Use Case Diagrams

Tutorial

What is a use case?

- A requirements analysis concept
- A case of a use of the system/product
- Describes the system's actions from a the point of view of a user
- ***Tells a story***
  - A sequence of events involving
  - Interactions of a user with the system
- Specifies one aspect of the behavior of a system, ***without specifying the structure of the system***
- Is oriented toward satisfying a user's goal

How do we describe use cases?

- Textual or tabular descriptions
- User stories
- Diagrams

Use Case Descriptions

- **actors** - something with a behavior or role, e.g., a person, another system, organization.
- **scenario** - a specific sequence of actions and interactions between actors and the system, a.k.a. a *use case instance*
- **use case** - a collection of related success and failure scenarios, describing actors using the system to support a goal.

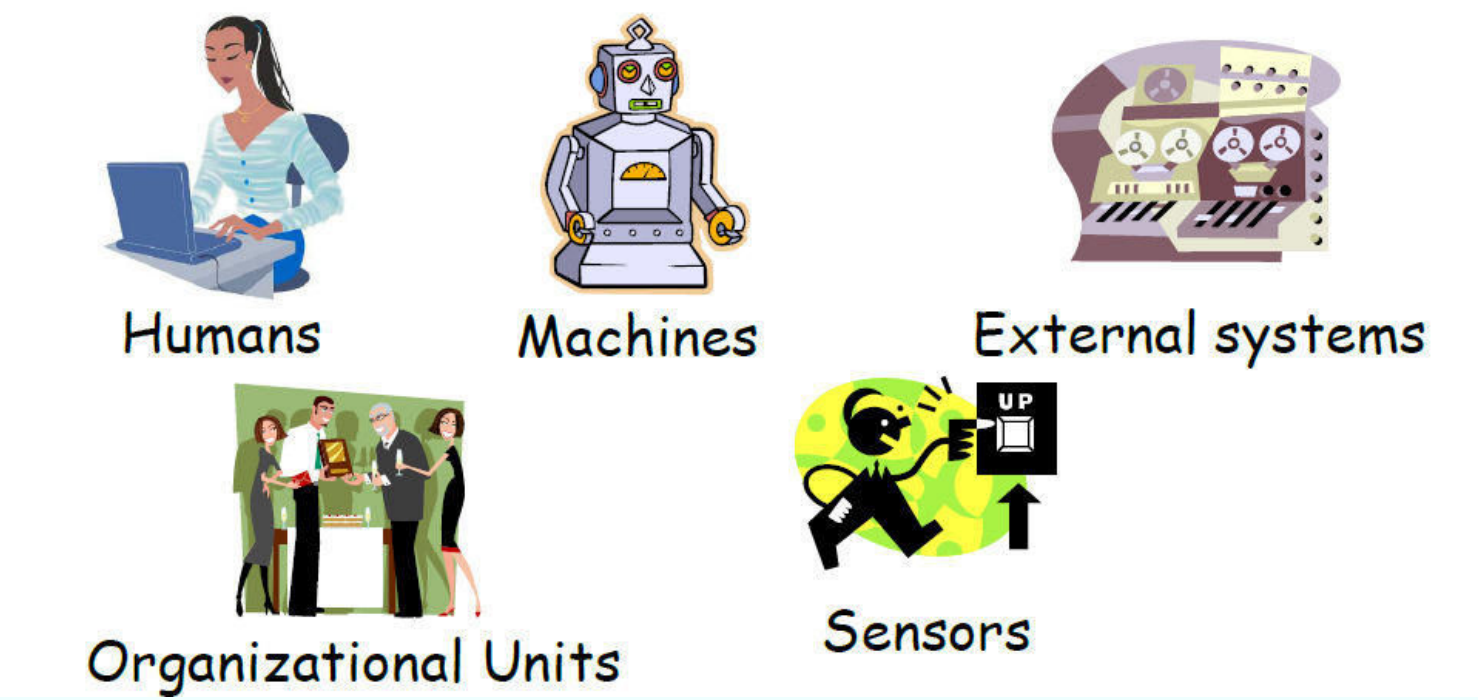
What is an Actor?

- Include all user roles that interact with the system
- Include system components only if they responsible for initiating/triggering a use case.
  - For example, a timer that triggers sending of an e-mail reminder
- ***primary*** - a user whose goals are fulfilled by the system
  - importance: define user goals
- ***supporting*** - provides a service (e.g., info) to the system
  - importance: clarify external interfaces and protocols
- ***offstage*** - has an interest in the behavior but is not primary or supporting, e.g., government
  - importance: ensure all interests (even subtle) are identified and satisfied

Finding Actors [1]

External objects that produce/consume data:

- Must serve as sources and destinations for data
- Must be external to the system



Finding Actors [2]

Ask the following questions:

- Who are the system's primary users?
- Who requires system support for daily tasks?
- Who are the system's secondary users?
- What hardware does the system handle?
- Which other (if any) systems interact with the system in question?
- Do any entities interacting with the system perform multiple roles as actors?
- Which other entities (human or otherwise) might have an interest in the system's output?

What is a user story?

- An abbreviated description of a use case
- Used in agile development

Answers 3 questions:

1. Who?
2. Does what?
3. And why?

As a <type of user>,
I want <some behavior from the system>
so that <some value is achieved>

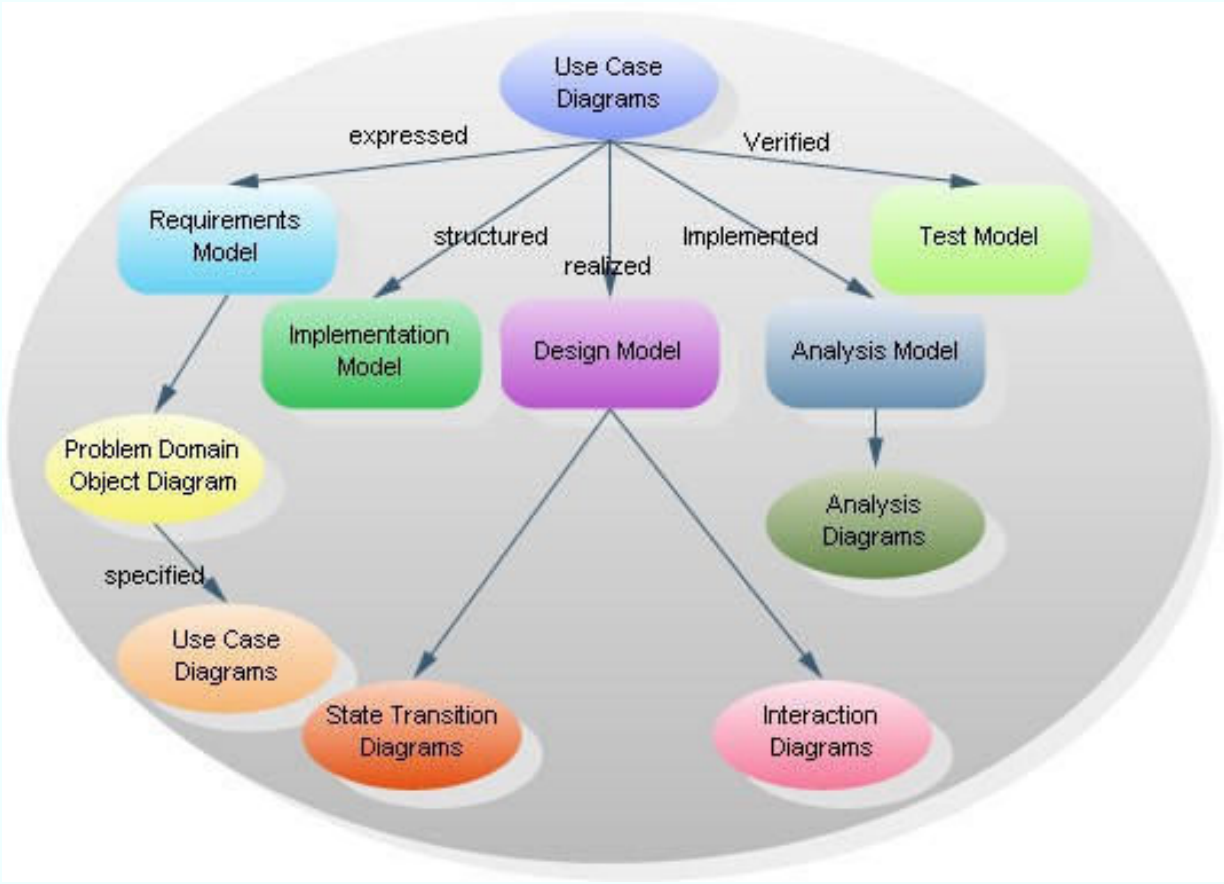
Use Case Diagrams

- A picture
  - describes how actors relate to use cases
  - and use cases relate to one another
- Diagrams are not essential
- They are helpful in giving an overview, but only secondary in importance to the textual description
- They do not capture the full information of the actual use cases
- In contrast, text ***is*** essential

Use Case Diagram Objective

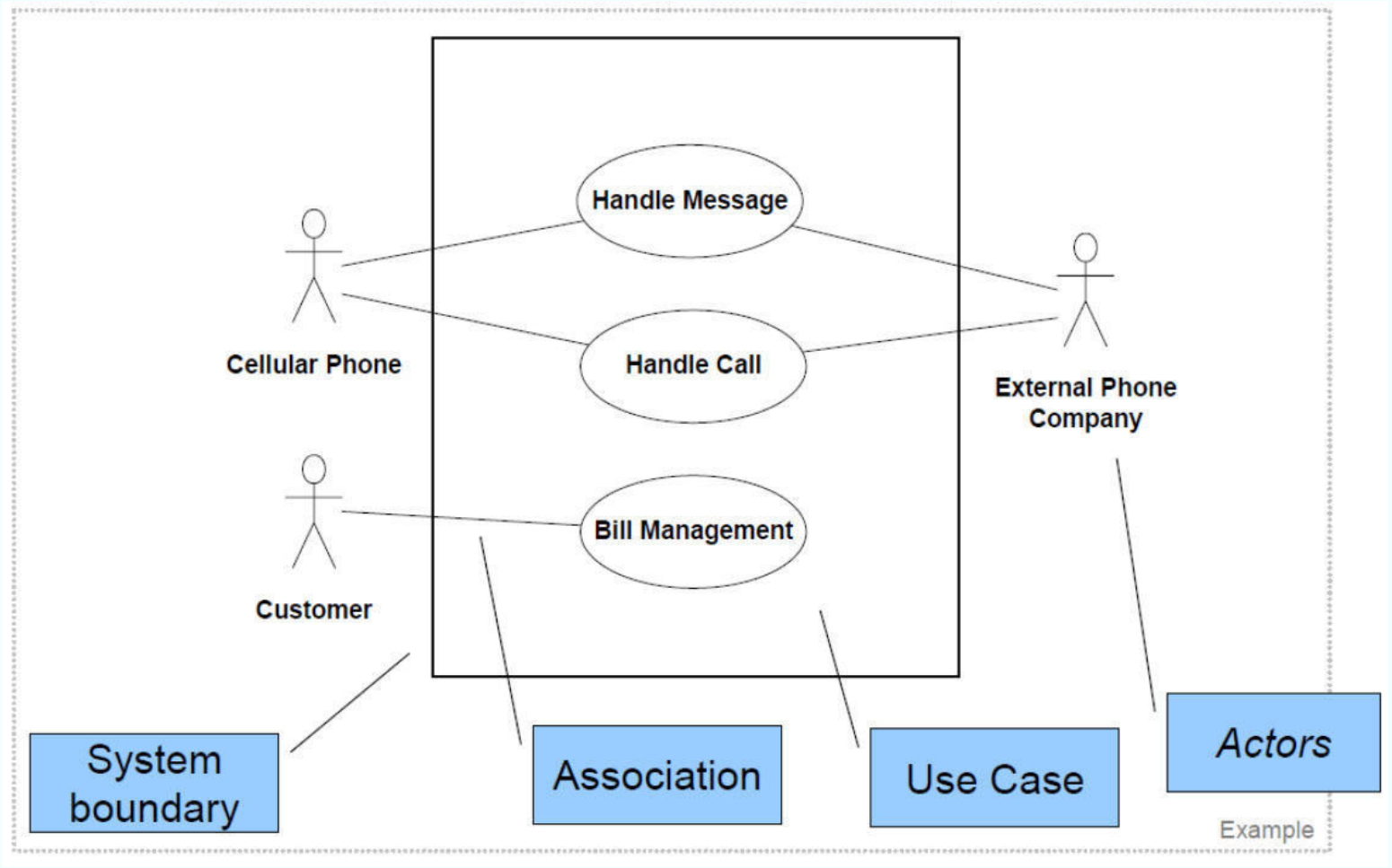
- Built in early stages of development
- Purpose
  - Specify the context of a system
  - Capture the requirements of a system
  - Validate a systems architecture
  - Drive implementation and generate test cases
  - Developed by analysts and domain experts

How do use case diagrams fit in?



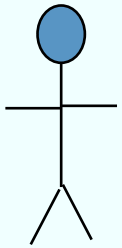
This applies also to use case descriptions.
Diagram reproduced from www.edrawsoft.com.

Example Use-Case Diagram



A standard form of use case diagram is defined in the Unified Modeling Language.

Elements of use case diagram: Actor



Name

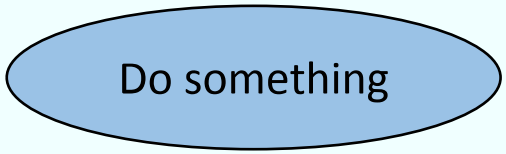
- *Actor* is someone interacting with use case (system function). Named by noun.

- Similar to the concept of user, but a user can play different *roles*; (example: a prof. can be instructor and researcher – plays 2 roles with two systems).

- Actor *triggers* use case.

- Actor has responsibility toward the system (inputs), and Actor have expectations from the system (outputs).

Elements of use case diagram: Use Case



- System function (process – automated or manual).
- Named by verb.
- Each Actor must be linked to a use case, while some use cases may not be linked to actors.

USER/ACTOR	USER GOAL = Use Case
Order clerk	Look up item availability Create new order Update order
Shipping clerk	Record order fulfillment Record back order
Merchandising manager	Create special promotion Produce catalog activity report

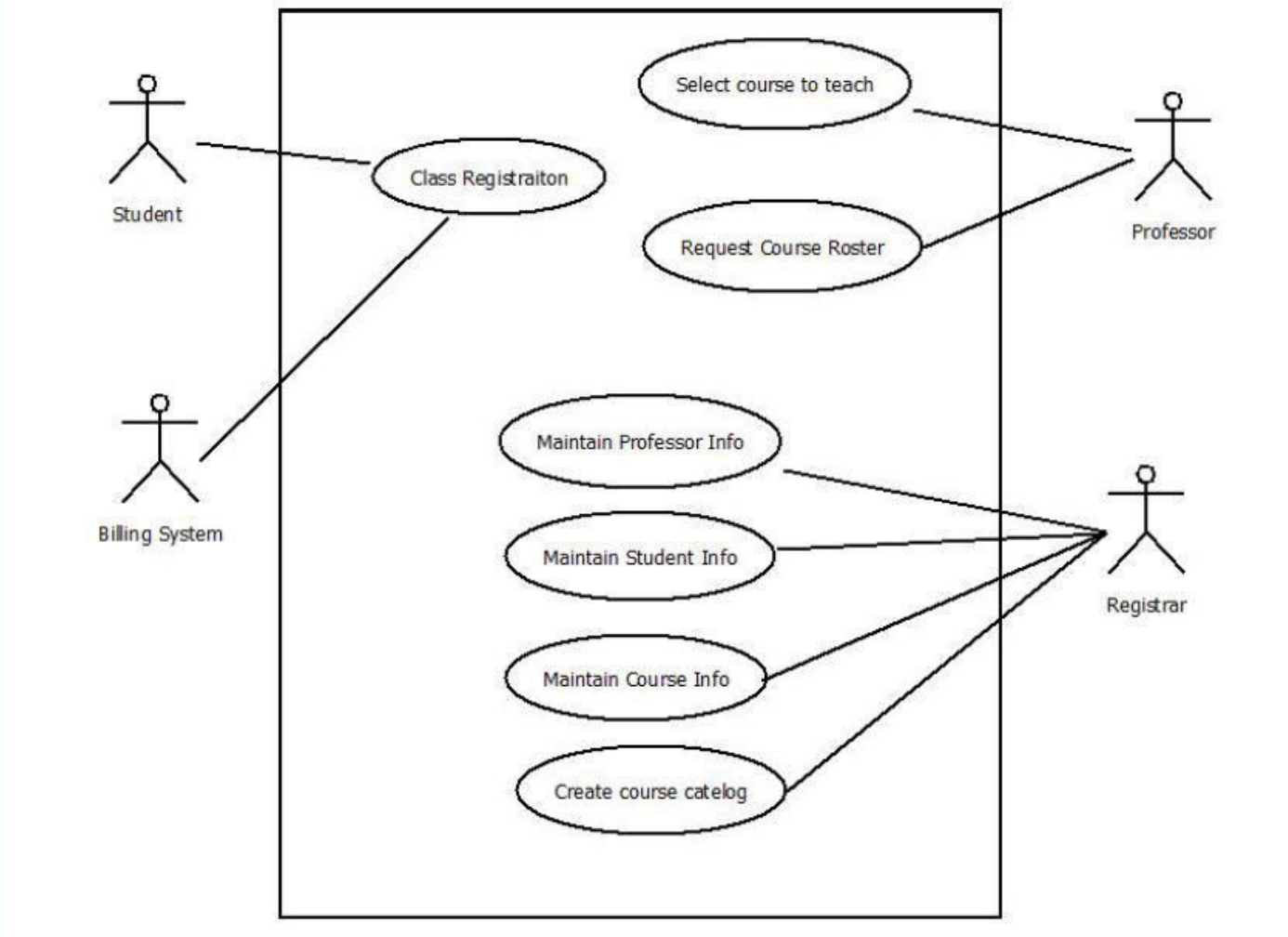
Example #2

Altered State University (ASU) Registration System

1. Professors indicate which courses they will teach on-line.
2. A course catalog can be printed
3. Allow students to select on-line four courses for upcoming semester.
4. No course may have more than 10 students or less than 3 students.
5. When the registration is completed, the system sends information to the billing system.
6. Professors can obtain course rosters on-line.
7. Students can add or drop classes on-line.

Example #2 cont.

Altered State University (ASU) Registration System



How to create use case diagram

1. List main system functions (use cases) in a column:
   - think of business events demanding system's response
   - users' goals/needs to be accomplished via the system
   - Create, Read, Update, Delete (CRUD) data tasks
   - Naming use cases – user's needs usually can be translated in data tasks
2. Draw ovals around the function labels
3. Draw system boundary
4. Draw actors and connect them with use cases (if more intuitive, this can be done as step 2)
5. Specify include and extend relationships between use cases (yes, at the end - not before, as this may pull you into process thinking, which does not apply in UC diagramming).

Use-Case Diagrams: Example [1]

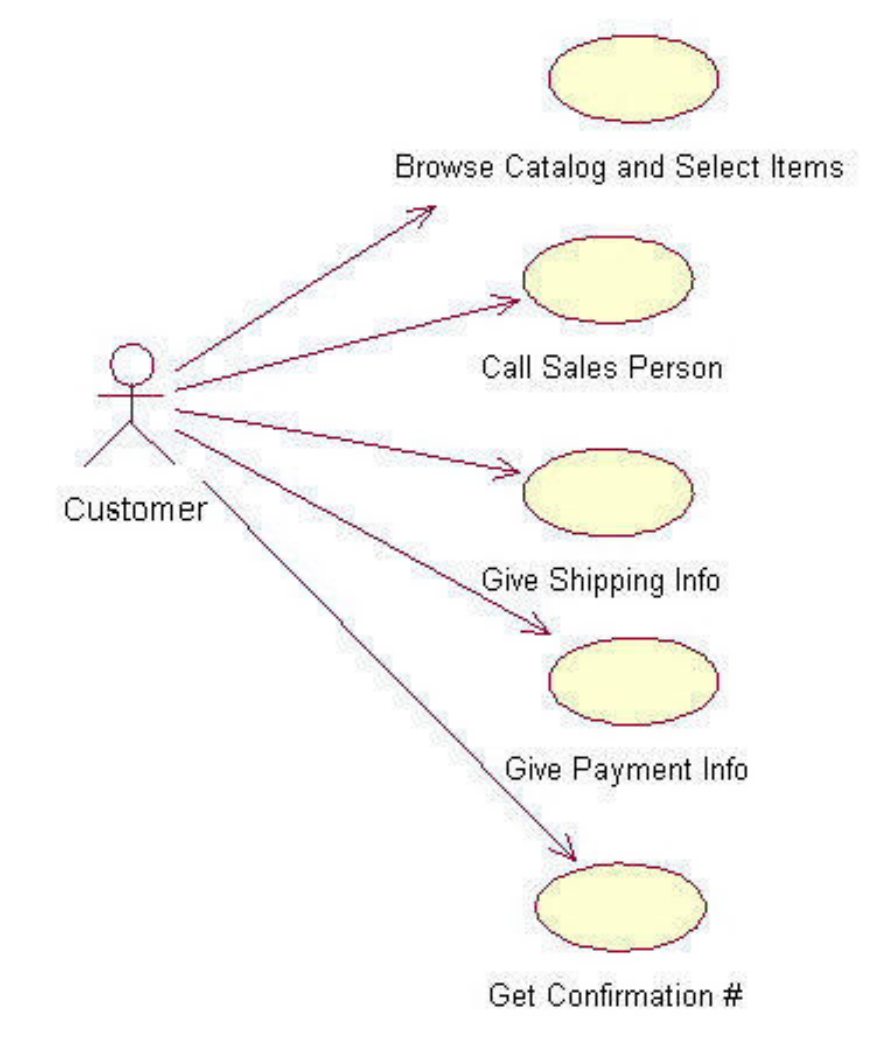
I. Begin with a Use Case!

A user placing an order with a sales company might follow these steps :

1. Browse catalog and select items.
2. Call sales representative.
3. Supply shipping information.
4. Supply payment information.
5. Receive conformation number from salesperson.

II. Then translate Use Case sequence into Diagram

Use-Case Diagrams: Example [2]



The salesperson could also be included in this use case diagram because the salesperson is also interacting with the ordering system.

EXERCISE 1: RESTUARANT USE CASE

Draw a use case diagram example for a restaurant system. This example has four actors, which are the waiter, the client, the chef, and the cashier.

The client's main tasks are "order food", "eat food", and "pay for food". They can also "order wine", "drink wine", and "pay for wine".

The waiter's tasks are to "order food", "serve food", and "pay for food" since they are the ones that facilitate the payment. If the client also orders wine, they also have the tasks of "order wine", "serve wine", and "pay for wine".

The chef interacts with the tasks "order food" and "prepare food". Finally, the cashier only accepts the payment, so their main task is "pay for food". In some cases, they also have the extended task of "pay for wine".