

Information System Design

IT60105

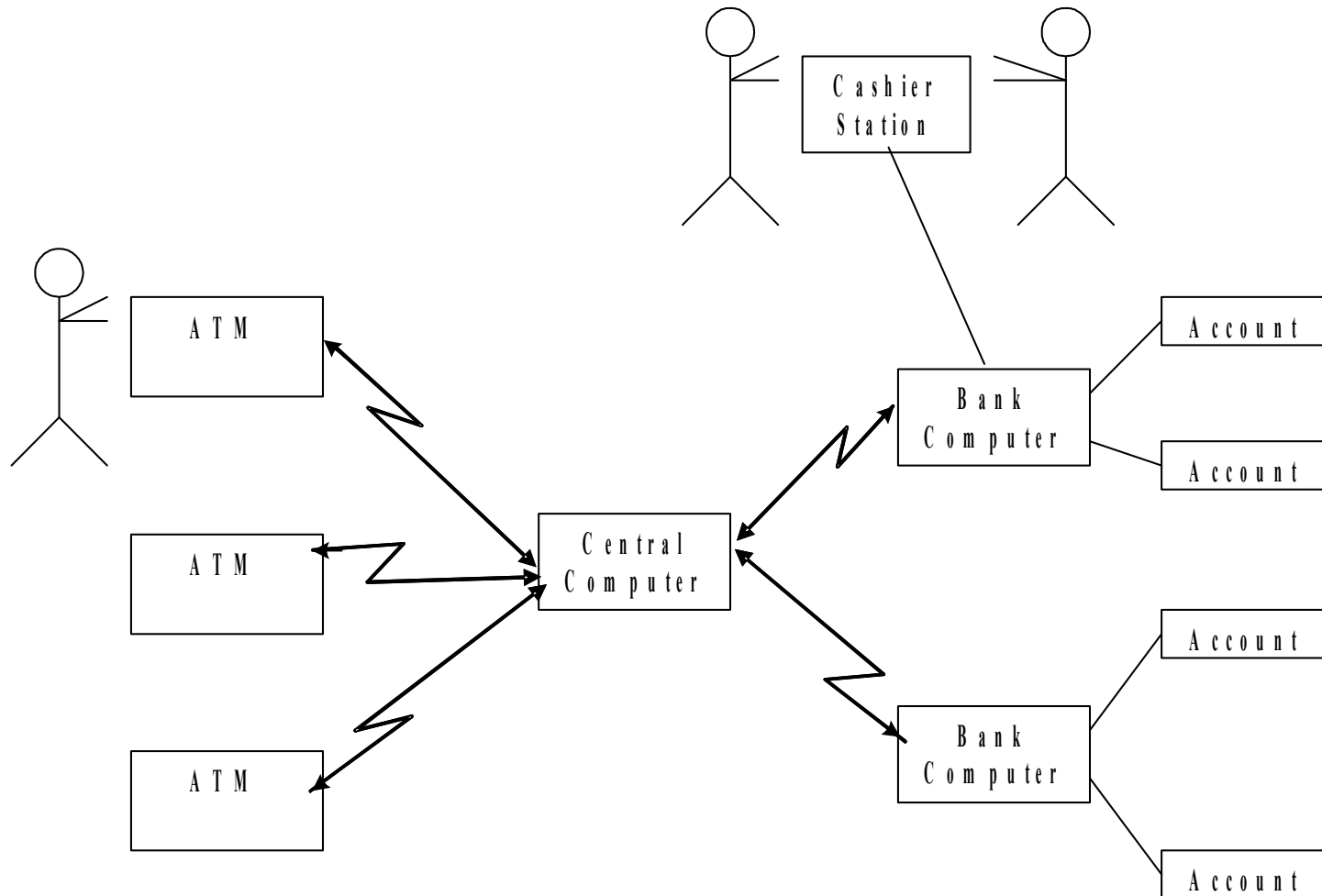
Lecture 12

Class Modeling

Lecture #12

- Domain class model
 - Finding classes
 - Keeping the right classes
 - Finding associations
 - Keeping the right associations
 - Finding attributes
 - Keeping the right attributes
 - Finding methods
- Grouping classes into packages

Case Study: ATM System

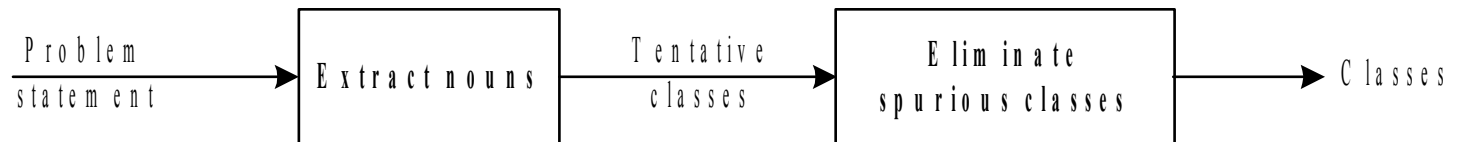


ATM System: Problem Statement

- Design the software to support a computerized banking network including both human cashiers and automatic teller machines (ATMs) to be shared by a consortium of banks. Each bank provides its own computer to maintain its own accounts and process transactions against them. Cashier stations are owned by individual banks and communicate directly with their own bank's computers. Human cashiers enter account and transaction data.
- Automatic teller machines communicate with a central computer that clears transactions with the appropriate banks. An automatic teller machine accepts a cash card, interacts with the user, communicates with the central system to carry out the transaction, dispenses cash, and prints receipts. The system requires appropriate recordkeeping and security provisions. The system must handle concurrent accesses to the same account correctly.
- The banks will provide their own software for their own computers; you are to design the software for the ATMs and the network. The cost of the shared system will be apportioned to the banks according to the number of customers with cash cards.

Finding Classes

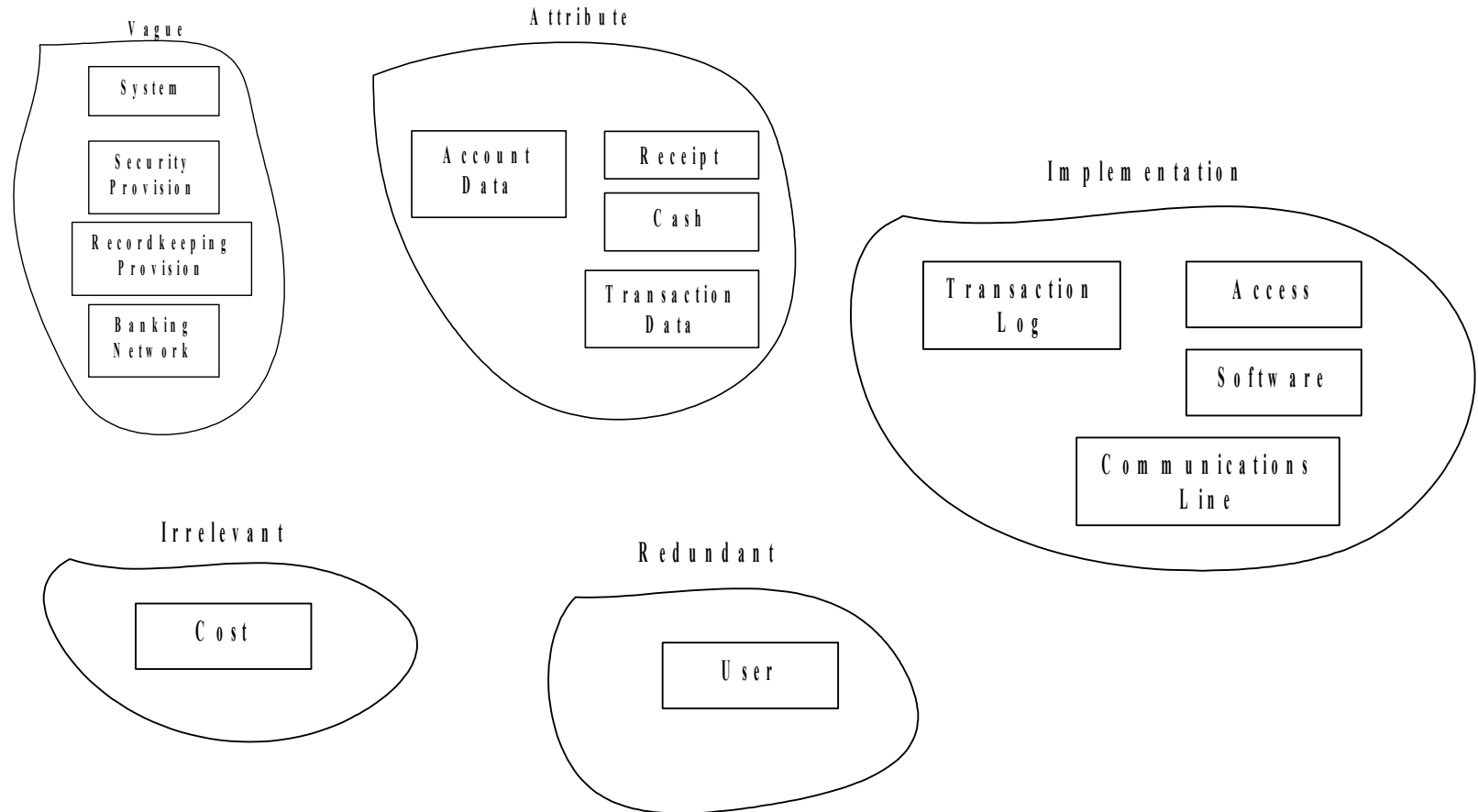
- To find the relevant classes for objects from the problem domain
 - Not all classes are explicit in the problem statement
 - Some are implicit in the application domain or general knowledge
- We can find many classes by considering nouns



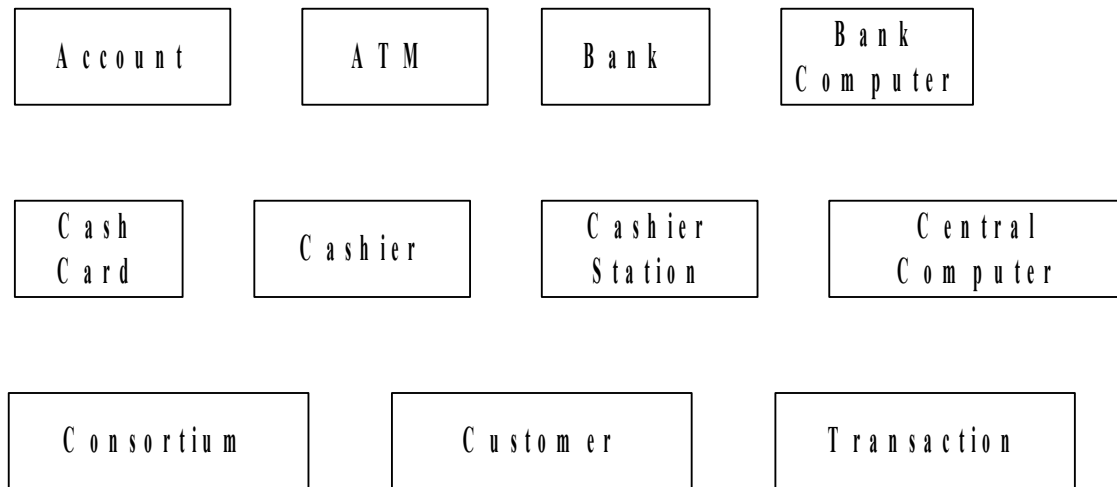
ATM System: Find Classes

- Design the **software** to support a computerized **banking network** including both human **cashiers** and **automatic teller machines (ATMs)** to be shared by a **consortium** of **banks**. Each bank provides its own **computer** to maintain its own **accounts** and process **transactions** against them. **Cashier stations** are owned by individual banks and communicate directly with their own bank's computers. Human cashiers enter **account data** and **transaction data**.
- Automatic teller machines communicate with a **central computer** that clears transactions with the appropriate banks. An automatic teller machine accepts a **cash card**, interacts with the **user**, communicates with the central system to carry out the transaction, dispenses **cash**, and prints **receipts**. The **system** requires appropriate **recordkeeping** and **security provisions**. The system must handle concurrent **accesses** to the same account correctly.
- The banks will provide their own software for their own computers; you are to design the software for the ATMs and the network. The **cost** of the shared system will be apportioned to the banks according to the number of **customers** with cash cards.

ATM System: Bad Classes



ATM System: Good Classes



Finding Associations

- Apply all or appropriate
 - Find Verb phrases
 - Implicit verb phrases
 - Knowledge of problem domain

ATM System: Find Associations

Verb phases

- Banking Network includes cashier stations and ATMs
- Consortium Shares ATMs
- Bank provides bank computer
- Bank computer maintains accounts
- Bank computer process transaction against account
- Bank owns cashier Station
- Cashier station communicates with bank computer
- Cashier enters transaction for account
- ATMs communicate with central computer about transaction
- Central computer clears transaction with bank
- ATM accepts cash card
- ATM interacts with user
- ATM dispense cash
- ATM prints receipts
- System handles concurrent access
- Bank provide software
- Cost apportioned to banks

ATM System: Find Associations

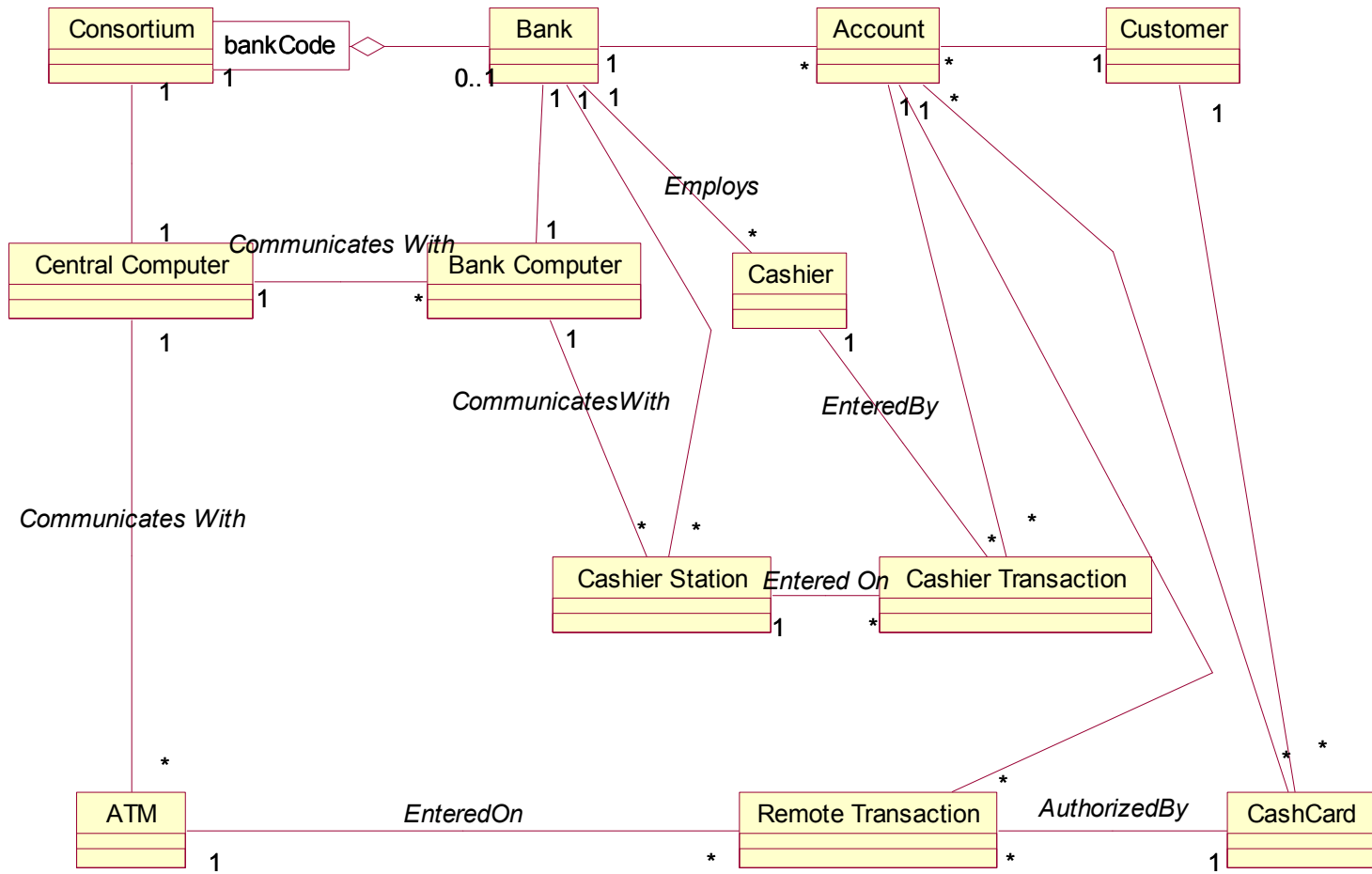
Implicit verb phases

- Consortium consists of banks
- Bank holds account
- Consortium owns central computer
- System provides recordkeeping
- System provides security
- Customers have cash cards

Knowledge of problem domain

- Cash card accesses accounts
- Bank employs cashiers

ATM System: Class Diagram



Finding Attributes

- Consider attributes directly relevant to the application
 - Get the most important attributes first
 - Add fine details later
 - Avoids attributes those are solely for implementation
 - Be sure to give each attribute a meaning full name

Keeping Right Attributes

- Eliminate unnecessary and incorrect attributes with the following criteria
 - Objects
 - Careful about right objects as attributes in the class
 - Qualifiers
 - Which actually matters
 - Names
 - A name is an attribute when its use does not depends on context

Find Methods

- Decide the operations require to manipulate the members within and other objects
- Use other models as well
 - Sequence diagrams
 - Statechart diagrams