INF 111 / CSE 121: Software Tools and Methods

Lecture Notes for Fall Quarter, 2007

Michele Rousseau

Set 17

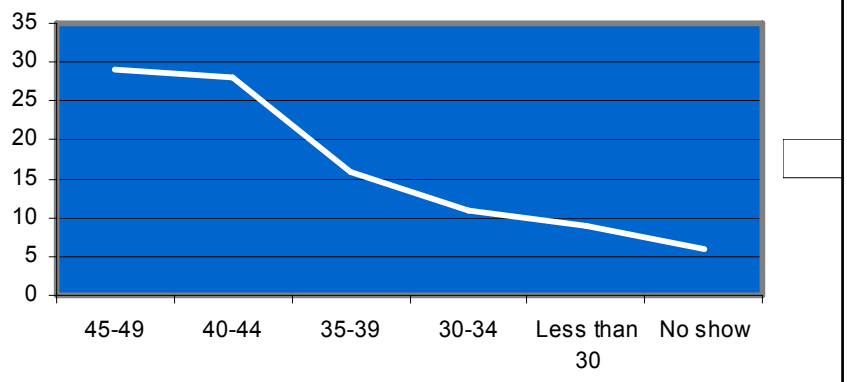
(Some notes adapted from Susan E. Sim & UML Distilled)

Announcements

- **Homework Due 11/21 @ 3p**
- **TA will be available for questions in class on Friday & in discussion on Monday**
- **Lab 6 will be posted on Thursday**
- **Quiz #3 scores have been posted**
  - Not distributed yet
- **UML Links:**
- http://dn.codegear.com/article/31863#use-case-diagram

Quiz # 3

- **Range**
  - High Score: 49
  - Low Score: 9
- **Median: 41**



Previously in INF 111…

- **UML**
  - Generalization
    - Inheritance
    - Polymorphism

Today's Lecture

- **UML**
  - Class Diagrams
  - Use Case Diagrams
  - Sequence Diagrams

Class Diagrams

- **Association**

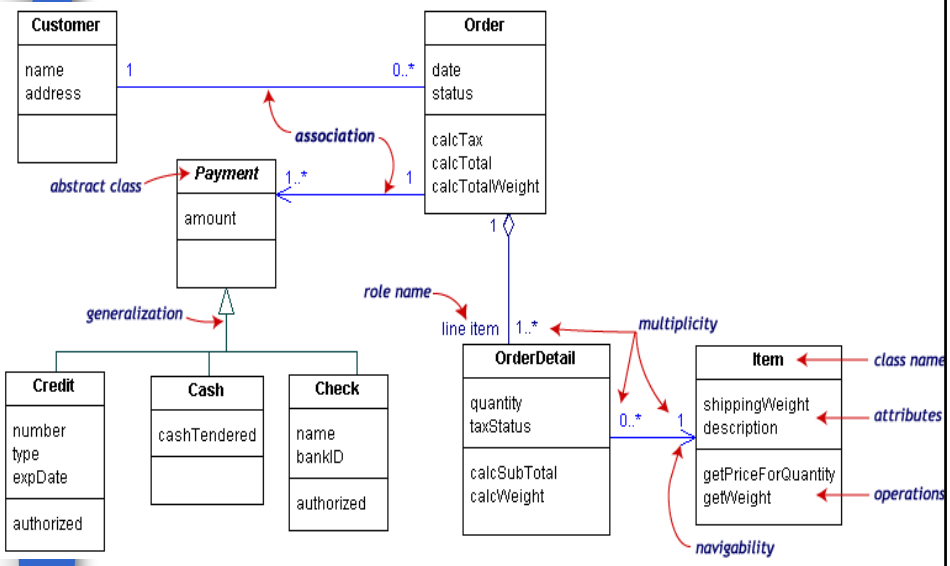
  There is an ***association*** between two classes if an instance of one class must know about the other in order to perform its work.
  - A relationship between instances of the two classes.
  - In a diagram, an association is represented by a link connecting two classes.
  - may have a *role name* to clarify the nature of the association
  - A *navigability arrow* on an association indicates which direction the association can be traversed or queried.
    - no navigability arrows are bi-directional.

- **Aggregation**
  - An association in which one class belongs to a collection.
  - In a diagram, an aggregation is represented with a diamond end pointing to the part containing the whole.
    - **"is a part of"**

- **Generalization**
  - An inheritance link indicating one class is a superclass of the other
    - "is a" or "is like a"
  - A generalization is represented with a triangle pointing to the superclass.

Class Diagrams provide a static model view of the system
Describes the Structure

Class Diagrams



Types of UML Diagrams

Structure (6 types)

- Class diagrams
- Object diagram
- Package diagram
- Composite structure diagram
- Component diagram
- Deployment Diagram

Behavior (4 types)

- Activity diagram
- Use Case diagram
- State machine diagram
- Interaction diagrams
  - **Sequence diagram**
  - **Communication diagram**
  - **Interaction overview diagram**
  - **Timing diagram**

If the appropriate diagram is not part of UML
use it anyways

Scenarios

- **Describes the system from an *external* viewpoint**
- **A Scenario is an example of what happens when someone interacts with the system**
- **EXAMPLE Scenario – Medical Clinic:**
  - "A patient calls the clinic to make an appointment for a yearly checkup. The receptionist finds the nearest empty time slot in the appointment book and schedules the appointment for that time slot. "

Use Cases

- **Again - describes the system from an *external* viewpoint**
  "provides an outsider's view"

Use Case Diagrams describe the dynamic behavior of the system

- **A Use Case is a summary of scenarios for a single task or goal.**
  - Represented as an oval
- **Actors**
  - who or what initiates the events involved in that task
  - roles that people or objects play
  - Represented as stick figures
- **Communication (or Communication Association)**
  - A Connection between the actor and the use case
  - Represented as a line

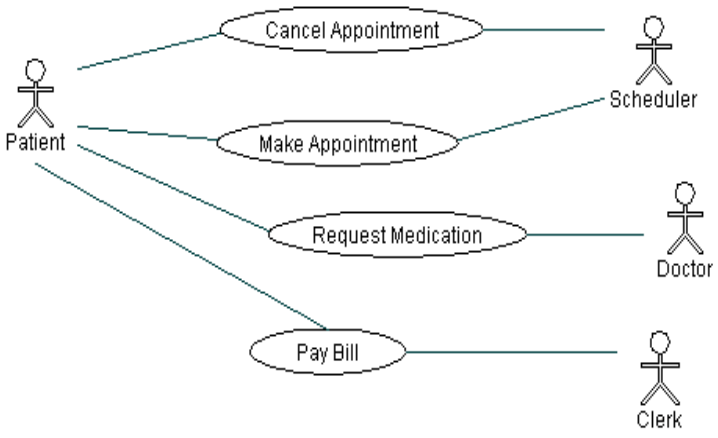
Use Case Diagrams

- **A collection of actors, use cases, and their associations**

Use case diagrams are helpful in three areas

- **Determining features (requirements)**
  - New use cases often generate new requirements.
    - Can happen during design and system analysis
- **Communicating with clients**
  - Simple notation makes them easy to understand
- **Generating test cases**
  - The collection of scenarios for a use case may suggest a suite of test cases for those scenarios

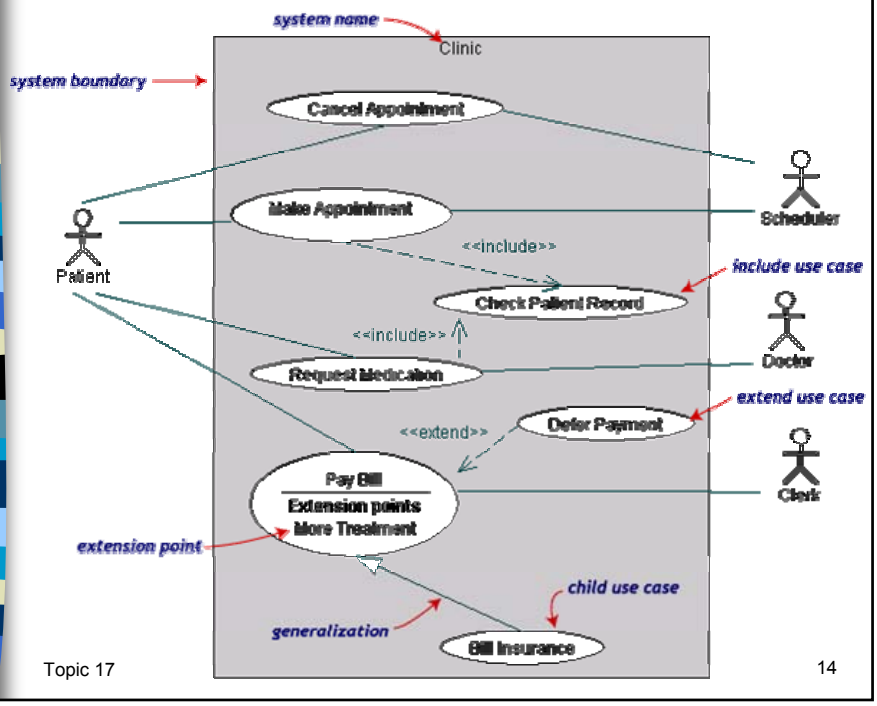
Use Case Diagram – Medical Clinic



Expanding Use Cases

- **A simple use case diagram can be expanded to display more information**
- **Use Cases can be developed iteratively and incrementally**

- **System boundaries**
  - separates the system from the external actors
  - Represented as a rectangle
- **Generalizations**
  - shows that one use case is simply a special kind of another
  - Represented with an open triangle
- **Includes**
  - factor use cases into additional ones
  - For example, a subtask that other use cases may use
  - Represented as a dotted line beginning at base use case ending with an arrows pointing to the include use case.
    - The dotted line is labeled <<include>>.
- **Extensions**
  - One use case is a variation of another
  - Represented as a dotted line, labeled <<extend>>, and with an arrow toward the base case.
  - The extension point determines when the extended case is appropriate, is written inside the base case.



Sequence Diagrams

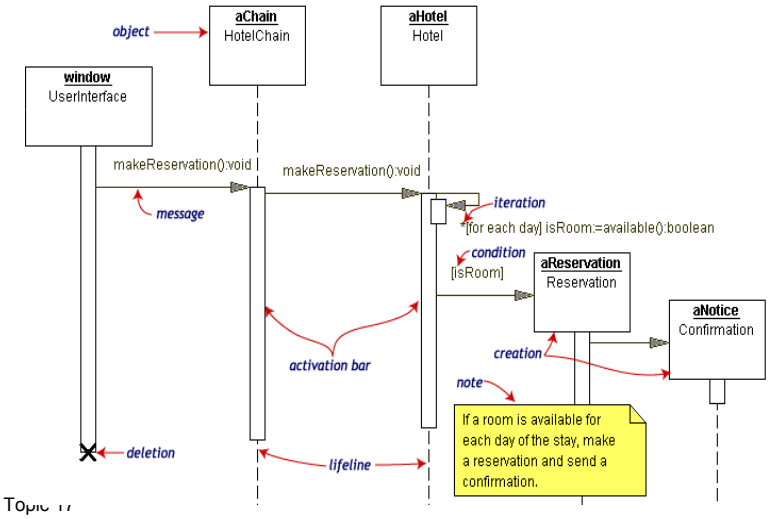
- **One type of Interaction Diagram**
- **Also describe the behavior of the system**
- **Details how operations are carried out**
  - What messages are sent when
- **Organized according to time**
- **Objects listed from left to right**
  - According to when they take part in the message sequence

Sequence Diagrams: Terms

- **A lifeline, representis the time that an object exists**
  - Represented as a vertical line.
- **An activation bar represents the duration of execution of the message**
  - Represented by a vertical rectangle
- **A message call is represented by an arrow between activation bars**
  - A simple message return is represented by a dashed arrow
- **A self call is when an object calls itself**
- **A note is used to clarify details**
  - Represented with a dog-eared rectangle

(Notes can be put into any kind of UML diagram)

Sequence Diagram Example: Hotel Reservation



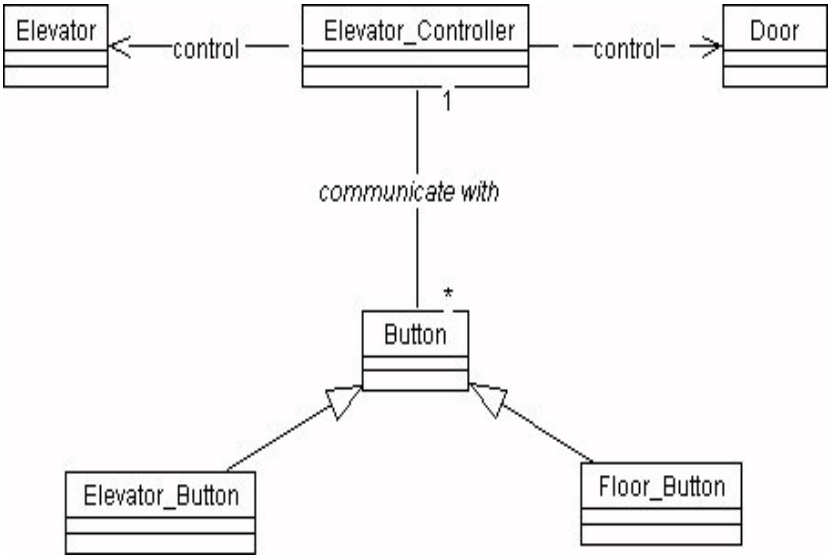
Putting them together

- **Class Diagrams**
- **Scenarios**
- **Use Cases**
- **Sequence Diagrams**

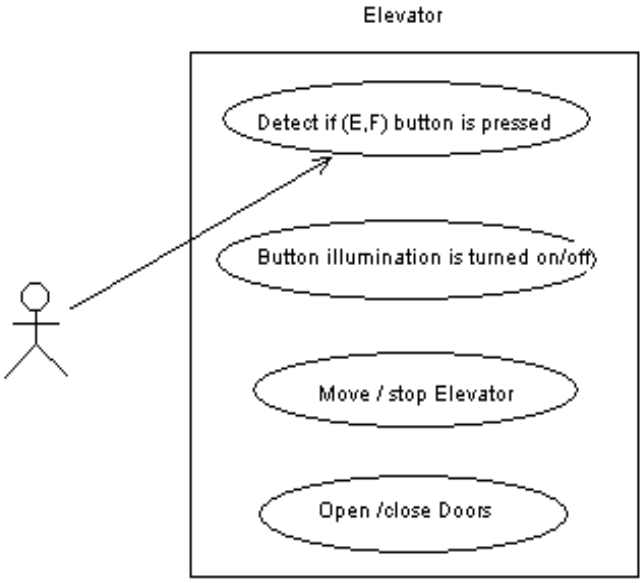
- **How do they all work together**

UML is iterative & Incremental

Elevator Example: Basic Class Diagram



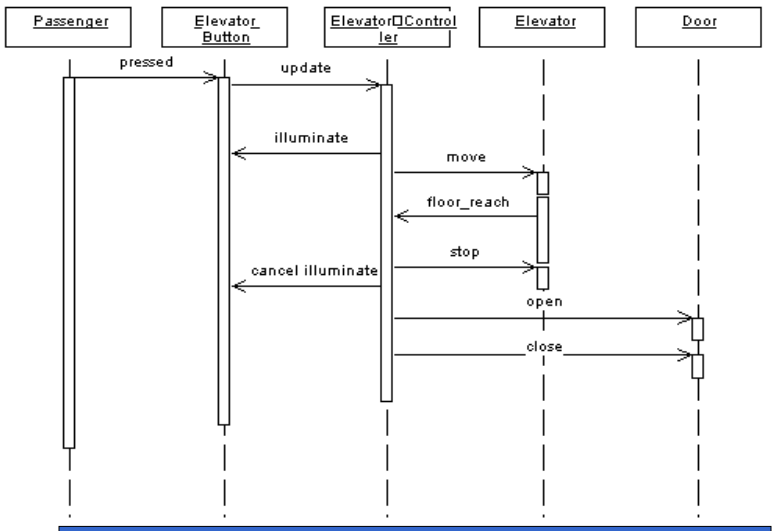
Elevator Example: Use Case



Elevator Example: Scenario

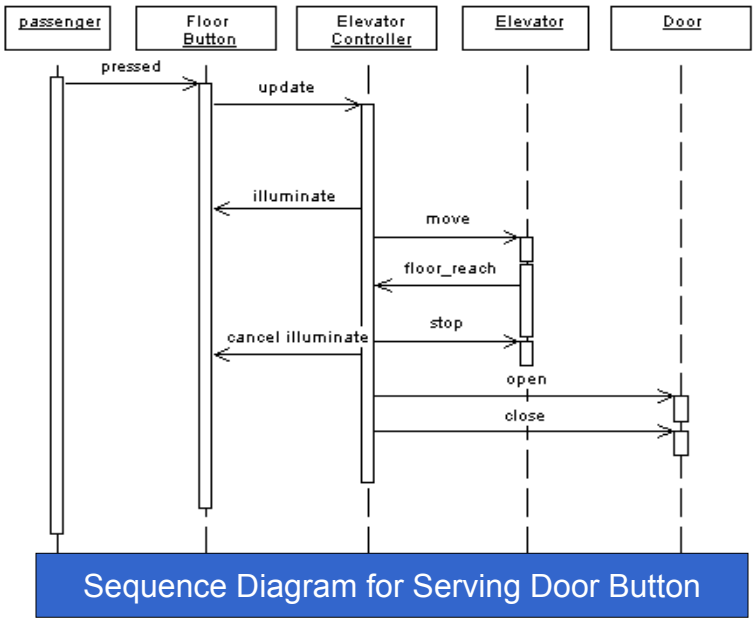
- Passenger pressed floor button
- Elevator system detects floor button pressed
- Elevator moves to the floor
- Elevator doors open
- Passenger gets in and presses elevator button
- Elevator doors closes
- Elevator moves to required floor
- Elevator doors open
- Passenger gets out
- Elevator doors closes

Elevator Example: Sequence Diagram



Sequence Diagram for Serving Elevator Button

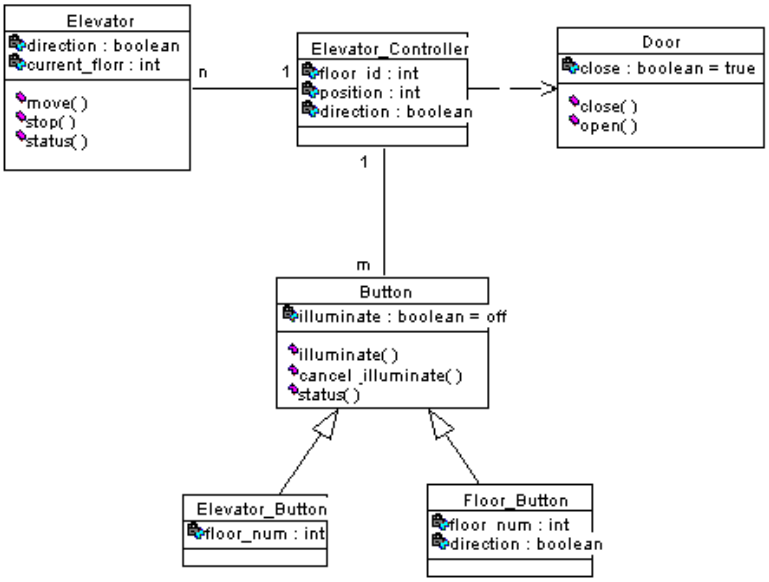
Elevator Example: Sequence Diagram



Sequence Diagram for Serving Door Button

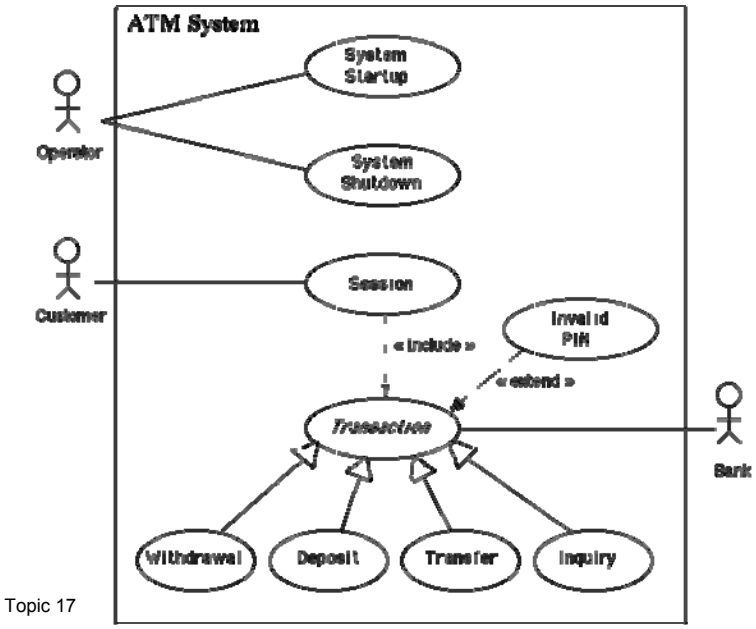
23

Elevator Example: Revising the Class Diagram



24

ATM Example



Session Use Case Description

A session is started when a customer inserts an ATM card into the card reader slot of the machine.

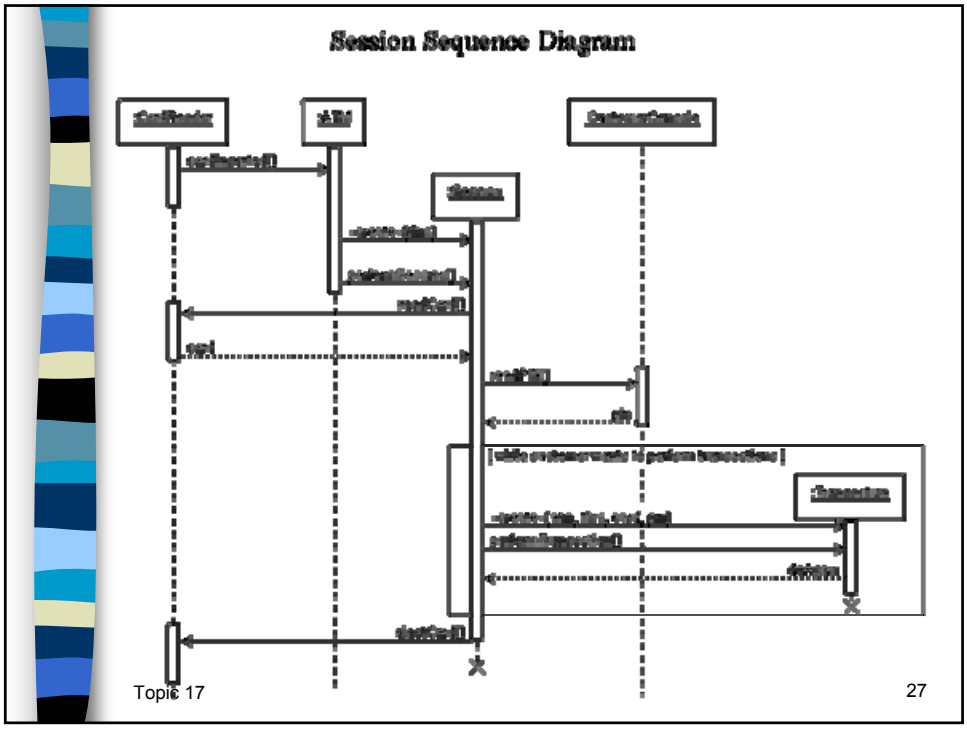
The ATM pulls the card into the machine and reads it. (If the reader cannot read the card due to improper insertion or a damaged stripe, the card is ejected, an error screen is displayed, and the session is aborted.)

The customer is asked to enter his/her PIN, and is then allowed to perform one or more transactions, choosing from a menu of possible types of transaction in each case.

After each transaction, the customer is asked whether he/she would like to perform another. When the customer is through performing transactions, the card is ejected from the machine and the session ends.

If a transaction is aborted due to too many invalid PIN entries, the session is also aborted, with the card being retained in the machine.

- **The customer may abort the session by pressing the Cancel key when entering a PIN or choosing a transaction type.**


Session Sequence Diagram
Topic 17
27