

## Chapter 2

# The Origins of Software

### Chapter Overview

The primary purposes of Chapter 2 are to show students that not all of the software associated with a systems development project is developed in-house and to emphasize that analysts should consider several design strategies before choosing one to pursue for further development in design. The secondary purpose is to emphasize that the consideration of a packaged software solution should be done after the analysis efforts are complete, not as a substitute for analysis.

The chapter discusses six sources for software: Information Technology Services Firms, packaged software producers, enterprise-wide solutions (ERP), application and managed service providers (ASPs and MSPs), open-source and in-house developers. In addition, the reasoning that should be followed when choosing among the many options available to an analysis team for developing design strategies. The point is that the “make-versus-buy” decision is not a choice of one or the other, but is in reality a spectrum of choices ranging from make at one end and buy at the other. Just as important, more choices these days are made toward the buy end of the scale. The Request for Proposal (RFP) is shown as an important element to understanding how analysis and design requirements must be translated into a document that external source organizations can review for bid. The chapter includes a discussion of outsourcing, an option for systems development and management that may not occur to many students in their first systems development course. The chapter explains reuse and the four different approaches to reuse.

### Instructional Objectives

Specific student learning objectives are included at the beginning of the chapter. *From an instructor’s point of view, the objectives of this chapter are to:*

1. Show students that in-house developed systems are not the only source for software.
2. Six sources for software need to be understood: Information Technology Services Firms, packaged software producers, enterprise-wide solutions (ERP), Cloud Computing Services, open-source and in-house developers. An understanding of the advantages and disadvantages of each must be shown.
3. Cloud Computing is likely to be new to students and even though they may have heard the term it is important to ensure that they understand what it is and its advantages. Emphasize that it enables customers (firms) to use software they have not installed on hardware they do not own to access services over the Internet on a pay-for-use basis. Emphasize the three key advantages of cloud computing (1) freeing internal IT staff, (2) gaining access to applications faster than via internal development, and (3) achieving lower-cost access to corporate-quality applications. Also mention that cost savings are achieved from not having to purchase and install infrastructure or pay annual licensing or maintenance fees. Another critical aspect is instant scalability; that is customers can temporarily acquire and use more resources than they actually have and thus run very large applications and processes at reasonable costs and in reasonable time frames.

4. Show students how to evaluate off-the-shelf software and why it is important to do thorough analysis first. Also explain the claims made by software and hardware vendors need to be validated by someone outside the vendor organization such as current users and independent software testing centers. Emphasize that vendor information may be biased and even trade publication articles may describe software in a more than realistic positive light. When in doubt check the information out.
5. Discuss the importance of software reuse and how object-oriented and component-based development are the two most common reuse methods. Also emphasize that reuse must be aligned with the organization's overall strategic goals. Explain the four levels of adoption currently being seen in the industry: *ad hoc*, facilitated, managed and designed (Table 2-3 lists the four approaches).

## Classroom Ideas

1. Use Table 2-1 and an updated version of the same information from the most recent Software Magazine survey to begin a discussion of the many, varied sources of software in the marketplace.
2. Use Table 2-2 to summarize the alternative sources for software and how to choose among them for specific software needs. This table can serve as the basis for a discussion of the make-versus-buy decision and can be expanded to include the "not invented here" syndrome.
3. Have students research the proper format and contents for Requests for Proposal and have them create and/or present an RFP (see Problem and Exercise 1). RFP preparation should include discussion of the hardware, software, and organizational issues presented in this chapter.
4. Find a local guest speaker (from your alumni base or a recruiter coming to your campus) from an organization that employs object-oriented design. Have them come in to discuss the level of reuse and the real world issues involved in their organization with promoting the concept, given some of the startup costs and constraints.

## Answers to Key Terms

Suggested answers are provided below. These answers are presented top-down, left-to-right.

3. Outsourcing	5. Reuse
2. Enterprise resource planning (ERP) systems	1. Cloud Computing
4. Request for proposal (RFP)	

## Answers to Review Questions

1. Six sources of software are identified in the text. These include Information Technology Services Firms, packaged software producers, enterprise-wide solutions, application and managed service providers, open-source and in-house developers.

Information Technology Services firms are among the world's leading software producers, and the software they make is not limited only to operating systems and utilities, within this segment custom

software producers, sometimes called consulting firms, develop specific, customized software that matches a client's particular needs.

In contrast, packaged software producers develop all kinds of software for all kinds of markets.

Enterprise solutions (ERP) consist of a series of integrated modules; these modules are integrated to focus on business processes rather than on business functional areas.

Cloud Computing is the provision of applications over the internet such that customers do not need to invest in hardware and software infrastructure and can pay on a per-use basis. A key advantage is that server and storage capacity can be ordered on demand in any quantity required.

Open-source software has risen in popularity in the recent past. This software is available free and is developed and maintained by a community of like-minded people dedicated to improving source code access; with Linux being the most prevalent example.

In-house development requires the resources, especially trained staff, to develop software targeted to an organization's own specific needs. Table 2-2 compares the six sources of software components.

2. When deciding what off-the-shelf software to buy, you should compare products and vendors. Additional criteria include (among others that are more situation-specific): cost, functionality, vendor support, vendor viability, flexibility, documentation, response time, and ease of installation.
3. An RFP is a formal document that provides detailed specifications about a target information system and asks vendors for information on how they would develop the system. Analysts use RFPs as a way to get vendors to perform the research to determine what application design will meet user requirements and the hardware and systems software vendors believe are necessary for developing the new system.
4. To verify vendor claims about a software package, an analyst can ask for a software demonstration, use the software (and its documentation and training materials) personally, talk with other users of the software, and consult independent software testing and abstracting services (surveys available for a fee). It is important to make sure that the system fits your organizations' needs and is not just being applied because "everyone else is doing it."
5. Enterprise resource planning systems consist of a series of integrated modules; these modules pertain to specific, traditional business functions. However, these modules are integrated to focus on business processes rather than on business functional areas. Enterprise resource planning systems advantages include a single repository of data for all aspects of a business process, flexible modules, less maintenance, more consistent and accurate data, and ease of adding and integrating new modules into the existing system. Possible disadvantages of this approach include complexity, lengthy implementation, lack of in-house expertise, expense, and changing how the organization conducts its business. These projects when accomplished successfully are most often approached as an institutional change project, not simply an IT project.
6. Reuse is most often applied to object-oriented and component-based development. Creating, storing and maintaining objects and components that can be drawn on again and again for new applications is the objective. Reuse should in theory increase programmer productivity, decrease development time, minimize errors and schedule overruns. Ultimately it should produce higher quality work with fewer defects and thus reduce overall implementation and maintenance time. In current practice, due to high initial start up costs, lack of good quality methods for labeling, storage, combined with lack

of senior management commitment, reuse is not practiced as often as it could be. Additionally, lack of incentives and rewards to design for and apply reuse concepts and the overall difficulty in measuring economic gain from its application all conspire against reuse in the real world. Note Figure 2-5 on the high initial startup costs when a high level of reuse is planned. As more organizations achieve success and as more componentization takes place in the *for purchase* arena more organizations will have incentive to integrate reuse into their business strategy.

7. In comparing and contrasting the four approaches to reuse, the student should note the advantages and disadvantages listed in Table 2-3. Note also that no one type yields the best possible solution.

## Answers to Problems and Exercises

1. An organization uses the Request for Proposal (RFP) to solicit proposals from several competing vendors. Usually, RFPs first provide background information on the company and the business units involved in the request, an explanation of the information systems needs, a description of what is wanted from the vendors (i.e., what information they must provide or other actions they must take), and an explanation of any rules or procedures for the RFP and system development process. The bulk of the document then describes the mandatory, essential, and desirable requirements in the areas of need (e.g., functionality, hardware, software, and service). Students' RFP outlines should include these key features.
2. In addition to cost, functionality, vendor support, vendor viability, flexibility, documentation, response time, and ease of installation, a number of other "real-world" criteria might be included. People often choose application packages, such as word processors and spreadsheets, based solely on their familiarity with the packages and/or their bias toward one hardware platform or operating system over another. To a certain extent this is functional. On the other hand, this can be dysfunctional. For example, it is useful to consider the current staff's familiarity with the new application software and the resulting need for retraining. However, it is dysfunctional if a company does not choose new software because of the employees' lack of familiarity with the software. Eventually, software will evolve and the market will change, and the employees (and the company) will be left behind using antiquated technology. Additional criteria include compatibility with currently used application software (so, for example, data can be shared), compatibility with existing hardware and system software, ability to support a range from novice to experienced (or power) users, and appeal of the user interface (ease of use).
3. The list for evaluating alternative custom software developers is similar to that for selecting off-the-shelf application software or for computer hardware and system software. In addition to cost, functionality, vendor support, vendor viability, flexibility, documentation, response time, and ease of installation, you might include the current staff's familiarity with the software, need for retraining, compatibility and connectivity with current systems, and the track record of the vendor in successfully implementing similar software in other organizations. Such vendors should have an established track record for developing similar software in other organizations. Their references should be checked thoroughly, including visits to other sites. If the developer's role ends after the application is accepted, then the vendor's reputation for handling this transition from external development to internal maintenance is important. From a legal point of view, you may want to select a custom developer based on the willingness to sign a non-disclosure agreement, so that he is not allowed to develop a similar system for one of your competitors, at least for a certain amount of time.
4. The project team can use the advantages of the enterprise resource planning design as part of its strategy for selling this system. The team can stress that this solution consists of a series of

integrated modules; these modules are integrated to focus on business processes, and the firm can integrate all parts of a business process. This approach includes a single repository of data, thus providing more consistent and accurate data and less maintenance. These modules are flexible and new modules are easily integrated into an existing system.

Also, an enterprise resource planning (ERP) system might be justified on the following grounds: (1) it is a complete enterprise-wide solution that models all aspects of each transaction, supposedly seamlessly and within a single system; (2) an ERP system is based on a single repository of all corporate data, which implies consistency, accuracy and flexibility of the data; and (3) adding new modules should be relatively painless as all modules are specifically designed to work together. On the other hand, some might counter that ERP systems are complex and require expensive outside expertise for implementation, and that the organization itself has to adjust to fit the software's model of how organizations should operate rather than the reverse.

## **Guidelines for Using the Field Exercises**

1. Business people are likely to use this list of criteria in some formal or informal way. Have your students present their findings to the class so that they can learn about a variety of companies. It is useful for students to see how companies actually use these criteria and methods in purchasing off-the-shelf software.
2. Obtaining RFPs may be difficult for your students. It may be necessary for you to obtain copies of RFPs either from business contacts or from the university. It is very useful for students to see real RFPs. Students are amazed at how lengthy and detailed these are for larger, more complex systems and at how complicated RFPs can become for governmental agencies.
3. Chances are any organization that a student contacts about its ERP implementation will have a lot to say, provided the IT staff is willing to share the information. Typically, ERP implementations take several years and cost quite a bit in terms of consultant fees. There are many reasons to move to an ERP system, just as there are many reasons not to move to an ERP system. The organization was probably attracted to the promise of uniformity and consistency made by ERP vendors, although the exact reasons differ from firm to firm. Chances are good that the organization has made some internal changes, such as realigning departments internally to take advantage of the opportunities ERP systems offer, as well as to meet the demands ERP systems make in order to operate effectively. The chances are also good that most of the implementation work has been done by outside consultants, so for firms not used to managing large numbers of contractors, an ERP implementation is a new and different experience. The implementation is likely still going on at whatever firm a student happens to talk with, but it has probably been going on for many months or years, as each ERP implementation is a learning experience for the consultants and the adopting firm.