer to use, then it will have an immediate impact on its use and subsequently the bottom line.

- **Hardware and software resources.** The computer type (e.g., desktop, laptop, mainframe) and the operating system (e.g., Windows, LINUX, Mac) required by the package must be compatible with the existing platform. The CPU and storage requirements must also be compatible with existing specifications or easily accommodated.

- **Installation.** The installation effort required to implement the package is another important consideration. Some packages are complex and their installation requires extensive consultation. The installation process may also take a considerable amount of time and expertise.
Part 7: Auctions and Application Development

- **Maintenance services.** Because EC application requirements are changing constantly, ongoing maintenance is required. It is important to consider how often the package needs to be updated 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 who has a good reputation and track record than from one with a less-than-stellar or unknown reputation. A vendor’s quality can be indicated by its experience in the particular application and its sales and financial records, as well as its responsiveness to clients’ requests. Vendor support may include online help and customer relationship management (CRM) programs, as well as partner relationship management (PRM) tools. To minimize risk, minor applications should be acquired 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, training, 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 acquired from external consultants can be expensive.

- **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, 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 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 because 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 because 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 when 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.** Latency is the time required to complete an operation, such as downloading a Web page. It is an indicator of the users’ experience with the merchant server system.

  - **Throughput.** 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 directly affects the other.

- **Reliability.** Reliability is an essential requirement for a successful system. System failures and downtime are costly. When an EC application fails, business is interrupted: At best the company loses sales; at worst it loses customers. System reliability can be enhanced through the use of backup systems.

- **Security.** Security is critical to 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, not after their implementation. In addition to technological solutions such as firewalls and encryption, physical and procedural security measures must also be enforced.

DEVELOPMENT TOOLS
To implement Web 2.0 applications, users sometimes need a development framework for building rich media Internet applications—Web-based programs that run like they are on a desktop, refreshing page views without resetting the page through the server. A number of major hardware and software vendors have created software development environments to assist programmers in creating and deploying Web Services. These frameworks come in different styles, including Flash, a multimedia development platform, and JavaScript, a Web-development language. Organizations are using these tools to build Web applications faster and cheaper than ever before. Several other tools exist. The development environments provided by three of the leaders in this arena—IBM, Microsoft, and Sun—are described briefly here.

- **IBM WebSphere.** The foundation of IBM’s Web-based applications, including their EC offerings, is the WebSphere Application Server. Over the past couple of years, IBM has integrated various Web Services technologies (e.g., SOAP and WSDL) into its application server. To assist software and application developers with the design, development, and deployment of Web Services on the WebSphere platform, IBM has enhanced its existing development environment—the WebSphere Studio Application Developer—to support Web Services and has created a new development environment called the Emerging Technologies Toolkit. A comprehensive discussion of the IBM WebSphere EC suite is provided in the section on EC suites.

- **Microsoft .NET.** Microsoft’s .NET framework provides the foundation for Web Services that can be created and deployed on Windows 2000 and XP. The development environment for the .NET framework is Visual Studio .NET. Visual Studio .NET enables software developers to design, develop, and deploy Web Services with the major Windows programming languages—C++, C# (C Sharp), and Visual Basic .NET.

- **J2EE Architecture.** With J2EE Architecture (java.sun.com/j2ee), developers have access to a complete development platform specifically designed to meet the needs of enterprise application development. J2EE makes all Java enterprise functionality available and accessible in a well-integrated fashion, simplifying complex problems in the development, deployment, and management of multi-tier server-centric enterprise solutions.

One example of the use of Web 2.0 development tools is that of the American Cancer Society. The American Cancer Society (cancer.org) redesigned its online bookstore using Adobe Systems’ Flex 2, a development environment based on Flash that lets visitors read book descriptions and drag selections into a shopping cart without waiting for a server to refresh the page. Before the redesign, the society’s Web presence—built on HTTP—was inefficient, because it was necessary to call back to the server each time a user clicked a different link. The process was cumbersome. The new solution (cancer.org/docroot/pub/pub_0.asp) is flashy, catches attention, and makes an emotional impression, as well as an educational one, for the user.

Section 18.3 REVIEW QUESTIONS
1. What are the major factors driving the SaaS option for e-commerce development?
2. Consider the differences between utility computing, ASP, and SaaS. Compare and contrast the benefits and drawbacks of each of these outsourcing approaches.
3. Consider and describe other development options.
4. List some reasons why it is difficult for applications running in different environments on different computers to communicate with one another.
5. Describe the key technologies underlying Web Services.
6. What types of Web Services do Amazon.com and Google offer application developers?
7. Discuss some of the advantages of Web Services.
8. Discuss some of the factors limiting the adoption of Web Services.
9. Define service-oriented architecture (SOA) and list its advantages. Describe the relationship between SOA and Web Services EC.

### 18.4 TECHNOLOGY SUPPORT: FROM BLOGGER.COM TO INFRASTRUCTURE SERVICES

A variety of tools have emerged to enhance the Web 2.0 experience. These include **Ajax**, a Web development technique for creating interactive Web applications; **Really Simple Syndication (RSS)**, a family of Web-feed formats used to publish frequently updated digital content; and **Atom**, an RSS Internet standard created by the Internet Engineering Task Force (IETF). These are described next.

#### AJAX

**Ajax** is an acronym for **Asynchronous JavaScript and XML**. It is a group of interrelated Web development techniques used to create interactive Web applications or rich Internet applications. With Ajax, Web applications can retrieve data from the server asynchronously in the background without interfering with the display and behavior of the existing page. The use of Ajax has led to an increase in interactive animation on Web pages. See en.wikipedia.org/wiki/AJAX for the history of Ajax and the advantages and disadvantages of its use.

#### REALLY SIMPLE SYNDICATION (RSS) AND ATOM

**Really Simple Syndication (RSS)** is an XML format for syndicating news and other Web content. Through RSS, a user can subscribe to a site and be notified of new content available for download. At a given site, the presence of an RSS feed is denoted by an orange subscription icon with waves or a small orange rectangle with the letters XML or RSS inside. The acronym RSS refers generally to the concept of syndicated news feeds. RSS has evolved through four different versions (RSS 0.9, RSS 0.91, RSS 1.0, and the current version, RSS 2.0). As a result, there is no agreed-upon RSS standard, making it difficult to develop standard components and tools to work with the incompatible RSS news feeds. To address this incompatibility issue, the Internet Engineering Task Force (IETF) developed two related Web-feed standards: **Atom** (**Atom Syndication Format**), an XML language used for Web feeds, and **Atom Publishing Protocol (APP)**, a simple HTTP-based protocol for creating and updating Web resources. Most of the major blog servers either support or plan to support Atom, which is designed to ease the difficulties associated with developing applications with Web syndication feeds. Web publishers use RSS to easily create and distribute links, headlines, and news summaries. Among the many sites that now deliver updated online content via RSS are CNN, The Christian Science Monitor, The New York Times, BBC News, and The Motley Fool.

#### SOFTWARE TO SUPPORT SOCIAL NETWORKS, BLOGS, AND WIKIS

Many Web 2.0 applications are social networks (Facebook, MySpace, LinkedIn). Social networks were introduced in Chapter 9. The current move to an enterprise use of Web 2.0 that harnesses the power of social networks and the marketplace to allow enterprises to expand their contact base and/or advertise and sell their products and services has been termed **Enterprise 2.0** and its use is illustrated at Del Monte (opening case).

**Social Software**

Many advocates of social software believe the software helps to create actual communities. The more specific term, **collaborative software**, applies to cooperative work systems and is usually narrowly applied to software that enables work functions. Distinctions between the usage of the terms **social** and **collaborative** are in the applications, not the tools, although there are some tools that are only rarely used for work collaboration. **Social software** enables people to rendezvous, connect, and collaborate through computer-mediated communication. Exhibit 18.9 shows a
number of Web 2.0 applications that can be used by enterprises to disseminate information to customers, suppliers, and employees and allow them to interact. Intel offers a collaboration software suite for SMEs. Called SuiteTwo, the package includes software from Six Apart (a free blogging service that lets bloggers decide who gets to see what—if anything—on their blogs); Socialtext (an enterprise wiki system); NewsGator (an RSS platform and readers); and SimpleFeed (an RSS-based feed parsing that enables users to subscribe to topics of interest). These are small software companies that provide applications for blogs, RSS feeds, wikis, and social networking.

Some of these tools are described next.

## Software for Blogs and Wikis

Blogs and wikis were introduced in Chapter 7. Many tools are available for bloggers and wiki writers.

ProBlogger ([problogger.net](http://problogger.net)) lists dozens of blog tools, including statistical packages, blog editors, news aggregators, e-mail subscription and newsletter services, blog poll tools, and others. Notable blogging applications include:

- Blogger ([offered free from Google](http://blogger.com)) is a blog publishing system. For details, see [blogger.com](http://blogger.com) and [en.wikipedia.org/wiki/Blogger](http://en.wikipedia.org/wiki/Blogger).
- Digg ([digg.com](http://digg.com)) and Del.icio.us ([del.icio.us](http://del.icio.us)) are social bookmarking tools for sourcing stories and linking to encourage traffic.

Representative wiki tools include the following:

- EditMe ([editme.com](http://editme.com))
- Seed Wiki ([seedwiki.com](http://seedwiki.com))
- Socialtext ([socialtext.com](http://socialtext.com))
- Eurekster Swicki ([eurekster.com](http://eurekster.com))
- TeamFlux ([webseitz.fluxent.com/wiki/TeamfluxCom](http://webseitz.fluxent.com/wiki/TeamfluxCom))
- OpenWiki ([openwiki.com](http://openwiki.com))

### EXHIBIT 18.9 Sample Enterprise Use of Social Software

<table>
<thead>
<tr>
<th>Feature</th>
<th>Activity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ratings</td>
<td>Consumers create profiles and rate and comment on the company products.</td>
</tr>
<tr>
<td>Forums</td>
<td>Consumers create their own topic areas and build communities around shared interests.</td>
</tr>
<tr>
<td>Blogs</td>
<td>Staff editors lead company-formatted essays and discussions that allow customer comments.</td>
</tr>
<tr>
<td>Social networks</td>
<td>Customers, employees, and suppliers can establish a sense of community to exchange ideas about products and processes.</td>
</tr>
<tr>
<td>E-newsletters</td>
<td>Available on an opt-in basis; flow information to customers, partners, and employees.</td>
</tr>
<tr>
<td>Streaming video</td>
<td>Videos created by consumers showing the use of the company products, and the company’s TV commercials are all available for viewing.</td>
</tr>
<tr>
<td>Contests</td>
<td>Sweepstakes, do-it-yourself challenges, incentives, and other initiatives involve customers from product development to marketing campaigns.</td>
</tr>
<tr>
<td>Search engine</td>
<td>Using search engines (e.g., Google AdWords and Google Desktop) to attract new customers and increase traffic.</td>
</tr>
</tbody>
</table>
Representative companies that support RSS and support integration with Microsoft’s Exchange e-mail platform and Microsoft’s Outlook include the following:

- NewsGator (newsgator.com)
- Knownow.com (knownow.com)

An increasing number of companies are providing RSS-based applications to their employees in order to manage news feeds that they pick up from the outside and distribute inside the enterprise. Tools to support enterprise wikis and blogs will be discussed in a later section.

**Personalization Tools: From My Yahoo! to Netvibes**

Users can create highly personalized pages that are constantly updated with information (such as news articles and stock prices), view photos, use a calculator, and perform similar actions, all in one page. Users can also post necessary tools as modules, which appear as small square or rectangular objects, with the content or functionality inside. Users can arrange the modules on their sites. Users can also produce a wide variety of modules and upload them on My Yahoo! as well as similar software, making them available for free. Pages can be personalized online or offline.

One such personalization tool is My Yahoo!. Users can use My Yahoo! to combine page segments featuring Yahoo!’s own news and information with segments containing RSS feeds. Microsoft’s My MSN is another tool.

On the desktop, the best known mini-application is Apple’s Dashboard, which allows Macintosh users to install tiny programs called **widgets** that perform searches, display photo slideshows, track stocks, play music, and more. Microsoft’s Windows Vista operating system has a comparable system called **Sidebar**. Netvibes (netvibes.com) offers the best features of My Yahoo! and Dashboard. Modules can be added easily and are arranged in a menu (for details, see Mossberg 2007). For graphics-rich content, users can use Pageflakes (pageflakes.com).

**Social Bookmarking**

**Social bookmarking** is a Web Service for sharing Internet bookmarks. It is a popular way to store, classify, share, and search links through folksonomy techniques on the Internet and intranets. Examples of such sites are Reddit, Digg, and Del.icio.us. For details, see en.wikipedia.org/wiki/Social_bookmarking.

**FILE-SHARING TOOLS**

Several new services enable people to exchange large digital files (some for free, some for a fee):

- Glide Presenter from Transmedia (transmediacorp.com) can be used to store and share digital media.
- MediaMax (online-storage-service-review.toptenreviews.com/mediamax-service.html) can be used to store digital photos and movies on the Web.
- Myfabrik (myfabrik.com) can be used to send links to shared files stored on the Web.
- Pando (pando.com) bypasses e-mail attachment limits for P2P transfers.
- YouSendIt (yousendit.com) enables users to send links to uploaded files.
- Zapr (zapr.net) turns any file on a PC into a shareable Web link.

**Mobile Phones and Social Networks**

Mobile computing was introduced in Chapter 8. Mobile phones are heavily used by members of social networks. On other continents, such as Europe and Asia, devices with mobile Internet capabilities outnumber desktop PCs and laptops. Large companies, including Ericsson, Google, Microsoft, Nokia, Samsung Electronics, T-Mobile, and Vodafone, have invested heavily in dotMobi (mtld.mobi), which allows companies to buy top-level domains and tools to develop Web applications to mobile devices simply and efficiently. Thus, the
mobile e-commerce opportunities are enormous. For example, the mobile phone company Vodafone allows its customers to access the popular online video service YouTube. The launch initially focused on the United Kingdom and involved YouTube providing a daily selection of videos. Vodafone says that customers can forward links of their favorite clips and upload their own content from the phones. Vodafone also said it will roll out a service with MySpace. That deal will allow its customers to access the social network via cell phones. Vodaphone also offers its users access to eBay via their phones. Similarly, Verizon has made a deal with YouTube to bring popular videos from YouTube to cell phones using Verizon’s V-cast service.

These so-called Web 2.0 applications are more commonly associated with development of individual Web sites. However, corporations are increasingly using blogs, wikis, and social networking applications to communicate within their supply chain. This move to corporate development of Web Services and Web 2.0 is discussed next.

**SUPPORTING WEB 2.0 IN THE ENTERPRISE**

Not surprisingly, new software tools, such as IBM’s WebSphere Community Edition, Intel’s SuiteTwo, Microsoft’s Expression Web, and Oracle’s WebCenter, have been introduced to support the Enterprise 2.0 environment as companies realize its potential to engage customers and generate content for their Web sites in a way not previously possible. The potential to add an additional dimension to a flat consumer Web site and to develop closer connections to a company’s customer base is becoming very attractive to an increasing number of businesses. The latest generation of the Web does not just link information; it links people. For example, a company can allow consumers who visit its Web site to create their own profiles and rate and comment on products. Consumers can also create their own topic areas within forums and build online communities around shared interests pertaining to the company’s products. Those sites with streaming video capabilities can also encourage consumers to provide user-supplied videos of customers using their products and solicit consumer feedback on beta versions of product commercials before final release. Just imagine the immeasurable value to small firms that are unable to afford large market research and advertising campaigns; essentially, a Web 2.0 environment provides an ongoing 24/7 customer focus group, supplying firms with an unending stream of data to analyze and act upon. The move to the development of Web 2.0 tools for the enterprise is shown in Exhibits 18.10 and 18.11.

---

**EXHIBIT 18.10 Product Development 2.0: Using the Web to Put Users in Control and Co-Create Better, Richer Products Faster**

Source: Dion Hinchcliffe Web 2.0 Blog at web2.wsj2.com.
Companies such as Traction Software (tractionsoftware.com) and Socialtext (socialtext.com) offer enterprise versions of wiki and blog software that include security controls, archiving, and identity management tools. Many companies with grassroots adoption of wikis and blogs use open-source versions of the technology, but CIOs may prefer deploying tools developed specifically for the enterprise.

As a result, enterprises are now experimenting with blogs. Team members now can publish items to the blog that were formerly sent as e-mail and copied to “My Whole
Division.” Newsreader clients can capture these feeds using RSS or Atom. Users subscribe to the feeds and receive results in a common Web browser or customized reader application. Feed subscribers can be more selective about what they receive than users of traditional e-mail. For example, rather than downloading entire articles, users can first see headlines and summaries only.

At SAP, the world’s largest business software company and the third largest software supplier overall, blogging has been a change for the better. Until 2006, SAP paid a clipping service to collect relevant news stories and then e-mail the articles to large groups. Now, the Six Apart Movable Type Weblog system is used to send news articles to field staff. SAP’s blog also acts as a central source for competitive data, with field personnel contributing content. Their comments, tags, and data create a virtual conversation around the way the team works. It is much better than a stream of “FYI” e-mailed articles. Because the blog is field generated, the content is highly relevant. The blog’s value is not so much about saving money, but rather improving the depth of competitive news. Many companies are using wikis and blogs to facilitate online training.

**Software to Support Viral Marketing**

The video dimension of Web 2.0 capabilities is referred to as viral video. The term viral video refers to video clip content that gains widespread popularity through the process of Internet sharing, typically through e-mail or IM messages, blogs, and other media-sharing Web sites. Popular sites that embody this concept include YouTube (youtube.com), AOL Comedy (video.aol.com/category/comedy), Treemo (treemo.com), and VEOH (veoh.com). The marketing potential of viral video has yet to be fully explored, but companies that offer viral video services to businesses are beginning to emerge, and companies are beginning to use preexisting social networks to produce exponential increases in brand awareness.

A classic example of how viral marketing works is Hotmail.com, one of the first free Web-based e-mail services. Their strategy is simple and straightforward:

1. Give away free e-mail addresses and services;
2. Attach a simple tag at the bottom of every free message sent out: “Get your private, free e-mail at www.hotmail.com;”
3. Then stand back while people e-mail to their own network of friends and associates;
4. Who see the message;
5. Sign up for their own free e-mail service; and then
6. Propel the message still wider to their own ever-increasing circles of friends and associates.

Six simple principles of viral marketing that are applicable to most companies have been proposed by Ralph Wilson, an e-commerce consultant, and are shown in Exhibit 18.12.

Whether a company opts to purchase or lease its EC applications, two basic categories of software are available to support the development of e-commerce components: functional packages (e.g., electronic catalogs, shopping carts, electronic payment mechanisms) and merchant server software, such as EC suites.
**FUNCTIONAL SOFTWARE PACKAGES**

Standard functional software packages are available from a large number of vendors specifically for storefront construction (see Chapter 15). An example can be found at monstercommerce.com/shopping_cart_features_new.asp. This site offers dozens of software packages in the following areas: setting up your site, merchandising, inventory, payment options, hosting, shopping, tax, sales analysis, databases and systems, customer service, site design and layout, repeat customer accommodation, and security. Each includes a wide variety of features.

MonsterCommerce (monstercommerce.com) offers software both for small businesses and for large corporations (see demo at the site). The site also offers integrated solutions that are offered by several other vendors (e.g., Storefront.com; storefront.com). An up-to-date list of vendors and the EC software packages they offer can be found at thesoftwarenetwork.com/eCommerce%2DSoftware.

**Electronic catalogs** are the virtual version of traditional product catalogs (see Chapter 2). 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 are included in merchant server software, which includes features that make it simple and relatively inexpensive to develop (usually less than $10,000). A catalog operation includes a straightforward pricing and product configuration.

**Merchant Server Software**

**Merchant server 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 18.13 outlines the major components of merchant server software. As shown, a single server is used to handle product presentation, order processing, and payment processing.
(Treese and Stewart 2003). Likewise, 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, it is not necessary to store the product descriptions in a database. Instead, the pages of the Web catalog can be created ahead of time.

**EC SUITES**

An **EC suite** is a type of merchant server software that consists of an integrated collection of a large number of EC tools and components that work together for EC applications development. 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 with less sophisticated merchant server systems. The elements displayed in Exhibit 18.14 are indicative of the components contained in a typical 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 major products that remain on the market are Microsoft’s Commerce Server 2007 and soon to be available Commerce Server 2009 Release Candidate (microsoft.com/commerceserver/default.mspx), IBM’s WebSphere Commerce suite (www-306.ibm.com/software/websphere), and Oracle’s EC applications. Up-and-coming EC suites include SalesPro (webusbusconnect.com/serve.aspx?page=ec_ecomm_suite), OnLine Suites (onlinesuites.com), and AIT, Inc.’s E-Commerce Suite (ait.com/ecommercesuite.php).

**Microsoft’s Commerce Server 2007**

Microsoft’s Commerce Server 2007/Commerce Server 2009 Release Candidate (microsoft.com/commerceserver/default.mspx) offers a comprehensive framework for building tailored EC solutions. The framework consists of six main systems:

- **Product catalog system.** Enables the creation, management, and syndication of customer-specific and location-specific catalogs with specialized pricing and sophisticated search capabilities.
- **Targeting system.** Creates and deploys multilingual merchandising and advertising campaigns that support advanced discounting, cross-selling, and customer profiling.
- **Profiling system.** Enables catalog personalization, pricing, business processing, merchandising, and advertising to specific needs of customers, suppliers, and partners.
- **Business processing pipelines system.** Enables firms to tailor order and merchandising processes to handle currency conversions, multiple shipments, and complex discounting.
- **Business analytics system.** Offers businesses the ability to analyze, forecast, and mine the business data resulting from EC activities and processes, including clickstream usage, purchase histories, browsing behaviors, campaign effectiveness, and currency preferences, so that they can make informed decisions about the success of their online business.
- **User management console: Business desk.** Enables business managers to respond quickly to changing customer and business needs through a centralized intuitive console.

**EXHIBIT 18.14 Major Components of an EC Suite**

<table>
<thead>
<tr>
<th>Process Supported</th>
<th>Back-End Databases</th>
<th>Operational Systems Utilized</th>
</tr>
</thead>
<tbody>
<tr>
<td>Catalog creation and update</td>
<td>Catalog</td>
<td>Fulfillment systems</td>
</tr>
<tr>
<td>Customer management</td>
<td>Customer Order</td>
<td>Financial network</td>
</tr>
<tr>
<td>Order management</td>
<td>Payment</td>
<td></td>
</tr>
<tr>
<td>Accounts receivable</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*Source: Courtesy of Interworld Corporation, interworld.com. Used with permission.*
for managing customer and partner profiles, personalization rules, product catalogs and pricing, merchandising and advertising campaigns, and business analysis.

Microsoft’s Commerce Server comes in three editions—Standard, Developer, and Enterprise—which are designed to handle medium, large, and extremely high-traffic sites, respectively. All three editions are built on top of Microsoft’s Windows operating system, SQL Server database, and the Visual Studio .NET development environment. All three editions also operate seamlessly with Microsoft’s other .NET servers (e.g., Microsoft BizTalk Server and Microsoft’s Content Management Server). The new version of Commerce Server 2009 Release Candidate leverages SharePoint Commerce Services (microsoft.com/Sharepoint/default.mspx), a comprehensive e-commerce shopping feature-set, and technology integration between Commerce Server and SharePoint technologies. An example of a business portal built using Microsoft’s Commerce Server 2007 and BizTalk Server 2006 is described in Case 18.2.

IBM’s WebSphere Commerce Suite

IBM’s WebSphere Commerce suite (www-306.ibm.com/software/websphere) 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

CASE 18.2
EC Application

SEAL COMPANY EXPECTS $5 MILLION REVENUE BOOST FROM INNOVATIVE INTERNET ORDERING SYSTEMS

Gulf Coast Seal (www.gulfcoastseal.com) distributes the O-rings, seals, and gaskets that keep oil and gas pipelines secure and leak proof. The company’s business is built on lots of low-cost parts, some costing as little as 3 cents each. Some customers routinely place $1 orders, with the average order size being just $10. With orders bouncing between people, paper, faxes, and phones, Gulf Coast Seal calculated that its order-processing costs were between $50 and $75 per order, so these small orders were hurting the company’s profitability.

The problem was the lack of electronic links between its customers’ ERP systems and its own. A customer would create an order on its ERP and then phone or fax the order to Gulf Coast Seal. The company’s small sales staff, who doubled as order-entry and customer service staff, took the order, checked that the delivery date was possible, and cleared up any ambiguities. The salesperson then entered the order into Gulf Coast Seal’s character-based ERP system. The whole process was laborious and required 3 to 5 days before an order hit Gulf Coast Seal’s ERP system. The more orders, the bigger the backlog, and the louder customers grumbled. The company tried hiring more salespeople, but that only drove up operational and training costs.

To address these problems, the 130-person company in Houston, Texas, made a $37,000 investment in an e-commerce system that it expects to yield up to $5 million in additional revenue within 3 years by serving more customers and serving current customers better. Manual order processing had grown increasingly expensive as the company’s order volume grew. Gulf Coast Seal wanted to introduce electronic efficiencies to drive costs down and increase customer satisfaction.

Using Microsoft Commerce Server 2007 and Microsoft BizTalk Server 2006, Gulf Coast Seal created a full-service e-ordering site that enables customers to place orders, check order status, and self-service every aspect of their account. Not only does the system enable Gulf Coast Seal to handle more customers, but also to do so without increasing staff. Customers love the faster, easier ordering, and Gulf Coast Seal has reduced order-processing costs from as much as $75 to just 3 cents per order. Today, Gulf Coast Seal customers can visit the company’s Web site and sign in to a personalized site where they can place orders, view open invoices and orders, review their account history, and even look at Gulf Coast Seal’s inventory to see whether the company has needed parts in stock. Customers can manage their own account profiles, including account IDs, passwords, and shipping addresses. They can view accounts receivable and transaction summaries, as well as view or print invoices.

Sources: Compiled from gulfcoastseal.com (accessed January 2009) and Microsoft (2006).
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, Java Servlets, JavaServer Pages (JSP), Enterprise JavaBeans (EJB), and XML. These standards make it easier to integrate new products with existing back-office transaction systems and databases.

IBM WebSphere delivers application infrastructure and integration software that helps companies address key priorities in an on-demand world. The suite offers a flexible operating environment that can easily adapt to support companies' efforts to drive business growth. It provides tools to help companies streamline and extend business processes to deliver the right information to the right people at the right time and enhance the productivity of their employees.

**Oracle's EC Products**

Oracle E-Business Suite Release 12 (oracle.com/applications/crm/siebel/self-service-ebilling/siebel-e-commerce-8-1-1.html) provides applications and technology to assist companies in competing more effectively in the worldwide marketplace. This suite provides a vast array of applications aimed at supporting marketing, selling, and servicing of customers, suppliers, and partners online and enables businesses to think globally to make better decisions, work globally to be more competitive, and manage globally to lower costs and increase performance. With hundreds of cross-industry capabilities spanning enterprise resource planning, customer relationship management, and supply chain planning, this new release helps companies manage the complexities of global business environments. The following applications are aimed specifically at B2C and B2B operations:

- **Oracle iStore.** Enables merchants to build, deploy, manage, and personalize online Webstores. iStore is one of Oracle's key applications. It supports product catalog and content management, interactive and complex selling, personalized pricing, flexible check and payment options, account and contract management, and postsales order and shipping services. iStore integrates easily with Oracle's online marketing and eMerchandising functionality.

- **Oracle Marketing.** Provides automation and tools for the entire marketing process, ranging from initial marketing analysis to determine what and who should be targeted, campaign planning, budget and list maintenance, and multichannel execution, to campaign monitoring.

- **Oracle iPayment.** Offers risk management capabilities, transaction routing features, and a flexible payment architecture that supports every major online payment option.

- **Oracle Quoting.** Automates the creation and management of quotes for customized sales and service.

- **Oracle iSupport.** Provides customers with the ability to service and assist themselves over the Web.

- **Oracle Configurator.** Interactively captures, configures, and validates specialized manufacturing, sales, and service orders from customers, suppliers, and partners.

Like IBM's EC suite, Oracle's EC applications are built on open industry standards such as Java, Java Servlets, JSP, EJB, and XML. As one would expect, Oracle's applications rest on the market-leading Oracle database.

**OTHER EC SUITES**

The following are some additional EC suites that enable an organization to quickly build an e-business.
LiteCommerce 2.2
LiteCommerce 2.2 ([litecommerce.com/products.html?designer](http://litecommerce.com/products.html?designer)) enables Web designers to create EC stores without having to program. Users can download templates to a workstation and then design their EC store in WYSIWYG HTML editors such as MS FrontPage and Macromedia Dreamweaver.

Wireless E-Com Suite
Wireless E-Com Suite ([wm-us.com/e-commerce.htm](http://wm-us.com/e-commerce.htm)) is designed for small- and medium-sized Internet stores. With the Wireless E-Com Suite, a company can run one store with an unlimited number of categories and products. Features include customer care, product catalogs, product details, merchandising and inventory tools, security features, shipping and tax tools, sales analysis and tracking capabilities, and repeat-customer features.

AspDotNetStorefront
AspDotNetStorefront for Enterprise Suite (IS) ([commercemanager.info/magazine/news_h19828_aspdotnetstorefront_ecommerce_platform_to_power.html](http://commercemanager.info/magazine/news_h19828_aspdotnetstorefront_ecommerce_platform_to_power.html)) provides a complete state-of-the-art e-commerce storefront powered by the Microsoft ASP.NET platform. Interprise Solutions and AspDotNetStorefront have partnered to create AspDotNetStorefront for Enterprise Suite (IS), a powerful “Web to warehouse” solution that combines both e-commerce and back-office operations in one real-time solution. AspDotNetStorefront is fully integrated, so order, inventory, and customer information is shared by both the e-commerce Web site and Enterprise Suite. In addition to providing seamless interaction between the Web site and back-end operations, AspDotNetStorefront IS contains hundreds of industry-leading e-commerce features and supports search engine keyword integration and optimization.

ATG Commerce
ATG’s ([atg.com](http://atg.com)) comprehensive e-commerce product suite, rated number one by Forrester Research in 2006 (Mendelsohn 2006), enables customers to implement, monitor, and continuously improve personalized e-commerce applications. ATG helps keep customer interactions consistent across all contact channels and throughout the customer life cycle. ATG Commerce is a comprehensive, highly scalable solution to automate the complete life cycle of online sales, marketing, and service. Its flexible, component-based architecture enables firms to personalize the online buying experience for their customers, making it easy for them to find desired products, comparison shop, register for gifts, preorder products, redeem coupons, and execute many other useful features.

WHERE IS WEB 2.0 SOFTWARE GOING?
Currently, several hundred Web 2.0 products are on the market, with little agreement on standards. Many of these products will probably disappear, and others will be purchased by large companies (such as the purchases of Groove by Microsoft and Blogger by Google).

Large companies like to embed Web 2.0 tools in their existing collaboration products. For example, IBM is adding such tools to its Lotus/Domino Collaboration Suite. IBM is adding the following products to Lotus to make it easier for people to collaborate: Lotus Connection (social networking software for the enterprise, including Ning, Activities, Communities, Dogear for social bookmarking, and Profiles and Blog), Lotus Quickr, and Lotus Sametime. IBM WebSphere also has been redesigned to support collaboration. Oracle has an extensive collaboration suite that includes Web 2.0 tools. Another example is Intel’s Web 2.0 software suite. Even Acrobat 8.0 (from Adobe Systems) has collaboration capabilities in its professional version.

INFRASTRUCTURE SERVICES
With millions of members, and in the case of MySpace, close to 200 million members, heavy traffic demands stress the social networks’ computing infrastructure. The problem is becoming even more serious as sites add more functionalities, becoming all-in-one communities. In addition,
many companies are adding Web 2.0 tools, which also increases demand. Companies need to ana-
lyze the data that add more demand on information infrastructure and processing. To overcome
the demand, more servers that handle higher power are being installed.

For example, specialized servers are required to enable end users to add or modify Web
site content and handle RSS syndication and news feeds. Two servers that have become the
building blocks of many Web 2.0 sites are blog and wiki servers. These Web service tools and
specialized servers are described in greater detail in Technical Appendix A.

Also, the use of content delivery networks such as Akami (akamai.com), which are systems
of computers networked together across the Internet, are helpful in delivering the massive
amount of multimedia information being posted and accessed.

Luckily, social network growth is slow enough to enable appropriate infrastructure
upgrades. MySpace has managed to scale its Web site infrastructure to meet booming
demand by using a mix of time-proven and leading-edge information technologies.
MySpace was able to support more than 140 million accounts and more than 38 billion page
views a month.

Other examples of infrastructure support tools include the following:

- A number of companies will monitor online chats and provide subscribers with reports
  on topics such as the opinions, preferences, issues, and coming trends that are being
discussed. Companies offering such services include Nielsen BuzzMetrics–BrandPulse
  (www.nielsen-online.com), Cymfony–Orchestra (cymfony.com/Solutions/Our-
  Approach/Orchestra-Platform), Nstein–NtelligentEnterpriseSearch (nstein.com), and
  Dow Jones Factiva (factiva.com).
- Google Alerts can be used to monitor what the media has to say about any topic.
- Advanced search features, such as the linkdomain and allinanchor commands at Yahoo!,
  provide useful information, as does Amazon’s AG search engine.
- Popular RSS-based search engines include Technorati (technorati.com), IceRocket
  (icerocket.com), and BlogPulse (biogpulse.com). These can be used to search blogs as
  well (e.g., Technorati’s Top 100 blogs monitor the links from these for other bloggers).
  Monitoring more than 2 million blogs is not an easy task.
- Podcasts are difficult to search because the content is hard to index. Some interesting
  search engines are Podscape (podscope.com) and EveryZing (podzinger.com).
- Several companies, including Nielsen’s BuzzMetrics–BrandPulse, track comments on
  blogs in order to assess their popularity.
- Aggregation engines analyze link blogs, topical blogs, and community news sites. Sites
  such as Boing Boing (boingboing.net), MetaFilter (metafilter.com), Waxy.org (waxy.org),
  and ScienceBlogs (scienceblogs.com) are all link blogs and are major influencers in social
  media.

MySpace was successful with its infrastructure upgrades because the site is not rich
media. CondéNet Inc. has launched flip (flip.com), a rich media, social network for teens.
The new, rich-media style might make MySpace’s two-dimensional static pages obsolete as
users migrate toward Flash-heavy slideshows. VUVOX (vuvox.com) offers special authoring
tools that enable users to turn amateur Web page creators into Adobe Photoshop experts.

Another concept in IT infrastructure that has matured to enable the development of
e-commerce applications is virtualization.

**Virtualization**

**Virtualization** enables the sharing and/or aggregation of physical resources, such as operat-
ing systems, software, and IT services, in a way that hides the technical detail from the end
users and reduces the per unit service cost. Because the virtualization system sits between the
guest and the hardware, it can control the guest’s use of CPU, memory, and storage, even
allowing a guest OS to migrate from one machine to another.

Originally developed in the 1960s to partition mainframe hardware, virtualization declined in popularity as minicomputers and PCs provided a more efficient, affordable way
to distribute processing power. Recently, decreasing hardware costs, underutilization of

**virtualization**

A technique for hiding the physical characteristics of computing resources from the way
in which other systems, applications, or end users interact with those resources.
resources, escalating maintenance costs, and security issues have caused IT professionals to take another look at virtualization as a way to help businesses address issues of scalability, security, and management of their IT infrastructure. The benefits of virtualization are shown in Exhibit 18.15.

**Virtualization Products.** Not surprisingly, there has been a visible surge in virtualization-related startups, acquisitions, and partnerships, as well as related academic research. Dozens of virtualization products are available, offered by a number of small and large companies. Some examples in the operating systems and software applications space are VMware (vmware.com), Xen (xensource.com), and Microsoft Virtual Server (microsoft.com/windowsserversystem/virtualserver/default.mspx).

Major IT players also have shown a renewed interest in the technology. For example, IBM—a long-time pioneer in the area of virtual machines—has not only strengthened its own virtualization-related products but has also forged a relationship with VMware. Sun and HP also are improving their virtualization offerings (IBM 2009). Both Intel and AMD now make processors that explicitly support virtualization. Microsoft acquired Connectix Corporation, a developer of virtualization software for Windows and Macintosh platforms, and Softricity, an application streaming developer. EMC acquired VMware, and Cisco acquired several companies that develop products in the storage, file, and network virtualization spaces. An interesting example of virtualization is shown in Case 18.3.

**Cloud Computing**

An emerging infrastructure service that incorporates software as a service (SaaS), utility computing, grid computing, and Web 2.0 concepts is cloud computing. According to en.wikipedia.org/wiki/cloud_computing, cloud computing is Internet (cloud)-based development coupled with the use of computer technology (computing). The “cloud” metaphor refers to the Internet (based on how it is depicted in computer network diagrams) and is an abstraction for the complex infrastructure that it conceals. It is a style of computing that provides IT-related capabilities as a service. It allows users to access technology-enabled services from the Internet without any knowledge, expertise, or control of the technology infrastructure that supports them. According to the IEEE Computer Society, cloud computing allows information to be permanently stored in servers on the Internet and cached temporarily on client desktops, table computers, notebooks, and handhelds. With cloud computing, desktop-based computing is moved onto the Web, and programs and documents can be accessed from any computer that is connected to the Internet. Whether you need to share photographs for business or pleasure, coordinate volunteers for a community organization, or manage a large multifaceted project at work, you can do it more easily and quickly than ever before using cloud computing. According to Miller (2008), cloud computing is

---

**EXHIBIT 18.15 Benefits of Virtualization**

**Partitioning**
- Multiple applications and operating systems can be supported within a single physical system.
- Servers can be consolidated into virtual machines on either a scale-up or scale-out architecture.
- Computing resources are treated as a uniform pool to be allocated to virtual machines in a controlled manner.

**Isolation**
- Virtual machines are completely isolated from the host machine and other virtual machines. If a virtual machine crashes, all others are unaffected.
- Data do not leak across virtual machines, and applications can only communicate over configured network connections.

**Encapsulation**
- Complete virtual machine environment is saved as a single file; easy to back up, move, and copy.
- Standardized virtualized hardware is presented to the application, guaranteeing compatibility.
drastically changing the way that people use computers. Case 18.4 shows how Kenworth Truck Company was able to use cloud computing to maintain its sales levels and save its customers money. By allowing open access to thousands of computers and services, all linked and accessible via the Internet, the dependency on expensive programs stored on a single computer, data accessibility from only one computer, and work tied to a single computer will no longer be the norm as companies strive to maintain their competitive edge.

**Section 18.4 REVIEW QUESTIONS**

1. Describe some tools for blogging and wikis.
2. Define social software and list some examples.
3. Describe suites of Web 2.0 tools and the integration of these tools.
4. Describe the need for IT infrastructure with regard to the Web 2.0 revolution.
5. List the major features of an electronic catalog.
6. Describe the basic business systems in Microsoft’s Commerce Server 2007.
7. Describe the functions supported by IBM’s WebSphere Commerce suite.
8. Describe the key EC applications provided by Oracle for building B2C and B2B sites.
9. As the owner of a small- to medium-sized online business, which of the e-commerce suites described in this section would you find most useful? Explain.
10. Define virtualization and describe its relationship to SOA.
11. Discuss ways in which virtualization might assist a small manufacturing firm to offer an e-commerce application to its customers.
Part 7: Auctions and Application Development

18.5 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 such as Digital Dreamwave (digitaldreamwave.com), a Web development firm that provides state-of-the-art Web application and software solutions. Thus, a major aspect of developing an EC application revolves around the selection and management of these vendors and their software offerings. Exhibit 18.16 shows the flow and interaction of the following six steps used to select a software vendor and/or e-commerce package:

1. Identify potential vendors and software.
2. Determine evaluation criteria and weight the importance of each.
3. Evaluate vendors and packages.
4. Choose the vendor and package based on criteria, weights, and feedback.
5. Negotiate a contract and obtain legal advice.
6. Establish a mutually agreeable service-level agreement (SLA).
STEP 1: IDENTIFY POTENTIAL VENDORS

Potential 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 that have a questionable reputation. Also, packages may be eliminated if they do not have the required features or will not work with available hardware or the 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 vendor characteristics, functional requirements of the system, technical requirements the software must satisfy, the 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 to invite 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.
STEP 3: EVALUATE VENDORS AND PACKAGES

Vendor 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 at least 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 must 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.

Section 18.5 REVIEW QUESTIONS

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

18.6 CONNECTING TO DATABASES AND OTHER ENTERPRISE SYSTEMS

The major integration areas discussed in this chapter are connecting to databases and to back-end systems.
CONNECTING TO DATABASES

Nearly every EC application requires database access. For example, when a customer orders a product online, the product description, inventory count, and order information are likely to be retrieved from and stored in one or more databases. To check availability, the ordering system must be connected to the inventory system. An EC application can be connected to a back-end database in a variety of ways. Today, most of these connections are accomplished via a multi-tiered application architecture, which is an EC architecture that consists of four tiers: Web browsers, Web servers, application servers, and database. Customers with a Web browser can then access catalogs in the seller’s database, request specific data, and receive an instance response. Here the application server manages the client request. The application server also acts as the front end to complex databases.

This architecture has four tiers:

1. A Web browser that presents data and information to and collects data 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 in which the data are stored and managed and database requests are processed

CONNECTING TO BACK-END SYSTEMS

Several technologies can be used to integrate an EC application with a back-end database. Many 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 Active Server Pages [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’s WebLogic Server 9.2 (bea.com) is a 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 enterprise application integration (EAI) software. These products focus on the integration of large systems. TIBCO (tibco.com), WebMethods (www-01.ibm.com/software/integration/wbiserver/ics), and IBM’s WebSphere InterChange Server (ibm.com) are examples of companies that have offerings in the EAI arena.

CONNECTING TO BUSINESS PARTNERS

As discussed earlier, companies and organizations are now building enterprisewide EC systems by integrating previously independent applications together with new developments. EC applications also must be connected to items such as the partners’ systems or to public exchanges. Such connections are referred to as integration. EC users interact with Internet applications through a variety of devices whose characteristics and performance figures span an increasingly wide range. Applications use communication protocols and intermediate software that resides on top of the operating systems to perform the following functions:

- Hiding distribution (i.e., the fact that an application is usually made up of many interconnected parts running in distributed locations)
- Hiding the heterogeneity of the various hardware components, operating systems, and communication protocols
Part 7: Auctions and Application Development

Providing uniform, standard, high-level interfaces to the application developers and integrators so that applications can be easily composed, reused, ported, and made to interoperate

Supplying a set of common services to perform various general-purpose functions to avoid duplicating efforts and to facilitate collaboration between applications

The intermediate software layers have come to be known under the generic name of middleware. Middleware is essentially a separate program that provides an interface between diverse client and server systems. Its main function is to mediate interaction between the parts of an application or between applications. For more information, see middleware.objectweb.org.

IBM is the leading provider of middleware. The company offers a number of on-demand solutions for communication, government, retail, banking, financial markets, and many other industries. IBM Middleware (ibm.com/middleware) helps automate systems, integrate operations, connect people, and develop software.

Section 18.6 REVIEW QUESTIONS

1. Describe the basic elements of a multi-tiered application architecture.
2. List the ways in which an EC application can be connected to back-end databases and other transaction processing systems.
3. Define middleware and describe its attributes.

18.7 USAGE ANALYSIS AND SITE MANAGEMENT

To improve EC Web sites, it is advisable to monitor what customers are doing there (usage analysis). Both B2C and B2B Web sites require a thorough understanding of the usage and patterns—the who, what, where, when, and how. This can be done by analyzing Web log files.

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 in which each line of the file details an individual access. Regardless of the type of Web server, access logs use a common log file format. 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 EC software vendors provide free software for analyzing access log files. Commercial products that provide more sophisticated log analyses also are available (e.g., NetIQ’s 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 the following:

- **Pageviews by time slot.** Pageview statistics allow frequent review of the number of site accesses. Grouping pageviews by “time bucket” (time slot) also enables a company to ascertain the time slots, such as morning, afternoon, or evening, during which customers visit the site.

- **Pageviews by customers’ log-in status.** This information helps determine whether requiring customers to log in is worthwhile. 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 log-in 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’ interests 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 enable a company to obtain information on the hardware platform (e.g., Mac or PC) 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 sites. Knowing where customers are coming from can enable the company to target potential customers via popular hosts such as Yahoo!
Alexa: Web Traffic Information Provider

Alexa Internet (alexa.com) is a Web site owned by Amazon.com that provides information on Web traffic to other Web sites. Alexa collects information from users who have installed an Alexa toolbar, allowing them to provide statistics on Web site traffic, as well as lists of related links. Alexa ranks sites based on visits from users of its Alexa toolbar for Internet Explorer and from integrated sidebars in Mozilla and Netscape.

Some question how representative Alexa's user base is of typical Internet behavior (see en.wikipedia.org/wiki/Alexa_Internet). If Alexa's user base is a fair statistical sample of the Internet user population (e.g., a random sample of sufficient size), Alexa's ranking should be quite accurate. In reality, not much is known about the sample, and it might or might not have many sources of sampling bias. Another concern is whether Alexa ratings are easily manipulated. Some Webmasters claim that they can significantly improve the Alexa ranking of less popular sites by making them the default page, by exchanging Web traffic with other Webmasters, and by requiring their users to install the Alexa toolbar; however, such claims are often anecdotal and are offered without statistics or other evidence.

Competitors in the Internet market research space include Complete Inc., ComScore, Hitwise, Nielsen/NetRatings, and Netcraft.

Some of the marketing and business questions to which these statistics can be applied are listed in Exhibit 18.17.

<table>
<thead>
<tr>
<th>Area of Analysis</th>
<th>Business Questions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overall store performance</td>
<td>• What is the sales value for a specific period of time, say, 1 week?</td>
</tr>
<tr>
<td></td>
<td>• What is the number of customer visits for the day?</td>
</tr>
<tr>
<td></td>
<td>• What is the store conversion rate for the week?</td>
</tr>
<tr>
<td></td>
<td>• What is the sales value index for the week?</td>
</tr>
<tr>
<td>Advertising</td>
<td>• Which banner ads are pulling in the most traffic?</td>
</tr>
<tr>
<td></td>
<td>• How many sales are driven by each banner ad?</td>
</tr>
<tr>
<td></td>
<td>• What products do shoppers select from a particular banner?</td>
</tr>
<tr>
<td></td>
<td>• What is the conversion rate for each banner ad?</td>
</tr>
<tr>
<td>External referrals (from others to your site)</td>
<td>• Which portal sites are pulling in the most traffic?</td>
</tr>
<tr>
<td></td>
<td>• Which are generating the most sales?</td>
</tr>
<tr>
<td></td>
<td>• How many sales are generated by each referral site/search engine?</td>
</tr>
<tr>
<td></td>
<td>• What products do shoppers from a particular portal site purchase?</td>
</tr>
<tr>
<td>Shopper segmentation</td>
<td>• How many visitors are from a specific domain?</td>
</tr>
<tr>
<td></td>
<td>• What is the distribution of first-time versus repeat shoppers?</td>
</tr>
<tr>
<td></td>
<td>• What characterizes shoppers of a particular set of products?</td>
</tr>
<tr>
<td></td>
<td>• What characterizes shoppers who abandon shopping baskets?</td>
</tr>
<tr>
<td>Product grouping</td>
<td>• How much do cross-sells/up-sells contribute to gross revenue?</td>
</tr>
<tr>
<td></td>
<td>• What are the best performing cross-sell pairs? Worst?</td>
</tr>
<tr>
<td></td>
<td>• What is the overall conversion rate for cross-sells/up-sells?</td>
</tr>
<tr>
<td>Promotions and recommendations</td>
<td>• How much do promotions contribute to gross revenue?</td>
</tr>
<tr>
<td></td>
<td>• Which promotions are generating the most sales?</td>
</tr>
<tr>
<td></td>
<td>• What is the overall conversion rate for promotions?</td>
</tr>
<tr>
<td></td>
<td>• What is the overall conversion rate for recommendations?</td>
</tr>
<tr>
<td></td>
<td>• At what levels in the site hierarchy are the best promotions located?</td>
</tr>
<tr>
<td>Shopping metaphor</td>
<td>• What generates the most sales value: searching or browsing?</td>
</tr>
<tr>
<td></td>
<td>• How much does searching contribute to gross revenue?</td>
</tr>
<tr>
<td></td>
<td>• What is the conversion rate for searching?</td>
</tr>
</tbody>
</table>

(continued)
## 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. One of the more prominent vendors of IT and Web management tools is BMC Software, Inc. (bmc.com). Some of the products offered by BMC include:

- **Patrol for E-Business Management.** This package includes Patrol for Internet Services, which measures Web response time; Patrol for Firewalls, which provides firewall administration; and Patrol for Microsoft or Netscape application servers.

- **MainView for E-Business Management.** This package includes MainView for WebSphere, which aids in the management of mainframe-based EC applications; MainView for Network Management, to monitor mainframe network connections; and MainView for Systems Management, to provide systems administration.

Other EC management tools include site version control tools, combined utilities/tools, server management and optimization tools, and authoring/publishing/deployment tools that include significant site management or testing capabilities. A detailed list of vendors is provided by Hower (2009). The following are some examples:

- **COAST WebCentral.** This site management tool from Coast Software, Inc. (e-tamed.co.uk/products/coast4.html) monitors compliance standards related to privacy, information assurance, accessibility, and Web governance. Capabilities include high-level summary reports, site scores, and trend analyses using intuitive browser-based reports; roll-up of results from different business units or global Web sites for overall views of site compliance; complete inventory of Web sites; and comparison inventory reports to compare different versions of a Web site.

- **Maxamine Knowledge Platform.** This integrated site management tool from Maxamine Inc. (maxamine.com) combines site content, structure, and visitor traffic data into one searchable database, enabling detailed analysis of what is going on behind the scenes of Web operations.

- **WebCEO.** Companies can use this tool from Radiocom Ltd. (webceo.com) to maintain, promote, and analyze Web sites. It includes a link checker, a WYSIWYG editor, and FTP/publishing, traffic analysis, and site monitoring capabilities.

- **ManageEngine Applications Manager.** This site management tool from AdventNet (adventnet.com) works with a variety of Web servers, database servers, service types, and operating systems.

### Section 18.7 REVIEW QUESTIONS

1. Define access log files.
2. List some of the types of statistics provided by an access log.
3. Describe some of the uses of an access log.
4. Describe EC site management activities.
MANAGERIAL ISSUES

A number of managerial issues relate to the concepts presented in this chapter. Some of these are as follows:

1. **Do our business goals match well with e-business solutions?** When one thinks of the Web, one may immediately think of technologies and solutions. Surely technology is necessary; however, we should not forget about identifying our goals. It is very important to map a combination of solutions that will lead to our goals. The spectrum of solutions covers goals related to e-marketplace, ERP, CRM, SCM, data warehouse, data mining, and enterprise application integration (EAI) tools. Since it is not easy to recruit experts on all these solutions, it is unavoidable that we rely on the external business and solution consulting for the right judgment.

2. **What is the development strategy for our enterprise information system, including EC?** Since the technologies for EC have become standardized and available, the most critical factors in developing EC systems are the right selection of EC solutions (such as e-procurement, SCM, and CRM) and integrating them with ERP, existing modules, and external partners. To be compatible with external partners, adopting global standards such as Rosettanet and ebXML is necessary. However, following these sophisticated standards is not cost-effective unless there is a high volume of transactions. This is true for most small and medium enterprises; designing the right combination of tight integration with loose integration is a very important factor in making the EC system cost-effective.

3. **What is the outsourcing strategy?** Many large-scale enterprises are capable of building and running their own EC Web sites. However, EC Web sites require complex integration, security, and excellent performance. Outsourcing has become the major trend in order to reduce the large development investment. These services enable companies to start small and evolve to full-featured functions through the use of ASPs, Internet malls, and software vendors that offer merchant server and EC applications. Outsourcing is strongly recommended, particularly for small companies. Nevertheless, some parts should be built in-house (insourced) to secure more direct control over data management. Thus, the integration of internal systems and outsourced systems is a challenging issue. EAI can enhance the flexibility of integration with multiple external systems that adopt different standards.

4. **How should Web Services be deployed?** Many organizations face problems of integrating systems, applications, and data. The IT costs associated with getting applications and databases running on different hardware and operating systems are substantial. Web Services offers the means to secure this interoperability in an open, straightforward, and efficient fashion without the need for new hardware or application reprogramming. The platform makes the service-oriented architecture (SOA) possible. However, integration of complex processes is not easy, and the standard for high-level Web orchestration has not yet gained momentum.

5. **Which strategy should we choose for vendor selection: inside-out or outside-in approach?** Because most EC applications are built from packaged applications and components, or are outsourced to a third party, the success of the EC application rests on choosing the best vendor and package. Two approaches are the inside-out approach (an ERP package provided by a vendor expands its scope to encompass the e-business components like SCM and CRM) and outside-in approach (the best-of-the-breed of e-business components are integrated with an ERP package). Recently, the solution of inside-out tends to win the market because design by a single vendor provides a more consistent architecture.

6. **Have we analyzed the usage data?** Most 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 the same user. The patterns discovered can be used for the customized service for one-to-one marketing.

7. **How do we balance the experts’ views with users’ requirements?** The direct and indirect users of an EC system are likely to be the most knowledgeable individuals concerning requirements and which alternatives will be the most effective. Users are also the most affected by a new information system. IS analysts and designers, however, are likely to be the most knowledgeable individuals concerning technical and data management issues. These professionals, too, are likely the most experienced in arriving at viable EC systems solutions. Thus, the right mixture of user involvement and information systems expertise is crucial.
In this chapter, you learned about the following EC issues as they relate to the chapter's learning objectives.

1. **The major steps in developing an EC system.** 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, an EC application portfolio is defined based on an organization’s strategy. Second, the EC architecture is created. 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 application development strategies, along with their advantages and disadvantages.** EC sites and applications are rarely built from scratch. Instead, enterprises buy a packaged EC suite and customize it to suit their needs, or they outsource the development to a third party. A new generation of Web tools is taking the programmer out of the development process and empowering more users to develop their own Web sites. The selection of one option over another should be based on a systematic comparison of a detailed list of requirements that examines important considerations such as flexibility, information needs, user friendliness, hardware, and software resources.

3. **The varied EC application development methods, along with their benefits and limitations.** Once a strategy has been determined, numerous development methods can be used to develop an EC system. These include Web 2.0, software as a service, utility computing, cloud computing, and many others listed and detailed within the chapter. Depending on the resources available to the organization developing the EC system and the requirements of the system, one or more of the different development methods will be chosen to create the most efficient and effective solution.

4. **EC application outsourcing options.** Many enterprises elect to outsource the development and maintenance of their EC sites and applications. The most common type of EC applications outsourcing is the use of software as a service (SaaS). Utility computing is another popular option, and the emerging concept of cloud computing is growing in popularity. 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, the costs involved, the time frame, and the available IT resources.

5. **The major components of software packages and EC application suites.** 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 Webstores and other types of EC sites (e.g., B2B, exchanges, etc.) can be built from comprehensive EC suites such as Microsoft’s Commerce Server or IBM’s WebSphere Commerce suite. A payment gate and a site search engine are useful.

   Every type of EC application has a long list of functional requirements. Fortunately, most of these requirements can be met by packaged applications. Online Webstores 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 components that have the listed functionalities.

6. **The rise of Web Services and XML.** Web 2.0 is the newest technology aimed at solving integration and interoperability problems (getting applications running in different computer environments to communicate with one another). Web Services generally rely on open standards, including XML, SOAP, WSDL, and UDDI, to overcome the problems. WSDL defines the operations that a Web Service can perform. To invoke a Web Service, an application sends a SOAP or XML message to the service, which in turn responds with a SOAP/XML message. Companies can publish their Web Services in a UDDI registry, so that any application can take advantage of the operations the services perform. Although Web Services require minimal changes and reprogramming of existing systems and applications, they still require advanced programming skills to implement and deploy them. Toward this end, hardware and software vendors such as Microsoft and IBM have provided software development environments to ease the task.

7. **Understand service-oriented architecture and its relationship to EC.** With service-oriented architecture (SOA) and Web Services, functions within EC applications such as order taking and billing can be automatically invoked and executed anywhere in the world based on business rules. Developers can swap out one EC service and replace it with another service without having to worry about the inner workings of the two services. SOA is expected to transform the way we build EC systems in the future—from slow, error-prone manual coding to an automated discipline.

8. **Virtualization.** Virtualization allows the delivery of an operating system or software application functions over the Web. Examples of virtualization include virtual desktop infrastructure, traditional server-based
solutions, Web-based solutions, terminal server solutions, and more recently, application streaming and image-level services. An underlying assumption of virtualization is that storage and networking are available and may also be virtualized.

9. **Criteria used in selecting software vendors and packages.** A systematic process should be used in selecting a third-party tool or an outsourcing company. 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 an SLA to define who is responsible for specific aspects of the development and maintenance and what quality metrics will be used for the services to be rendered.

10. **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). 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, Active Server Pages [ASP], or JSP), employing specialized application servers, employing a full-blown EAI tool, or using XML-based technologies.

11. **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.

12. **The importance of usage analysis and site management.** Usage analysis is about monitoring customer activities at EC Web sites. Web log files provide a variety of statistics about Web site usage patterns that can be useful for analyzing and improving marketing and business strategies.

---

**KEY TERMS**

- Acceptance testing 7
- Access log 42
- Ajax 24
- Application service provider (ASP) 12
- Atom 24
- Cloud computing 36
- EC architecture 5
- EC suite 31
- Electronic catalog 30
- Enterprise application integration (EAI) 9
- Insourcing 8
- Integration testing 7
- Interoperability 8
- Latency 22
- Mashup 21
- Merchant server software 30
- Middleware 42
- Multi-tiered application architecture 41
- Outsourcing 9
- Policy-based resource management tools 12
- Policy-based service-level management tools 12
- Really Simple Syndication (RSS) 24
- Representational State Transfer (REST) 20
- Request for proposal (RFP) 39
- Reusability 8
- Scalability 6
- Service-level agreement (SLA) 40
- Service-oriented architecture (SOA) 19
- Social bookmarking 26
- Social software 24
- Software as a service (SaaS) 13
- Throughput 22
- Turnkey approach 9
- Unit testing 7
- Usability testing 7
- Utility (on-demand) computing 11
- Viral video 29
- Virtualization 35
- Virtualization tools 12
- Web-oriented architecture (WOA) 20
- Web Services 16
- Web 2.0 17
- Widget 21

---

**QUESTIONS FOR DISCUSSION**

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 in-house?

3. A large chemical manufacturing company is interested in starting an online exchange. What are some of the ways it could achieve this goal?
INTERNET EXERCISES

1. Access choicemall.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. Go to 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?

4. Visit ibm.com. Find its WebSphere product. Read some recent customer success stories. What makes this software so popular?

5. Go to the World Wide Web Consortium’s discussion of Web Services architecture (w3.org/TR/ws-arch/wsa.pdf). Based on this discussion, what role does a UDDI play in Web Services? If a company wants to publicize its Web Services, what means are available?

6. Visit Sun Microsystems (sun.com). What type of development platform does Sun provide for creating and deploying Web Services? What are the capabilities and benefits of the platform?

7. Visit microsoft.com. Find its BizTalk product. What kind of software is this? What role could it play in an EC application?

8. Enter webservices.org and find material on Web Services and EC. Prepare a report.

9. Enter monstercommerce.com and ecommerce-shoppingcart.biz and write a report comparing their products.

10. Search the Internet to find a mashup site that is not mentioned in the text. Determine the components that it has “mashed” and write a report on the value added that the mashup provides.

11. Access techsoup.org/toolkits/web2. Read the article “Seven Blogging Tools Reviewed” and visit the Web sites of four of the blogging tools. Determine the features and ease of use of the tools that you personally review and compare your assessment with that of the review authors.

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 online vendor wants to hook its shopping cart application to a credit card authorization Web site. How could this be done with Web Services? How would the authorization site advertise its services for other sites to use?

6. An enterprise wants to modify its EC site so that it conforms more closely with the company’s 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?

7. Define Web Services. List the major areas of Web Services applications. Discuss three major advantages and three major disadvantages of Web Services.

8. In what ways do you think a Web site’s log files may violate consumers’ privacy?

9. 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.

10. Identify the major reasons why utility computing tools may become the next big thing in EC applications development.

11. Do you think the technology has a lot of catching up to do before utility computing will become a reality for companies with many computing needs?

12. Discuss the relationship between service-oriented architecture (SOA) and Web Services.
TEAM ASSIGNMENT, PROJECTS, AND CLASS DISCUSSIONS

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 several 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 Webstores. The Web sites of these vendors usually list those online stores that currently use 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 18.2 for a list of functionalities). If the functionalities of the applications are not sufficient, explain what additional functionalities are needed.

4. Amazon.com provides Web Services for its Associates (amazon.com/webservices), as well as other product sellers and vendors. Assign one team to explore the services for Associates and another to find out about the services for sellers and vendors. Describe the services provided by each. What are the benefits of these services? What companies are currently using these services? Go to their sites and describe how they are using these services.

5. Each team selects a software vendor that supports integration (e.g., see oracle.com, intersystem.com, icode.com, tibco.com, microsoft.com, or webmethods.com). Prepare a report on how they connect EC applications (such as ordering) to back-office systems.

6. Each team brainstorms a new and innovative mashup site. Discuss and document the value it would provide and define the market to which the site would appeal. Present (sell) your site to the other teams and take a vote on the best mashup created by all teams.

7. Address the following topics in a class discussion:
   a. Is there a pattern to the functions that are provided by different types of Web Sites?
   b. What capabilities are offered by B2B and B2C EC applications? How do they differ?
   c. Are the functions provided by integration software vendors adequate in today’s web marketplace?
   d. How are Web Services adding value to companies that use them?

Closing Case

TOMTOM DEVELOPS A NEW E-COMMERCE INFRASTRUCTURE

The Business Opportunity

TomTom NV (tomtom.com) is a leading provider of personal navigation products and services, founded in 1991 in Amsterdam. It has offices in Europe, Australia, the United States, and Asia. Its products include all-in-one navigation devices that enable customers to navigate right out of the box, as well as navigation software products that integrate with third-party devices such as PDAs and smartphones. TomTom’s portfolio of products and services includes the TomTom GO family of integrated navigation devices, software for PDAs, TomTom MOBILE navigation software for smartphones, and a location-based content and services offering for TomTom’s navigation products.

TomTom had been using a simple Notes-based system to handle e-mail and conventional FAQs to answer customers’ questions via the Internet. However, the company soon found itself with a backlog of e-mails and broadening phone queues during the holiday season. The following summer, TomTom initiated a product upgrade that brought forth a huge market response. It was obvious that this company had to strengthen its customer service operation in expectation of a growing number of inquiries as more products entered the marketplace and sales continued to grow. There were fears that if TomTom did not have a first-class contact center in place before the next holiday season, the new customers that had just been won over to this company would be alienated. After a careful analysis of all vendor solutions available on the market, it was clear that RightNow’s multi-channel contact center solution was the most advantageous resolution.

The Project

The intuitive Web self-service interface offered by RightNow was particularly appealing to TomTom because
ease-of-use is such an important design factor in its own products. In addition, an online knowledge base was made extremely effective because some standard questions could be answered via the Web. RightNow has been able to provide TomTom with the contact center solution that was needed to provide integrated e-mail management and call tracking, since it supports customers via whichever channel they prefer. For this service to be useful, it was essential that the service agent be able to access the same knowledge base that the customer accesses. TomTom’s 170+ call center agents are empowered with information right at their fingertips so that they can quickly answer customer queries via e-mail and phone, using the information that is held centrally in the knowledge base. Reporting and analytics give visibility into customers’ changing issues and needs, helping the service team to proactively post information onto the Web site, perhaps anticipating questions before they are asked. Not only was RightNow’s call center solution able to provide all of the functionality that TomTom needed, but it was also very easy to implement. Immediately, the contact center was able to reduce e-mail volume by 25 percent and stay on top of the expected spike of questions and issues that were sure to arise from the impeding holiday season.

The new contact center employees were able to get up to speed on the CRM software system so quickly that it has made it easier and less expensive for TomTom to grow its contact center. This ease-of-use allows TomTom to manage knowledge base content, reducing the company’s cost of ownership as well as enabling the company to stay on top of emerging issues. TomTom has experienced such success with the customer service solutions provided by RightNow that the company is taking advantage of RightNow’s marketing automation solutions. TomTom will be able to execute newsletter-style mailings to support its evolving customer loyalty programs. Integration between RightNow’s customer service solutions and marketing automation solutions ensures the right information will be sent efficiently and reliably to the right customers at the right time. In addition to providing TomTom with world-class technology, RightNow has also provided the insight, support, and expertise the company required to address the various challenges that come with fast growth and rapid market expansion.

**Results**

The world’s leading portable navigation solutions provider, TomTom, has a continued commitment to provide “An Outstanding Customer Service Experience.” This company recently received the prestigious J.D. Power and Associates Certified Call Center recognition for call center customer satisfaction excellence for the second year in a row. J.D. Power and Associates also conducted a random survey of TomTom customers who had contacted the call center. The evaluation criteria of the customer support representatives include courtesy, concern for the customer, knowledge, convenience of operating hours, usefulness of the information provided, ease of reaching a representative, and timely resolution. TomTom is very proud to have been the only GPS company to have ever received this honor. TomTom call center operations handled more than 900,000 e-mail, telephone, and fax interfaces in 2008 from its call centers in Concord, Massachusetts, and Watertown, New York. TomTom President Jocelyn Vigreux said, “The recognition is a true testament to the dedication we’ve had and success we’ve achieved in making TomTom customer experience a consistently positive one.”

**Sources:** TomTom News (2008) and RightNow (2008).

**Questions**

1. List and describe three reasons why TomTom decided to look to e-commerce opportunities to meet its needs.
2. Do you think TomTom had to make any sacrifices by switching from a simple Notes-based system to RightNow’s contact center solution? Explain what, if anything, you think was given up.
3. What other companies/industries do you feel would most benefit from the services offered by RightNow and other e-commerce vendors who offer similar services?

**ONLINE RESOURCES**

**available at pearsonhighered.com/turban**


Online Tutorial T1 on business plans
Online Tutorial T2 on eCRM

Online Appendix
A technical appendix on Web site design and creation is available at the book’s Web site (pearsonhighered.com/turban).
Comprehensive Educational Web Sites

e-commercedevelopmentcenter.com/services.asp: E-Commerce Development Center resources and links
fstc.org: Financial Services Technology Consortium
itbriefingcenter.com/programs/gartner_tandemseven.html: User interface innovations
ech.com/casesearch/product_details.cfm?id=79647: Ways virtualization can cut costs

REFERENCES


