Chapter 2
The Sources of Software

Chapter Overview

Systems Analysis and Development involves developing software, but also involves purchasing or in other ways obtaining various types of software to accomplish various SDLC activities. Chapter 2 discusses where software can be obtained, advantages and disadvantages of various sources, and how the systems analyst decides on a vendor and manages software procurement decisions.

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. Identify and discuss information systems development options, including IT services firms, packaged software producers, enterprise-wide solutions, application service providers, open-source software, and in-house developers.

2. Define and understand outsourcing and its implications.

3. Discuss choosing off-the-shelf software.

4. Make comparisons between off-the-shelf solutions and in-house solutions and validate purchased software.

5. Understand how and why reusing software can be a useful option.

Classroom Ideas

1. This is a relatively short chapter. It may be easier to use the end of chapter elements for lectures. The review questions and problems and exercises provide targeted topics for discussion in terms of discussing the many sources of software.

2. Emphasize how many sources of software are available. Students should understand that being a systems analyst does not necessarily mean building code from scratch. Illustrate that, while a systems analyst might need a unique solution for their particular organization, there are many sources where the analyst can acquire already existing pieces that can be brought together to build their unique solution.

3. It can be useful to discuss the history of systems acquisition and show how changes in technology and availability have influenced where software can now be acquired.

Copyright © 2015 Pearson Education, Inc.

4. If possible, try to obtain off-the-shelf validation documents (RFP or RFQ) from an actual organization to illustrate what these documents contain.

5. In lecturing about this chapter, you might want to use the tables and figures in the chapter to point out to students the major issues at stake in sourcing software.

Lecture Notes

Introduction

Software is available from many sources. Some firms still do in-house programming, but many others acquire software from a variety of sources. This chapter discusses the various sources of software, from outsourcing to open source and off-the-shelf options. This chapter illustrates how systems analysts can make decisions about choosing from the many sources, and also discusses reusing software from previous sources.

Systems Acquisition

Considering how important information systems are to organizations, when did they begin to be used, and where did they come from? GE produced an in-house payroll option in 1954. Since then, in-house development has become a smaller component of many organizations' systems development budgets as other sources of software have become available and more strategically advantageous than developing software in-house.

Outsourcing

Outsourcing is a practice where another organization develops or runs a computer application for your organization. Outsourcing can be very cost effective, and many organizations choose to outsource much, if not all, of their IT functions. This has resulted in a very large market for outsourcing—$930 billion in 2006. Considering the availability and cost-effectiveness of outsourcing, analysts should be aware of the option and carefully consider it as a potential alternative.

Sources of Software

Methods of acquiring software can be grouped into 6 major categories—(1) **information technology services firms**, (2) **packaged software providers**, (3) **vendors of enterprise solution software**, (4) **application service providers and managed service providers**, (5) **open-source software**, and (6) **in-house development**. Each of these sources are presented in **Figure 2–1** and **Table 2–2** provides a comparison of the 6 sources of software**.**

Choosing Off-The-Shelf Software

If an analyst chooses to purchase software, s/he should understand what criteria they should evaluate before making a decision, and how to compare the options available to them. The most common criteria for evaluating off-the-shelf software relative to in-house

development are shown in **Figure 2–2.** Each of the criteria may be more important than the others depending on various aspects of the project.

Another important consideration when evaluating off-the-shelf options is validating the software information. This can be done by asking the vendor, but can also be carried out by issuing a **Request for Proposal (RFP)**, where vendors can propose hardware and software that will meet the requirements of your new system. Other users of the software are also good sources for feedback and validation.

Reuse

In addition to the sources of software already identified, some organizations may be able to **Reuse** software, or various components from already developed software. Reuse may apply to many pieces of previously developed software, but most often applies to object-oriented and component-based technologies. Reuse requires the organization to be able to leverage the cost of components across many users and projects. **Figure 2–3** shows a graph illustrating this concept. If an organization makes the decision to reuse, there are four approaches to reuse, which are presented in **Table 2–3.**

Key Terms Checkpoint Solutions

Answers for the Key Terms Checkpoint section are provided below. The number following each key term indicates its location in the key term list.

1. Outsourcing (3)

2. Enterprise resource planning (ERP) systems (2)

3. Request for proposal (RFP) (4)

4. Reuse (5)

5. Cloud Computing (1)

Review Questions Solutions

2-1. Describe and compare the six sources of software.

The six sources of software identified in the textbook are: (1) information technology services firms, (2) packaged software providers, (3) vendors of enterprise solution software, (4) application service providers and managed service providers, (5) open-source software, and (6) in-house development. IT services firms help companies develop custom information systems for internal use; they develop, host, and run applications for customers; or they provide other services. An IT services firm may be chosen if the system can't be developed internally or requires customer support. Packaged software providers are companies that produce software exclusively, like Microsoft or Intuit, and are preferable if the task needing the system is generic. Vendors of enterprise solution software create a system that is composed of a series of integrated modules. Each module supports a business function, such as accounting, or human resources. ERP systems may be appropriate if a complete system is required that can cross functional boundaries. A more intense option for larger,

more customizable solutions are Managed service providers who can provide more services than application service providers. ASPs and MSPs may be appropriate when instant access to an application is desired, and in the case of ASPs, when the task is generic. Open-source software is a type of software that is developed by a community of interested people and it is freely available. Open-source software is preferable when the task is generic and cost is an issue. Finally, in-house development is still an option. In-house development refers to the organization developing software with its own IT assets, rather than purchasing software elsewhere. In-house development may be appropriate when a system must be built from scratch and as long as the personnel and resources are available to the organization.

2-2. How can you decide among various off-the-shelf software options? What criteria do you use?

To decide what off-the-shelf software to buy, compare products and vendors. Use the following criteria (among others that may be more situation-specific): cost, functionality, vendor support, viability of vendor, flexibility, documentation, response time, and ease of installation.

2-3. What is an RFP and how do analysts use one to gather information on hardware and system software?

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 necessary research into specific design strategies and the hardware and system software vendors believe are necessary for developing the new system.

2-4. What methods can a systems analyst employ to verify vendor claims about a software package?

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, ask specific questions via a questionnaire, and consult independent software testing and abstracting services.

2-5. What are ERP systems? What are the benefits and disadvantages of such systems as a design strategy?

ERP is a system that is composed of a series of integrated modules. Each module supports a business function, such as accounting, or human resources. An advantage of ERP systems is that the modules are integrated across business functional areas, so that, for instance, all aspects of a single transaction can occur seamlessly within one information system, where it would have had to have been processed multiple times across several systems in a more traditional environment focused on functional areas. Other benefits include having a single repository of data for all aspects of a business process and the flexibility of the modules; other modules can be added as needed and are immediately integrated into the existing system. Disadvantages include complexity, lack of knowledge

among organization's personnel and reliance on consultants, and expense. In some cases, an organization may be required to change how it does business to more efficiently utilize ERP systems.

2-6. Explain reuse and its advantages and disadvantages.

Reuse refers to using previously written software resources, especially objects and components, in new applications. It can be more efficient to reuse objects or components, and research has shown that it can increase productivity, reduce defects, and reduce the amount of rework needed. Disadvantages include organizational lack of commitment, lack of proper training and rewards needed to promote reuse, and it is difficult to measure the economic gains from reuse.

2-7. Compare and contrast the four approaches to reuse.

"Ad Hoc" reuse refers to individuals finding ways to reuse objects and components on their own. Facilitated reuse refers to reuse that is encouraged by the organization, but with limited resources, infrastructure, and policies that would make reuse easier. Managed reuse refers to reuse that occurs because the organization enforces reuse through resources, infrastructure, policies, and people. Designed reuse refers to an organization carefully designing assets for reuse.

2-8. Why would a company rely on cloud computing for its software needs?

Cloud computing frees a company's internal IT from developing application which results in a lower cost application as well as faster development time. In addition, since cloud computing in general is a pay as you go type service, applications can be quickly scaled up or down depending upon the needs of the company.

Problems and Exercises Solutions

2-9. Research how to prepare a request for proposal.

The request for proposal (RFP) is used when the organization wants to solicit proposals from several competing vendors. RFPs usually first give some background information on the company and the business units involved in the request, an explanation of the information system 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).

2-10. Review the criteria for selecting off-the-shelf software presented in this chapter. Use your experience and imagination and describe other criteria that are or might be used to select off-the-shelf software in the real world.

For each new criterion, explain how its use might be functional (i.e., it is useful to use this criterion), dysfunctional, or both.

In addition to cost, functionality, vendor support, viability of vendor, flexibility, documentation, response time, and ease of installation, there are a number of other real world criteria that 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 can be functional. However, this could also be dysfunctional. For example, it would be useful to consider the current staff's familiarity with the new application software and the resulting need for retraining. However, it would be dysfunctional if a company did not ever choose new software because of the employees' lack of familiarity with it. Eventually, software will evolve and the market will change, and the employees (and the company) will be left behind using antiquated technology.

Some other 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).

2-11. In the section on choosing off-the-shelf software, eight criteria are proposed for evaluating alternative packages. Suppose the choice was between alternative custom software developers rather than prewritten packages. What criteria would be appropriate to select and compare among competing bidders for custom development of an application? Define each of these criteria.

The list for evaluating alternative custom software developers would be very similar to that for selecting off-the-shelf application software or for computer hardware and system software. In addition to cost, functionality, vendor support, viability of vendor, flexibility, documentation, response time, and ease of installation, we 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 of developing similar software in other organizations. Their references should be checked thoroughly, including visits to these other sites. If the developer's role will end after the application is accepted, then the reputation of the vendor 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 they are not allowed to develop a similar system for one of your competitors, at least for some amount of time.

2-12. How might the project team recommending an ERP strategy justify its recommendation as compared with other types of design strategies?
The project team would need to justify an ERP strategy by highlighting the strategic advantage to reengineering business processes to integrate with ERP modules. Doing so will provide a single repository for data, which can streamline all IT within the organization and ensure that data is consistent and accurate.

Furthermore, if expansion is required in the future, additional modules are able to be integrated immediately. Finally, considering the regulatory climate, the ERP vendor (and consultants) will likely be better at ensuring the organization's compliance with regulations, such as Sarbanes-Oxley, HIPAA, etc.

Petrie's Electronics Case Questions Solutions

2-16.How do information systems projects get started in organizations?

The two primary ways in which information systems projects get started in organizations are, simply put, top-down or bottom-up. Top-down projects start with management, where managers determine that a new system is needed or an existing system needs to be updated or replaced. Bottom-up projects start with lower-level employees, who determine that systems they use are inadequate and either need to be replaced or updated.

2-17. How are organizational information systems related to company strategy? How does strategy affect the information systems a company develops and uses?

A company's strategy should drive what it does and the goals the company should pursue. A company's strategy is implemented through many tactical decisions, each of which relies on tools and technologies to fulfill. Information systems are some of the tools and technologies companies use to fulfill decisions and hence implement strategy.

2-18. Research customer loyalty programs in retail firms. How common are they? What are their primary features?

Customer loyalty programs are very common in retail firms. Most students probably belong to several such programs. The most primary feature involves awarding points for purchases, points which can later be redeemed for discounts or prizes. Research into these programs will generate lists of other features, and students may also find numbers on how many such programs actually exist.

2-19. What do you think Jim's next step would be? Why?

This answer will differ with each student, but looking ahead in the systems development life cycle, the next step should involve some preliminary data gathering and analysis, especially in terms of feasibility analysis. Jim and his team need to determine if the proposed system makes sense for the company, from a variety of perspectives, before proceeding.

2-20. Why would a systems analyst new to a company be a good choice to lead an important systems development effort?

An analyst new to the company would not have any preconceptions about the system and how it might fit into the company's strategy. Having a fresh perspective might allow for new insights and a new approach that could lead to a better and more creative system than would otherwise result.