Analysis and Design of Information Systems

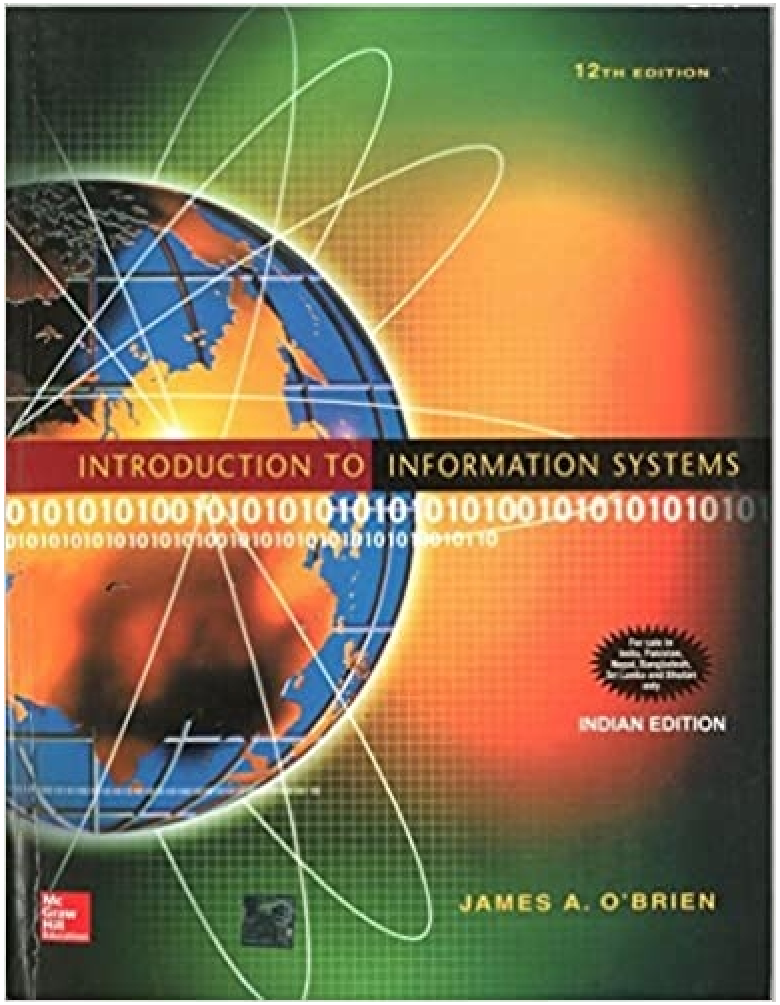
Nguyen Manh Hung
The posts and telecommunications Institute of technology (PTIT)

CHAPTER 1

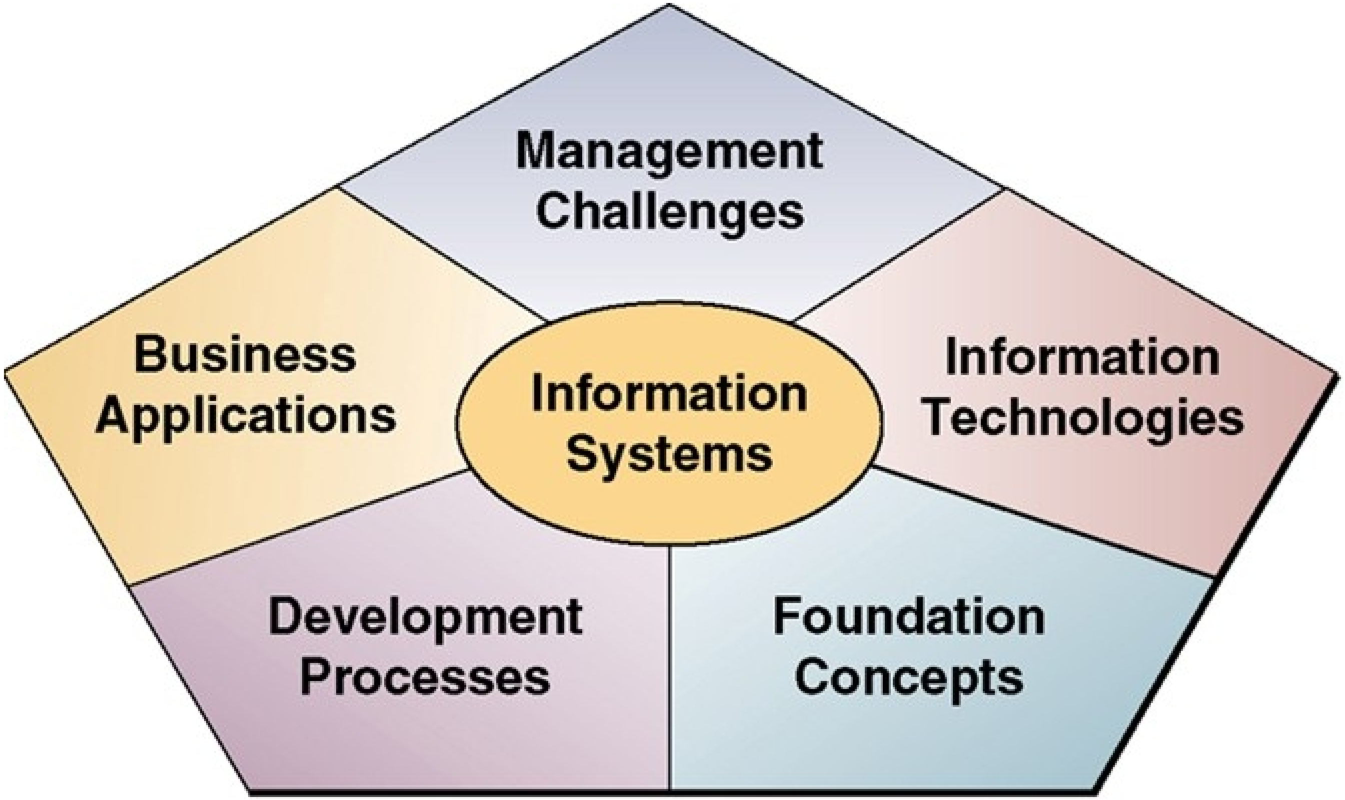
INTRODUCTION TO INFORMATION SYSTEM

Reference

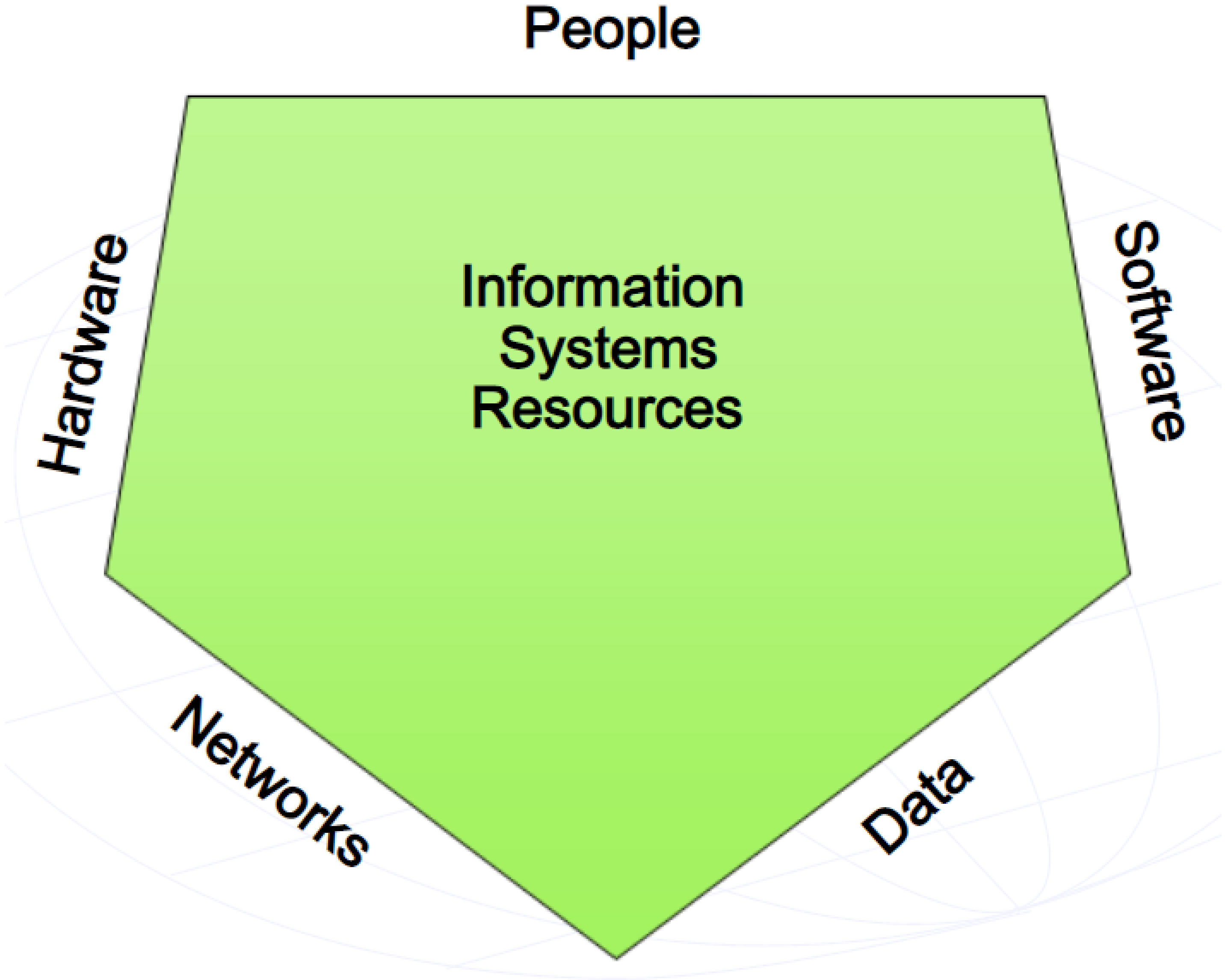
- This chapter refers from the book: Introduction to information systems, 12th edition. James A. O'Brien. McGrow Hill, 2005.



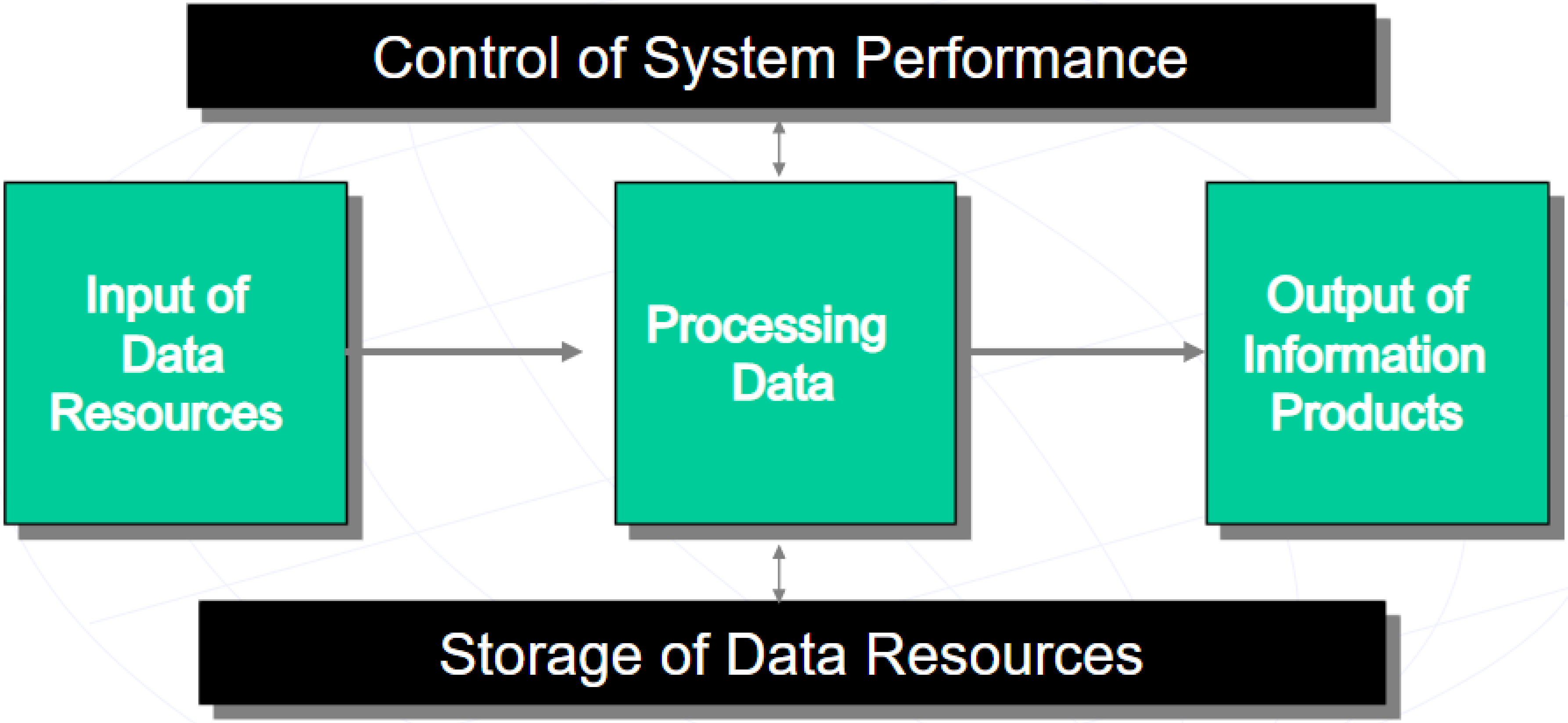
IS framework



Components of an IS



An IS



Control of System Performance

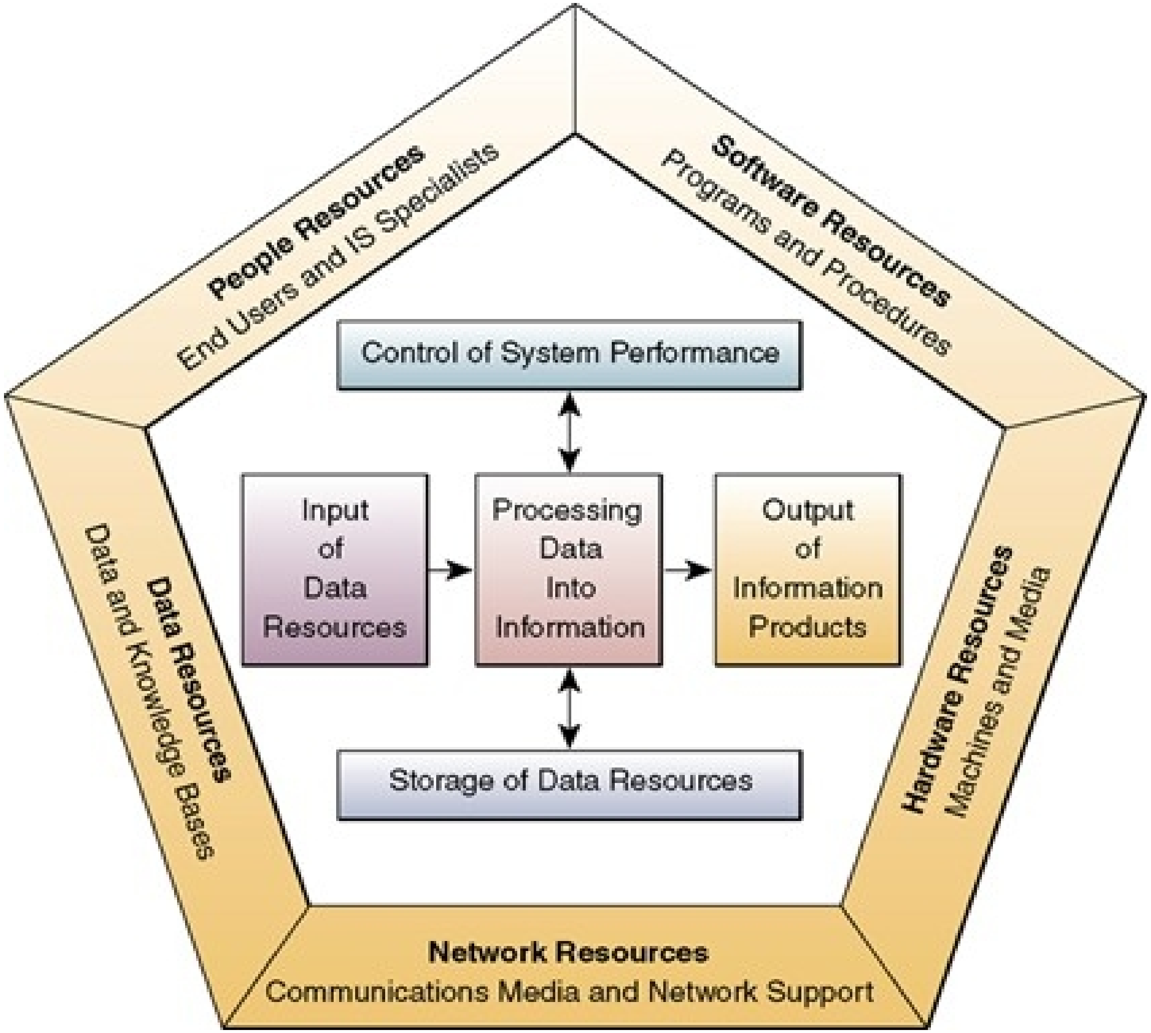
Input of Data Resources

Processing Data

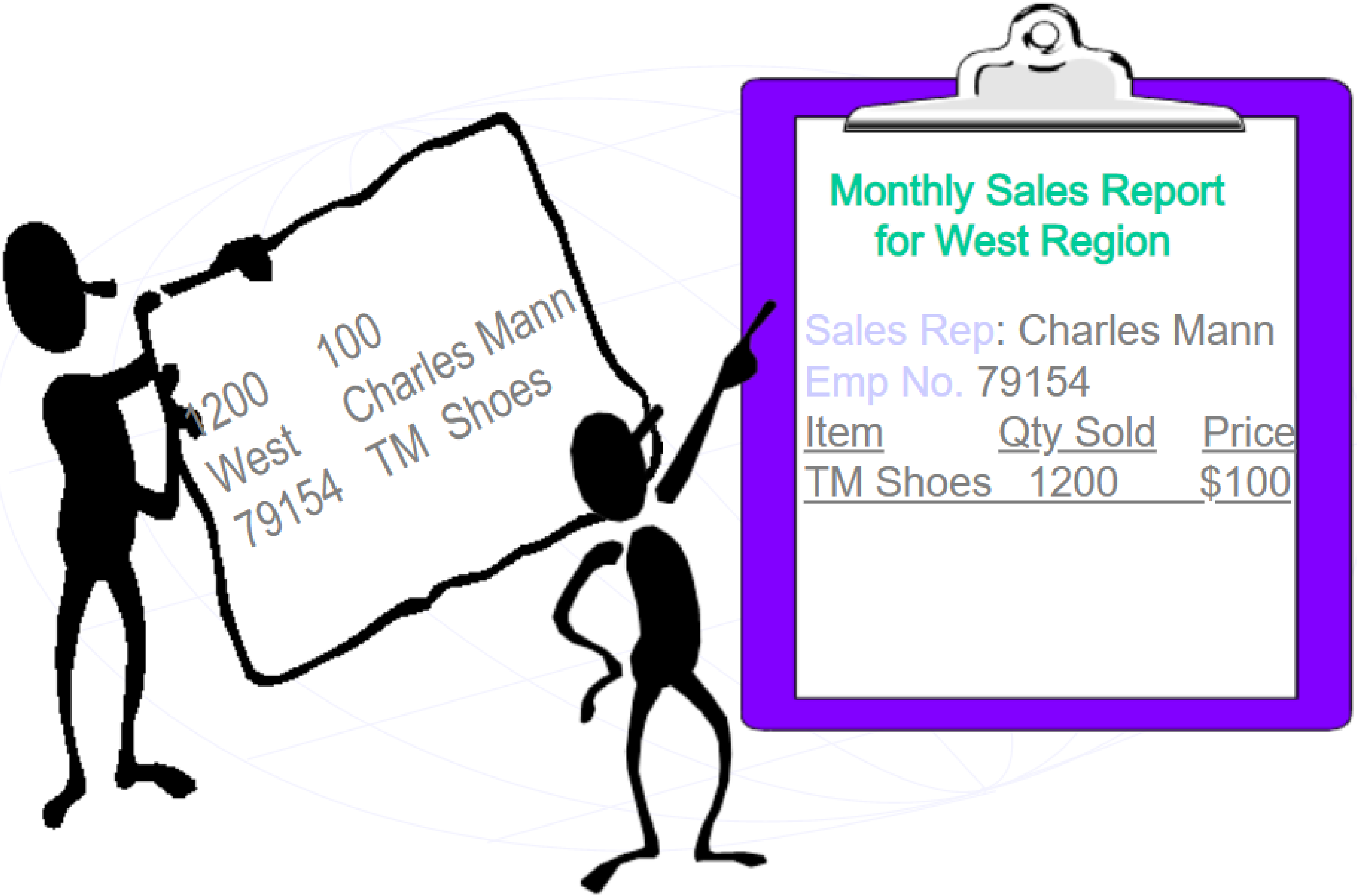
Output of Information Products

Storage of Data Resources

Components of an IS


People Resources
End Users and IS Specialists
Software Resources
Programs and Procedures
Control of System Performance
Input of Data Resources
Processing Data Into Information
Output of Information Products
Storage of Data Resources
Data Resources
Data and Knowledge Bases
Hardware Resources
Machines and Media
Network Resources
Communications Media and Network Support

Data vs. Information

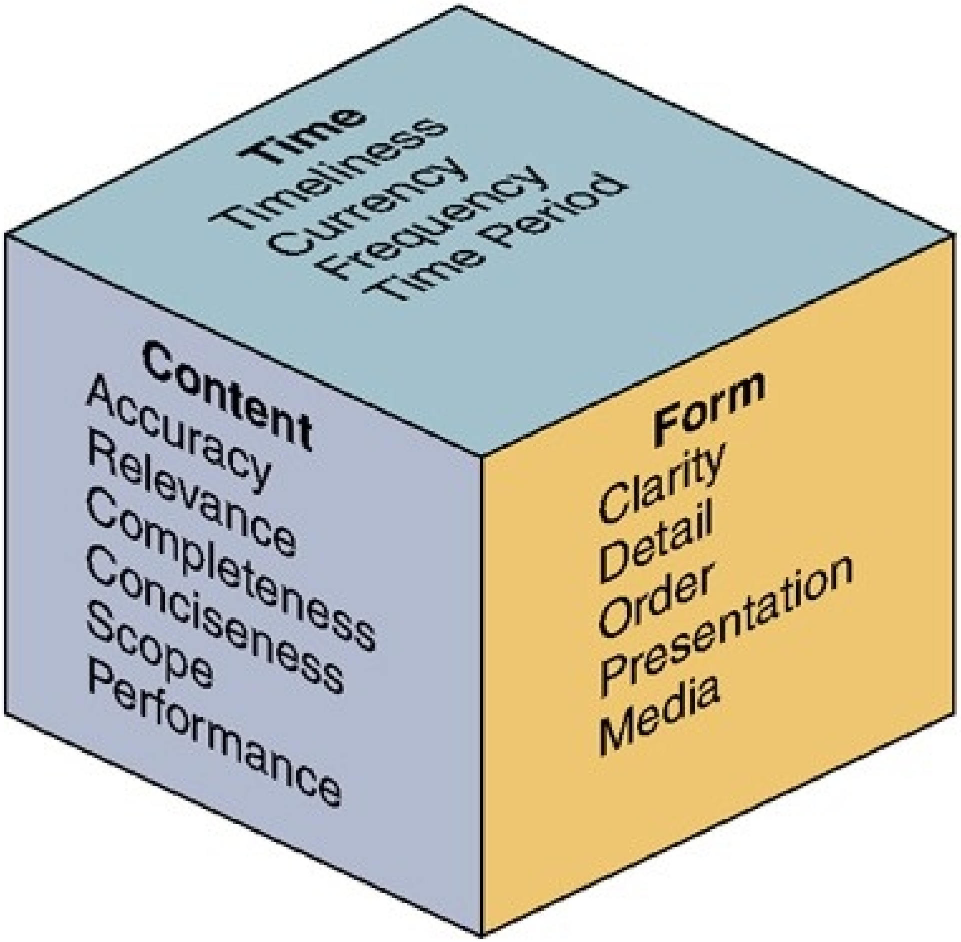


Monthly Sales Report
for West Region

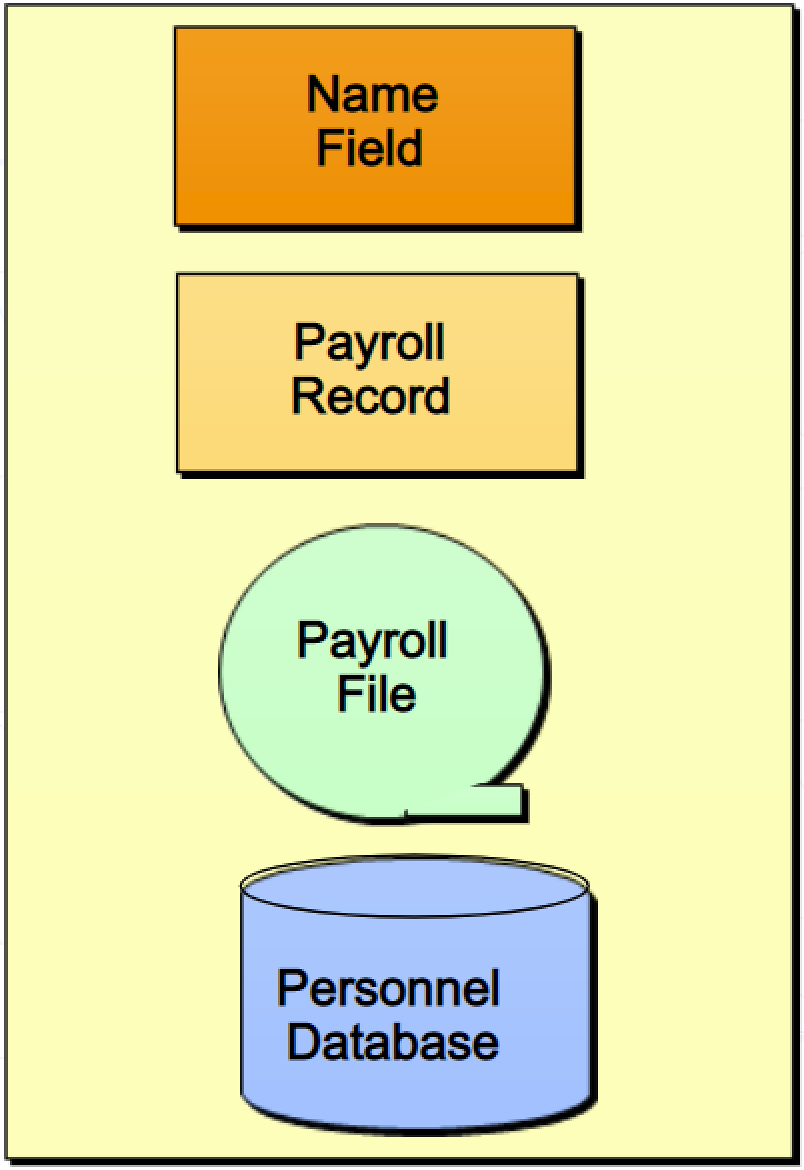
Sales Rep: Charles Mann
Emp No. 79154

Item	Qty Sold	Price
TM Shoes	1200	$100

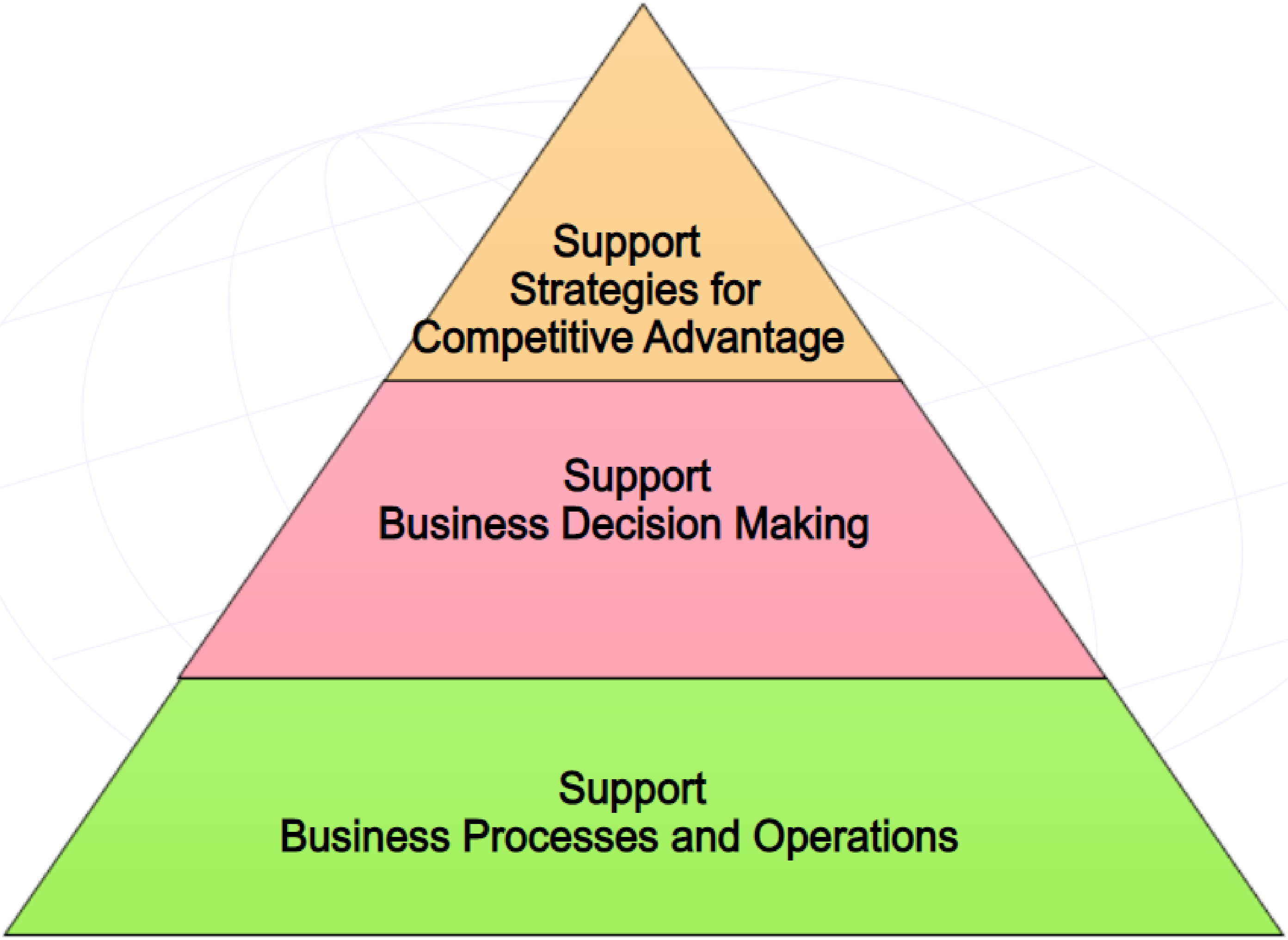
Information quality



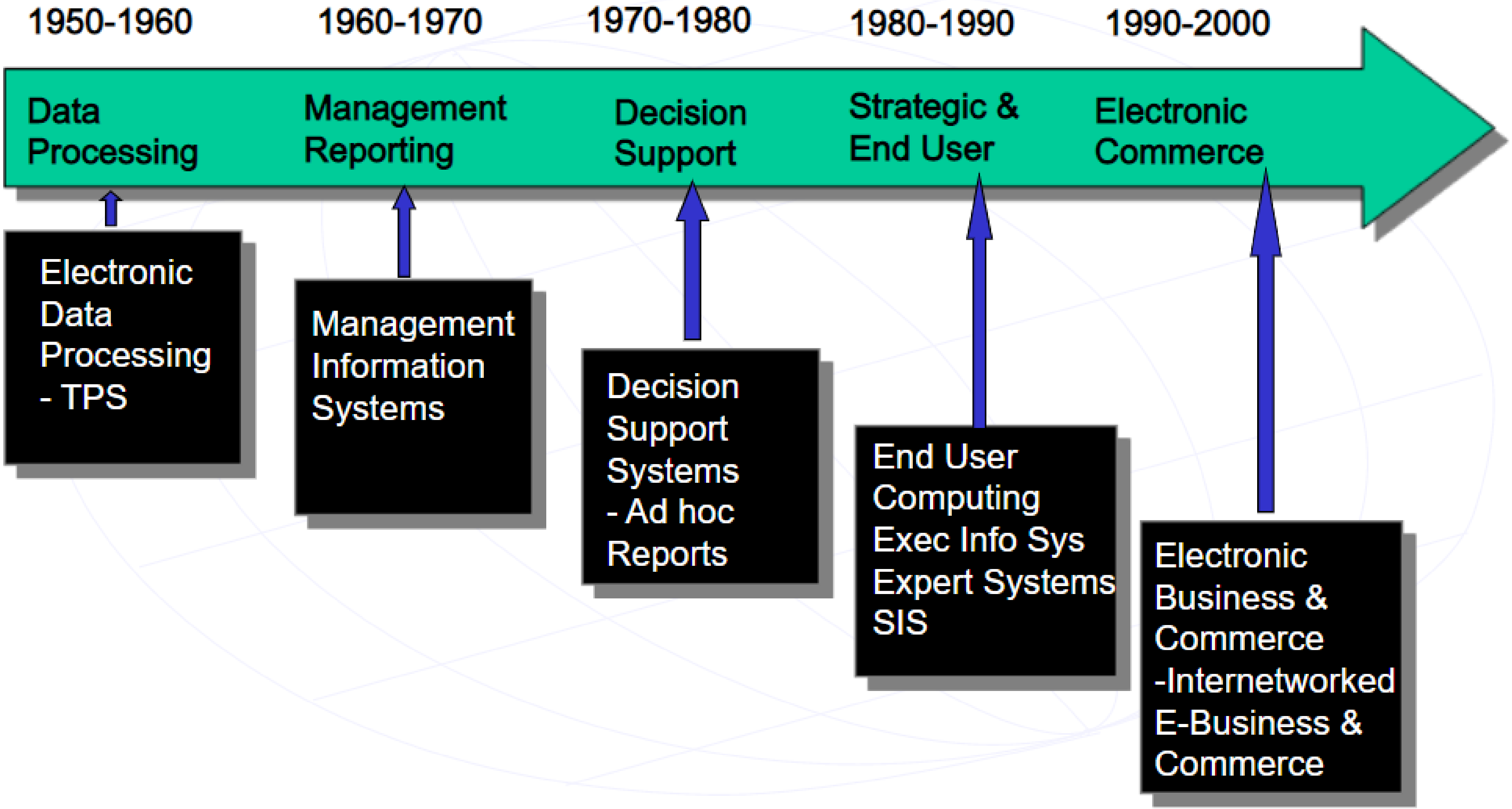
Logical data elements



Roles of IS


Support
Strategies for
Competitive Advantage
Support
Business Decision Making
Support
Business Processes and Operations

Roles of IS: history



1950-1960

Data Processing

Electronic Data Processing - TPS

1960-1970

Management Reporting

Management Information Systems

1970-1980

Decision Support

Decision Support Systems - Ad hoc Reports

1980-1990

Strategic & End User

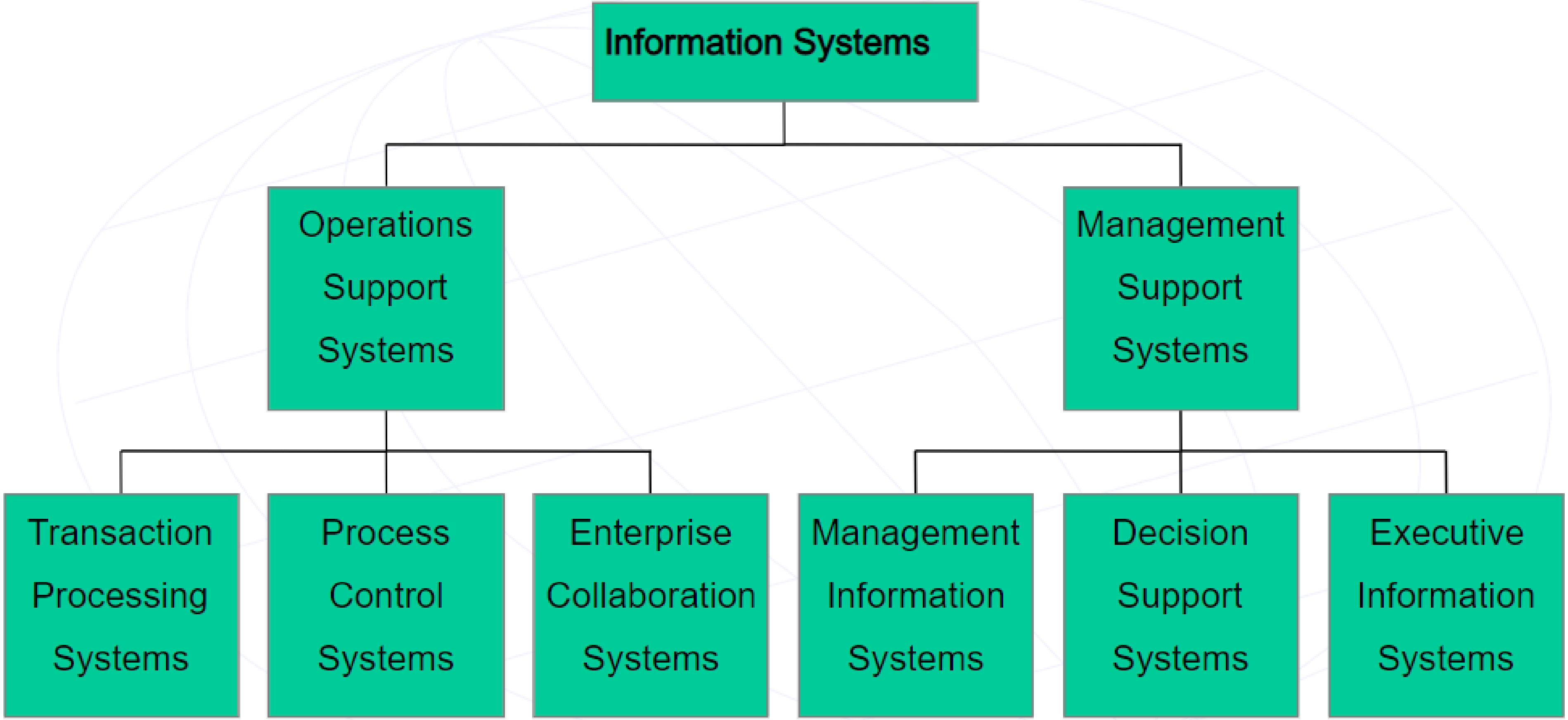
End User Computing Exec Info Sys Expert Systems SIS

1990-2000

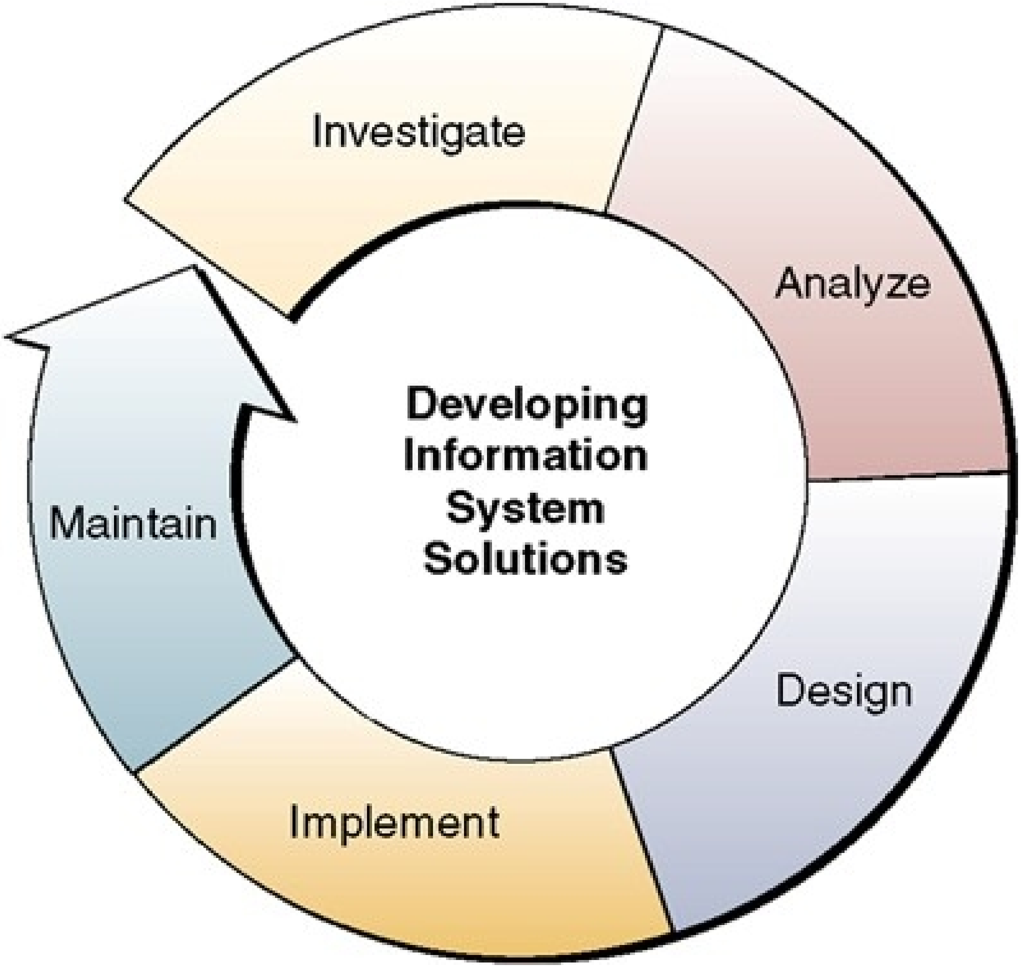
Electronic Commerce

Electronic Business & Commerce -Internetworked E-Business & Commerce

Types of IS


Information Systems
Operations Support Systems
Management Support Systems
Transaction Processing Systems
Process Control Systems
Enterprise Collaboration Systems
Management Information Systems
Decision Support Systems
Executive Information Systems

IS development process



Analysis and Design of Information Systems

Nguyen Manh Hung

The posts and telecommunications Institute of technology (PTIT)

CHAPTER 2

INTRODUCTION TO UML

IS development process

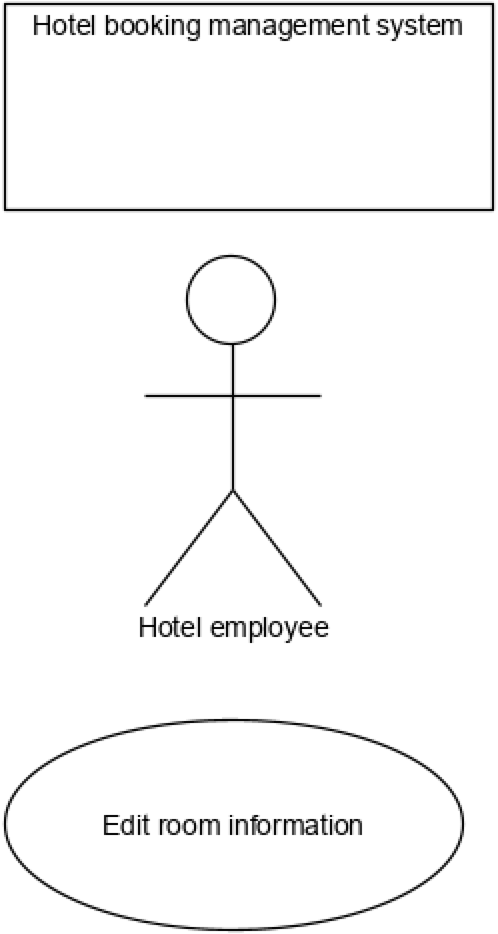
- Requirement
- Analysis
- Design
- Implementation
- Testing

Requirement

- Use case diagram
  - Elements
  - Relationships among elements

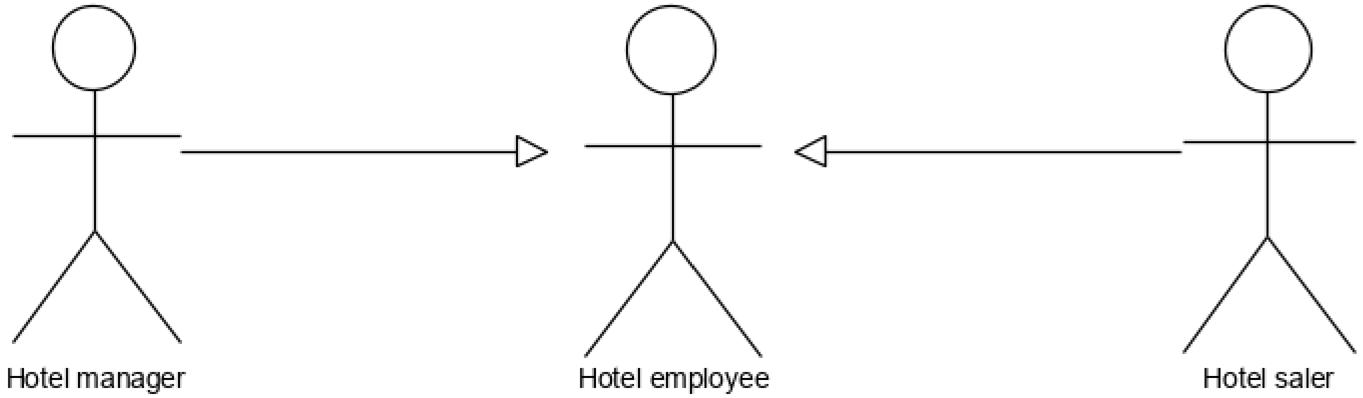
Use case diagram: elements

- System
- Actor
- Use case



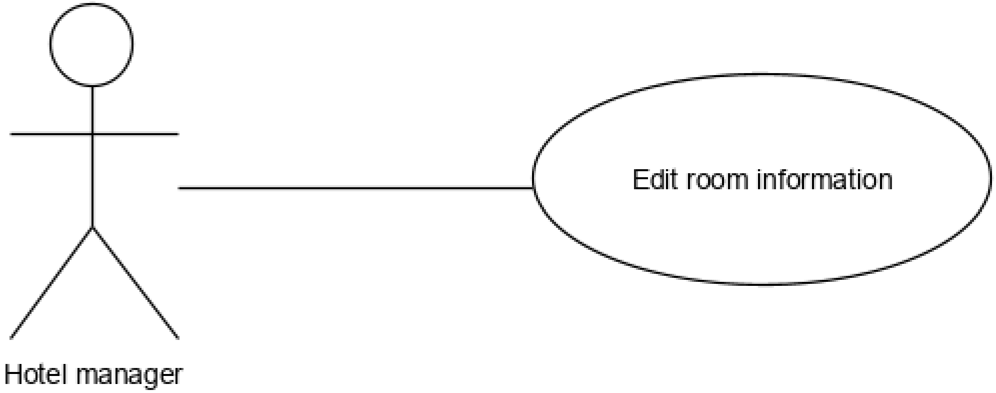
Use case diagram: Actor (1)

- Generalization relationship



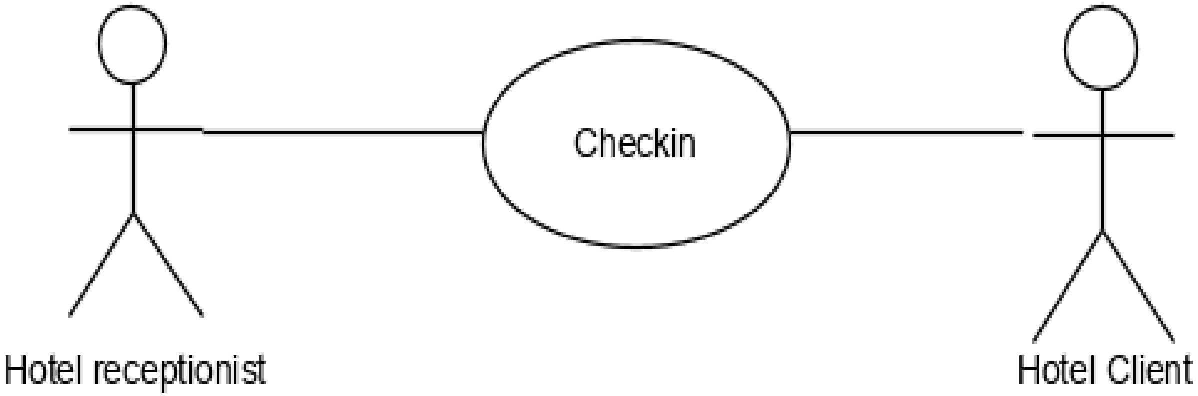
Use case diagram: Actor (2)

- Use case has one actor



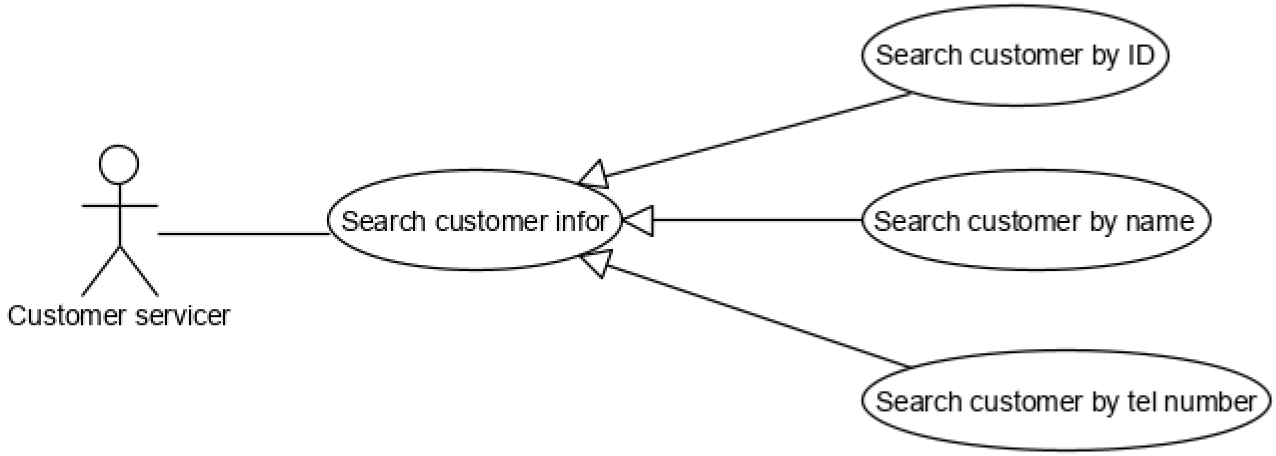
Use case diagram: Actor (3)

- Use case has two actors



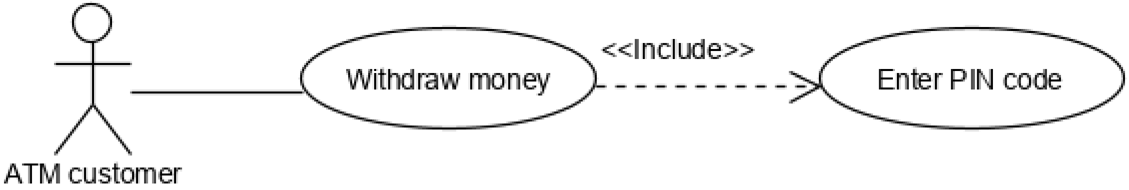
Use case diagram: use case (1)

- Generalization relationship



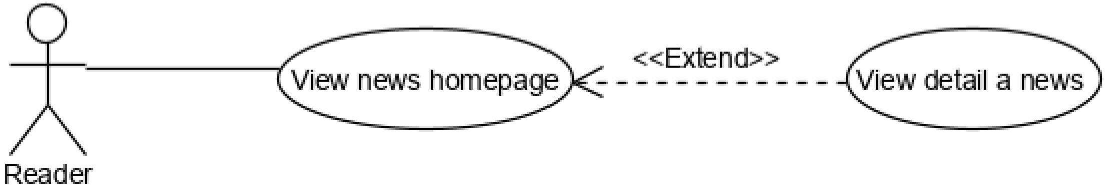
Use case diagram: use case (2)

- Include relationship



Use case diagram: use case (3)

- Extend relationship

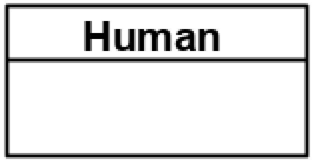


Analysis

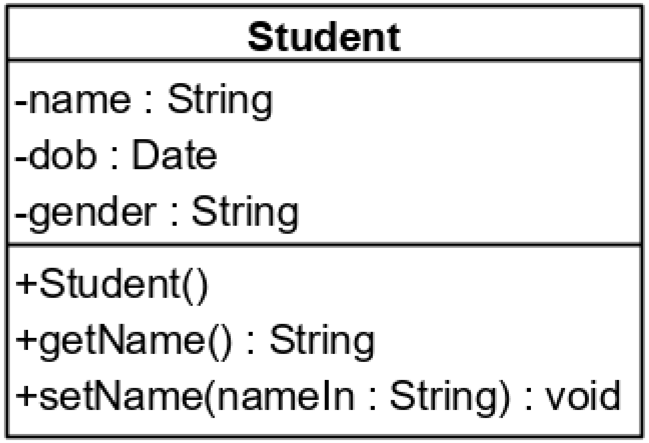
- Class diagram
- State diagram
- Sequence diagram
- Collaboration/communication diagram

Class diagram: elements

- Class



Human



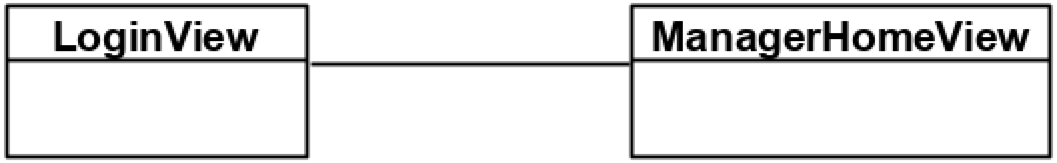
Student

-name : String
-dob : Date
-gender : String

+Student()
+getName() : String
+setName(nameIn : String) : void

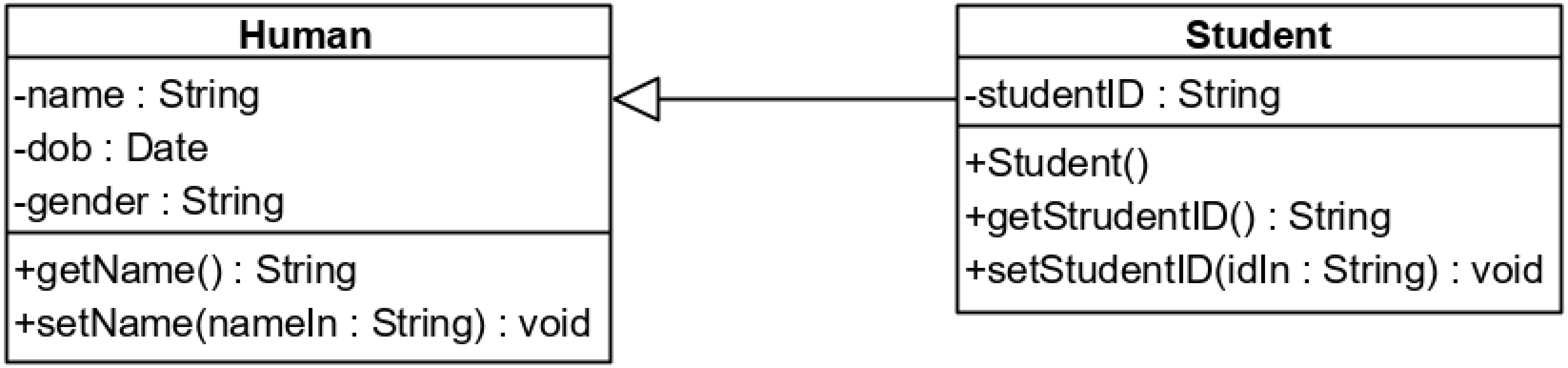
Class diagram: relationship (1)

- Interaction



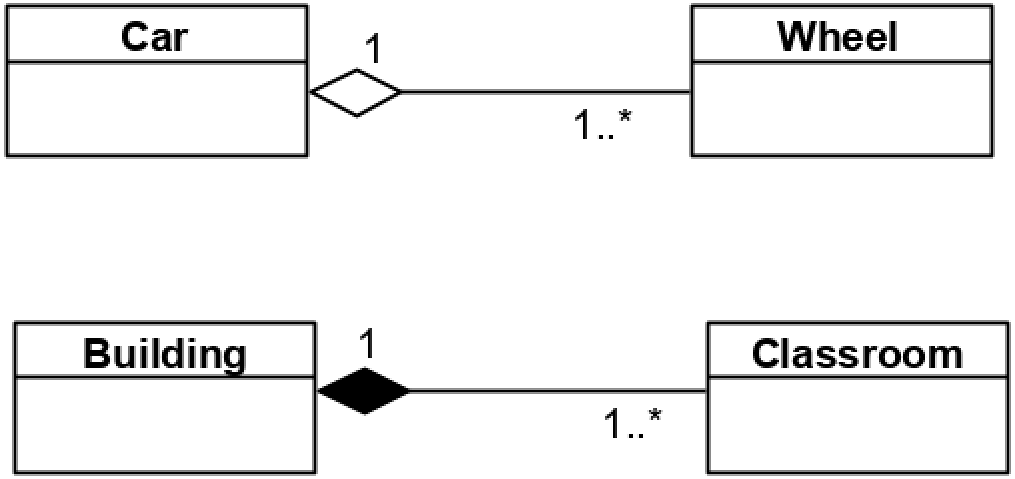
Class diagram: relationship (2)

- Generalization



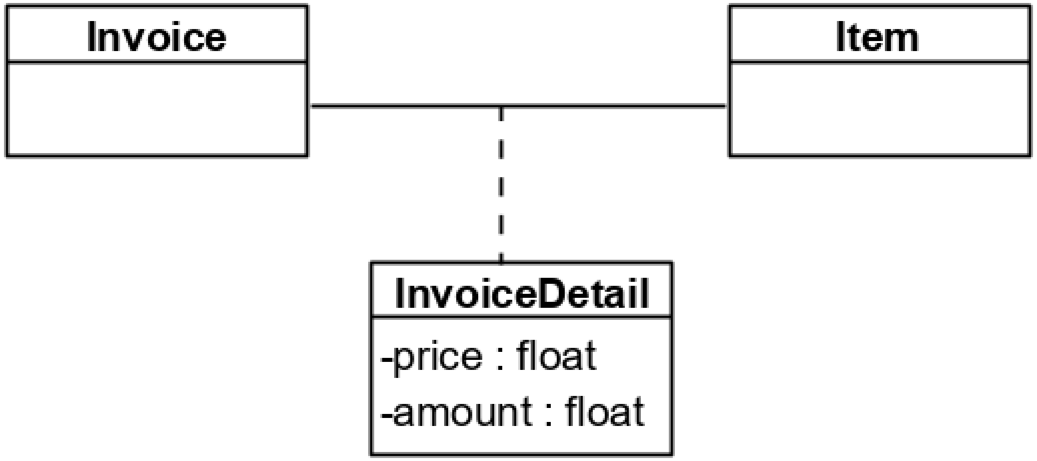
Class diagram: relationship (3)

- Aggregation vs. composition



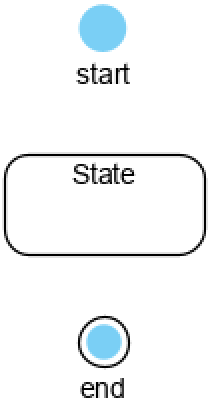
Class diagram: relationship (4)

- Association



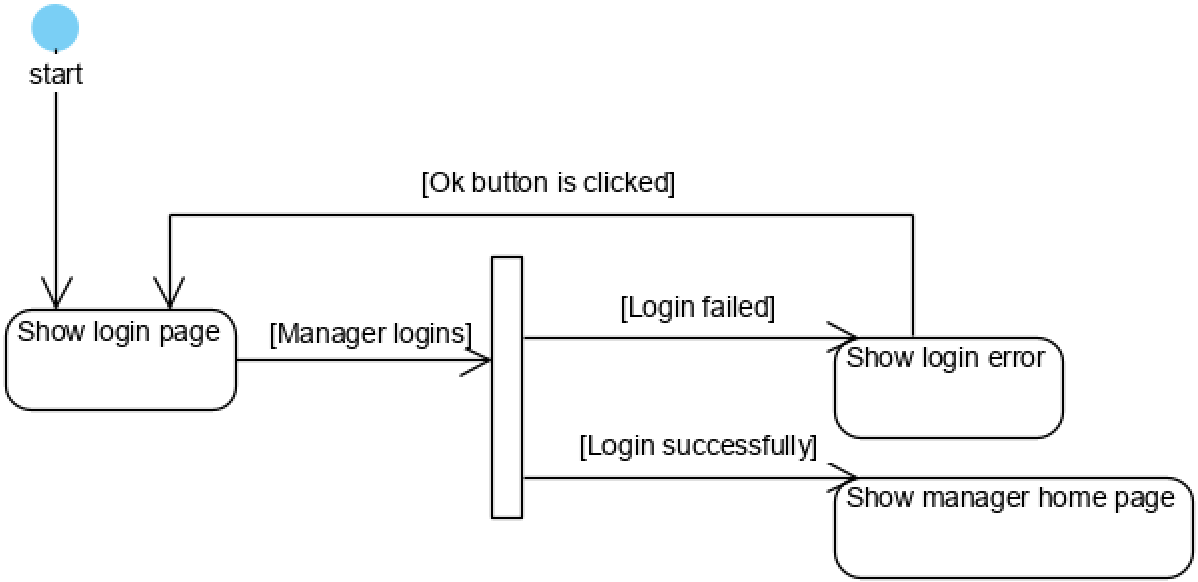
State diagram: elements

- Start
- State
- End



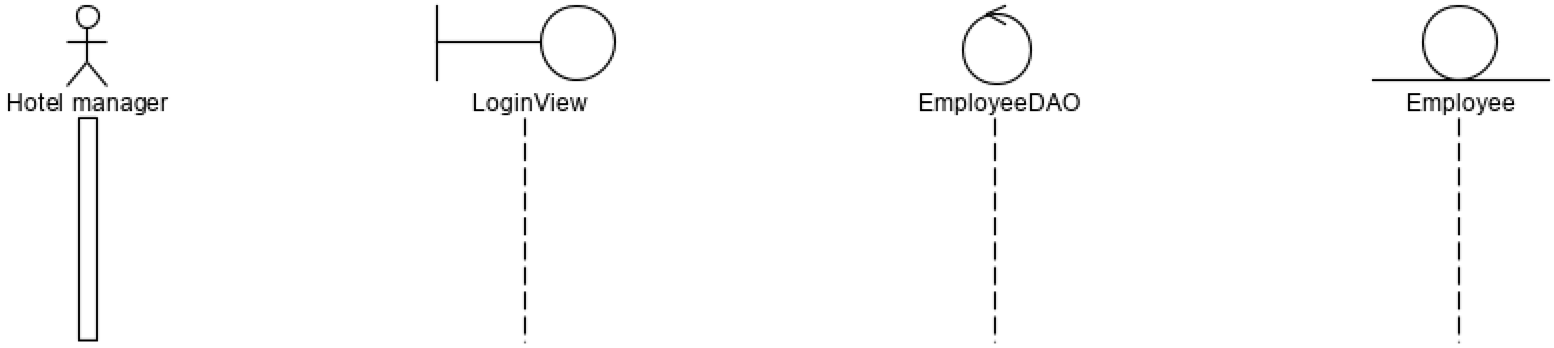
State diagram: relationship

- Change state



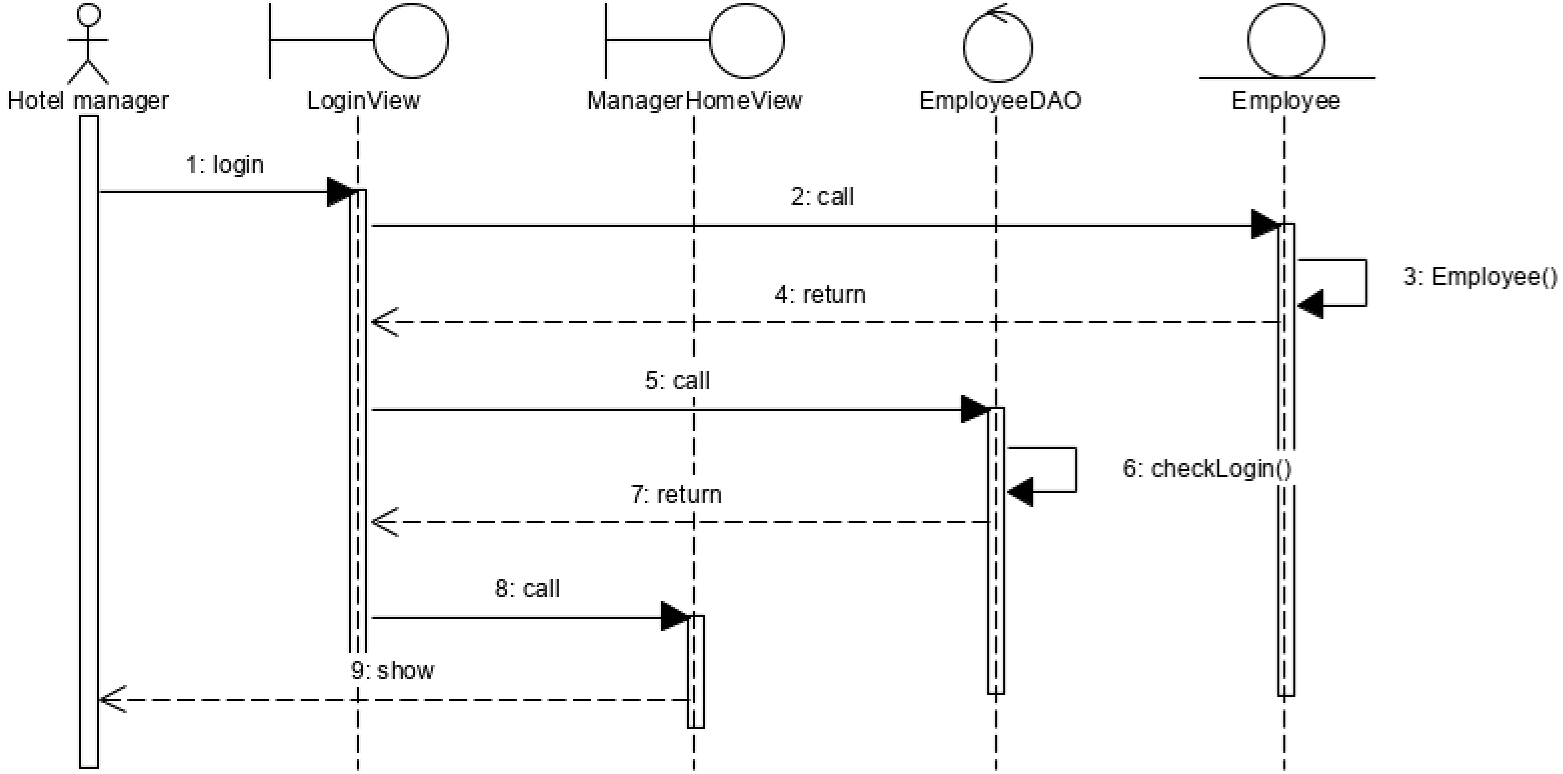
Sequence diagram: elements

- Actor
- View/interface/boundary class
- Control/business class
- Model/entity class



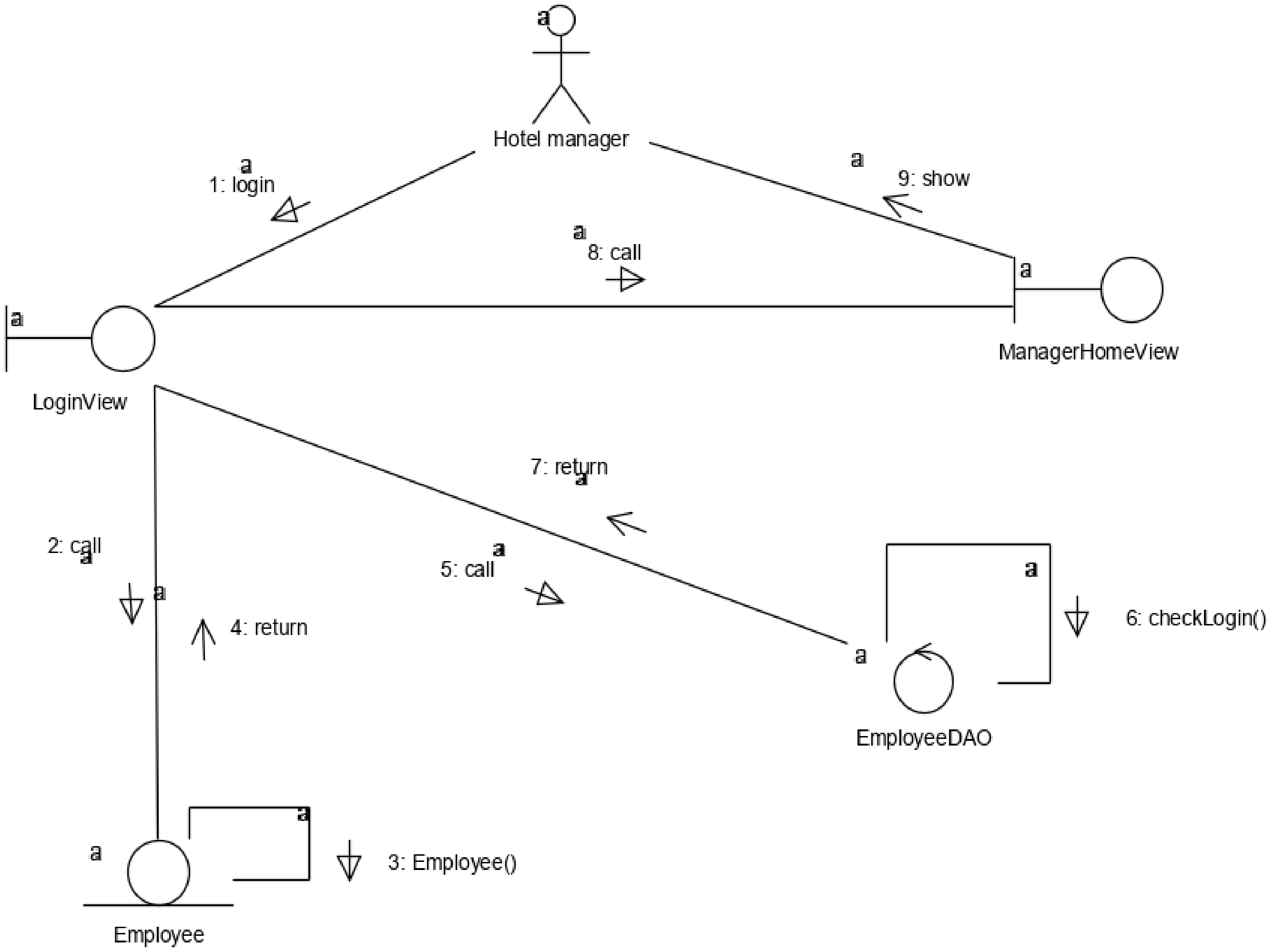
Sequence diagram: event steps

- Ex: login



Communication diagram: event steps

- Ex: login

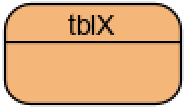


Design

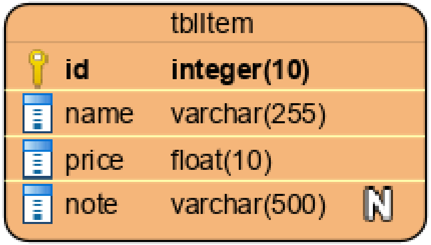
- Class diagram (entity, full detail – presented)
- Database diagram
- Activity diagram
- Sequence/communication diagram (presented)
- Package diagram
- Deployment diagram

Database diagram: elements

- table



tblX

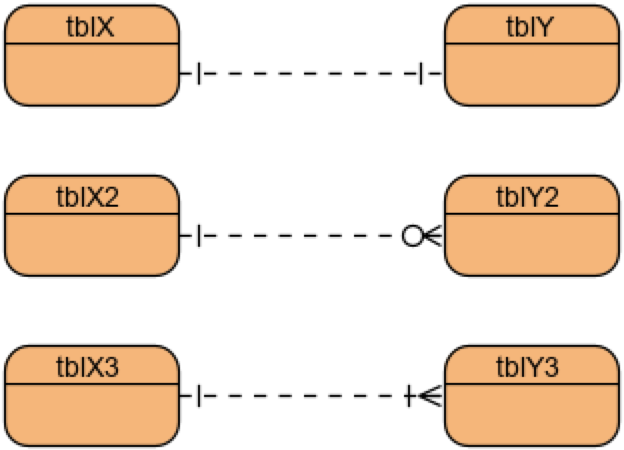


tblItem

	id	**integer(10)**	
	name	varchar(255)	
	price	float(10)	
	note	varchar(500)	N

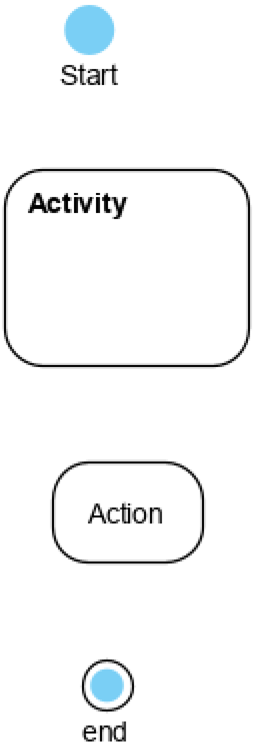
Database diagram: relationships

- 1-1
- 1-n
- n-n (convert to many 1-n relationships



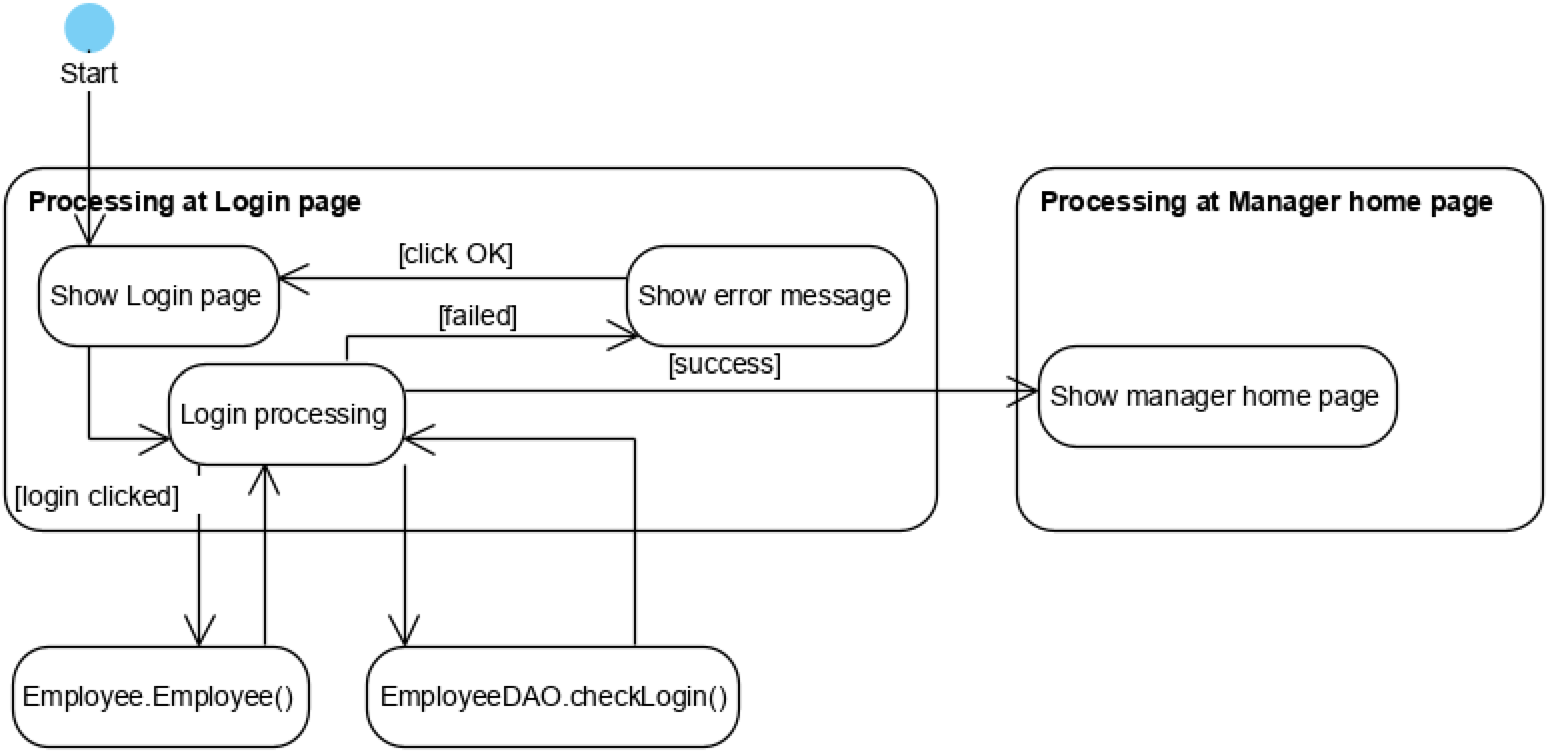
Activity diagram: elements

- Start
- Activity
- Action
- End



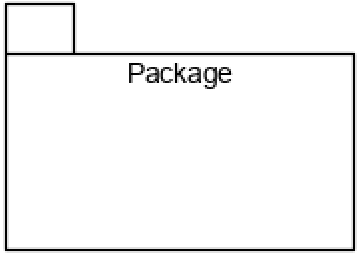
Activity diagram: relationships

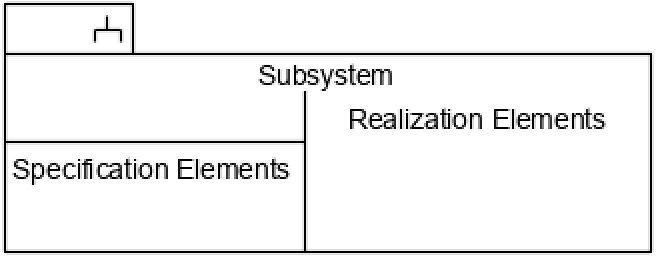
- Change action



Package diagram

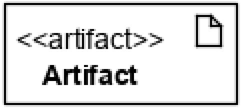
- Package
- sub-system

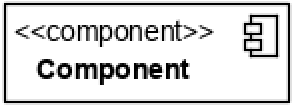


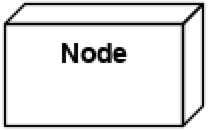


Deployment diagram

- artifact
- node
- component


<<artifact>>
Artifact


<<component>>
Component


Node

Analysis and Design of Information Systems

Nguyen Manh Hung

The posts and telecommunications Institute of technology (PTIT)

CHAPTER 3

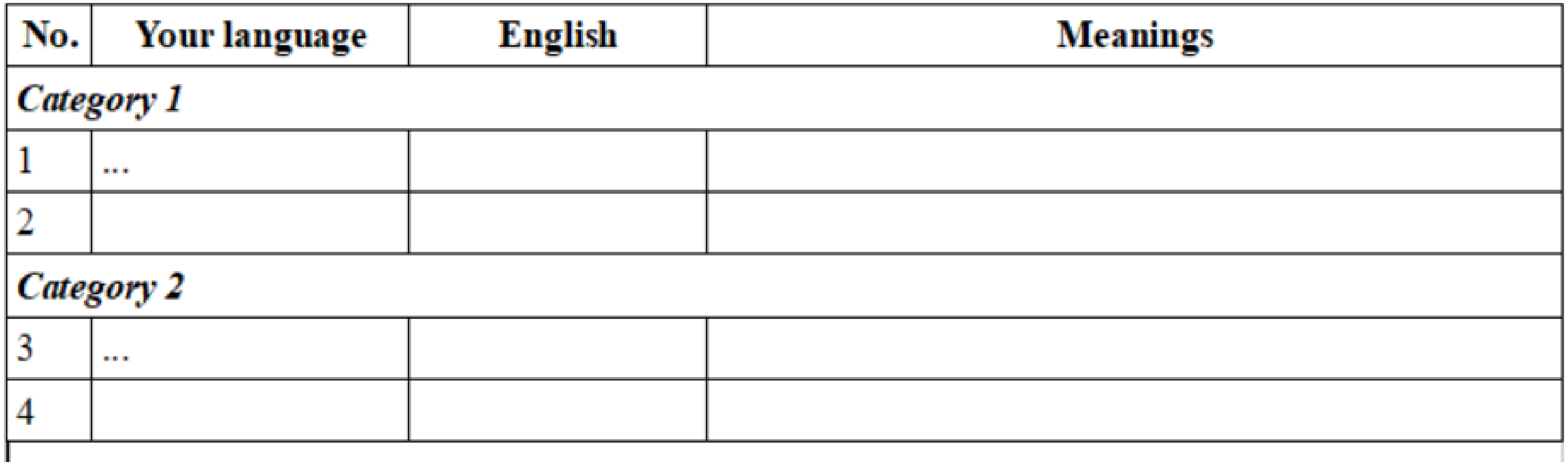
REQUIREMENTS

Requirement steps

- Concept exploration
  - Discover term/concepts in the application domain
  - Build the glossary list
- Business model
  - Description by natural language
  - Description by UML

Concept exploration

- Glossary list
  - Discover: brain storming, teamwork
  - Organise into glossary list



No.	Your language	English	Meanings
Category 1			
1	...		
2			
Category 2			
3	...		
4			

BM: natural language (1)

- Objective?
- Scope?
- How do the modules work?
- Information about objects?
- Relationships among objects?

BM: natural language (1)

- Objective
  - Description about the system
- Scope
  - Which type of application? (web, descktop, mobile)
  - Who can directly use the system?
  - Who can indirectly use the system?
  - What are the functions that each user could do?

BM: natural language (2)

- Operating in each function
  - Description about the order of steps to process in the function
  - Description the information displayed in each step
  - Description the action of the user in each step
  - Description all possible cases could happen after an user action at each step

BM: natural language (3)

- Object information in the system
  - Detect all entities/objects need to be managed or used in the system
  - Detect all necessary attributes for each entity/object in the system
  - Detect the data type and the value range of each attribute of entity/object.

BM: natural language (4)

- Relationships among objects in the system
  - Detect all possible quantity relationships among entities/objects
  - 1-1 relationship (zero or one, exactly one)
  - 1-n relationship (zero or more, one or more)
  - n-n relationship (zero or more, one or more)

BM: UML (1)

- General use case diagram (for the whole system)
- Detail use case diagram (for each function)

BM: UML (2)

- General use case diagram (for the whole system)
  - Detect actors of the system
  - Detect use cases for each actor
  - Refine the diagram

BM: General use case (1)

- Detect actors of the system
  - Input: the scope of the system by natural language
  - Each (direct/indirect) user → create an (direct/indirect) actor
  - Proposal some abstract actors if necessary

BM: General use case (1)

- Detect actors of the system
  - Input: the scope of the system by natural language
  - Each (direct/indirect) user → create an (direct/indirect) actor
  - Proposal some abstract actors if necessary

BM: General use case (2)

- Detect use cases of actor
  - In: the scope of the system by natural language
  - Each function of an user → create an use case for the corresponding actor
- Refine use case:
  - Proposal some abstract use cases if necessary

BM: Detail use case (1)

- Extract the main use case from the general UC
- Detect related sub use cases
- Detect the relationship to the main usecase
- Description each use case

BM: Detail use case (2)

- Extract the main use case from the general UC
  - Extract the actor(s) and the main use case from the general use case diagram
  - Extract the relationships among extracted actor(s) and the extracted main UC
- Detect related sub use cases
  - Input: description of the function operating in NL
  - Each interface with user → propose a sub use case
  - Ignore all alerting, confirmation or simple messages

BM: Detail use case (3)

- Detect relationships to the main UC
  - For each new sub use case, detect if it has include/extend relationship to the main UC
  - Some similar use cases may have the same abtract parent use case
- Description of use case
  - Each use case has a brief description: This use case enables who (someone) to do what (something)

Requirement

APPLY TO THE CASE STUDY

Analysis and Design of Information Systems

Nguyen Manh Hung
The posts and telecommunications Institute of technology (PTIT)

CHAPTER 4

ANALYSIS

Analysis steps

- Scenarios
- Entity class extraction
- State diagram
- Module class diagram
- Sequence/communication diagram

Scenarios

- The standard scenario
  - There is no error in the operation of the system
  - There is no error or illogic in the manipulation of the actor
- All exception scenarios
  - In case of error or inexpected results

Entity class diagram

- Extract entity class
  - Extract all nouns from all related scenarios
  - Consider if the noun could represent an entity class or not
  - Detect all neccesary attribute of each entity class
- Detect relationships among entity classes
  - 1-1 relationship: could be merged
  - 1-n relationship: let's it be
  - n-n relationship: propose more intermediate class(es) between them, if neccesary

State diagram

- Propose states
  - An interface to interact to user → a state
  - Ignore simple message
- Relationships among states
  - Trigger to change state: user action

Class diagram of module(1)

- Boundary/view/interface classes
  - Input: all scenarios + state diagram
  - An interface to interact to user or a state → a view class
  - Detect attributes of each view class:
    - Input attribute
    - Output attribute
    - Control/redirect attribute
    - Combined of them

Class diagram of module(2)

- Processing at the lower level
  - Each data processing → create a method
  - Detect input/output data
  - Assign the method to a related entity class:
    - Output related entity
    - Input related entity
- Relationships among classes
  - Extract all related relationships among entity classes of the module
  - Create a relationship if there are interaction between two view classes or between a view class and an entity class,

Sequence/communication diagram

- Scenario v.2
  - The user clicks on the interface of class X
  - Class X calls/requires class Y to do A
  - Class Y does method A...
- Diagram
  - A scenario has a diagram
  - The entity classes are also control classes (have actions/methods)
  - Convert from sequence to communication diagram

Analysis

APPLY TO THE CASE STUDY

Analysis and Design of Information Systems

Nguyen Manh Hung

The posts and telecommunications Institute of technology (PTIT)

CHAPTER 5

DESIGN

Design steps

- Design of entity classes
- Design of database
- Design of interface
- Detail design module class diagram
- Activity diagram of module
- Sequence/communication diagram
- Package diagram
- Deployment diagram

Entity class diagram at the design phase

- Input
  - The entity class diagram at the analysis phase
- Process
  - Add id attribute to all classes that are not inherited from any class
  - Design the datatype to all attributes
  - Convert association relationships to aggregation/composition relationships
  - Add the object attributes corresponding to the aggregation/composition relationships

Design of database (1)

- Input
  - The entity class diagram of the design phase
- Process
  - An entity class → create a table
  - Non-object attribute of the class → attritue of the corresponding table
  - Quantity relationships between two classes → quantuty relationships between the two corresponding tables
    - 1-1: should merged
    - 1-n: let's it be
    - n-n: return to the entity class diagram to correct it

Design of database (2)

- Process
  - Key attributes
    - Primary key: the id of the tables which have it
    - Foreinger key: if tblA – tblB have an 1-n relationship → the tblB must have a FK which refers to the PK of the tblA.
  - Remove redondant attributes
    - Doublicate attributes
    - Secondary attributes

Design of interface

- Process
  - An interface to interact to user
  - Combination of some simple interfaces into one
  - Message/dialogue/confirmation/Alert

Class diagram of module (1)

- View classes
  - Input: interface design
  - An interface → a view class
  - Design explicite attributes view class:
    - Input attribute
    - Output attribute
    - Control/redirect attribute
    - Combined of them
  - Design implicite attributes of view class
    - To receive data from previous class

Class diagram of module (2)

- Processing at the lower level
  - Each data processing → create a method
  - Design input parameters
  - Design output parameters
  - Assign the method to a related DAO class:
    - Output related entity
    - Input related entity
- Relationships among classes

Activity diagram of module

- Process
  - Processing at an interface → an activity
  - Each method → action
  - Consider all possible cases

Sequence/communication diagram

- Scenario v.3
  - The user clicks on the interface of class X
  - Class X calls/requires class Y to do A
  - Class Y does method A...
- Diagram
  - A scenario has a diagram
  - Each method has a sub-life line
  - Convert from sequence to communication diagram

Package and deployment diagram

- Package
  - Present all packages
  - All classes included in each package
- Deployment
  - Site of database
  - Site of server(s)
  - Site of client

Design

APPLY TO THE CASE STUDY