Topic 6:
Requirements Analysis

References

- [Pressman] Pressman, Roger S., Software Engineering: A Practitioner's Approach, 7th Edition, Mc. Graw Hill International, USA, 2010.

- [Dennis] Dennis, Alan, et. al., System Analysis and Design with UML 3rd Edition, John Wiley & Sons, 2010.

- [Sommerville] Sommerville, Ian, Software Engineering, 9th Edition, Pearson-Addison Wesley, England, 2011.

"Designing and building computer software is challenging, creative, and just plain fun. In fact, building software is so compelling that many software developers want to jump right in before they have a clear understanding of what is needed."
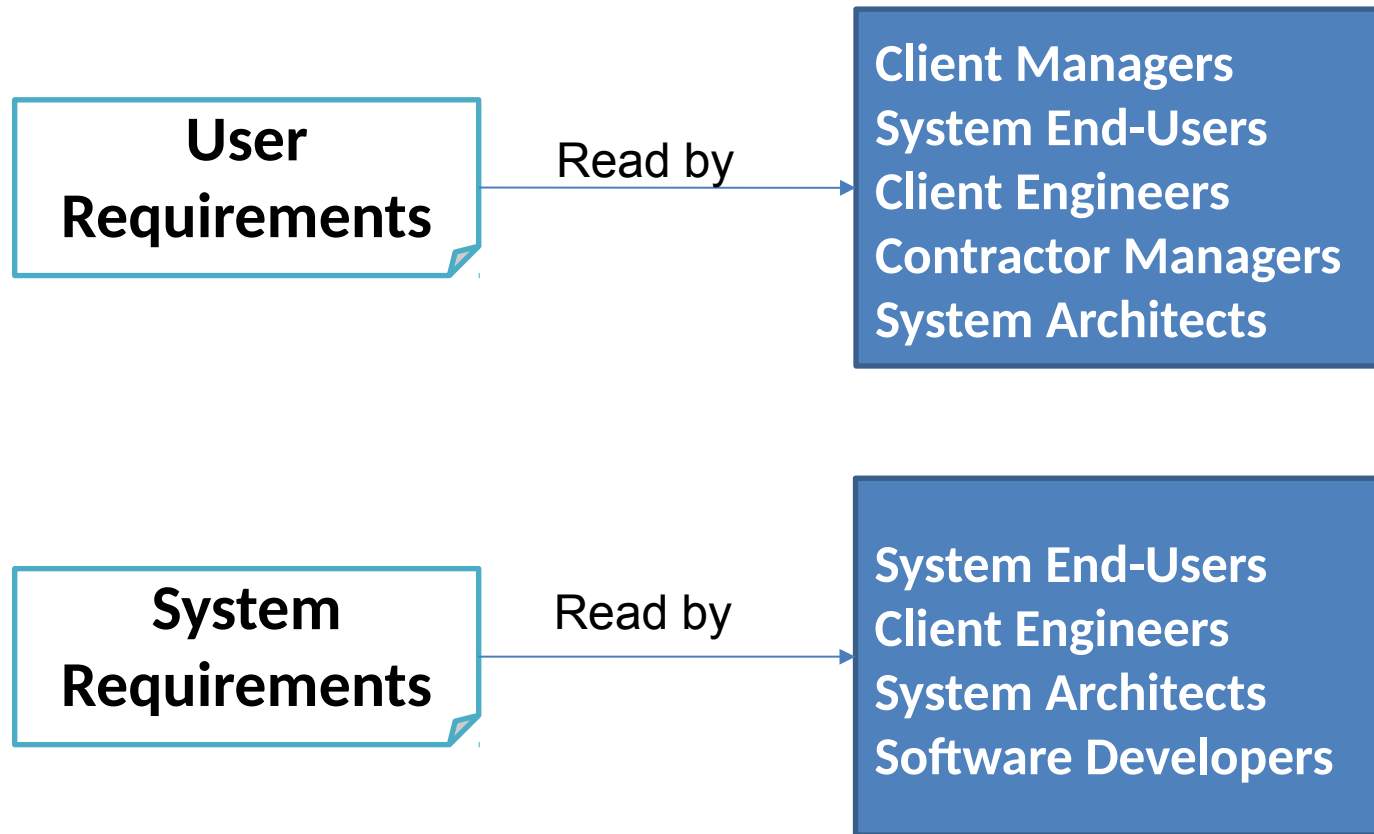
Roger S. Pressman

The SDLC and Requirements

- The SDLC transforms the existing (as is) system into the proposed (to be) system

- Requirements determination step is the single most critical step of the entire SDLC
  - ❖ Studies show that more than half of all system failures are due to problems with requirements

Defining a Requirement

- Requirement: "A statement of what the system must do or what characteristic it must have."
- During **analysis**, requirements:
  - ❖ are written from the perspective of the business person
  - ❖ focus on the "what" of the system
  - ❖ are usually called **business requirements** or **user requirements**
- Later in **design**, user requirements evolve to become **system requirements**
- However, some companies use the terms interchangeably

Readers of Requirement Specification

User Requirements Read by **Client Managers**
System End-Users
Client Engineers
Contractor Managers
System Architects

System Requirements Read by **System End-Users**
Client Engineers
System Architects
Software Developers

User & System Requirements

User requirements definition

1. The Mentcare system shall generate monthly management reports showing the cost of drugs prescribed by each clinic during that month.

System requirements specification

1.1 On the last working day of each month, a summary of the drugs prescribed, their cost and the prescribing clinics shall be generated.
1.2 The system shall generate the report for printing after 17.30 on the last working day of the month.
1.3 A report shall be created for each clinic and shall list the individual drug names, the total number of prescriptions, the number of doses prescribed and the total cost of the prescribed drugs.
1.4 If drugs are available in different dose units (e.g. 10mg, 20mg, etc.) separate reports shall be created for each dose unit.
1.5 Access to drug cost reports shall be restricted to authorized users as listed on a management access control list.

Types of Requirements

- Two kinds of requirements:
  - ❖ Functional
    - **processes** that the system has to perform
    - **information** that the system needs to contain
    - **functions** that the system must have
  - ❖ Nonfunctional
    - **behavioral properties** that the system must have
    - **quality attributes**, **quality requirements**, or the "**-ilities**" of a system
    - E.g. **URPS+** (usability, reliability, performance, supportability) in [Larman]

Requirements Definition

- A strightforward text report that simply lists the functional and nonfunctional requirements

Functional Requirements

Example
1. Manage Appointments 　　1.1. Patient makes new appointment 　　1.2. Patient changes appointment 　　1.3. Patient cancels appointment
2. Produce Schedule 　　2.1. Office Manager checks daily schedule 　　2.2. Office Manager prints daily schedule
3. Record Doctor Availability 　　3.1. Doctor updates schedule

Nonfunctional Requirements

Requirement type	Example
Operational	• The system will operate in Windows environment. • The system should be able to connect to printers wirelessly. • The system should automatically back up at the end of each day.
Performance	• Any interaction between the user and the system should not exceed 2 seconds.
Security	• Only doctors can see their availability • Only manager can produce a schedule.
Cultural & Political	• The system should be able to distinguish between United States and European currency • The system shall comply with insurance industry standards.

Exercise: Requirement Definition

- List functional and nonfunctional requirements of academic staff creating course and student registration to course in the beginning of semester ("Pengisian IRS")
- Consider these roles:
  - ❖ Student
  - ❖ Academic advisor
  - ❖ Course teacher
  - ❖ Academic staff

Exercise: Nonfunctional Requirements

1. Operational
    1. The system's interface should be mobile responsive
2. Performance
    1. The system should be able to handle 1000 concurrent users
    2. Time to process request should not exceed 5 seconds
3. Security
    1. Student's study plan can only be viewed by his/her academic advisor
4. Cultural and political
    1. Course registration is limited by schedule and other rules defined in the academic regulations
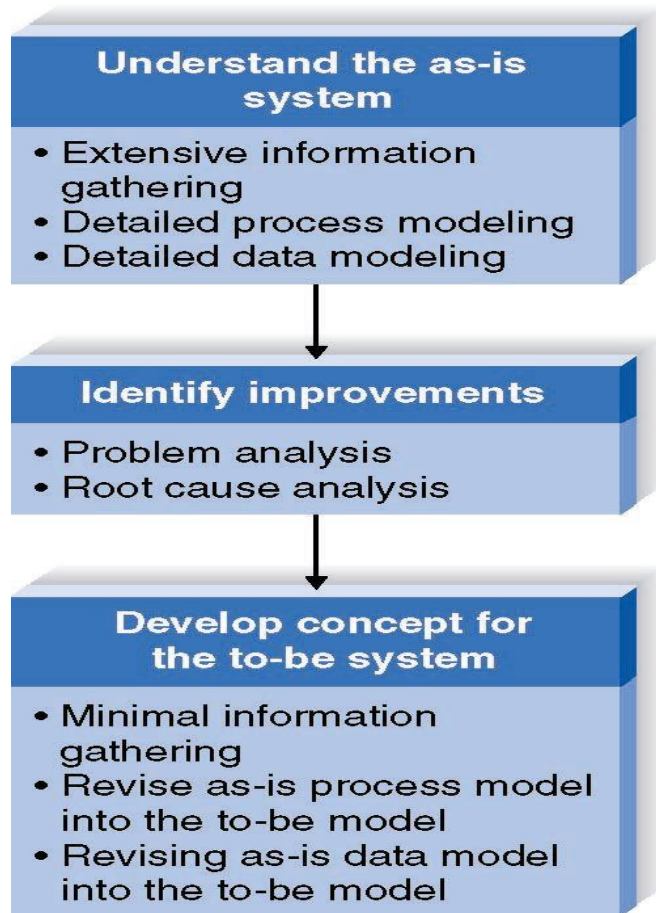
Determining Requirements

- Participation by business users/experts and IT analyst is essential
  - If done only by IT analyst, may not address true business needs
  - If done only by business experts, may not take advantage of technology

- Requirements are best determined by systems analysts **and** business people together

- Three techniques help users discover their needs for the new system:
  - Business Process Automation (BPA)
  - Business Process Improvement (BPI)
  - Business Process Reengineering (BPR)

Basic Process of Analysis (Determining Requirements)

- Understand the "As-Is" system
- Identify improvement opportunities
- Develop the "To-Be" system concept
- Techniques vary in amount of change
  - ❖ BPA – small change
  - ❖ BPI – moderate change
  - ❖ BPR – significant change
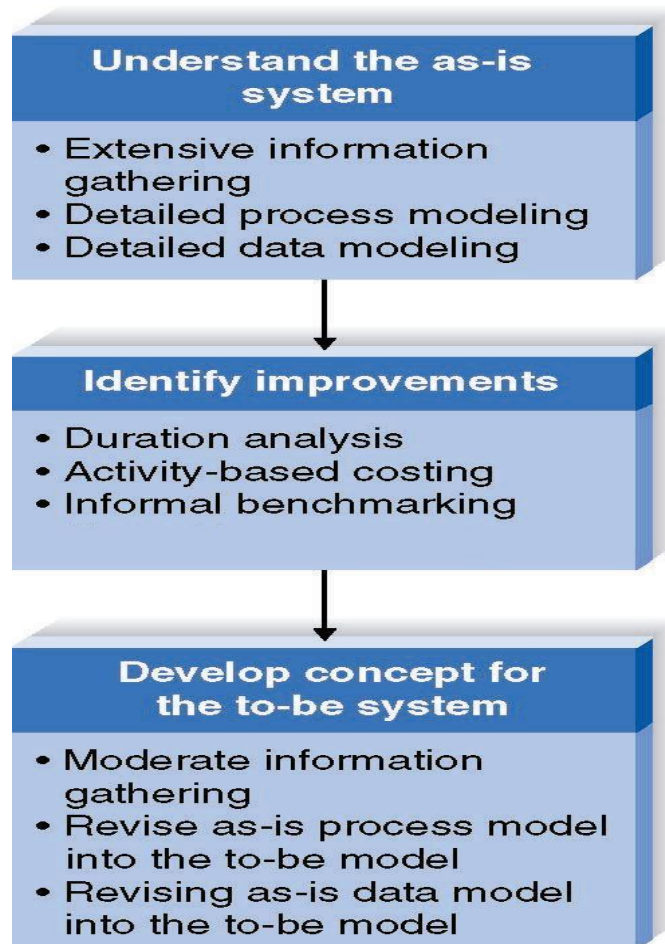-

Business Process Automation

Understand the as-is system
- Extensive information gathering
- Detailed process modeling
- Detailed data modeling

Identify improvements
- Problem analysis
- Root cause analysis

Develop concept for the to-be system
- Minimal information gathering
- Revise as-is process model into the to-be model
- Revising as-is data model into the to-be model

- Doesn't change basic operations
- Automates some operations

Goal:

Efficiency for users

Business Process Improvement

Understand the as-is system
- Extensive information gathering
- Detailed process modeling
- Detailed data modeling

Identify improvements
- Duration analysis
- Activity-based costing
- Informal benchmarking

Develop concept for the to-be system
- Moderate information gathering
- Revise as-is process model into the to-be model
- Revising as-is data model into the to-be model

Goal:

Efficiency and effectiveness for users

Business Process Reengineering (BRP)

Understand the as-is system
- Minimal information gathering
- Superficial process modeling

Identify improvements
- Outcome analysis
- Technology analysis
- Activity elimination

Develop concept for the to-be system
- Extensive imformation gathering
- Detailed to-be process model
- Detailed to-be data model
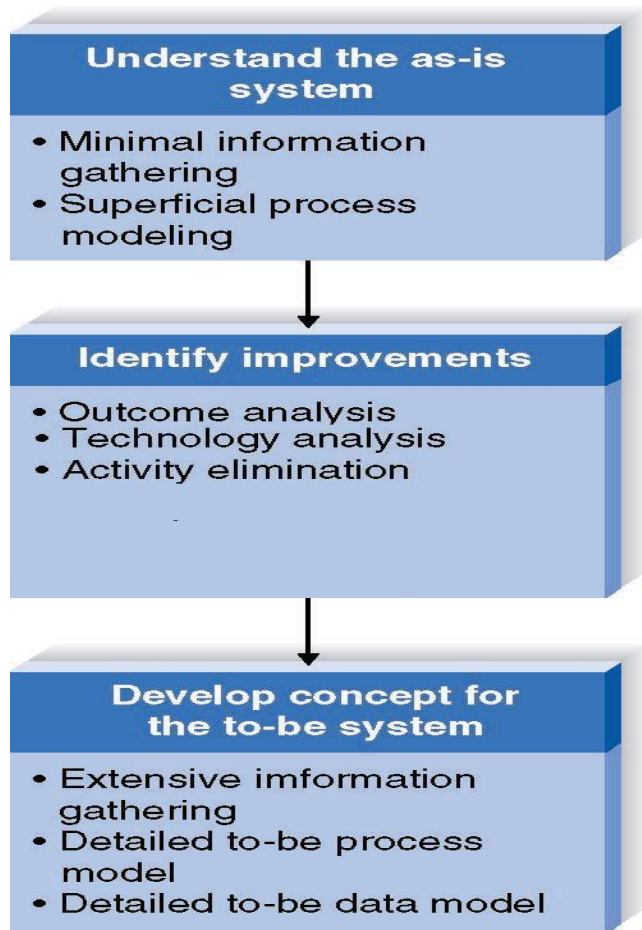
Changes fundamentally how the organization does certain operations

Goal:

Radical redesign of business processes

Requirements Engineering

- The broad spectrum of tasks and techniques that lead to an understanding of a requirement
- A major action that begins during communication and continues into modeling
- Build a bridge to design and construction

The journey across the bridge will allow us to see:

- The context
- Specific design & construction needs
- Priorities
- Information, function, behaviour

shutterstock.com • 1388367890

Requirements Engineering Phases

- Inception—establish basic understanding of …
  - ❖ the problem, the stakeholders, the desired solution, and the communication among stakeholders
- Elicitation—elicit requirements from all stakeholders
- Elaboration—expand and refine information gathered from elicitation. Create a requirement model (*e.g. use case diagrams, analysis class diagrams*) that identifies the software function, behavior, and information
- Negotiation—developers and customers discuss about priority, cost & risk of each requirement

Requirements Engineering Phases

- Specification—can be any one (or more) of the following:
  - ❖ A written document
  - ❖ A set of models
  - ❖ A formal mathematical
  - ❖ A collection of usage scenarios
  - ❖ A prototype
- Validation—examine the specification to ensure that each requirement is valid
- Requirements management – identify, control, and track requirements changes

Requirements Validation

To examine the specification to ensure that each requirement is:

- Correct
- Unambiguous
- Complete
- Consistent
- Verifiable
- Modifiable
- Traceable
- Ranked for importance

A Bad Requirement

Initial Specification: **Software will not be loaded from unknown sources onto the system without first having the software tested and approved.**

Critique:

- Ambiguous – if the software is tested and approved, can it be loaded from unknown sources?
- (not) Testable – it is stated as a negative requirement making it difficult to verify.
- (not) Traceable – a unique identifier is missing.

Re-specification: **3.4.5.2 Software shall be loaded onto the operational system only after it has been tested and approved.**

Unambigous requirement

- *REQ1 The system shall be implemented using ASP.*

- Does ASP mean Active Server Pages or Application Service Provider? To fix this, we can mention a full name and provide an acronym in parentheses:

- *REQ1 The system shall be implemented using Active Server Pages (ASP).*

Unambigous requirement (2)

- *REQ1 The system shall not accept passwords longer than 15 characters.*

- It is not clear what the system is supposed to do:
  - ❖ The system shall not let the user enter more than 15 characters.
  - ❖ The system shall truncate the entered string to 15 characters.
  - ❖ The system shall display an error message if the user enters more than 15 characters.

- *REQ1 The system shall not accept passwords longer than 15 characters. If the user enters more than 15 characters while choosing the password, an error message shall ask the user to correct it.*

Requirements-Gathering Techniques

- Interviews
- JAD (Joint Application Development)
- Questionnaires
- Document Analysis
- Observation

Five Basic Steps of Interviews

- Selecting interviewees, e.g:
  - ❖ Managers, Users, (Ideally all key stakeholders)
- Designing interview questions
- Preparing for the interview
- Conducting the interview
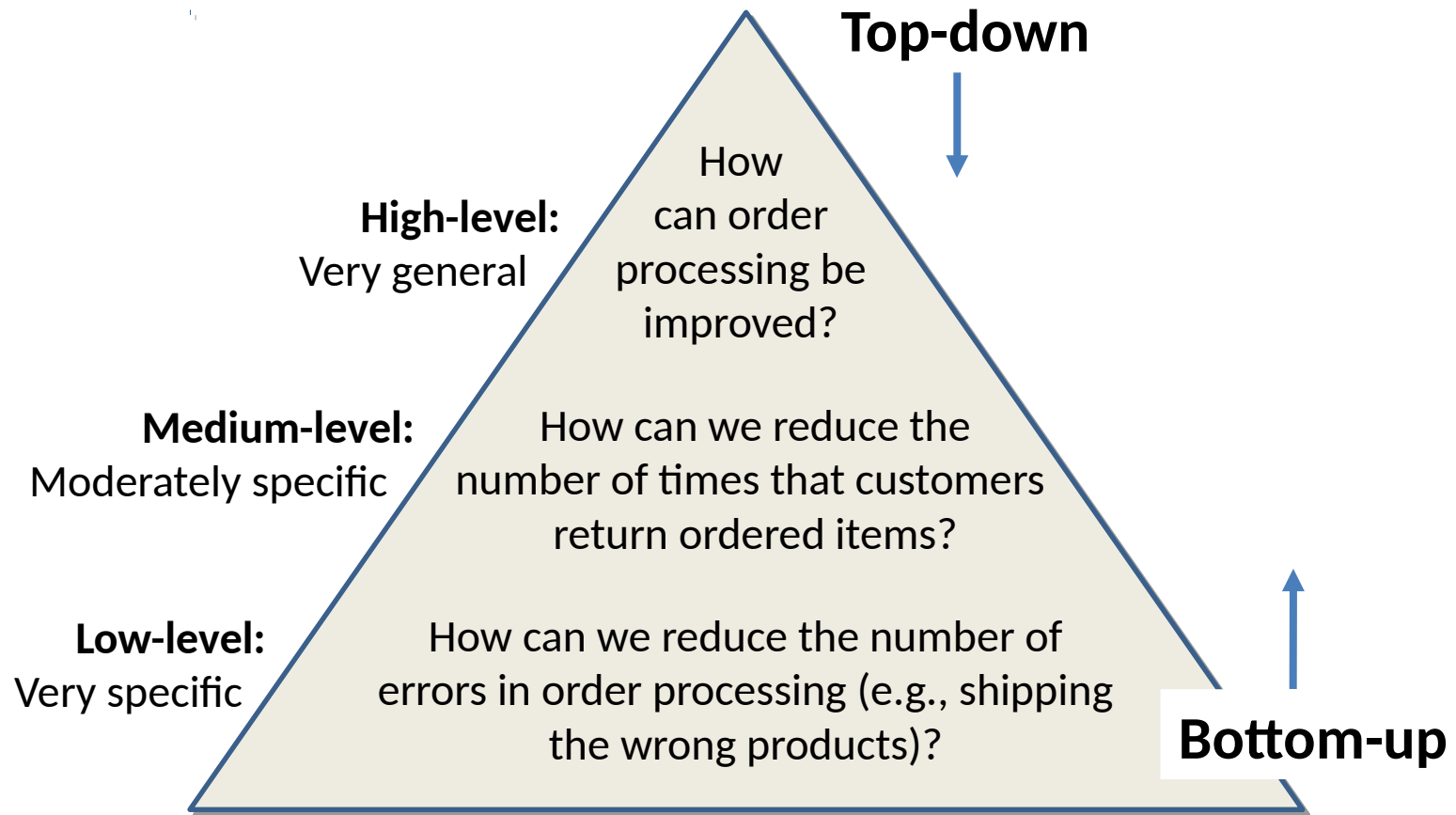- Post-interview follow-up

Selecting interviewees

Name	Position	Purpose of Interview	Meeting
Andria McClellan	Director, Accounting	Strategic vision for new accounting system	Mon., March 1 8:00–10:00 AM
Jennifer Draper	Manager, Accounts Receivable	Current problems with accounts receivable process; future goals	Mon., March 1 2:00–3:15 PM
Mark Goodin	Manager, Accounts Payable	Current problems with accounts payable process; future goals	Mon., March 1 4:00–5:15 PM
Anne Asher	Supervisor, Data Entry	Accounts receivable and payable processes	Wed., March 3 10:00–11:00 AM
Fernando Merce	Data Entry Clerk	Accounts receivable and payable processes	Wed., March 3 1:00–3:00 PM

Designing interview questions

Types of Questions	Examples
Closed-ended questions	• How many telephone orders are received per day? • How do customers place orders? • What information is missing from the monthly sales report?
Open-ended questions	• What do you think about the current system? • What are some of the problems you face on a daily basis? • What are some of the improvements you would like to see in a new system?
Probing questions	• Why? • Can you give me an example? • Can you explain that in a bit more detail?

Interviewing Strategies

Top-down

High-level:
Very general

How can order processing be improved?

Medium-level:
Moderately specific

How can we reduce the number of times that customers return ordered items?

Low-level:
Very specific

How can we reduce the number of errors in order processing (e.g., shipping the wrong products)?

Bottom-up

Post-interview: Interview Report

Interview Notes Approved by: Linda Estey

Person Interviewed: Linda Estey,
 Director, Human Resources

Interviewer: Barbara Wixom

Purpose of Interview:
- Understand reports produced for Human Resources by the current system
- Determine information requirements for future system

Summary of Interview:
- Sample reports of all current HR reports are attached to this report. The information that is not used and missing information are noted on the reports.
- Two biggest problems with the current system are:
  1. The data are too old (the HR Department needs information within two days of month end; currently information is provided to them after a three-week delay)
  2. The data are of poor quality (often reports must be reconciled with departmental HR database)
- The most common data errors found in the current system include incorrect job level information and missing salary information.

Open Items:
- Get current employee roster report from Mary Skudrna (extension 4355).
- Verify calculations used to determine vacation time with Mary Skudrna.
- Schedule interview with Jim Wack (extension 2337) regarding the reasons for data quality problems.

Detailed Notes: See attached transcript.

Joint Application Development

- Allows the project team, users, and management to work together to identify requirements for the system

- Often the most useful method for collecting information from users

- Key roles:
  - ❖ Facilitator
  - ❖ Scribe

JAD Meeting Room
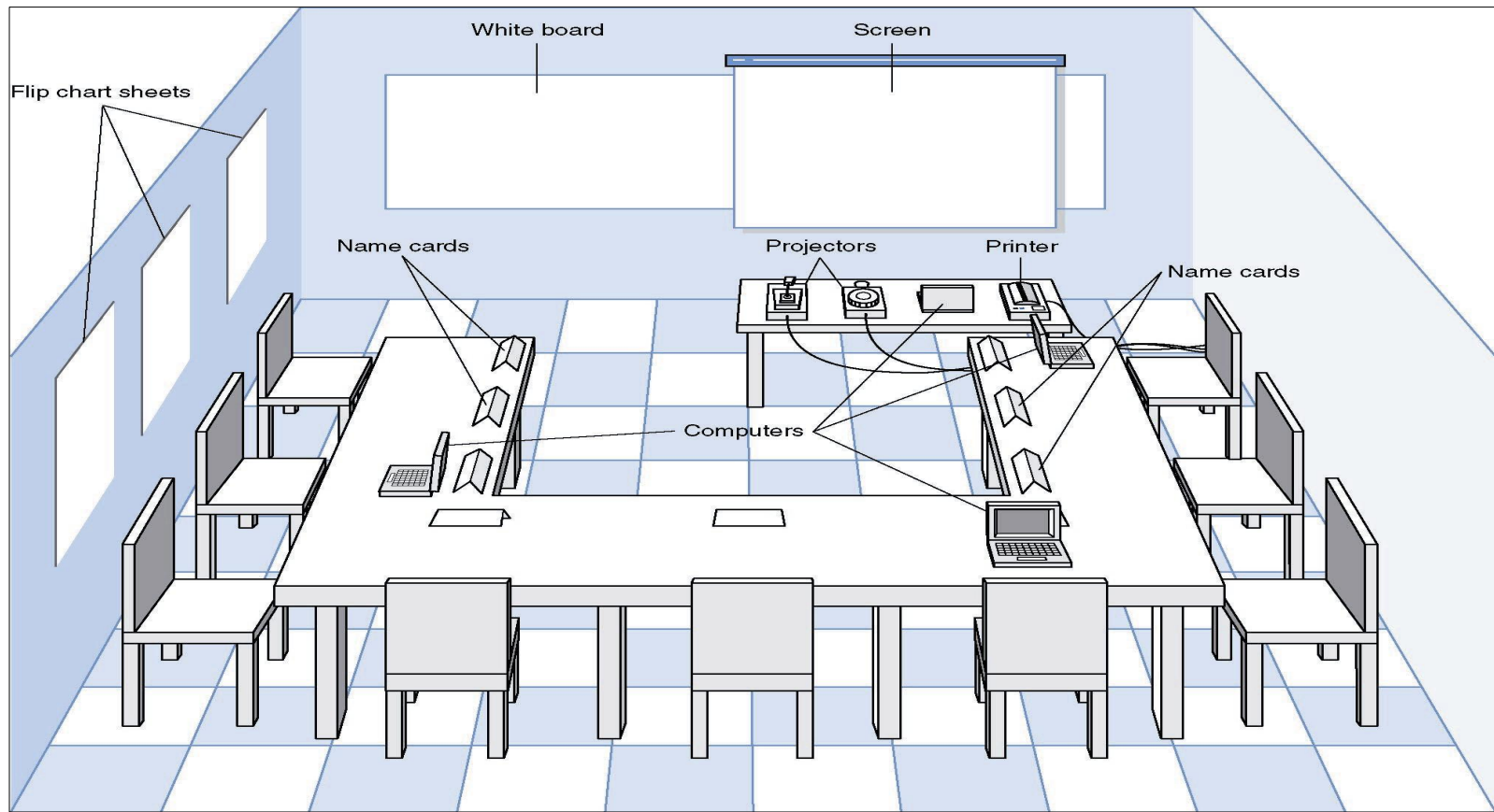
The JAD Session

- Tend to last 5 to 10 days over a three week period
- Prepare questions as with interviews
- Facilitator has to do:
  - ❖ Keep session on track
  - ❖ Help with technical terms
  - ❖ Record group input
  - ❖ Help resolve issues
- Post-session follow-up

Questionnaires

- A set of written questions used to obtain information from individuals

- Often used for large numbers of people from whom information and opinions are needed

- Common technique with systems intended for use outside the organization

- Response rates vary, but typically are significantly less than 50%

Questionnaire Steps

- Selecting participants
  - Using samples of the population
- Designing the questionnaire
  - Careful question selection
- Administering the questionnaire
  - Working to get good response rate
- Questionnaire follow-up
  - Send results to participants

Document Analysis

- Provides clues about existing "as-is" system
- The main document is a technical document. Unfortunately, most systems are not well documented
- Other helpful documents:
  - ❖ User training manuals
  - ❖ User interface of the existing system
  - ❖ Policy manuals
  - ❖ Forms, look for:
    - additional information (made by user) to an existing forms
    - unused form elements

Document Analysis

The customer made a mistake. This should be labeled **Owner's Name** to prevent confusion.

The staff had to add additional information about the type of animal and the animal's date of birth. This information should be added to the new form in the to-be system.

CENTRAL VETERINARY CLINIC
Patient Information Card

Name: ~~Buffy~~ Pat Smith

Pet's Name: Buffy   *Collie 7/6/99*

Address: 100 Central Court. Apartment 10

Toronto, Ontario K7L 3N6

Phone Number: *416-* 555-3400

Do you have insurance: yes

Insurance Company: Pet's Mutual

Policy Number: KA-5493243

The customer did not include area code in the phone number. This should be made more clear.

Observation

- Why observation? Because:
  - ❖ Users/managers often don't remember everything they do
- How to observe?
  - ❖ Observe the business system as it functions
  - ❖ Do not interrupt those who are working
  - ❖ Do not influence those being observed
  - ❖ Checks validity of information gathered other ways
- Watch out, Behaviors change when people are watched
- Observation is often used to supplement interview information

Other Requirement Gathering Techniques

- Throw-away prototyping
- Role playing CRC cards with use cases
- Mind/concept mapping

Selecting Appropriate Techniques

	Interview	JAD	Question-naires	Document Analysis	Observation
Type of information	As-is, improves, to-be	As-is, improves, to-be	As-is, improves	As-is	As-is
Depth of info	High	High	Medium	Low	Low
Breadth of info	Low	Medium	High	High	Low
Info integration	Low	High	Low	Low	Low
User involvement	Medium	High	Low	Low	Low
Cost	Medium	Low-medium	Low	Low	Low-medium

Building the Analysis Model as requirement modelling

● **Elements of the analysis model**
  ❖ Scenario-based elements
  - Functional—processing narratives for software functions
  - Use-case—descriptions of the interaction between an "actor" and the system
  ❖ Class-based elements
  - Implied by scenarios
  ❖ Behavioral elements
  - State diagram
  ❖ Flow-oriented elements
  - Data flow diagram

Q & A