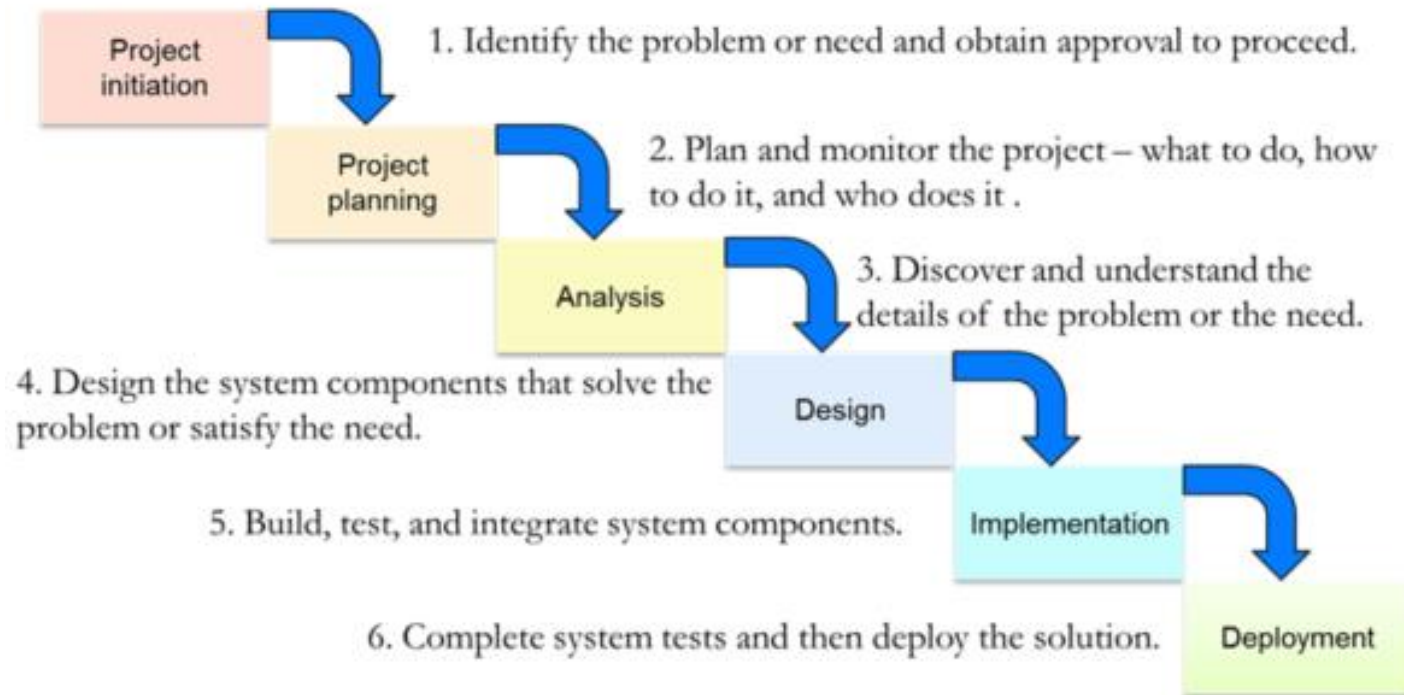# COMP1010 – WEEK
SOFTWARE DEVELOPMENT

Dr. Raymond Chiong, Dr. Mira Park, Dr. Mark Wallis

COMP1010 – Introduction to Computing

University of Newcastle

System Development Life Cycle (SDLC)

Project initiation

1. Identify the problem or need and obtain approval to proceed.

Project planning

2. Plan and monitor the project – what to do, how to do it, and who does it .

Analysis

3. Discover and understand the details of the problem or the need.

4. Design the system components that solve the problem or satisfy the need.

Design

5. Build, test, and integrate system components.

Implementation

6. Complete system tests and then deploy the solution.

Deployment

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

Chapter Objectives

- Describe systems analysis phase activities

- Explain joint application development (JAD), rapid application development (RAD), and agile methods

- Use a functional decomposition diagram (FDD) to model business functions and processes

- Describe the Unified Modeling Language (UML) and examples of UML diagrams

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

Chapter Objectives (Cont. 1)

- List and describe system requirements, including outputs, inputs, processes, performance, and controls

- Explain the concept of scalability

- Use fact-finding techniques, including interviews, documentation review, observation, questionnaires, sampling, and research

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

Chapter Objectives (Cont. 2)

- Define total cost of ownership (TCO)
- Conduct a successful interview
- Develop effective documentation methods to use during systems development

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

Systems Analysis Phase Overview

- Objectives
  - Understand the proposed project
  - Ensure that it supports business requirements
  - Build a solid foundation for system development

- **Systems Analysis Activities**
  - Requirements modeling
    - Involves fact-finding to describe the current system and identification of the requirements for new system (inputs, outputs, process, performance, and security)
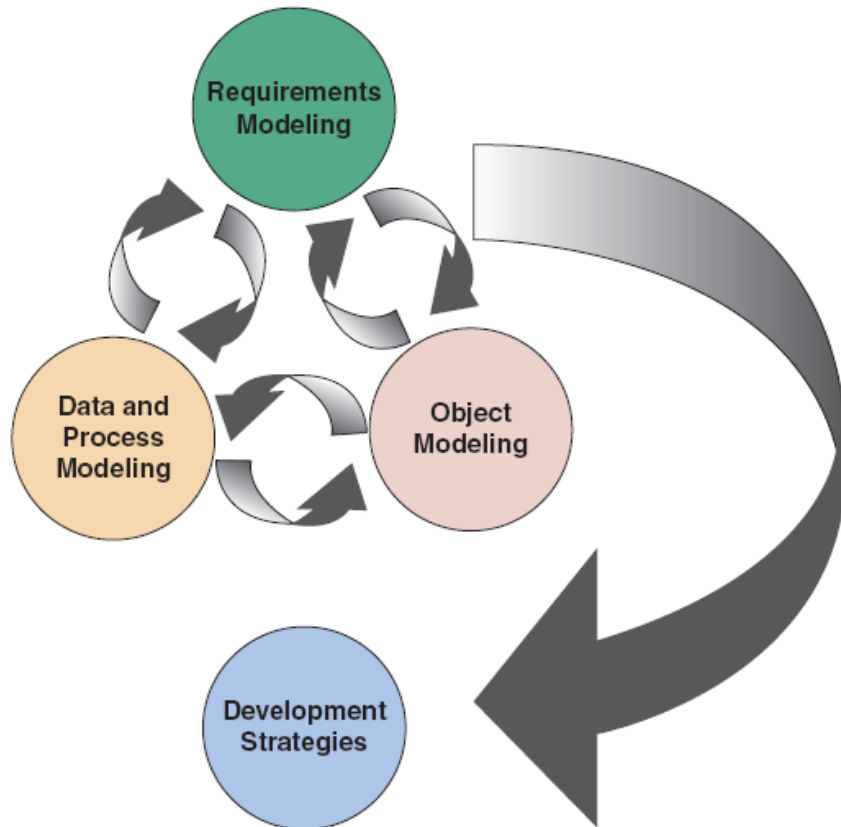   - It's not easy: https://www.youtube.com/watch?v=lXNu0VBVCUc

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

FIGURE 4-1 An illustration of the challenges inherent in understanding system requirements.

Systems Analysis Phase Overview (Cont. 1)

Systems Analysis Phase Tasks

– Data and process modeling
  • Graphically represents system data and processes

– Object modeling
  • Involves creation of objects to represent people, things, transactions, and events

– Development strategies
  • Include software trends, development alternatives, and outsourcing

FIGURE 4-2 The systems analysis phase consists of requirements modeling, data and process modeling, object modeling, and consideration of development strategies. Notice that the systems analysis tasks are interactive, even though the waterfall model generally depicts sequential development

THE UNIVERSITY OF NEWCASTLE AUSTRALIA

Systems Analysis Phase Overview (Cont. 2)

- **Systems Analysis Skills**
    - Strong analytical and interpersonal skills
- **Team-Based Techniques: JAD, RAD, and Agile Methods**
    - Goal - To deliver the best possible system at the lowest possible cost in the shortest possible time
        - Joint application development (JAD) brings users into the design process
        - Rapid application development (RAD) is a condensed version of the system development life cycle
        - Agile methods stress intense interaction between developers and users

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

Joint Application Development

- Brings users into the development process as active participants

- **User Involvement** (formal or informal)
  - Helps create a successful system

- ## JAD Participants and Roles
  - Project leader and one or more members
  - Participants should be insulated from distractions of day-to-day operations

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

Joint Application Development (Cont. 1)

JAD PARTICIPANT	ROLE
JAD project leader	Develops an agenda, acts as a facilitator, and leads the JAD session
Top management	Provides enterprise-level authorization and support for the project
Managers	Provide department-level support for the project and understanding of how the project must support business functions and requirements
Users	Provide operational-level input on current operations, desired changes, input and output requirements, user interface issues, and how the project will support day-to-day tasks
Systems analysts and other IT staff members	Provide technical assistance and resources for JAD team members on issues such as security, backup, hardware, software, and network capability
Recorder	Documents results of JAD sessions and works with systems analysts to build system models and develop CASE tool documentation

FIGURE 4-3 Typical JAD participants and roles

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

Joint Application Development (Cont. 2)

FIGURE 4-4 Typical agenda for a JAD session

Project leader	• Introduce all JAD team members • Discuss ground rules, goals, and objectives for the JAD sessions • Explain methods of documentation and use of CASE tools, if any
Top management (sometimes called the project owner or sponsor)	• Explain the reason for the project and express top management authorization and support
Project leader	• Provide overview of the current system and proposed project scope and constraints • Present outline of specific topics and issues to be investigated
Open discussion session, moderated by project leader	• Review the main business processes, tasks, user roles, input, and output • Identify specific areas of agreement or disagreement • Break team into smaller groups to study specific issues and assign group leaders
JAD team members working in smaller group sessions, supported by IT staff	• Discuss and document all system requirements • Develop models and prototypes
Group leaders	• Report on results and assigned tasks and topics • Present issues that should be addressed by the overall JAD team
Open discussion session, moderated by project leader	• Review reports from small group sessions • Reach consensus on main issues • Document all topics
Project leader	• Present overall recap of JAD session • Prepare report that will be sent to JAD team members

NEWCASTLE
AUSTRALIA

Joint Application Development (Cont. 3)

- **JAD Advantages and Disadvantages**
  - Disadvantages
    - More expensive than traditional methods
    - Can be cumbersome if the group is too large
  - Advantages
    - Allows key users to participate effectively
    - Users are more likely to feel a sense of ownership
    - Produces a more accurate statement of system requirements, a better understanding of common goals, and a stronger commitment to the success of the new system

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

Rapid Application Development

- Uses a group approach like JAD
- End product - New information system
- Complete methodology
  - Includes a four-phase life cycle that parallels the traditional SDLC
  - Reduces cost and development time
  - Increases the probability of success
  - Relies on prototyping and user involvement
    - Prototypes are modified based on user input

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

Rapid Application Development (Cont. 1)

Requirements Planning Tasks
- Users, managers, and IT staff agree upon business needs, project scope, and systems requirements
- Obtain approval to continue

Requirements Planning

Continuous interaction between the user design and construction phases

User Design Tasks
- Interact with users
- Build models and prototypes
- Conduct intensive JAD-type sessions

User Design

Construction

Construction Tasks
- Program and application development
- Coding
- Unit, integration, and system testing

Cutover

Cutover Tasks
- Data conversion
- Full-scale testing
- System changeover
- User training

FIGURE 4-5 The four phases of the RAD model are requirements planning, user design, construction, and cutover. Notice the continuous interaction between the user design and construction phases.

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

Rapid Application Development (Cont. 2)

- **RAD Objectives**
  - Cut development time and expense
    - Involve users in every phase of systems development
    - Must have the right IT resources, skills, and management support
- **RAD Advantages and Disadvantages**
  - Advantage – Helps develop systems quickly with significant cost savings
  - Disadvantages
    - Does not emphasize the company's strategic business needs
    - Less time to develop quality, consistency, and design standards

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

Agile Methods

- Attempt to develop a system incrementally, by building a series of prototypes and adjusting them to user requirements regularly

- Developers revise, extend, and merge earlier versions into the final product

- Emphasize continuous feedback
  - Each incremental step is affected by what was learned in the prior steps

- See https://www.youtube.com/watch?v=AsFMHnSfI2I

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

Agile Methods (Cont. 1)

- **Scrum**
  - A rugby term
  - Pigs include the product owner, the facilitator, and the development team
  - Chickens include users, other stakeholders, and managers
  - Scrum sessions
    - Have specific guidelines that emphasize time blocks, interaction, and team-based activities that result in deliverable software

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

Agile Methods (Cont. 2)

- **Agile Method Advantages and Disadvantages**
  - Advantages
    - Very flexible and efficient in dealing with change
    - Frequent deliverables constantly validate the project and reduce risk
  - Disadvantages
    - Team members need a high level of technical and interpersonal skills
    - Lack of structure and documentation can introduce risk factors
    - May be subject to significant change in scope

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

Modeling Tools and Techniques

- Involve graphical methods and nontechnical language that represent the system at various stages of development

- Systems analysts:
  - Build fact-finding results into models
  - Study the models to determine whether additional fact-finding is needed

- **Functional Decomposition Diagrams (FDD)**
  - Top-down representation of a function or process
  - Help analysts show business functions and how they are organized into lower-level processes

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

Modeling Tools and Techniques

(Cont. 1)
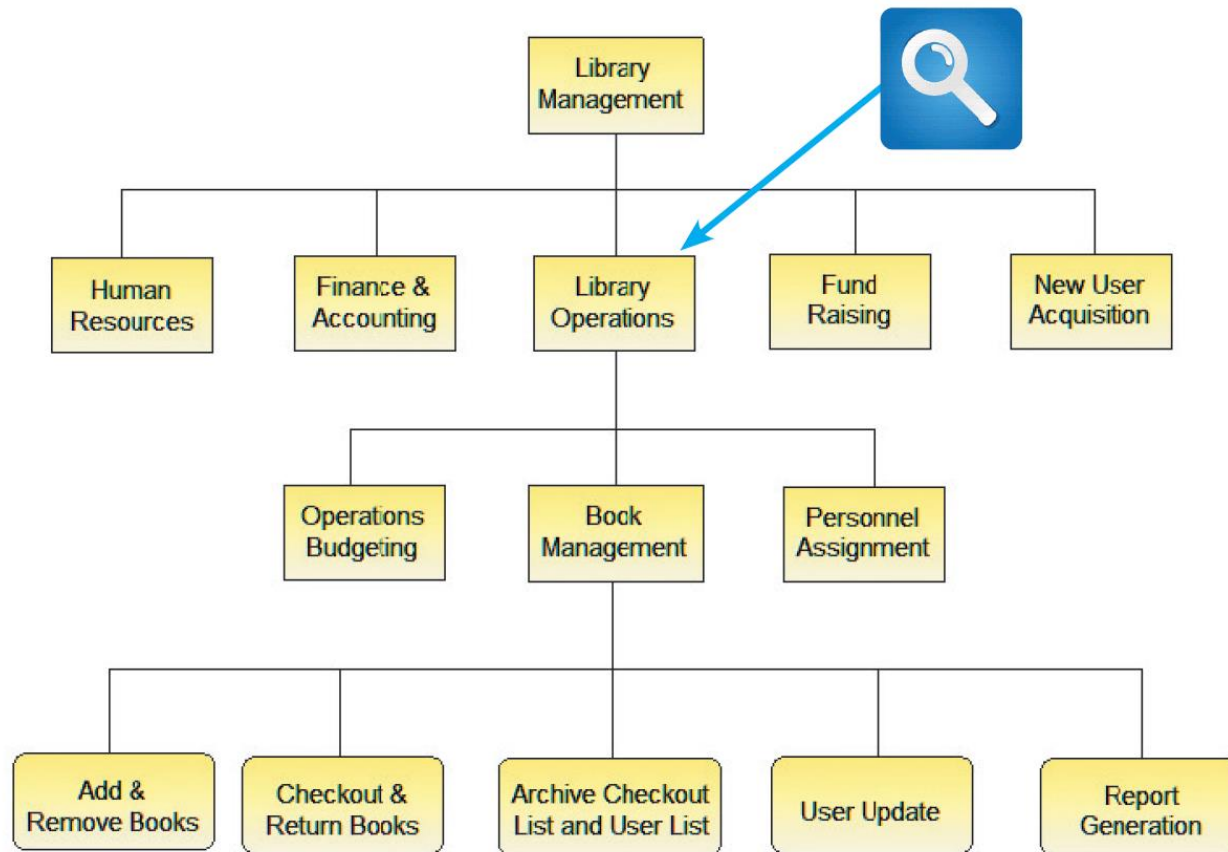
FIGURE 4-8 This Visible Analyst FDD shows a library system with five top-level functions. The Library Operations function includes two additional levels of processes and sub processes. Source: Visible Systems Corporation.

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

Modeling Tools and Techniques
(Cont. 2)

- **Business Process Modeling (BPM)**
  - Represents one or more business processes
  - Business process modeling notation (BPMN)
    - Models that use a standard language
    - Includes shapes and symbols to represent events, processes, and workflows
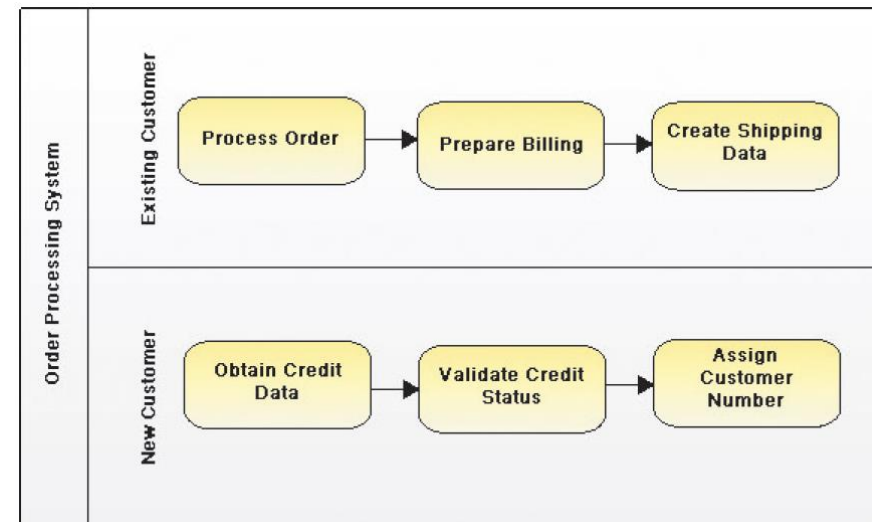
FIGURE 4-9 Using the Visible Analyst CASE tool, an analyst can create a business process diagram. The overall diagram is called a pool, and the two separate customer areas are called swim lanes.
Source: Visible Systems Corporation.

Modeling Tools and Techniques
(Cont. 3)

FIGURE 4-10 This Visible Analyst DFD shows how books are added and removed in a library system. Source: Visible Systems Corporation.

▸ **Data Flow Diagrams (DFD)**
- Show how the system stores, processes, and transforms data
- Additional levels of information and detail are depicted in other, related DFDs

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

Modeling Tools and Techniques - UML

(Cont. 4)
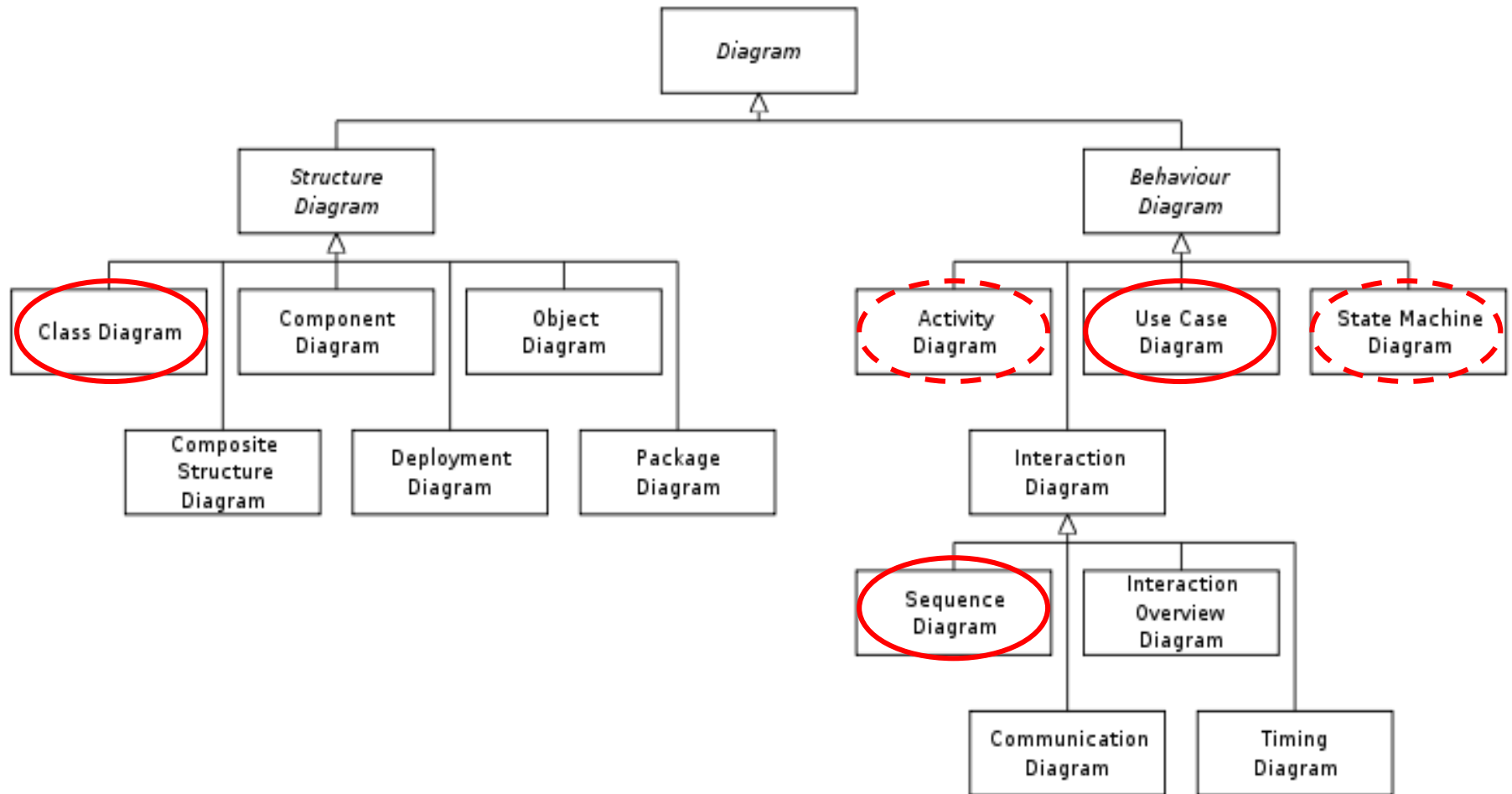
Modeling Tools and Techniques
(Cont. 5)

- ## Use Case Diagrams
  - Represent the interaction between
    users and the system

Name of Use Case:	Credit card validation process
Actor:	Customer
Description:	Describes the credit card validation process
Successful Completion:	1. Customer clicks the input selector and enters credit card number and expiration date 2. System verifies card 3. System sends authorization message
Alternative:	1. Customer clicks the input selector and enters credit card number and expiration date 2. System rejects card 3. System sends rejection message
Precondition:	Customer has selected at least one item and has proceeded to checkout area
Postcondition:	Credit card information has been validated Customer can continue with order
Assumptions:	None

FIGURE 4-12 This table documents the credit card validation use case shown in Figure 4-11.
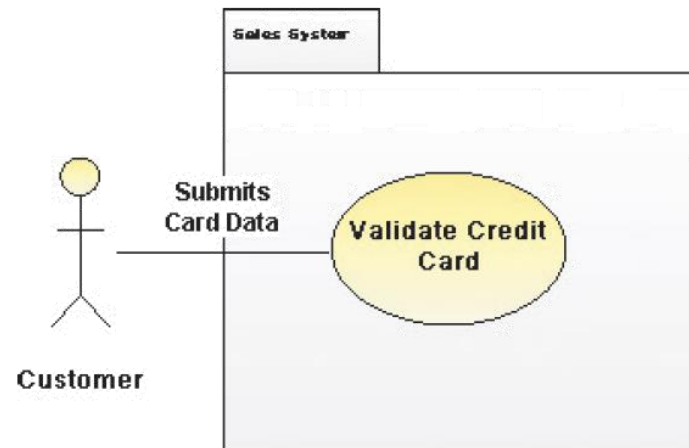
FIGURE 4-11 This Visible Analyst use case diagram shows a sales system, where the actor is a customer and the use case is a credit card validation.

Source: Visible Systems Corporation

THE UNIVERSITY OF
NEWCASTLE
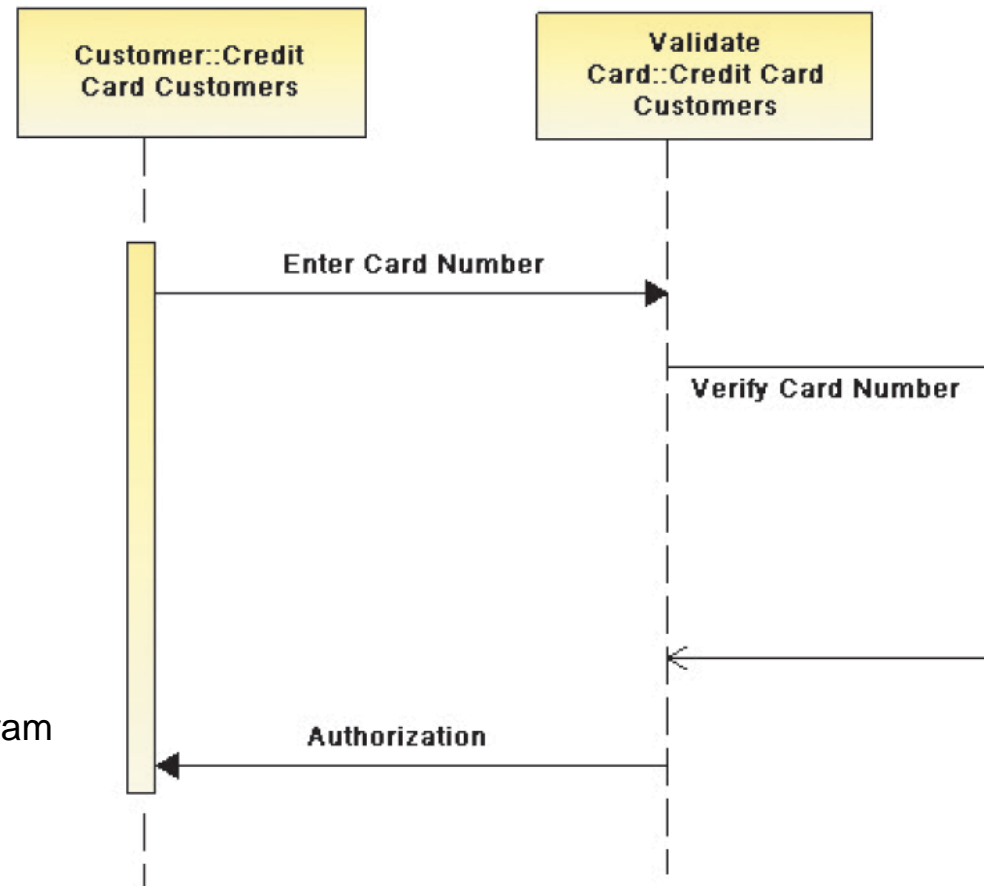AUSTRALIA

Modeling Tools and Techniques

(Cont. 6)

- ## Sequence Diagrams
 - – Show the timing
    of interactions
    between objects
    as they occur

FIGURE 4-14 This Visible Analyst sequence diagram
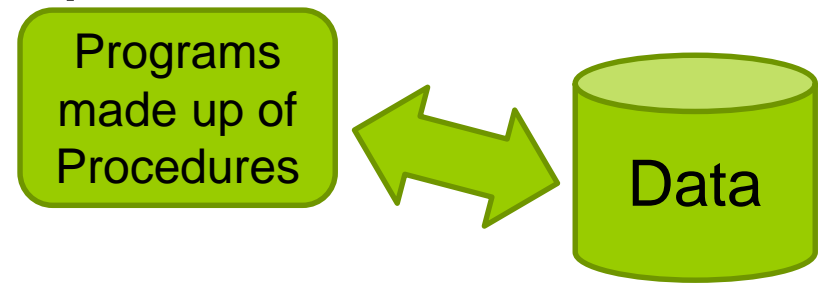shows a credit card validation process.
Source: Visible Systems Corporation

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

OO Design vs. Traditional Structured Approaches

OO
Encapsulated Structure & Behaviour

Data

Procedures

Traditional
Separate Structure & Behaviour

Programs made up of Procedures

Data
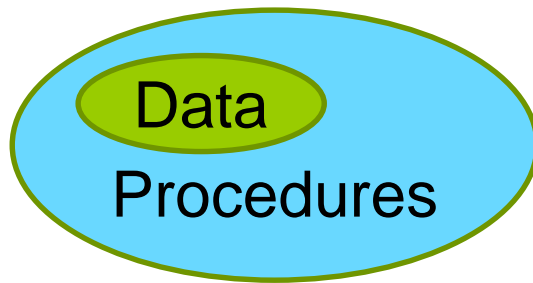
- **Structure** Implemented as Classes
  - **Attributes/Properties**

- **Behaviour** implemented
  - Internally in class by
    - **Methods** (Procedures/Subroutines/Functions)
    - Procedure calls (with arguments/parameters)
  - Externally between classes by
    - **Messages** (with arguments/parameters) passed/returned between Classes
  - Note all of these can send and receive values or objects as **parameters** (a Function returns one Value):
    - The parameters have persistence (local and global
    - The parameters can be absolute or relatively addressed, i.e., ByRef or ByVal)

- Structure modelled by an ER diagram and converts "easily" to a relational database for permanent storage
- Behaviour modelled by a series of Processes in Data Flow Diagrams (with pseudocode or structured English procedure definitions for each process description to indicate sequence and timing of activities
- Behaviour is <u>implemented</u> as a software program consisting of:
- Procedures/Subroutines/Functions
- Procedure calls (with arguments/parameters)
- Typically ODBC Calls to relational database (using SQL queries)

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

System Requirements Checklist

- **Output Examples**
  - The Web site must report online volume statistics every four hours, and hourly during peak periods
  - The contact management system must generate a daily reminder list for all sales reps
  - The purchasing system must provide suppliers with up-to-date specifications

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

System Requirements Checklist (Cont. 1)

- **Input Examples**
  - The department head must enter overtime hours on a separate screen
  - Student grades must be entered on machine-readable forms prepared by the instructor
  - Each input form must include date, time, product code, customer number, and quantity

- **Process Examples**
  - The student records system must calculate the GPA at the end of each semester
  - The human resources system must interface properly with the existing payroll system

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

System Requirements Checklist (Cont. 2)

– The prescription system must automatically generate an insurance claim form

- **Performance Examples**
  – The system must support 25 users online simultaneously
  – Response time must not exceed four seconds
  – The system must be operational seven days a week, 365 days a year

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

System Requirements Checklist (Cont. 3)

- **Control Examples**
  - The system must provide logon security at the operating system level and at the application level
  - The system must maintain separate levels of security for users and the system administrator
  - All transactions must have audit trails
  - The system must create an error log file that includes the error type, description, and time

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

Future Growth, Costs, and Benefits

- **Scalability**
  - A system's ability to handle increased business volume and transactions in the future
    - A scalable system offers a better return on the initial investment
  - Information required to evaluate scalability
   - Projected future volume for all outputs, inputs, and processes

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

Future Growth, Costs, and Benefits (Cont.)

- **Total Cost of Ownership**
  - Important if the development team is evaluating several alternatives
  - Problem - Cost estimates tend to understate indirect costs
    - Systems analysts should try to identify indirect costs and include them in TCO estimates

FIGURE 4-15 Total cost of ownership when migrating to the cloud can be significantly less than current computing platforms.

TCO for Cloud Computing

14% Higher

29% Same

57% Lower

THE UNIVERSITY OF NEWCASTLE AUSTRALIA
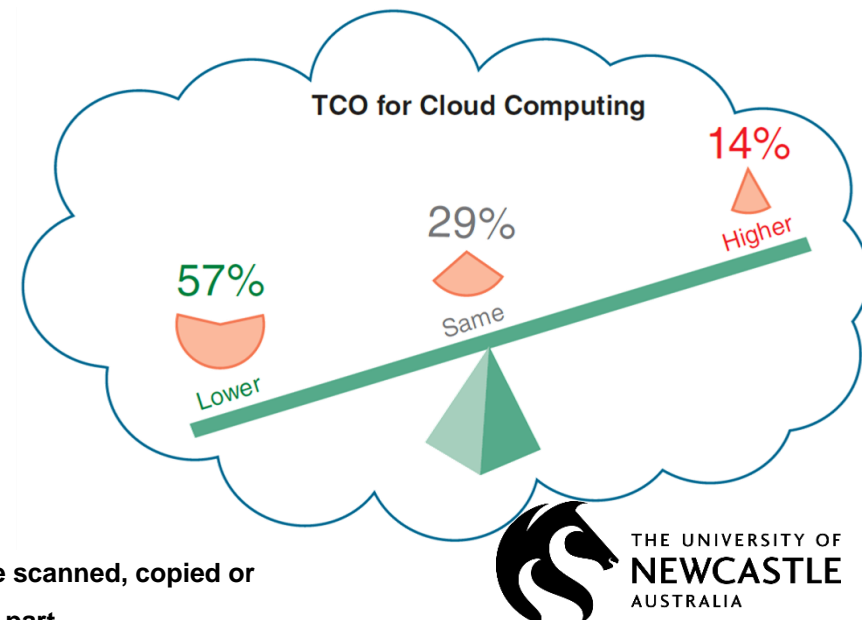
Fact Finding

▶ **Fact-Finding Overview**

  ◦ Identify the required information - Typical questions to ask

    • What business functions are supported by the current system?

    • What strategic objectives and business requirements must be supported by the new system?

    • What are the benefits and TCO of the proposed system?

    • What transactions will the system process?

    • Must the new system interface with legacy systems?

  ◦ Develop a fact-finding plan

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

Fact Finding (Cont. 1)

- **Who, What, Where, When, How, and Why?**
  - Systems analyst must first understand the current situation
   - Will help him/her tackle the question of what should be done

CURRENT SYSTEM		PROPOSED SYSTEM
Who does it?	Why does this person do it?	Who should do it?
What is done?	Why is it done?	What should be done?
Where is it done?	Why is it done there?	Where should it be done?
When is it done?	Why is it done then?	When should it be done?
How is it done?	Why is it done this way?	How should it be done?

FIGURE 4-17 Sample questions during requirements modeling as the focus shifts from the current system to the proposed system.

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

Fact Finding (Cont. 2)

- **The Zachman Framework**
  - Helps managers and users understand the model
  - Ensures that overall business goals translate into successful IT projects
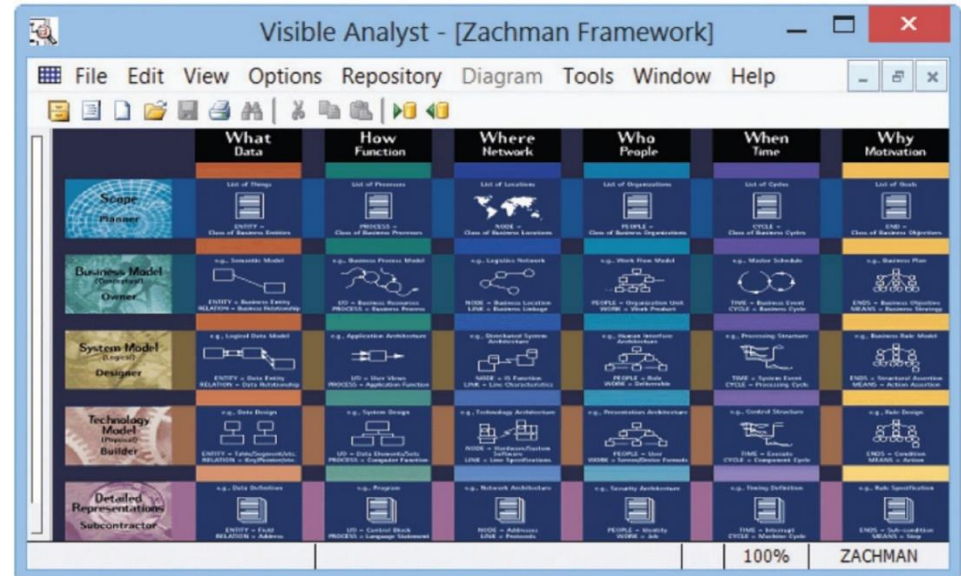
FIGURE 4-17 Visible Analyst uses the Zachman Framework for Enterprise Architecture. The Zachman concept presents traditional fact-finding questions in a systems development context.
Source: Visible Systems Corporation

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

Interviews

- Steps involved
  - Step 1 - Determine the people to interview
  - Step 2 - Establish objectives for the interview
  - Step 3 - Develop interview questions
  - Step 4 - Prepare for the interview
  - Step 5 - Conduct the interview
  - Step 6 - Document the interview
  - Step 7 - Evaluate the interview

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

Interviews (Cont. 1)

- **Step 1 - Determine the People to Interview**
  - Select the right people and ask the right questions
    - Consider candidates from both formal and **informal structures**
  - Decide on group and/or individual interviews

- **Step 2 - Establish Objectives for the Interview**
  - Determine the areas to be discussed
  - List the facts that need to be gathered
  - Objectives depend on the role of the person being interviewed

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

Interviews (Cont. 2)

- **Step 3 - Develop Interview Questions**
  - Decide what to ask and how to phrase the question
    - Avoid **leading questions**
    - **Open ended questions** encourage spontaneous and unstructured responses
    - **Close ended questions** limit the response
      - Used to verify facts
    - **Range-of-response questions** limit the response
      - Use a numeric scale

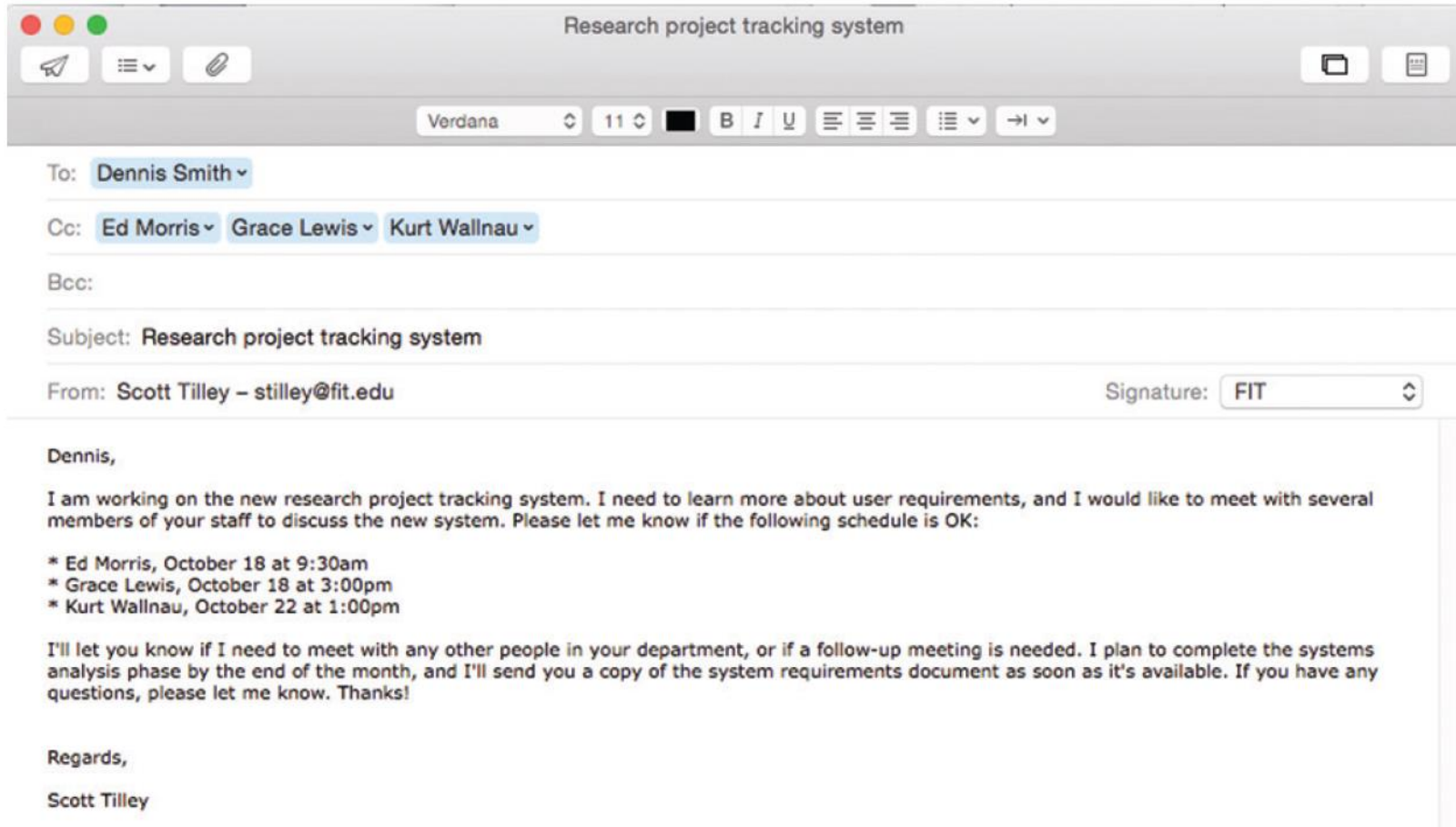THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

Interviews (Cont. 3)

- **Step 4 - Prepare for the Interview**
  - Careful preparation is essential
  - Limit the interview to no more than one hour
  - Verify time, place, length, and topics via e-mail
  - If there are questions about documents, ask the interviewee to have samples available at the meeting

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

Interviews (Cont. 4)

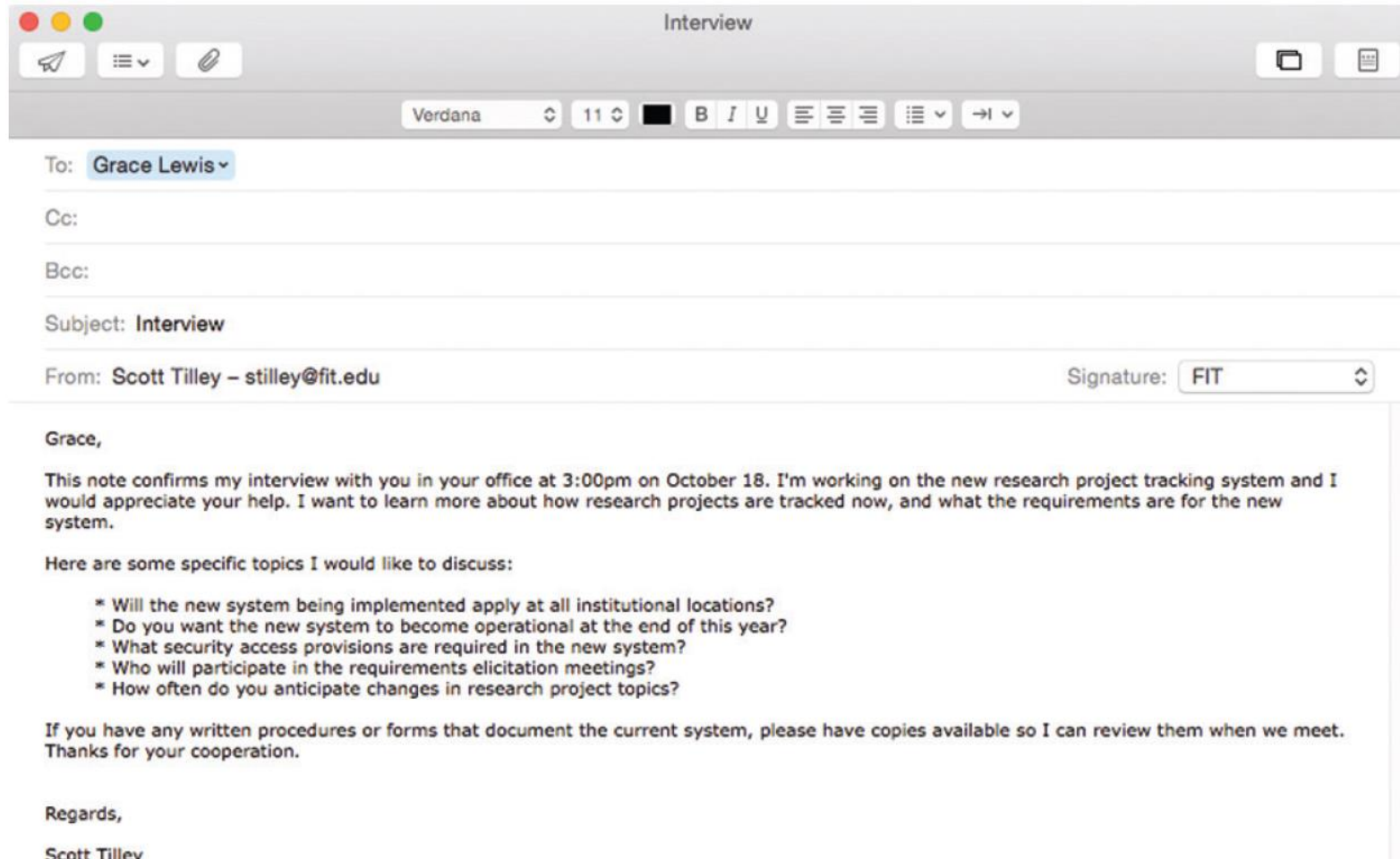**FIGURE 4-19** Sample message to a department head about interviews.
Source: 2015 Apple

Interviews (Cont. 5)

FIGURE 4-20 Sample message to confirm an interview.
Source: 2015 Apple

Interviews (Cont. 6)

- **Step 5 - Conduct the Interview**
  - Develop a specific plan for the meeting
  - Begin by introducing yourself, describing the project, and explaining your interview objectives
  - Practice **engaged listening**
  - Allow the person enough time to think about the question and arrive at an answer
  - After an interview, summarize the session and seek a confirmation

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

Interviews (Cont. 7)

- **Step 6 - Document the Interview**
  - Note taking should be kept to a minimum
  - After conducting the interview:
    - Record the information quickly
    - Send memo to the interviewee expressing your appreciation
      - Note the date, time, location, purpose of the interview, and the main points you discussed so the interviewee has a written summary and can offer additions or corrections

- **Step 7 - Evaluate the Interview**
  - In addition to recording the facts obtained in an interview, try to identify any possible biases

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

Interviews (Cont. 8)

- **Unsuccessful Interviews**
  - No matter how well you prepare for interviews, some are not successful
  - Misunderstanding or personality conflict could affect the interview negatively, or the interviewee might be afraid that the new system will eliminate or change his or her job

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

CASE IN POINT 4.3: FASTPAK OVERNIGHT PACKAGE SYSTEM

FastPak, the nation's fourth-largest overnight package system, is headquartered in Los Angeles, California. Jesse Evans is a systems analyst on an IT team that is studying ways to update FastPak's package tracking system. Jesse prepared well for her interview with Jason Tanya, FastPak's executive vice president. Mr. Tanya did not ask his assistant to hold his calls during the meeting, however. After several interruptions, Jesse tactfully suggested that she could come back another time, or perhaps that Mr. Tanya might ask his assistant to hold his calls. "No way," he replied. "I'm a very busy man and we'll just have to fit this in as we can, even if it takes all day." Jesse was unprepared for his response. What are her options? Is an analyst always in control of this kind of situation? Why or why not?

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

Other Fact-Finding Techniques

- **Document Review**
  - Review of baseline documentation
  - Helps an analyst understand how the current system is supposed to work

- **Observation**
  - Provides additional perspective and a better understanding of the system procedures
  - Should be planned in advance

Figure 4-21 The Hawthorne study suggested that worker productivity improves during observation. Always consider the Hawthorne Effect when observing the operation of an existing system.
Monkey Business Images/Shutterstock.com

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

- **Questionnaires and Surveys**
  - Make sure that the questions collect the right data in a form that can be used to further the fact finding effort
  - Can be traditional forms, fill-in forms, or forms from online survey websites
    - **Fill-in form**: Template used to collect data on the Internet or a company intranet

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

Other Fact-Finding Techniques (Cont. 2)

– Suggestions for designing a questionnaire

- Keep the questionnaire brief and user-friendly
- Provide clear instructions
- Arrange the questions in a logical order
- Phrase questions to avoid misunderstandings
- Try not to lead the response
- Limit the use of open-ended questions that are difficult to tabulate
- Limit the use of questions that can raise concerns about job security or other negative issues
- Include a section for general comments
- Test the questionnaire on a small test group before finalizing it and distributing to a large group

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

Other Fact-Finding Techniques (Cont. 3)

PURCHASE REQUEST QUESTIONNAIRE

Pat Kline, Vice President, Finance, has asked us to investigate the purchase requisition process to see if it can be improved. Your input concerning this requisition process will be very valuable. We would greatly appreciate it if you could complete the following questionnaire and return it by March 15 to Stef Ting in Information Technology. If you have any questions, please call Stef at x7045.

A. YOUR OBSERVATIONS
Please answer each question by checking one box.

1. How many purchase requisitions did you process in the past five working days?

2. What percentage of your time is spent processing requisitions?
○ Under 20%
○ 21 - 39%
○ 40 - 59%
○ 60 - 79%
○ 80% or more

3. Do you believe too many errors exist on requisitions?
○ Yes
○ No

4. Out of every 10 requisitions you process, how many contain errors?
○ Fewer than 5
○ 5 to 9
○ 10 - 14
○ 15 - 19
○ 20 or more

5. What errors do you see most often on requisitions?
☐ Incorrect charge number
☐ Missing charge information
☐ Arithmetic errors
☐ Missing authorization
☐ Other

B. YOUR SUGGESTIONS
Please be specific and give examples if possible.

1. If the purchase requisition form was redesigned, what changes would you recommend?

2. Would you be interested in meeting with an information technology representative to discuss your ideas further? If so, please complete the following information:

Name   Department

Telephone   E-mail

[Submit]

FIGURE 4-22 Online version of a sample questionnaire. Does it follow the suggested guidelines?
Created by author using Adobe Online Forms, Adobe Systems Incorporated

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

Other Fact-Finding Techniques (Cont. 4)

- **Interviews versus Questionnaires**
  - Interview is more familiar and personal
    - Costly and time-consuming process
  - Questionnaire gives people the opportunity to provide input and suggestions
    - Recipients can answer the questions at their convenience
- **Brainstorming**: Small group discussion of a specific problem, opportunity, or issue
  - **Structured brainstorming**
  - **Unstructured brainstorming**

THE UNIVERSITY OF NEWCASTLE AUSTRALIA

Other Fact-Finding Techniques (Cont. 5)

- ## Sampling
  - **Systematic sample**: Selection of every tenth customer for review
  - **Stratified sample**: Selection of five customers from each of four postal codes
  - **Random sample**: Selection of any 20 customers
  - Objective of a sample - To ensure that it represents the overall population accurately

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

Other Fact-Finding Techniques (Cont. 6)

- ## **Research**
  - The Internet, IT magazines, and books to obtain background information, technical material, and news about industry trends and developments
  - Attending professional meetings, seminars, and discussions with other IT professionals
  - **Site visits**

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

CASE IN POINT 4.4: CYBERSTUFF

Ann Ellis is a systems analyst at CyberStuff, a large company that sells computer hardware and software via telephone, mail order, and the Internet. CyberStuff processes several thousand transactions per week on a three-shift operation and employs 50 full-time and 125 part-time employees. Lately, the billing department has experienced an increase in the number of customer complaints about incorrect bills. During the preliminary investigation, Ann learned that some CyberStuff representatives did not follow established order entry procedures. She feels that with more information, she might find a pattern and identify a solution for the problem.

Ann is not sure how to proceed. She came to you, her supervisor, with two separate questions. First, is a questionnaire the best approach, or would interviews be better? Second, whether she uses interviews, a questionnaire, or both techniques, should she select the participants at random, include an equal number of people from each shift, or use some other approach? As Ann's supervisor, what would you suggest, and why?

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

Documentation

- ## The Need for Recording the Facts
  - Principles for documentation
    - Record information as soon as it is obtained
    - Use the simplest recording method
    - Record findings in a way that they can be understood by someone else
    - Organize documentation so related material is located easily

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

Documentation _(Cont. 1)

- **Software Tools**
  - CASE tools
  - **Productivity software**
    - Word processing
    - Spreadsheets
    - Database management
    - Presentation graphics
    - Collaborative software programs
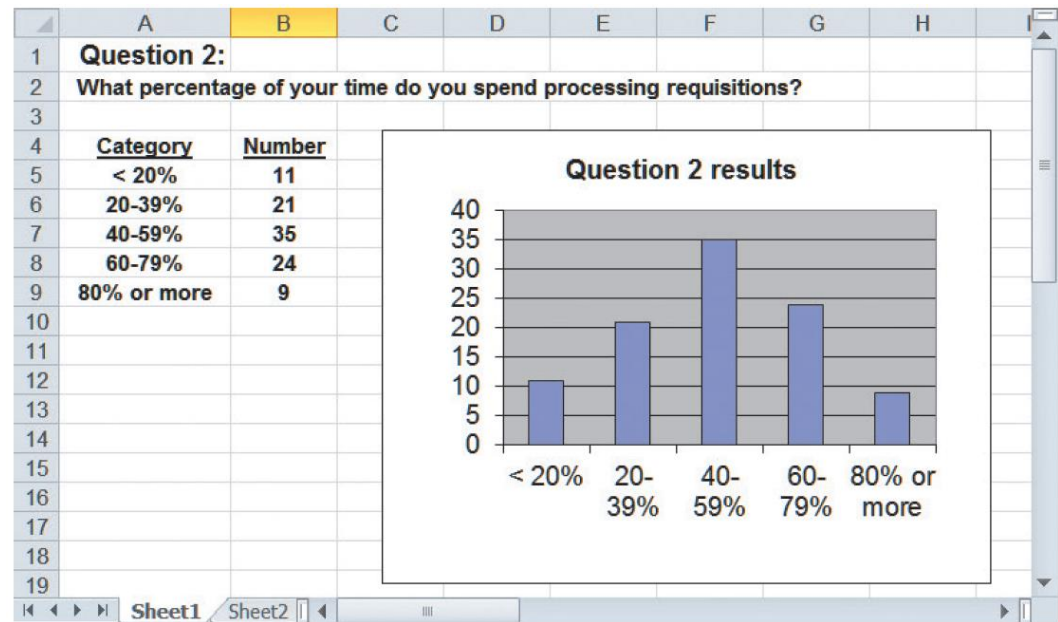
**FIGURE 4-24** This histogram displays the results from Question 2 in the questionnaire shown in Figure 4-22.

THE UNIVERSITY OF NEWCASTLE AUSTRALIA

Documentation (Cont. 3)

- ## Graphic Modeling Software
  - Help create charts and diagrams
  - Popular software
    - MS Visio

FIGURE 4-25 This Microsoft Visio drawing uses drag-and-drop shapes to represent a business process.
Source: Microsoft, LLC

NEWCASTLE AUSTRALIA
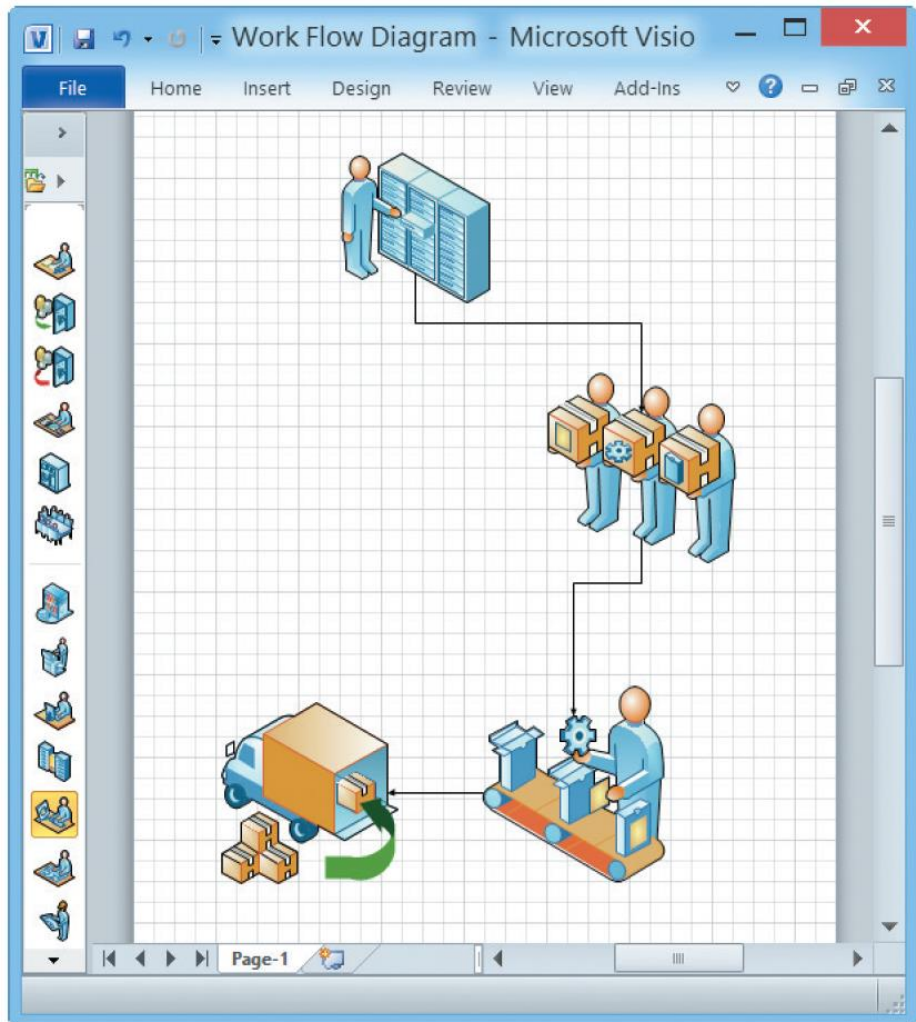
Information Management Software

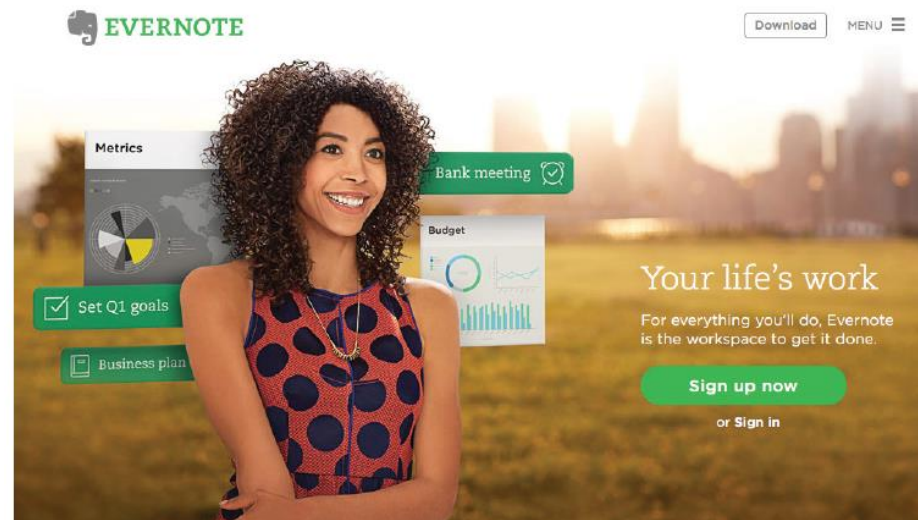- **Personal Data Management Software**
  - Microsoft Outlook
    - Includes a personal calendar, a to-do list with priorities and the capability to check off completed items, and powerful contact management features
    - Can manage email and appointments, and supports collaboration and team
  - Novell's GroupWise

Figure 4-27 Evernote offers a free version of its popular information management software for most computing platforms, including smartphones and on the web.
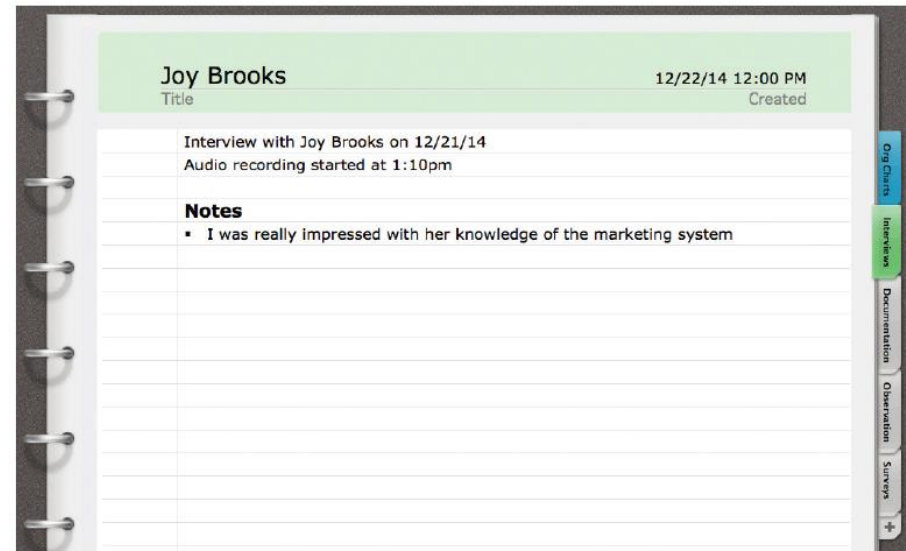Source: www.evernote.com

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

Information Management Software (Cont.)

- **Project Data Management**
  - Microsoft OneNote
    - Handles different types of input, including text, handwritten notes, images, audio and video recordings, and web links
  - Microsoft Word
    - Recent versions provide note taking feature

FIGURE 4-26 The analyst is using Microsoft Word to store fact-finding results. During the interview with Joy Brooks, the analyst recorded part of the discussion and stored it as a document annotation.
Source: Microsoft Corporation.

Joy Brooks
Title
12/22/14 12:00 PM
Created

Interview with Joy Brooks on 12/21/14
Audio recording started at 1:10pm

Notes
- I was really impressed with her knowledge of the marketing system

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

Preview of Logical Modeling

- At the conclusion of requirements modeling, systems developers should have a clear understanding of business processes and system requirements

- Next step - To construct a logical model of the system

- IT professionals have differing views about systems development methodologies, and no universally accepted approach exists

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

Chapter Summary

- The systems analysis phase includes requirements modeling, data and process modeling, and consideration of development strategies

    - Objective is to understand the proposed project, ensure that it will support business requirements, and build a solid foundation for the systems design phase

- Popular team-based approaches include JAD, RAD, and agile methods

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

Chapter Summary (Cont.)

- The fact-finding process includes interviewing, document review, observation, questionnaires, sampling, and research

- Systems analysts should carefully record and document factual information as it is collected, and various software tools can help an analyst visualize and describe an information system

THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA