

OBJECT-ORIENTED SYSTEMS DEVELOPMENT: USING THE UNIFIED MODELING LANGUAGE



Object Analysis: Classification

GOALS

- The concept of classification
- How to identify classes with the noun phrase approach
- How to identify classes with the common class patterns approach
- How to identify classes and object behavior analyzed by sequence/collaboration modeling
- Class responsibilities collaboration (CRC) approach



... Intelligent classification is intellectually hard work, and it best comes about through an incremental and iterative process Booch

..There is no such thing as the perfect class structure, nor the right set of objects. As in any engineering discipline, our design choice is compromisingly shaped by many competing factors.

~ Booch



CLASSIFICATION THEORY

- Classification is the process of checking to see if an object belongs to a category or a class and it is regarded as a basic attribute of human nature
- It guides us in making decision about modularization
- Classes are an important mechanism for classifying objects- the chief role of class is to define attributes, methods, and applicability of its instances



CONT...

○ Certain confusions:

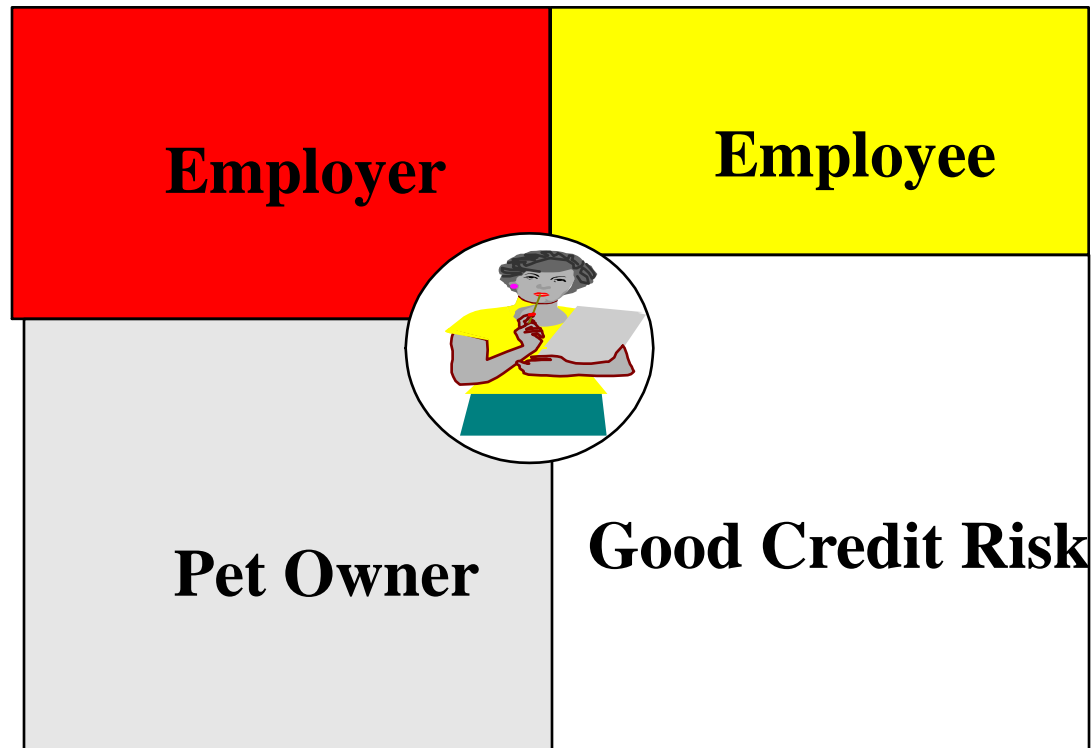
- Its useful to divide the world in objects having properties(attributes) and methods(behaviors)
- But we can also divide it into second dimension: distinguish classes from objects
- A class is a specification of structure, behavior, and the description of an object
- Hence classification is more concerned with identifying the class of an object than individual object within the system



THE CHALLENGE OF CLASSIFICATION

- Intelligent classification is intellectually hard work and may seem rather arbitrary
- Martin and Odell have observed in object-oriented analysis and design, that “In fact, an object can be categorized in more than one way”





- The same object, Betty, can be classified in many ways
- One person may regard Betty as a woman
- Her boss regards her as an employee. The person who mows her lawn classifies her as an employer
- The local animal control agency licenses her as a Pet Owner

THE PROBLEM

- The problem of classification may be regarded as one of the discriminating things, not between the individual objects but between classes
 - Of course via the searches for features or invariant attributes or behaviors among members of a class
- Again classification can be defined as categorization of input data into identifiable classes via extraction of significant features or attributes of the data from a background of irrelevant detail



APPROACHES FOR IDENTIFYING CLASSES

- Following are the alternative approaches to identifying classes
 - The noun phrase approach
 - The common class patterns approach
 - The use-case driven approach
 - The class responsibilities collaboration (CRC) approach



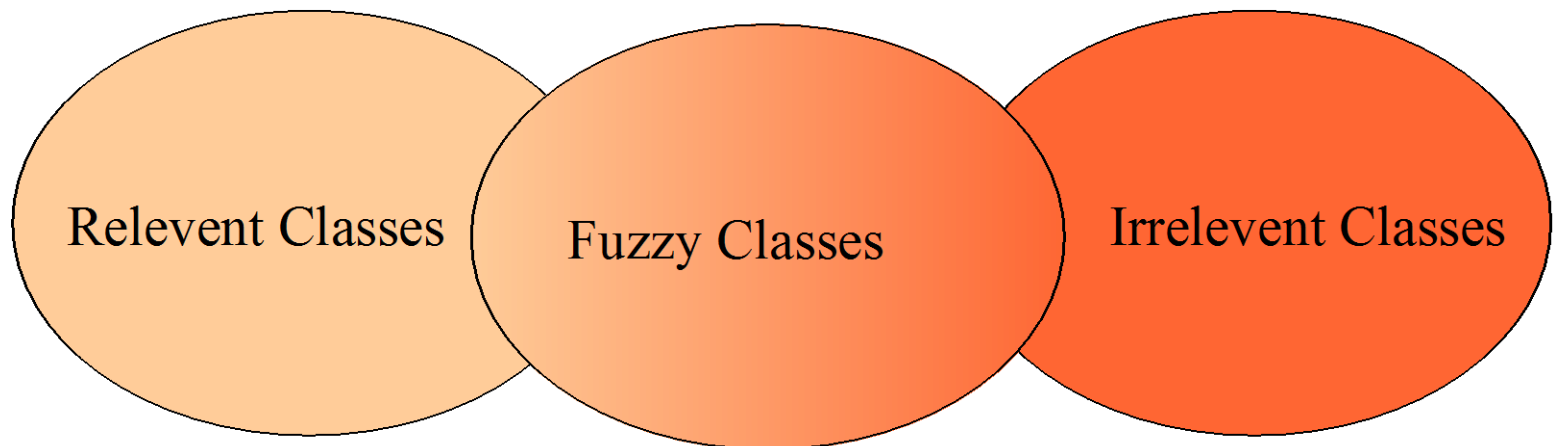
NOUN PHRASE APPROACH

- Using this method, you have to read through the Use cases, interviews, and requirements specification carefully, looking for noun phrases
- Nouns in the textual description are classes and verbs are methods of the classes



NOUN PHRASE STRATEGY (CON'T)

- Change all plurals to singular and make a list, which can then be divided into three categories



NOUN PHRASE STRATEGY (CON'T)

- It is safe to scrap the Irrelevant Classes
- You must be able to formulate a statement of purpose for each candidate class; if not, simply eliminate it
- You must then select candidate classes from the other two categories



GUIDELINES FOR IDENTIFYING CLASSES

- The following are guidelines for selecting classes in your application:
 - Look for nouns and noun phrases in the problem statement
 - Some classes are implicit or taken from general knowledge
E.g if library system : Book, member etc.
If Bank system : customer, Bank etc.
- A noun means a content word that can be used to refer to a person, place, thing, quality, or action
- All classes must make sense in the application domain
- Avoid computer implementation classes, defer it to the design stage
- Carefully choose and define class names



GUIDELINES FOR REFINING CLASSES

Redundant Classes:

- Do not keep two classes that express the same information
- If more than one word is being used to describe the same idea, select the one that is the most meaningful in the context of the system
- While choosing vocabulary always select the word that is used by the user of the system



Guidelines For Refining Classes (Cont...)

(ii) Adjective Classes:

- An adjective is a word that expresses an attribute of something or the word class that qualifies nouns.
- Adjectives can be used in many ways.
 - It modifies a noun. It describes quality, state or action that noun refers to
 - Adjective comes before nouns e.g. a new car
 - Size : big, tiny
 - Age : new, young, old
 - Shape : square, color, material etc.
 - Adjective can come after verbs e.g. your friend looks nice, dinner smells good
- Adjective can suggest different kind of object, different use of same object or can be irrelevant.



GUIDELINES FOR REFINING CLASSES

Adjective Classes:

- Does the object represented by the noun behave differently when the adjective is applied to it?

N.B. An adjective is a word that expresses an attribute of something or the word class that qualifies nouns

- If the use of the adjective signals that the behavior of the object is different, then make a new class
- For example, If *Adult Membership* and *Youth Membership* behave differently, then they should be classified as different classes



GUIDELINES FOR REFINING CLASSES

Attribute Classes:

- Tentative objects which are used only as values should be defined or restated as attributes and not as a class
- For example the demographics of Membership are not classes but attributes of the Membership class
- E.g. client status is attribute of client class.



GUIDELINES FOR REFINING CLASSES

Irrelevant Classes:

- Each class must have a purpose and every class should be clearly defined and necessary
- You must formulate a statement of purpose for candidate each class
- If you cannot come up with a statement of purpose, simply eliminate the candidate class

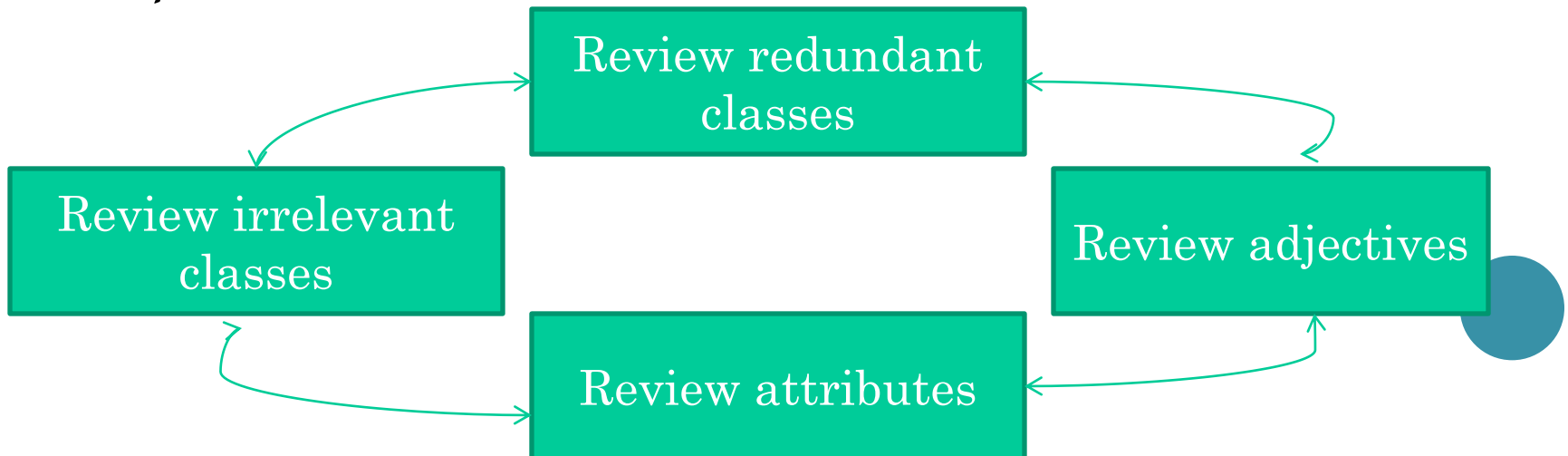


SUMMARY OF THE APPROACH

- This is an incremental approach- as you learn more about the problem, your design will have number of phases.
- There is no thing as “right” set of classes- but the process of identifying classes can be improved using incremental process.

i.e. some classes may be added or removed from model

Some classes will be missing while others can be eliminated or refined later.



ViaNet Bank ATM- Identifying classes-Noun Phrase Approach

Candidate classes

Account

Account balance

Amount

Approval process

ATM card

ATM machine

Bank

Bank client

Card

Cash

Check

Current

Current account

Client

Client's account

Currency

Envelope

Four digits

Fund

Invalid PIN

Message

Money

Password

PIN

PIN code

Record

Savings

Savings account

Step

System

Transaction

Transaction history



Con't

Read use case and description of system to identify noun phrases.

Remove irrelevant classes as they do not belong to problem statement.

Envelope, four digits and step



ViaNet Bank ATM- Irrelevant classes

Account

Account balance

Amount

Approval process

ATM card

ATM machine

Bank

Bank client

Card

Cash

Check

Current

Current account

Client

Client's account

Currency

Envelope

Four digits

Fund

Invalid PIN

Message

Money

Password

PIN

PIN code

Record

Savings

Savings account

Step

System

Transaction

Transaction history



Redundant Classes

Client, bank client – bank client

Account, client's account – Account

PIN, PIN code – PIN

Current, Current account – Current account

Savings, savings account - Savings account

Fund, Money – Fund

ATM card, Card – ATM card



After eliminating redundant classes

Account

Account balance

Amount

Approval process

ATM card

ATM machine

Bank

Bank client

Cash

Check

Current account

Currency

Fund

Invalid PIN

Message

Password

PIN

Record

Savings Account

System

Transaction

Transaction history



Reviewing classes containing adjectives

Does the object represented by the noun behave differently when the adjective is applied to it?

If yes, make a new class

Else class is irrelevant, we must eliminate it

We have no classes containing adjectives that we can eliminate.



Reviewing the Possible Attributes

Identifying the noun phrases that are attributes, not classes.

Amount : a value, not a class

Account Balance : An attribute of Account class.

Invalid PIN : It is only a value, not a class.

Password : An attribute of BankClient class.

Transaction history : An attribute of Transaction class

PIN : An attribute possibly of BankClient class.



Candidate classes-final

Account

Approval process

ATM card

ATM machine

Bank

Bank client

Cash

Check

Current account

Currency

Fund

Message

Record

Savings Account

System

Transaction



Reviewing the class purpose

- ATM Machine class: Provides an interface to the ViaNet bank.
- ATM Card class: Provides a client with key to an account.
- BankClient class: A client is an individual that has checking account, and possibly, a savings account.
- Bank class: Bank clients belong to the Bank. It is repository of accounts and processes the accounts transactions.
- Account class: An account class is a formal (or abstract) class, it defines the common behaviors that can be inherited by more specific classes such as Checking account and Savings Account.
- CheckingAccount class: It models a client's checking account and provides more specialized withdrawal service.
- SavingsAccount class: It models a client's savings account.
- Transaction class: Keeps track of transaction, time, date, type, amount, and balance.



PROBLEM

- The major problem is that it depends on the completeness and correctness of the document, which is almost no possible
- Large volume of text might lead to too many candidate classes
- Then also noun phrase approach can be very useful if combined with other approaches



COMMON CLASS PATTERNS APPROACH

- This approach is based on the knowledge-base of the common classes that have been proposed by various researchers



CANDIDATE CLASSES – CONCEPT CLASS

- A concept is a particular idea or understanding that we have of our world
- The concept class encompasses principles that are not tangible but use to organize business activities
- Privately held ideas are called conceptions- as when an understanding is shared by another , it becomes a concept
- To communicate with others we must share our conceptions and arrive at agreed concepts
- E.g. Performance is an example of concept class object



CANDIDATE CLASSES - EVENTS

- These are points in time that must be recorded and remembered
- Things happen, usually to something else, at a given date and time, or as a step in an ordered sequence
- Associated with things remembered are attributes such as who, when, what, where, how or why
- E.g. **order** which is an event that must be remembered



CANDIDATE CLASSES - ORGANIZATION

- The organizational units that people belong to: a collection of people, resources, facilities or groups
- E.g. accounting department might be considered as a potential class



CANDIDATE CLASSES - PEOPLE AND PERSON (ROLES AND ROLES PLAYED)

- Also known as person, role and roles played class
- The different roles users play in interacting with the application



CANDIDATE CLASSES - PEOPLE (CON'T)

- It can be divided into two types (Coad & Yourdon):
- Those representing users of the system, such as an operator, or a clerk;
- Those people who do not use the system but about whom information is kept
 - Some examples are Client, Employee, Teacher, Manager



CANDIDATE CLASSES - PLACES

- These are physical locations, such as buildings, stores, sites or offices that the system must keep information about




CANDIDATE CLASSES - TANGIBLE THINGS AND DEVICES

- Physical objects, or group of objects, that are tangible, and devices with which the application interacts
- E.g. Cars are tangible and pressure sensors are devices



The ViaNet Bank ATM System

Identifying classes by using common class patterns.

- Events Classes : Identify who what, when, where, how and why.
 - Account class: An account class is a formal (or abstract) class , it defines the common behaviors that can be inherited by more specific classes such as Checking account and Savings Account.
 - CheckingAccount class: It models a client's checking account and provides more specialized withdrawal service.
 - SavingsAccount class: It models a client's savings account.
 - Transaction class: Keeps track of transaction, time, date, type, amount, and balance.
 - Organization Class
 - Bank class: Bank clients belong to the Bank. It is repository of accounts and processes the accounts transactions.
 - People and Person Class
 - BankClient class: A client is an individual that has checking account, and possibly, a savings account.
 - Place Classes : Not applicable to this bank system.
 - Tangible and device classes
 - ATMMachine class: It allows access to all accounts held by a bank client.
- 

USE-CASE DRIVEN APPROACH

- The system described in terms of scenarios, the textual description or steps of each scenario can be used to determine what objects are needed for the scenario to occur.
- To identify objects of a system and their behaviors, the lowest level of executable use cases is further analyzed with a sequence and collaboration diagram pair
- By walking through the steps, you can determine what objects are necessary for the steps to take place



IMPLEMENTATION OF SCENARIOS

- The UML specification recommends that at least one scenario be prepared for each use case instance
- Each shows a different sequence of interaction between actors and the system
- This helps in understanding the behavior of the system
- Sequence diagrams are used to model scenarios in the systems
- Use cases and their descriptions offer a high level view of the system
- While sequence diagram enables you to model a more specific analysis and assists in design of the system by modeling the interactions between the objects of the system

THE VIA-NET BANK SYSTEM: DECOMPOSING A USE CASE SCENARIO WITH A SEQUENCE DIAGRAM : OBJECT BEHAVIOR ANALYSIS

- A sequence diagram represents the sequence and interactions of the given use case or scenario
- If used with object model or a class diagram, can capture most of the information about the system
- Developing sequence and collaboration diagrams requires to think about objects as they generate the events
- Sequence and collaboration diagrams represent the order in which events occur and how objects interact using message
- They provide a macro level analysis of the system
- Here sequence diagrams can be drawn for each scenario when a BankClient withdraws, deposit, or needs information on account

CONT...

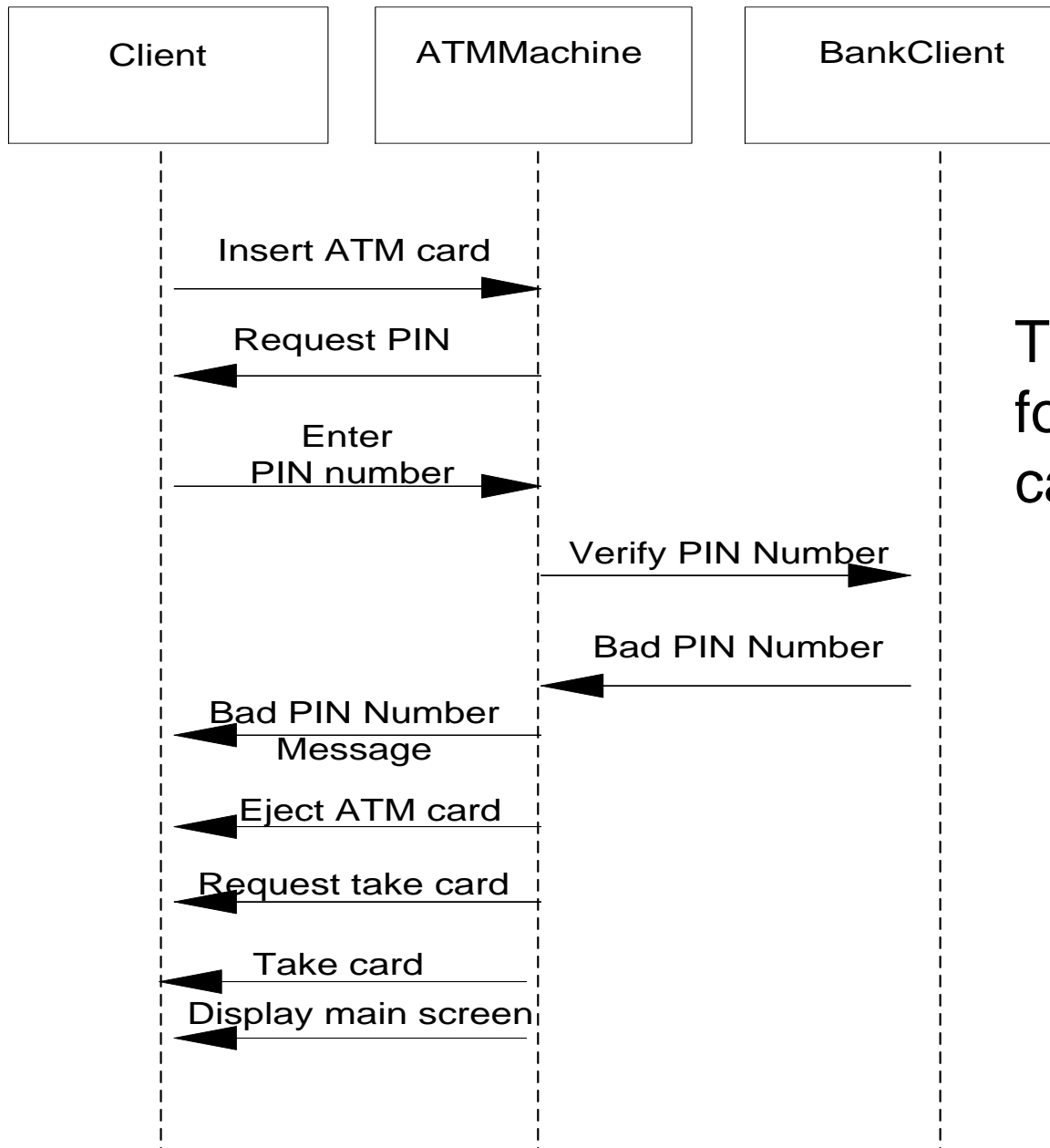
- In ch-6 we identified the use cases for the system:
 - Deposit Checking
 - Deposit Savings
 - Invalid PIN
 - Withdraw Checking
 - Withdraw More from Checking
 - Withdraw Savings
 - Withdraw Savings Denied
 - Checking Transaction History
 - Savings Transaction History



CONT...

- Create sequence/collaboration diagrams for following use cases:
 - Invalid PIN use case
 - Withdraw Checking use case
 - Withdraw Savings use case





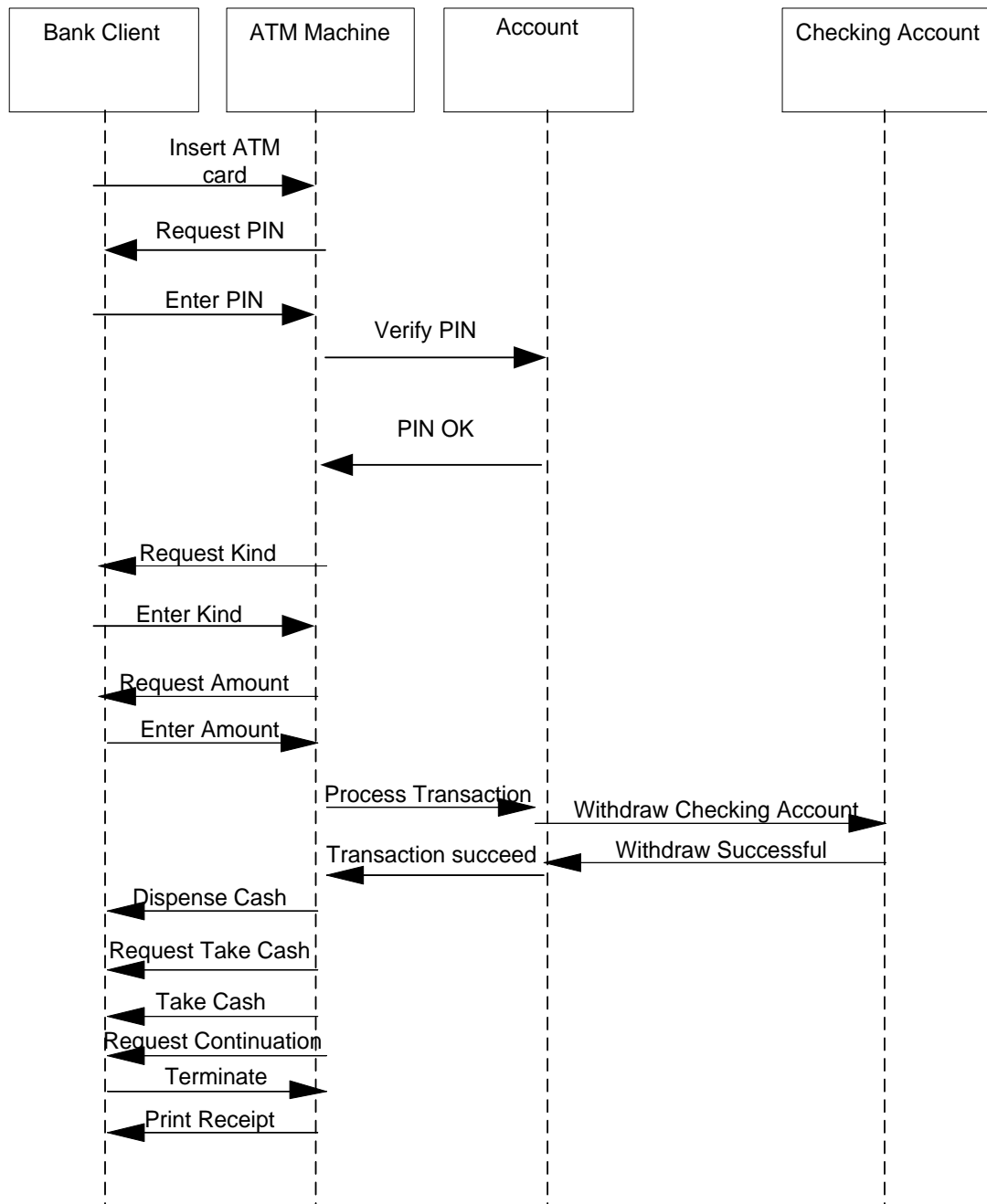
The sequence diagram
for the Invalid PIN use
case



CONT...

- The dotted lines are lifelines- an event arrow connects the objects i.e. a message is moving between them
- An event line can pass over an object without stopping at the object
- Each event must have a descriptive name
- In some cases several objects are active simultaneously or an object becomes active when it receives a message and becomes inactive as soon as it responds





Sequence diagram
for the withdraw
Checking use case



CONT..

- The collaboration diagrams are just another view of the sequence diagram and hence can be automatically created



CRC CARDS

- CRC stands for **Class, Responsibilities and Collaborators** developed by Cunningham, Wilkerson and Beck
- CRC can be used for identifying classes and their responsibilities and hence their methods and attributes



COLLABORATIONS

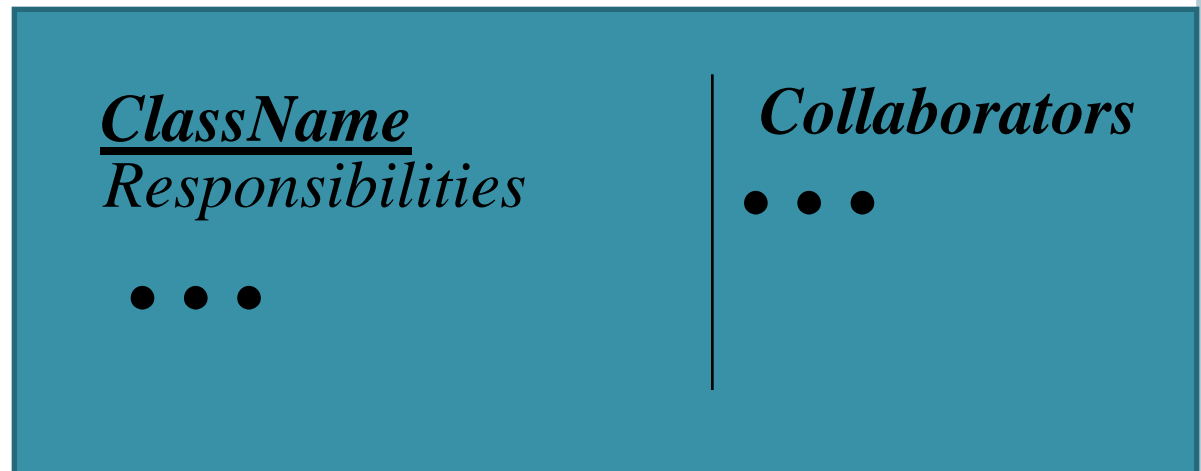
- An object can accomplish either a certain responsibility itself, or it may require the assistance of other objects
- If it requires an assistance of other objects, it must collaborate with those objects to fulfill its responsibility
- By identifying the objects responsibilities and collaborators you can identify its attributes and methods




CRC CARDS (CON'T)

- CRC cards are 4" x 6" index cards. All the information for an object is written on a card
- The class name should appear in the upper left corner, a bulleted list of responsibilities should appear under it and the list of collaborators should appear in the right third

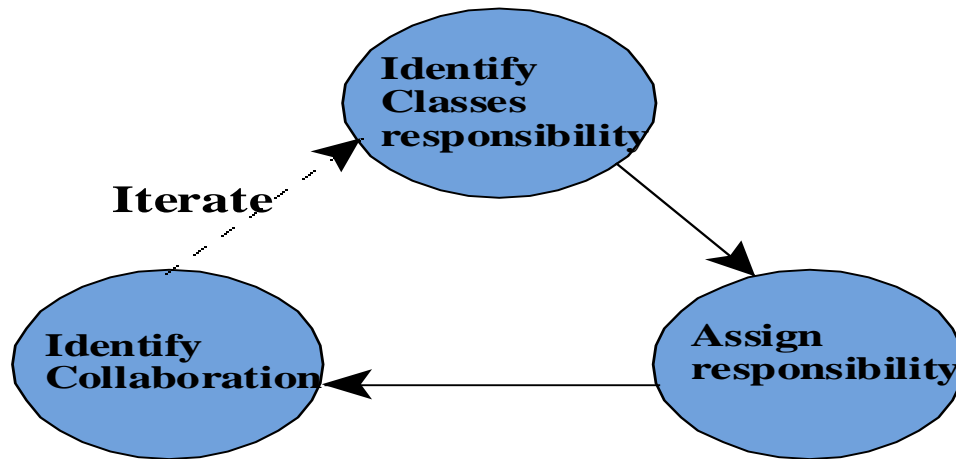
Fig: A Classes, Responsibilities, and Collaborators (CRC) index card



CRC PROCESS

- The process consist of three steps:
 - Identify classes' responsibilities(and identify classes)
 - Assign responsibilities
 - Identify collaborators
 - Classes are identified and grouped by common attributes
 - The class names then written onto classes, Responsibilities and Collaborators cards
 - The requirements are examined to find responsibilities
 - Then the responsibilities are distributed; they should be as general as possible
- 

PROCESS OF THE CRC TECHNIQUE



- The idea in locating collaborating classes is to identify how classes interact
- Classes that have a close collaboration are grouped together physically



CRC CARDS (CON'T)

- CRC starts with only one or two obvious cards
- If the situation calls for a responsibility not already covered by one of the objects:
 - Add, or
 - Create a new object to address that responsibility



THE VIA-NET BANKING ATM SYSTEM: IDENTIFYING CLASSES BY USING CRC

- The classes of the bank system are already identified so we will try to identify objects responsibilities such as attributes and methods in the system
- Account and transaction provide banking model.
- Account is responsible mainly for BankClient to keep track of account number, balance etc.
- These becomes attributes of account class.
- Also, account class provides certain services to BankClient like deposit, withdraw etc.



The ViaNet Bank ATM System

Identifying classes by using Classes, Responsibilities, and Collaborators

Classes, Responsibilities, and Collaborators for the Account Object.

Account <i>balance</i> <i>Number</i> ----- <i>deposit</i> <i>withdraw</i> <i>getBalance</i>	Checking Account (subclass) Savings Account (subclass) Transaction
--	--



GUIDELINES FOR NAMING CLASSES

- The class should describe a single object, so it should be the singular form of noun
- Use names that the users are comfortable with
- The name of a class should reflect its intrinsic nature
- By the convention, the class name must begin with an upper case letter
- For compound words, capitalize the first letter of each word - for example, LoanWindow



SUMMARY

- Finding classes is not easy
- The more practice you have, the better you get at identifying classes
- There is no such thing as the “right set of classes”
- Finding classes is an incremental and iterative process
- Unless you are starting with a lot of domain knowledge, you are probably missing more classes than you will eliminate
- Naming a class is also an important activity
- The class should describe a single object, so it should be a singular noun or an adjective and a noun

