

King Saud University

College of Engineering

IE – 462: “Industrial Information Systems”

Spring – 2020 (2<sup>nd</sup> Sem. 1440-41H)

Chapter 2

*Information System Development – p2*

Prepared by: Ahmed M. El-Sherbeeney, PhD

# Lesson Overview

- Sources of Software
- Criteria for Selecting Software

# Sources of Software



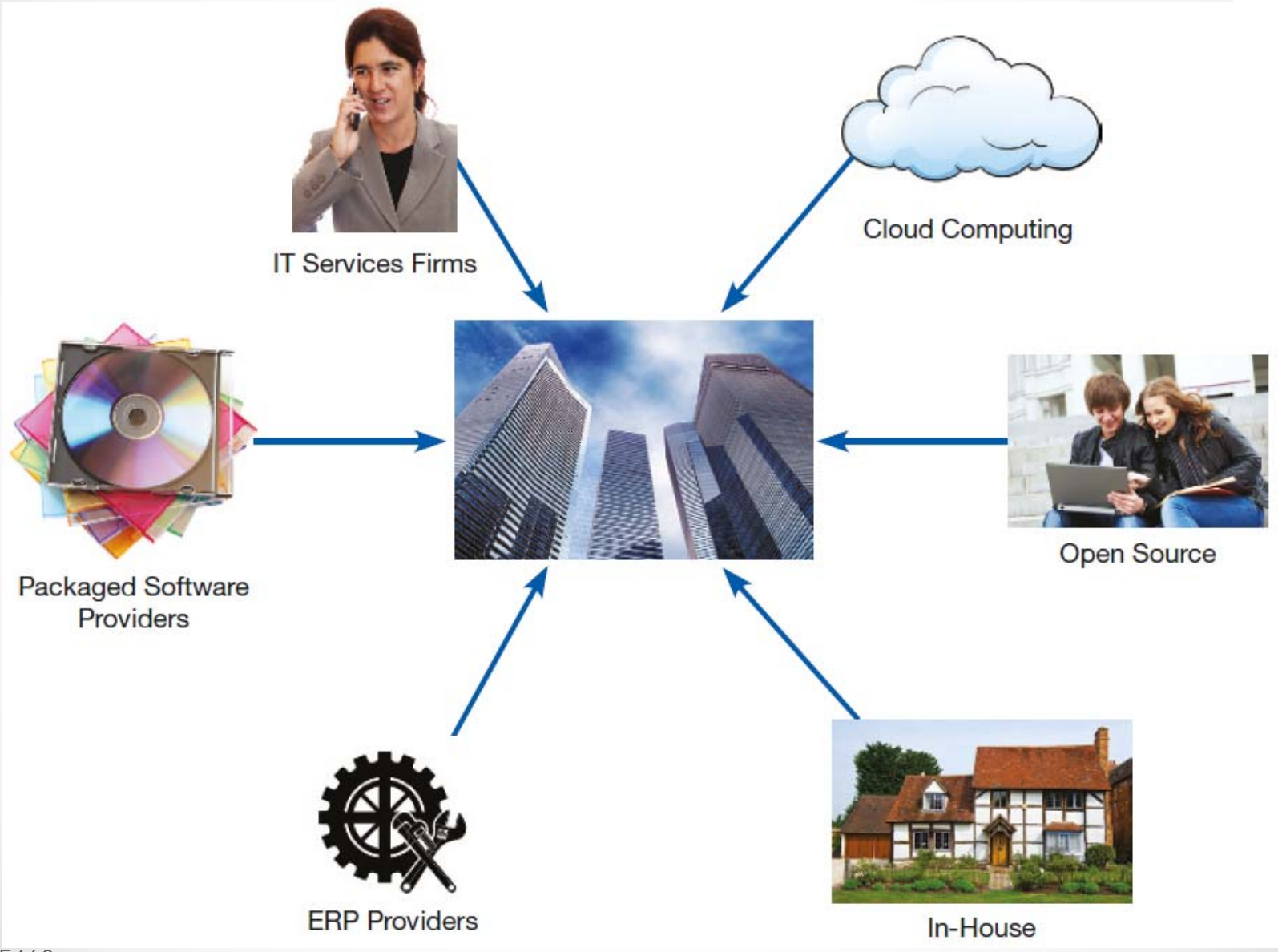
# Sources of Software

- There are various sources of software for organizations
- First administrative information system: US, General Electric's (GE) payroll system (1954) as *in-house development*
- There are various criteria used to evaluate software

# Sources of Software

1. Information technology services firm (**outsourcing**)
2. Packaged software producers
3. Enterprise-wide solutions (ERP)
4. Cloud computing
5. Open source software
6. In-house development

# Sources of Software



# Sources of Software

**TABLE 2-1** Leading Software Firms and Their Development Specializations

| Specialization                | Example Firms or Websites                                     |
|-------------------------------|---|
| IT Services                   | Accenture<br>Computer Sciences Corporation (CSC)<br>IBM<br>HP |
| Packaged Software Providers   | Intuit<br>Microsoft<br>Oracle<br>SAP AG<br>Symantec           |
| Enterprise Software Solutions | Oracle<br>SAP AG  |
| Cloud Computing               | Amazon.com<br>Google<br>IBM<br>Microsoft<br>Salesforce.com    |
| Open Source                   | SourceForge.net   |

# 1. Information Technology (IT) Services Firms

- **Outsourcing:** turning over responsibility of some or all of an organization's information systems applications and operations to an **outside firm**

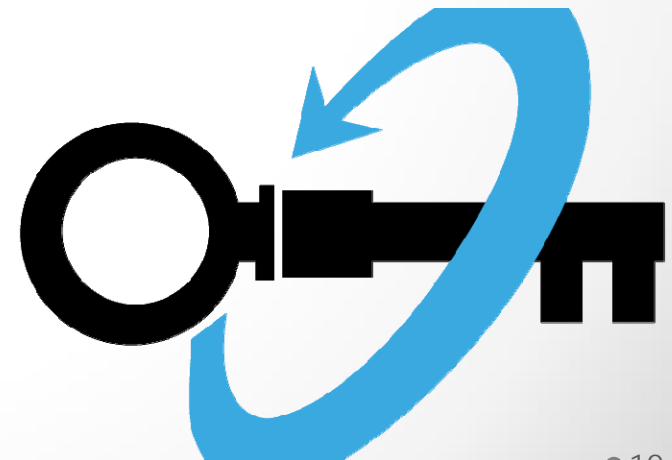


# 1. Information Technology (IT) Services Firms

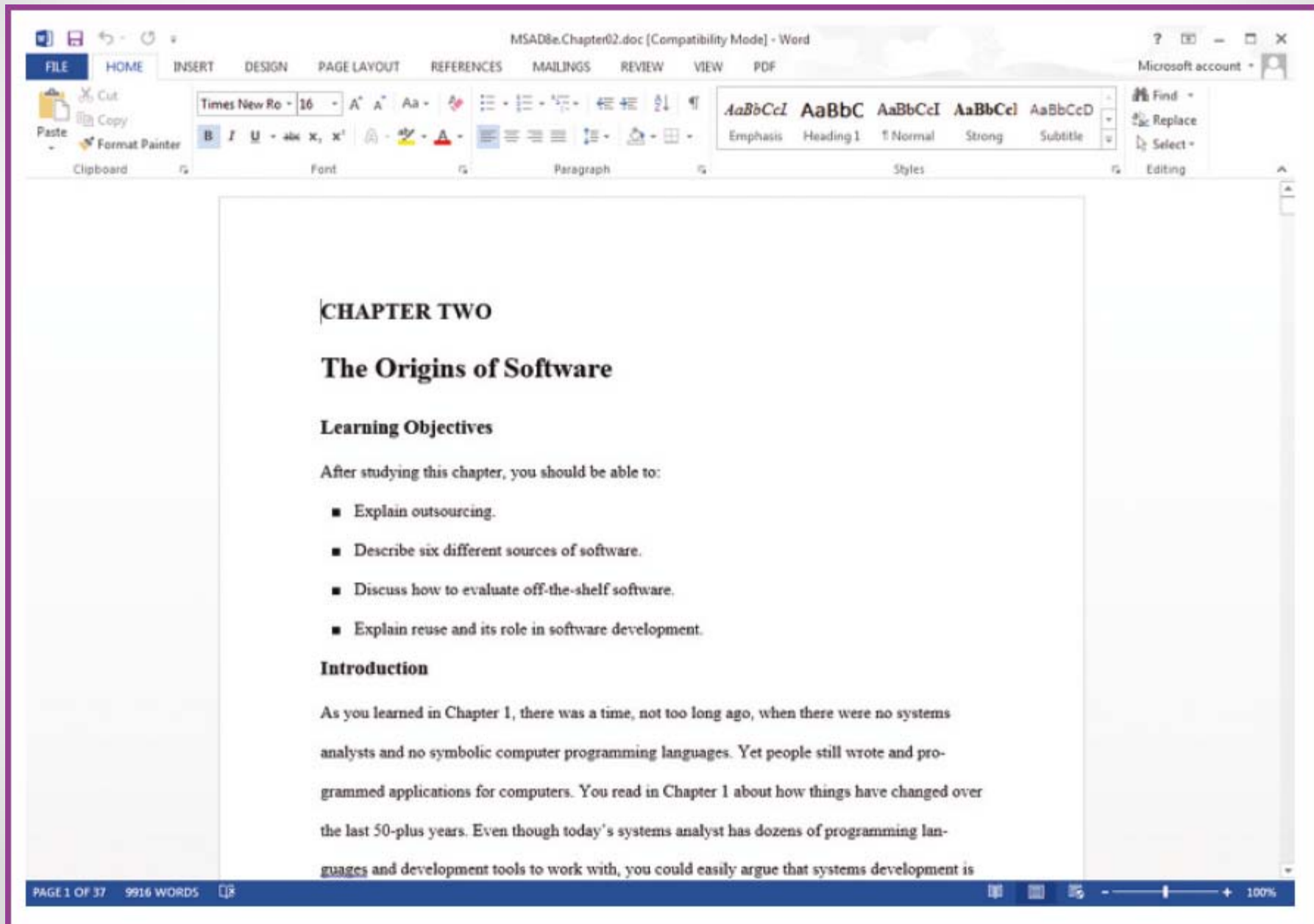
- **Reasons to outsource:**
  - Cost-effective
  - Take advantage of *economies of scale*
  - Free up internal resources
  - Reduce *time to market*
  - Increase process efficiencies
  - When system development is a non-core activity for the organization
  - Help companies develop custom IS for internal use
  - IT service firms develop, host, and run applications for customers

## 2. Packaged / Off-The-Shelf Software Producers

- Serve many market segments
- Provide variety of software:
  - broad-based packages (e.g. productivity tools)
  - specialized packages (e.g. software to manage small store)
- Software runs on all size computers, from *microcomputers* to large *mainframes*
- Prepackaged software is:
  - off-the-shelf (e.g. MS. Project)
  - *turnkey* software (i.e. not customizable\*); see [e.g.](#)



## 2. Packaged / Off-The-Shelf Software Producers



MSAD8e.Chapter02.doc [Compatibility Mode] - Word

FILE HOME INSERT DESIGN PAGE LAYOUT REFERENCES MAILINGS REVIEW VIEW PDF

Clipboard Font Paragraph Styles Editing

Find Replace Select

Microsoft account

**CHAPTER TWO**

**The Origins of Software**

**Learning Objectives**

After studying this chapter, you should be able to:

- Explain outsourcing.
- Describe six different sources of software.
- Discuss how to evaluate off-the-shelf software.
- Explain reuse and its role in software development.

**Introduction**

As you learned in Chapter 1, there was a time, not too long ago, when there were no systems analysts and no symbolic computer programming languages. Yet people still wrote and programmed applications for computers. You read in Chapter 1 about how things have changed over the last 50-plus years. Even though today's systems analyst has dozens of programming languages and development tools to work with, you could easily argue that systems development is

PAGE 1 OF 37 9916 WORDS 100%

# 3. Enterprise Solutions Software

- **Enterprise Resource Planning (ERP)** systems:
  - *integrate* individual traditional business *functions* into *modules*
  - thus, enabling a single *seamless transaction* to cut across functional boundaries
  - e.g. series of modules will support entire order entry process:
    - from receiving order
    - to adjusting inventory
    - to shipping
    - to billing
    - to after-the-sale service\*

## 3. Enterprise Solutions Software

- Top ERP producers:
  - **SAP AG:** German company; since 1972 is the leading vendor of ERP systems
  - **Oracle:** US company; famous for database software; both control 36% of ERP market



**ORACLE®**

# 3. Enterprise Solutions Software

- Advantages of ERP system:
  - single **repository of data** for *all* aspects of a business process
  - flexibility of the modules (easy to integrate new modules)
  - consistent and accurate data
  - less maintenance

# 3. Enterprise Solutions Software

- Disadvantages of ERP system:
  - systems are very complex
  - implementation can take a long time to complete
  - usually very expensive to hire consultants to install system
  - “migration” to new system involves changing how organizations do business

# 3. Enterprise Solutions Software

- Example here:
  - SAP's *Business ByDesign* product
  - designed for medium-sized companies





# 4. Cloud Computing

- Cloud Computing in brief:
  - it is the provision of computing resources, including applications, *over the Internet*
  - customers do not have to invest in the *computing infrastructure* needed to run and maintain the resources
  - third-party providers run applications at remote sites
  - users have access to applications through the Internet or through virtual private networks
  - application provider buys, installs, maintains, and upgrades the applications

## 4. Cloud Computing

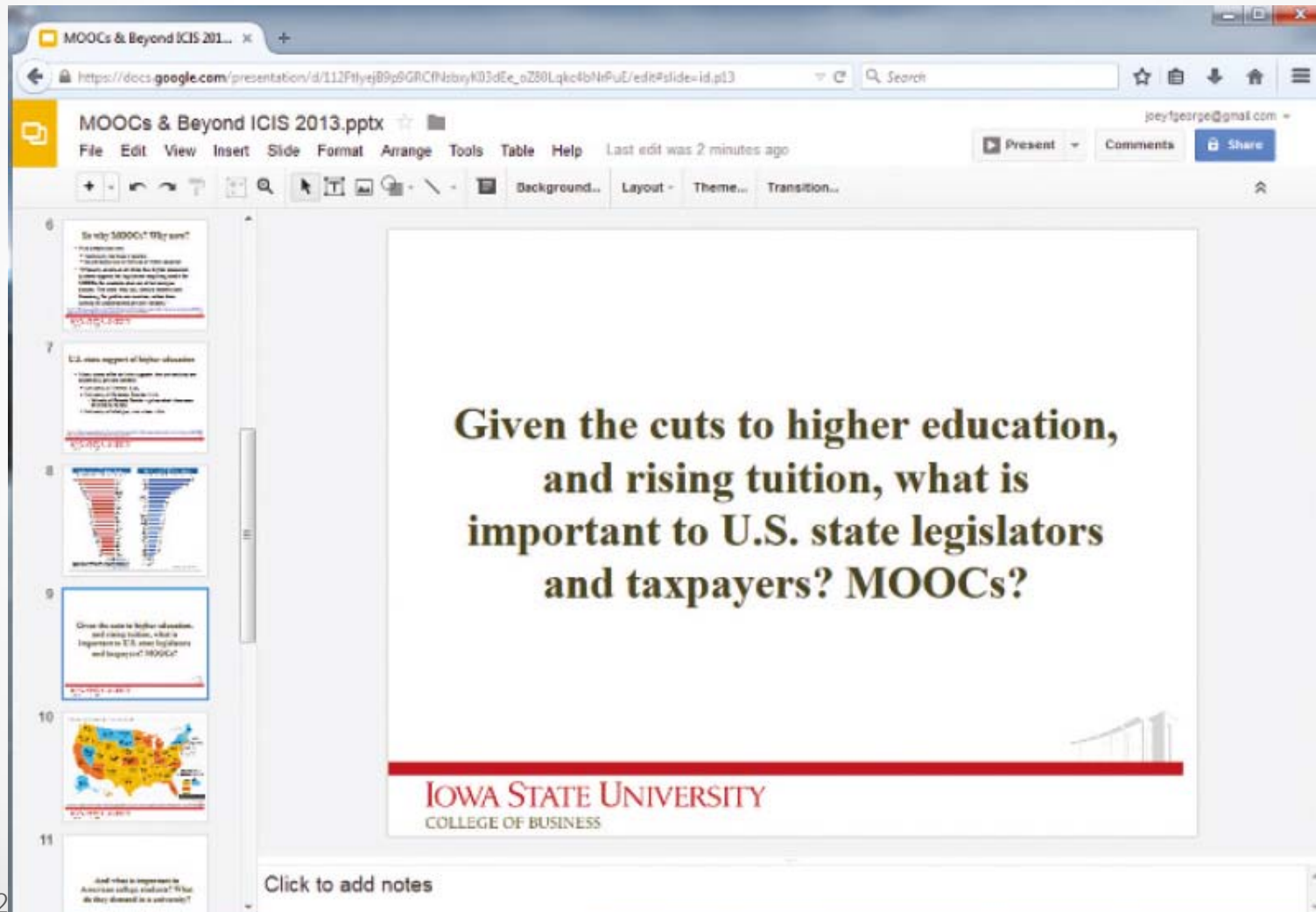
- Advantages of cloud computing:
  - freeing internal IT staff
  - gaining access to applications faster than via internal development
  - achieving low-cost access to corporate-quality applications
  - no expensive, time-consuming system implementation
  - cost effectiveness, speed to market, and better performance

# 4. Cloud Computing

- Disadvantages of cloud computing :
  - *security* concern: storing company data (e.g. customer information) on machines one does not own and that others can access
  - reliability: vulnerable to unexpected risks due to its complexity
  - compliance with government regulations

# 4. Cloud Computing

- A presentation edited in *Google Apps*:



## 5. Open Source Software

- Freely available, including source code
- Developed by a large community\* of interested people
- Performs the same functions as commercial software (e.g. operating systems, e-mail, database systems, web browsers)
- e.g. *Linux*, *mySQL* (database system), *Firefox*
- Note, companies/people can make money through maintenance, support, and selling fully-featured versions

## 6. In-House Development

- In-House Development in brief:
  - some or all of the system is developed by the organization's own staff
  - condition: sufficient system development expertise (for the chosen platform) exists in-house
  - becoming a progressively smaller piece of all systems development work that takes place in and for organizations (due to large maintenance burden)\*
  - *hybrid solutions* involving some purchased and some in-house components are common

# Criteria for Selecting Software



# Comparing Sources of Software Components

**TABLE 2-2 Comparison of Six Different Sources of Software Components**

| Producers                         | When to Go to This Type of Organization for Software   | Internal Staffing Requirements                                      |
|-----------------------------------|--|---|
| IT services firms                 | When task requires custom support and system can't be built internally or system needs to be sourced | Internal staff may be needed, depending on application              |
| Packaged software producers       | When supported task is generic   | Some IS and user staff to define requirements and evaluate packages |
| Enterprise-wide solutions vendors | For complete systems that cross functional boundaries  | Some internal staff necessary but mostly need consultants           |
| Cloud computing                   | For instant access to an application; when supported task is generic                                 | Few; frees up staff for other IT work                               |
| Open-source software              | When supported task is generic but cost is an issue  | Some IS and user staff to define requirements and evaluate packages |
| In-house developers               | When resources and staff are available and system must be built from scratch                         | Internal staff necessary though staff size may vary                 |



# Criteria for Selecting Software

- **Cost:** comparing the cost of developing in-house with the cost of purchasing or licensing the software pack
- **Functionality:** tasks that the software can perform
- **Vendor support:** how much support the vendor provides and at what cost
- **Viability of vendor:** can the software adapt to changes in systems software and hardware

## Criteria for Selecting Software (cont.)

- **Flexibility:** how easy it is to customize the software
- **Documentation:** is the user's manual and technical documentation understandable and up-to-date?
- **Response time:** how long it takes the software package to respond to the user's requests in an interactive session
- **Ease of installation:** a measure of the difficulty of loading the software and making it operational

# Source

- Modern Systems Analysis and Design. Joseph S. Valacich and Joey F. George. Pearson. Eighth Ed. 2017. Chapter 2.