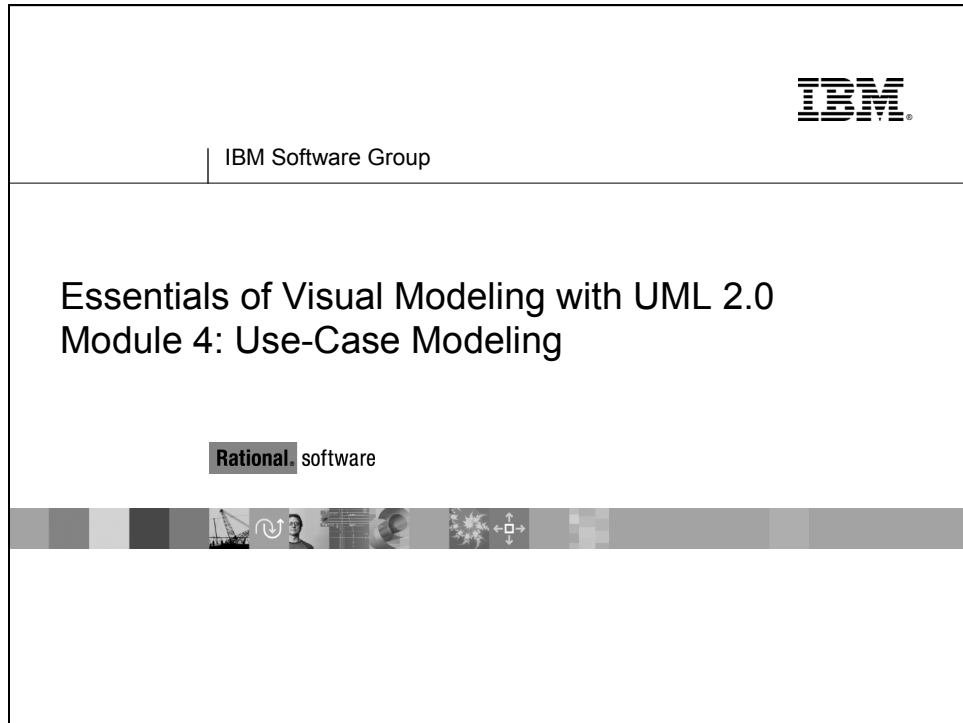


► ► ► **Module 4**

Use-Case Modeling



Topics

Major Concepts in Use-Case Modeling	4-7
Use Cases and Actors	4-11
What Is an Activity Diagram?	4-14

Objectives

Objectives

- ♦ Describe system behavior and show how to capture it in a model.
- ♦ Demonstrate how to read and interpret:
 - A use-case diagram
 - An activity diagram

Where Are We?

Where Are We?

- ☆ ♦ Concepts in use-case modeling
 - ♦ Use-case diagrams
 - ♦ Activity diagrams



3

IBM

What Is System Behavior?

What Is System Behavior?

- ♦ System behavior is how a system acts and reacts.
 - It comprises the actions and activities of a system.
- ♦ System behavior is captured in use cases.
 - Use cases describe the interactions between the system and (parts of) its environment.

4



No system exists in isolation. Every system interacts with people or automated systems for some purpose. These interactions result in some sort of predictable result. This predictable result is system behavior.

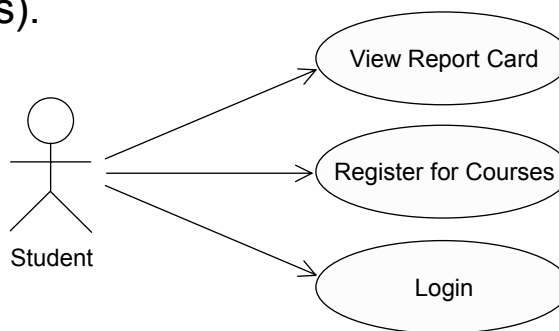
Use cases are the mechanism for capturing the desired behavior for the system that is under development, but do not specify how the behavior is to be implemented.

The UML specifies a model for communicating system behavior, the use-case model.

What Is a Use-Case Model?

What Is a Use-Case Model?

- ♦ A model that describes a system's functional requirements in terms of use cases.
- ♦ A model of the system's intended functions (use cases) and its environment (actors).



5

IBM

A **use-case model** describes a system's functional requirements in terms of use cases. The use-case model is a model of the system's intended functions and its environment and serves as a contract between the customer and the developers. Because it is a very powerful planning instrument, the use-case model is generally used in all phases of the development cycle.

The customer approves the use-case model. When you have that approval, you know the system is what the customer wants. You can also use the model to discuss the system with the customer during development.

Participants use it to better understand the system.

Designers use it as a basis for their work and to get a system overview.

Testers use it to plan testing activities (use case and integration testing) as early as possible.

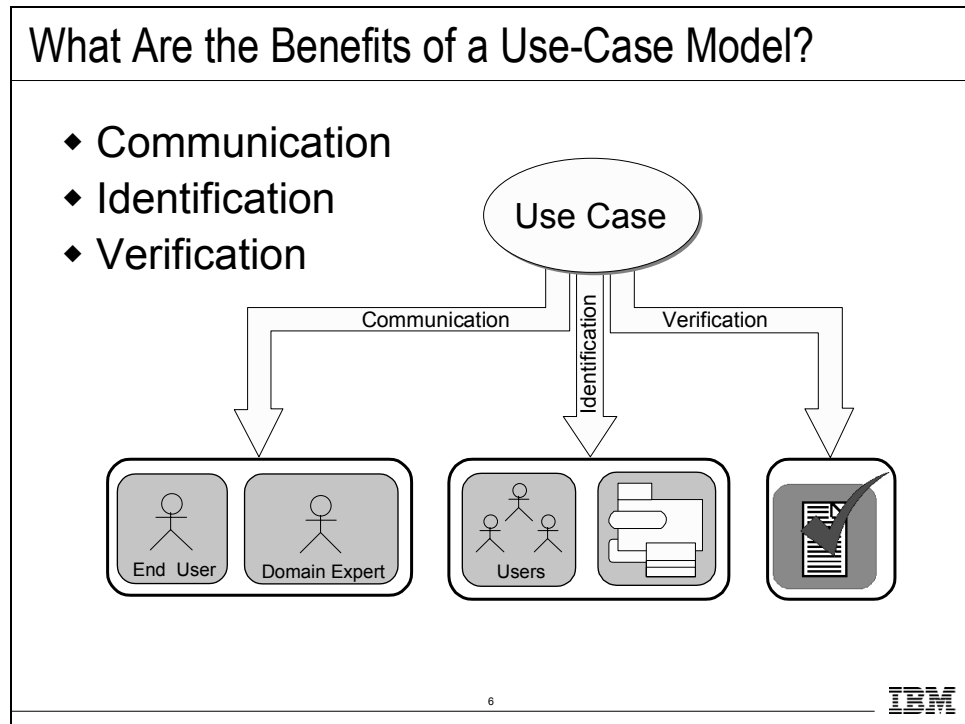
Those developing the next version of the system use it to understand how the existing version works.

Documentation writers review the use cases as a basis for writing the system's user guides.

The architect checks the use-case model to identify architecturally significant functionality.

The manager uses it to plan and follow up on the use-case modeling and also the subsequent design.

What Are the Benefits of a Use-Case Model?



There are many ways to model a system, each of which may serve a different purpose. However, the most important role of a use-case model is to communicate the system's behavior to the customer or end user. Consequently, the model must be easy to understand.

Communication with the end users and domain experts

- Provide buy-in at an early stage of system development
- Insure a mutual understanding of the requirements

Identification of system users and what the system should do

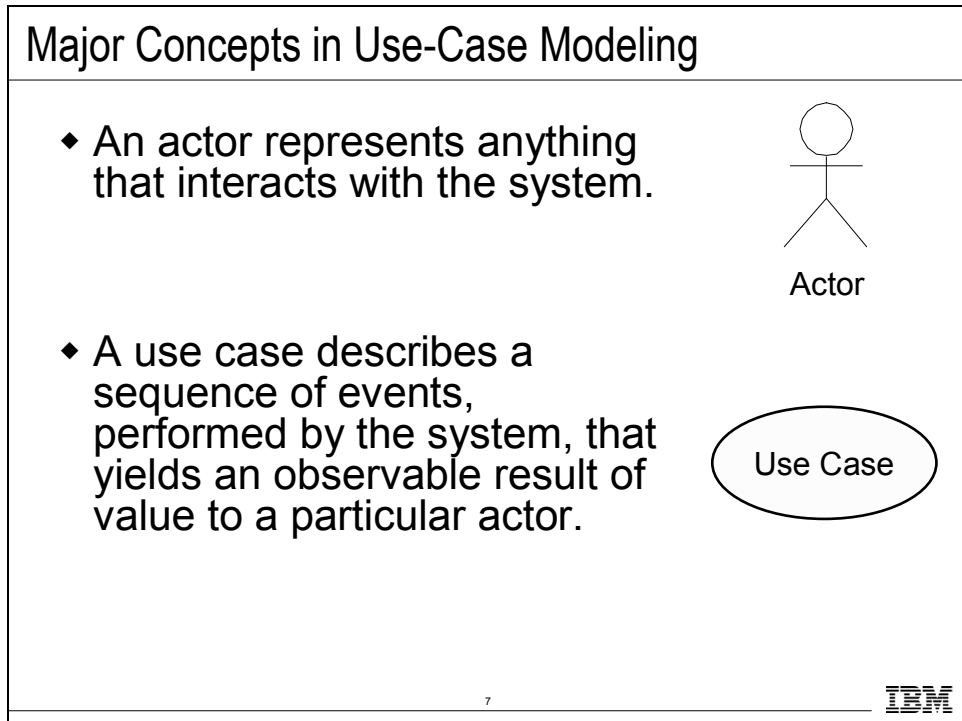
- The requirements for the system interfaces

Verification that all requirements have been captured

- The development team understands the requirements

Actors are the users and any other system that may interact with the system. Because they represent system users, actors help delimit the system and give a clearer picture of what it is supposed to do. Use cases are developed on the basis of the actor's needs, ensuring that the system turns out to be what the users expected.

Major Concepts in Use-Case Modeling



An **actor** represents a coherent set of roles that one plays when interacting with these use cases. Typically, an actor represents a role that a human, a hardware device, or even another system plays with a system.

A **use case** is a sequence of actions a system performs that yields an observable result of value to a particular actor. A use case describes *what* a system does, but it does not specify *how* it does it.

Where Are We?

Where Are We?

- ◆ Concepts in use-case modeling
- ☆◆ Use-case diagrams
- ◆ Activity diagrams



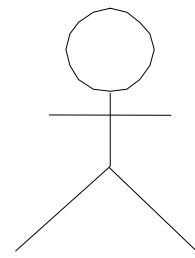
8

IBM

What Is an Actor?

What Is an Actor?

- ♦ Actors represent roles a user of the system can play.
- ♦ They can represent a human, a machine, or another system.
- ♦ They can actively interchange information with the system.
- ♦ They can be a giver of information.
- ♦ They can be a passive recipient of information.
- ♦ Actors are not part of the system.
 - Actors are EXTERNAL.



Actor

9



An **Actor** can be defined as:

Anything that exchanges data with the system and is external to the system.

- To fully understand the system's purpose you must know **who** the system is for. Different user types are represented as actors.
- An actor can be a user, external hardware, or another system. An actor may actively interchange information with the system, be a passive recipient of information, or can represent a human, a machine or another system.
- The difference between an actor and an individual system user is that an actor represents a particular class of user rather than an actual user. Several users can play the same role, which means they can be one and the same actor. In which case, each user constitutes an instance of the actor.
- In some situations, only one person plays the role modeled by an actor. For example, there may be only one individual playing the role of system administrator for a rather small system.
- The same user can also act as several actors. That is, the same person can take on different roles.

What Is a Use Case?

What Is a Use Case?

- ♦ Defines a set of use-case instances, where each instance is a sequence of actions a system performs that yields an observable result of value to a particular actor.
 - A use case models a dialogue between one or more actors and the system
 - A use case describes the actions the system takes to deliver something of value to the actor



10



A **Use-Case** can be defined as:

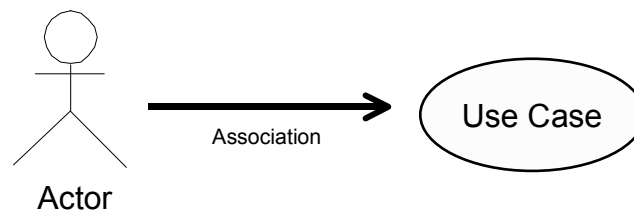
A sequence of **actions** a **system performs** that yields an **observable result of value** to a **particular actor**.

- **Actions** - An action is a computational or algorithmic procedure. It is invoked either when the actor provides a signal to the system or when the system receives a time event. An action may imply signal transmissions to either the invoking actor or other actors. An action is atomic. That is, it is performed either entirely or not at all.
- **System performs** - The system provides the use case. An actor communicates with a use-case instance of the system.
- **An observable result of value** - You can put a value on a successfully performed use case. A use case should ensure that an actor can perform a task that has an identifiable value. This is important to determine the correct level or granularity for a use case. Correct level refers to achieving use cases that are not too small. In certain circumstances, you can use a use case as a planning unit that includes individuals playing the role of an actor to the system.
- **A particular actor** - The actor is key to finding the correct use case because the actor helps to avoid use cases that are too large. As an example, consider a visual modeling tool. There are two actors to this application: a developer, someone who develops systems using the tool as support and a system administrator, someone who manages the tool. Each of these actors has its own demands on the system and requires its own set of use cases.

Use Cases and Actors

Use Cases and Actors

- ♦ A use case models a dialog between actors and the system.
- ♦ A use case is initiated by an actor to invoke a certain functionality in the system.



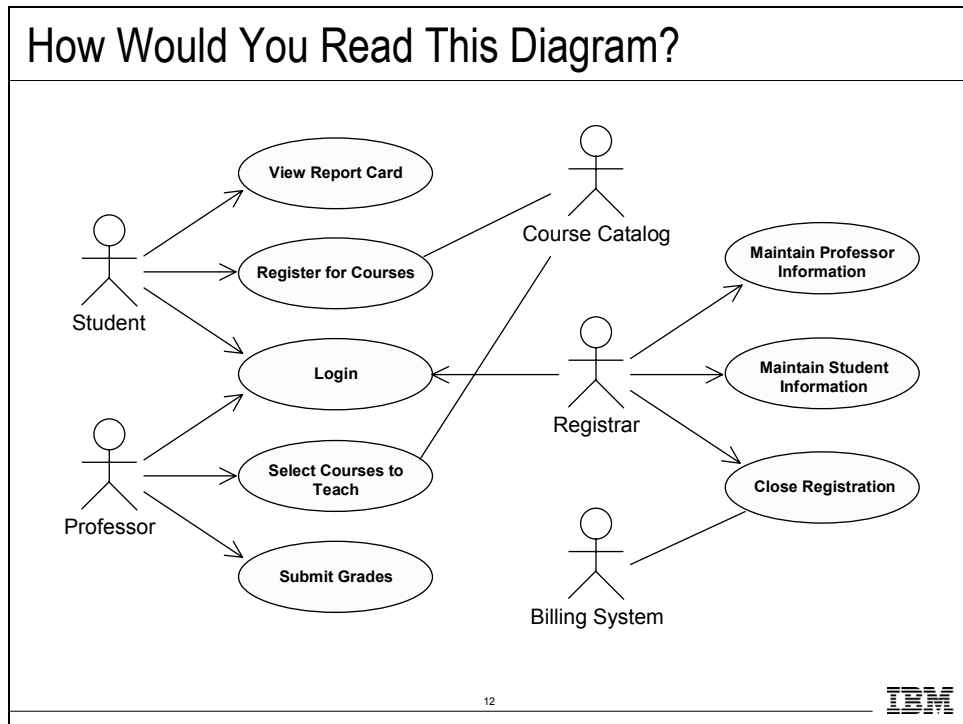
11



It is important to show how actors relate to the use case. Therefore, on finding a use case, establish the actors that interact with it. To do this, you must define an association that helps to clarify the communication between the actor and use case.

Actors may be connected to use cases only by an association. An association between an actor and a use case indicates that the actor and the use case instance of the system communicate with one another, each one able to send and receive messages. The arrow head is optional but it's commonly used to denote the initiator.

How Would You Read This Diagram?



Answer the following questions:

1. Which use cases can a student perform? A professor? The Course Catalog?
2. If Charlie is a student and professor, which use cases can he execute?
3. Describe the functionality of this system.
4. Describe the actor relationships for the Close Registration and Select Courses To Teach use cases.
5. Which use case needs to run first, Register for Courses or View Report Card?

Where Are We?

Where Are We?

- ♦ Concepts in use-case modeling
- ♦ Use-case diagrams
- ☆ ♦ Activity diagrams



13

IBM

What Is an Activity Diagram?

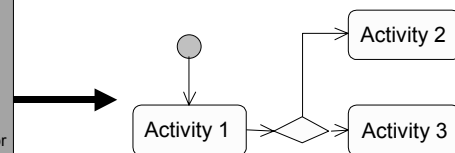
What Is an Activity Diagram?

- ♦ An activity diagram in the use-case model can be used to capture the activities and actions performed in a use case.
- ♦ It is essentially a flow chart, showing flow of control from one activity or action to another.

Flow of Events

This use case starts when the Registrar requests that the system close registration.

1. The system checks to see if registration is in progress. If it is, then a message is displayed to the Registrar and the use case terminates. The Close Registration processing cannot be performed if registration is in progress.
2. For each course offering, the system checks if a professor has signed up to teach the course offering and at least three students have registered. If so, the system commits the course offering for each schedule that contains it.



14

IBM

The workflow of a use case describes that which needs to be done by the system to provide the value the served actor is looking for.

It consists of a sequence of activities and actions that together produce something for the actor.

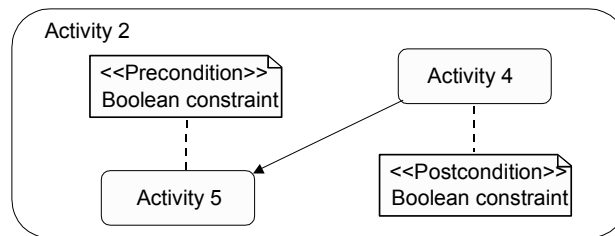
The workflow often consists of a basic flow and one or several alternative flows.

The structure of the workflow can be described graphically with the help of an activity diagram.

What Is an Activity?

What Is an Activity?

- ♦ A specification of behavior expressed as a flow of execution via sequencing of subordinate units.
 - Subordinate units include nested activities and ultimately individual actions.
- ♦ May contain boolean expression constraints when the activity is invoked or exited



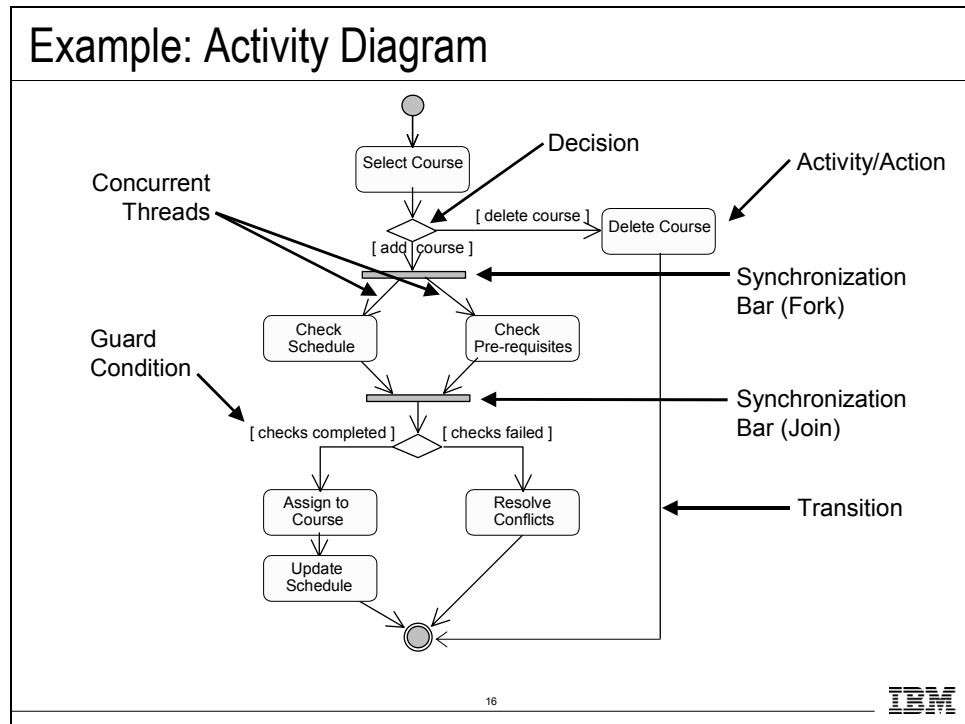
15



An activity is notated as an activity diagram. An activity definition is shown as a large rounded border containing a graph of node symbols and flow arrows representing the decomposition of the activity into its constituents. Activity preconditions and postconditions use the note notation with the keywords <<Precondition>> and <<Postcondition>> respectively.

An action is a primitive activity which is the smallest computation that can be expressed. An action is an activity that *does* something to the state of the system or extracts information from it. An action is drawn as a rectangle with rounded corners. Action preconditions and postconditions use the note notation with the keywords <<localPrecondition>> and <<localPostcondition>> respectively.

Example: Activity Diagram



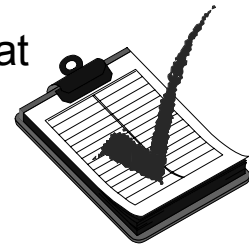
An activity diagram may include the following elements:

- **Activity/Action** represents the performance of a step within the workflow.
- **Transitions** show the activity/action that follows.
- **Decisions** evaluate conditions defined by guard conditions. These guard conditions determine which of the alternative transitions will be made and, thus, which activities are performed. You may also use the decision icon to show where the threads merge again. Decisions and guard conditions allow you to show alternative threads in the workflow of a use case.
- **Synchronization bars** show parallel sub-flows. They allow you to show concurrent threads in the workflow of a use case.

Review

Review

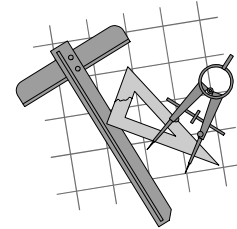
- ♦ What is system behavior?
- ♦ What is a use-case model? What are its benefits?
- ♦ What is an actor? A use case?
- ♦ What is an activity diagram?



Exercise

Exercise

- ♦ Given:
 - Use cases, actors and associations
- ♦ Draw:
 - A use-case diagram
- ♦ Given:
 - Action states and activity edges
- ♦ Draw:
 - An activity diagram



18

IBM

1. Draw a use-case diagram using the following elements:
 - There are four actors on the diagram: Prospective Buyer, E-Mail System, Loan System, and Credit Reporting System.
 - There are four use cases on the diagram: Find Realtor, Maintain Personal Planner, Search for a Home, and Apply for a Loan.
 - Document the following association;
 - Prospective Buyer to Find Realtor
 - Prospective Buyer to Maintain Personal Planner
 - Prospective Buyer to Search For A Home
 - Prospective Buyer to Apply For A Loan
 - Maintain Personal Planner to E-Mail System
 - Search for a Home to E-Mail System
 - Apply for a Loan to Loan System
 - Apply for a Loan to Credit Reporting System
- Study the diagram that you have drawn. What does it say? What doesn't it say? Describe the functionality that a Prospective Buyer expects from the system?
2. Draw an activity diagram that reflects the following four action states: Choose Profile, Find Buyer Profile, Log on, and Create New Profile. Starting with Choose Profile, go to Find Buyer Profile; then go from the Find Buyer Profile to Create New Profile, providing a profile does NOT exist. If a profile does exist, you can go to Log on.