Chapter 6: Structuring Systems Requirements: Use Case Description and Diagrams

Object-Oriented Systems Analysis and Design

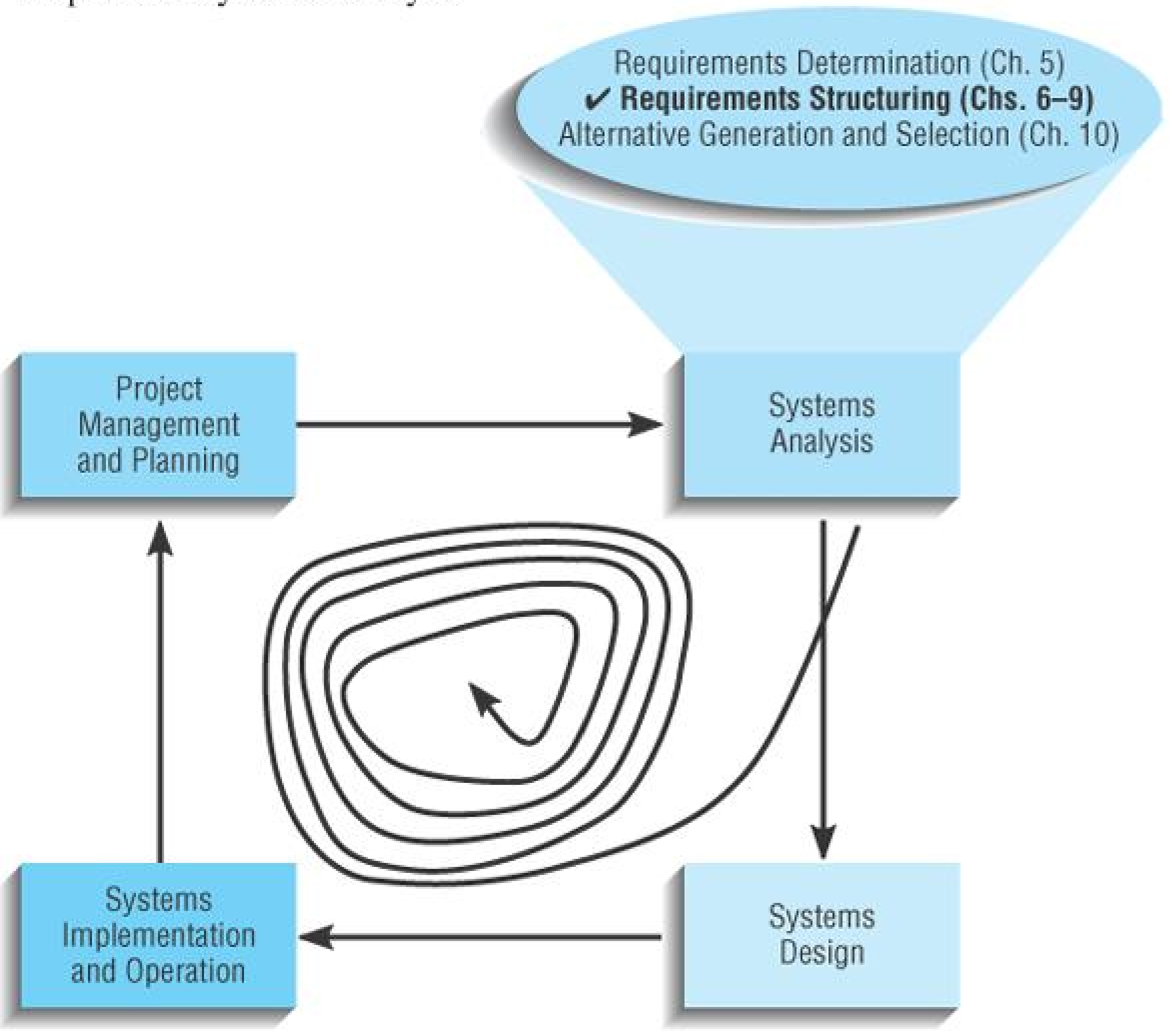
Joey F. George, Dinesh Batra,
Joseph S. Valacich, Jeffrey A. Hoffer

Chapter Objectives

After studying this chapter you should be able to:

- Understand how to structure requirements with use case diagrams.
- Explain the basics of use case construction using UML standards.
- Construct use case diagrams.
- Write text-based use cases.
- Construct Activity Diagrams

Figure 6.1 The Systems Development Cycle, Showing the Different Aspects of Systems Analysis


Requirements Determination (Ch. 5)
✔ Requirements Structuring (Chs. 6–9)
Alternative Generation and Selection (Ch. 10)
Project Management and Planning
Systems Analysis
Systems Implementation and Operation
Systems Design

What Is Requirements Structuring?

- The process of analyzing, organizing, and modeling the requirements obtained via interviews, questionnaires, observation, and document analysis

- Relevant UML models include use cases, class diagrams, interaction diagrams, and

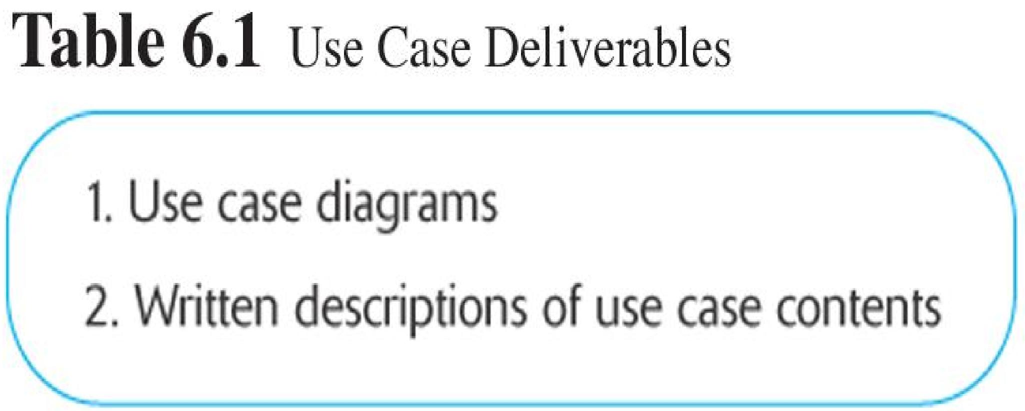
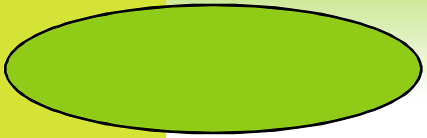


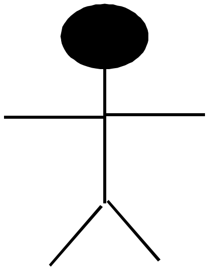
Table 6.1 Use Case Deliverables

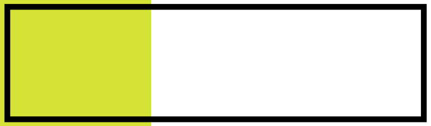
1. Use case diagrams
2. Written descriptions of use case contents

UML Use Case Diagram Symbols

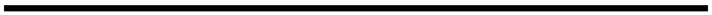


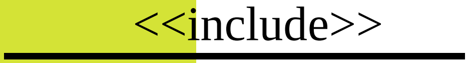
- Use Case
- Actor



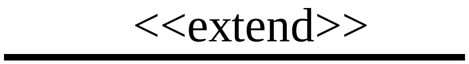


- Boundary
- Connection





- Include relationship
- Extend relationship



What Is a Use Case?

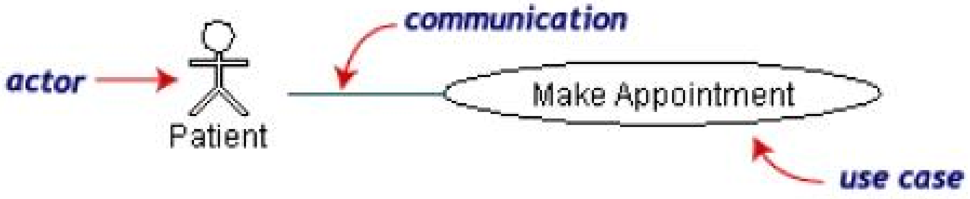
- A depiction of a system's behavior or functionality under various conditions as the system responds to requests from users functioning for a specific business purpose.

- **Use case diagrams** describe what a system does from the standpoint of an external observer. The emphasis is on *what* a system does rather than *how.*

- Use case diagrams are closely connected to scenarios. A **scenario** is an example of what happens when someone interacts with the system. Here is a scenario for a medical clinic:

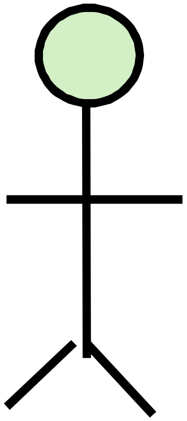
  "A patient calls the clinic to make an appointment for a yearly checkup. The receptionist finds the nearest empty time slot in the appointment book and schedules the appointment for that time slot. "

- A **use case** is a summary of scenarios for a single task or goal. An **actor** is who or what initiates the events involved in that task. Actors are simply roles that people or objects play. The picture is a **Make Appointment** use case for the medical clinic. The actor is a **Patient**. The connection between actor and use case is a **communication association** (or **communication** for short). *[From Borland's A Hands-on Introduction for developers]*



What Is an Actor?

- An external entity that interacts with the system
- It can be a:
  - Human
  - Peripheral device (hardware)
  - External system or subsystem
  - Time or time-based event
- Represented by stick figure
- An actor is a role, not a specific user; one user may play many roles, and an actor may represent many users.



What Is a Boundary?

- The dividing line between the system and its environment
- Use cases are within the boundary.
- Actors are outside of the boundary.

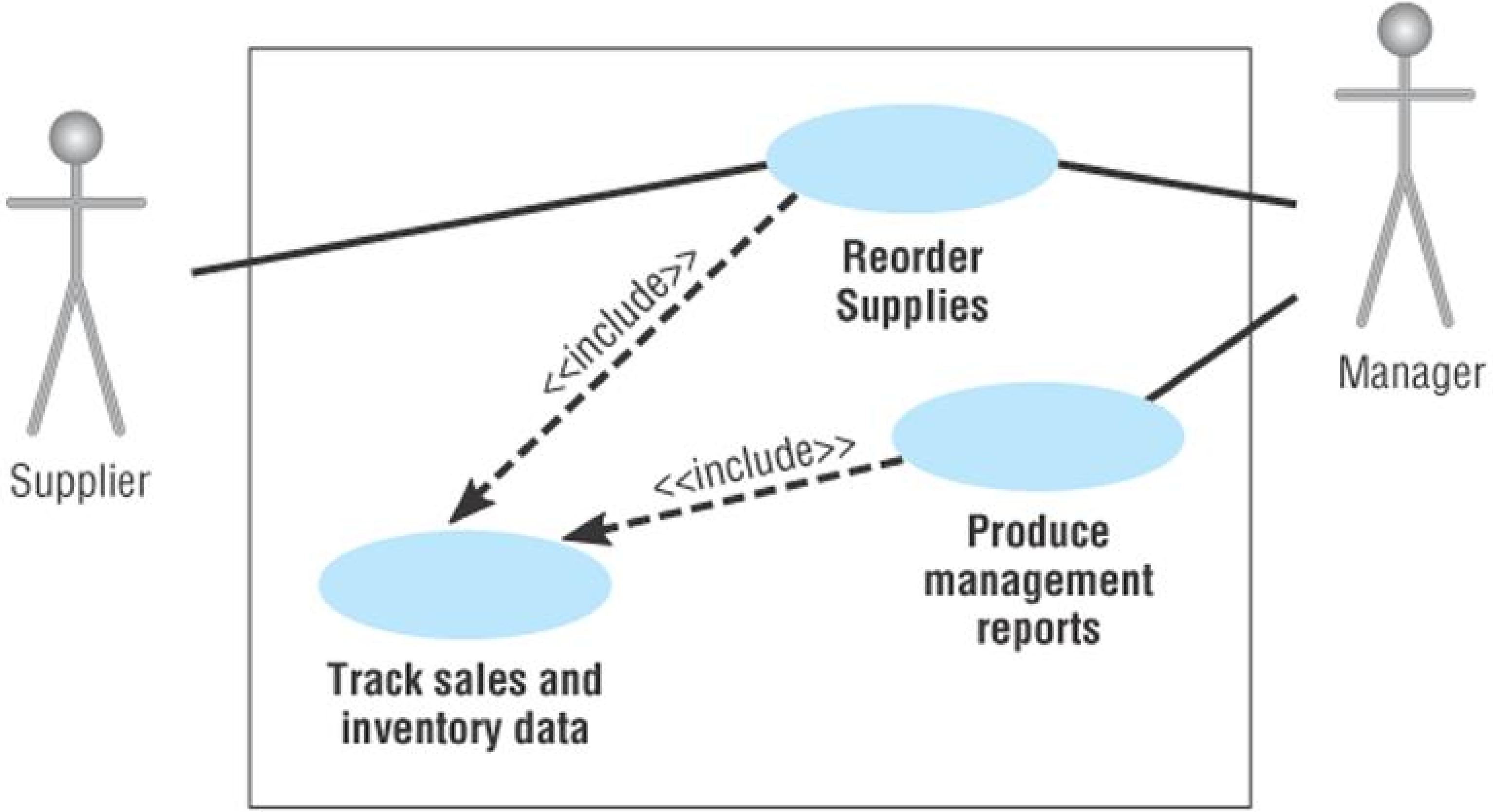
What Is a Connection?

- An association between an actor and a use case
- Depicts a usage relationship
- Connection does not indicate data flow

WHAT IS AN <<INCLUDE>> RELATIONSHIP?

- A connection between two use cases
- Indicates a use case that is used (invoked) by another use case
- Links to general purpose functions, used by many other use cases

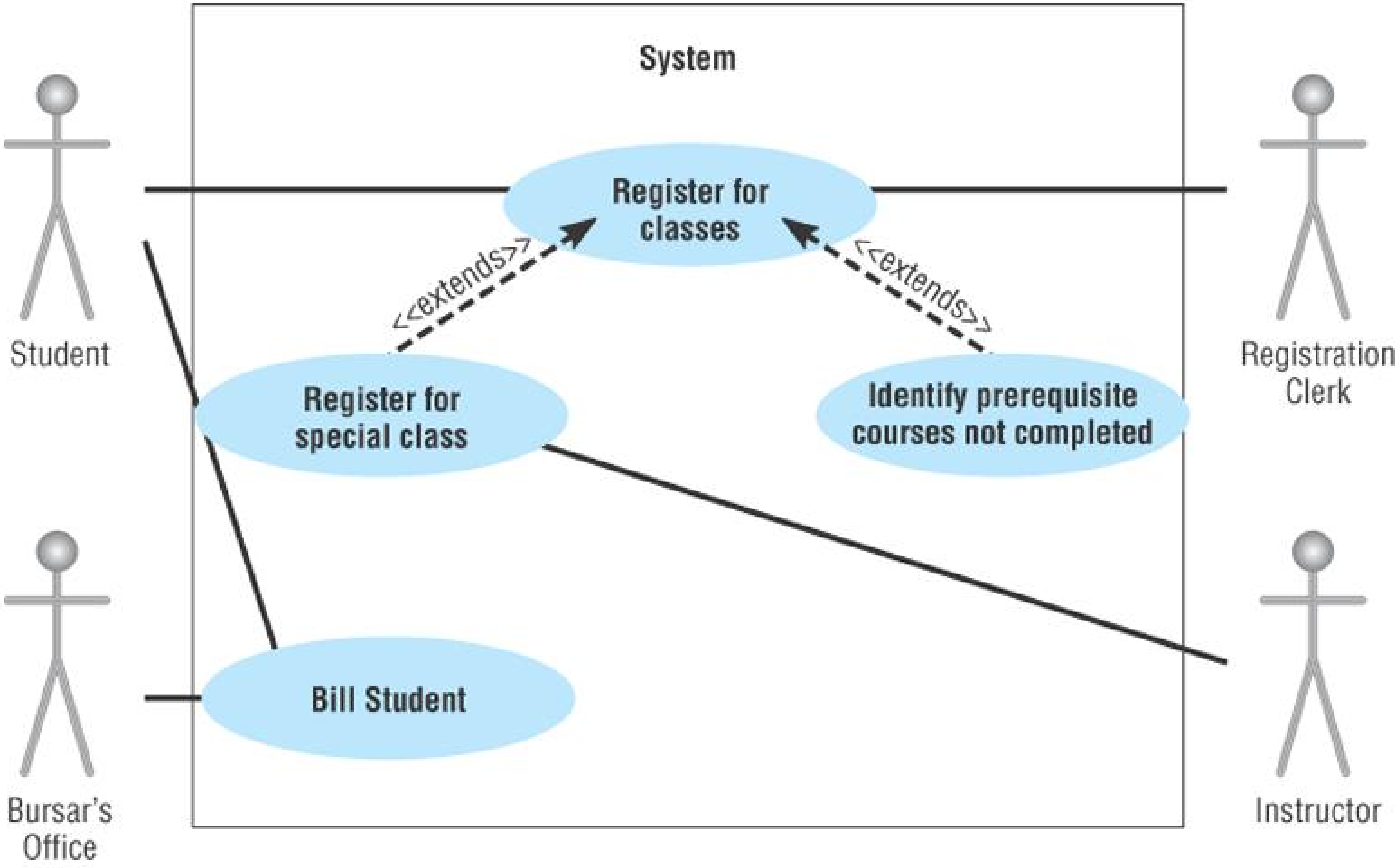
Figure 6.3 An Example of an Include Relationship Between Use Cases



What Is an <<extend>> Relationship?

- A connection between two use cases
- Extends a use case by adding new behavior or actions
- Specialized use case extends the general use case

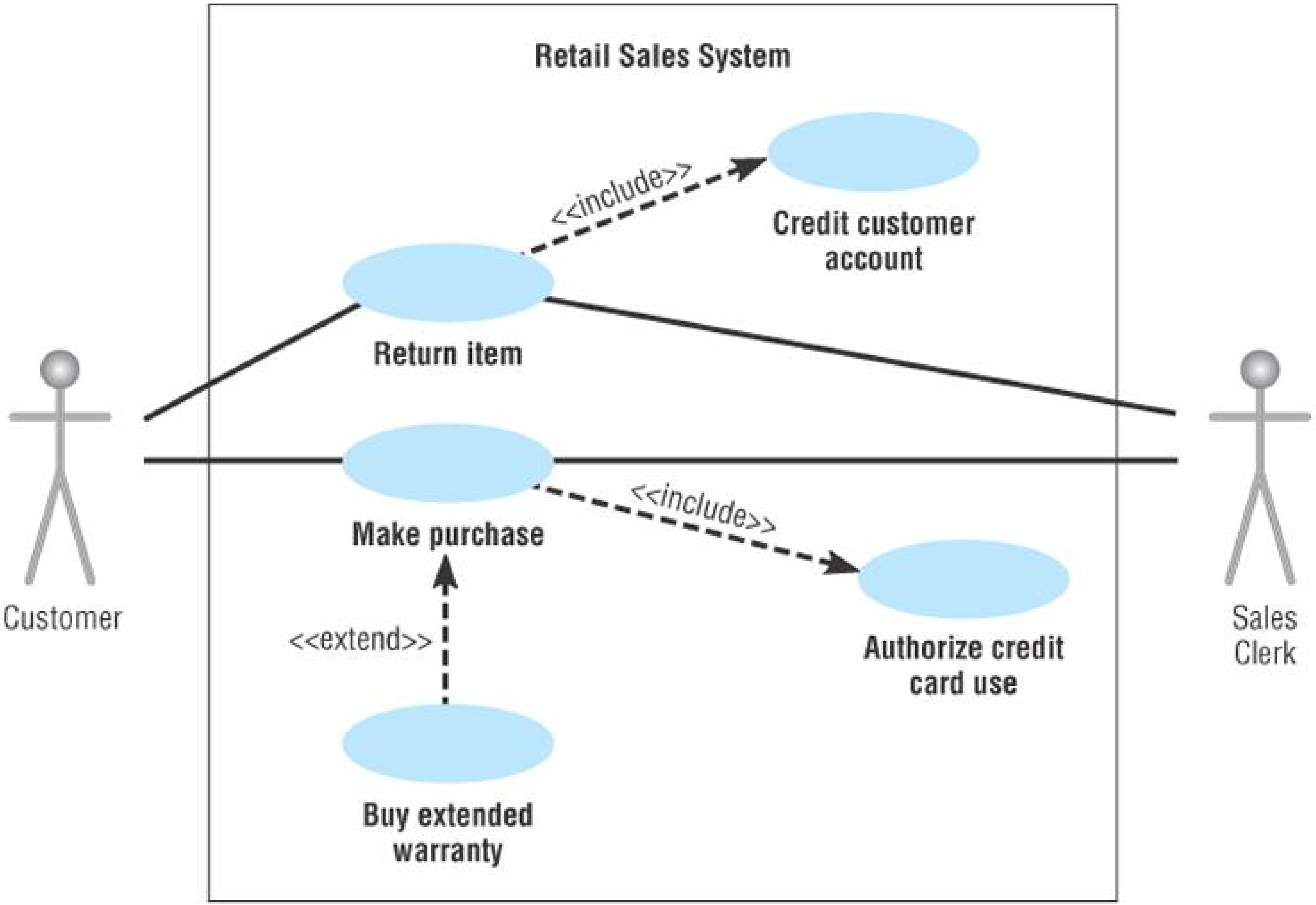
Figure 6.4 A Use Case Diagram for a University Registration System Represented with Microsoft's Visio



What Is a Stereotype

- A construct that extends the UML vocabulary
- Adds new meanings to existing entities
- Depicted with << >> delimiters
- <<extend>> and <<include>> are stereotypes

Figure 6.5 A Use Case Diagram of a Retail Sales System, Featuring Insert Relations and an Extend Relation



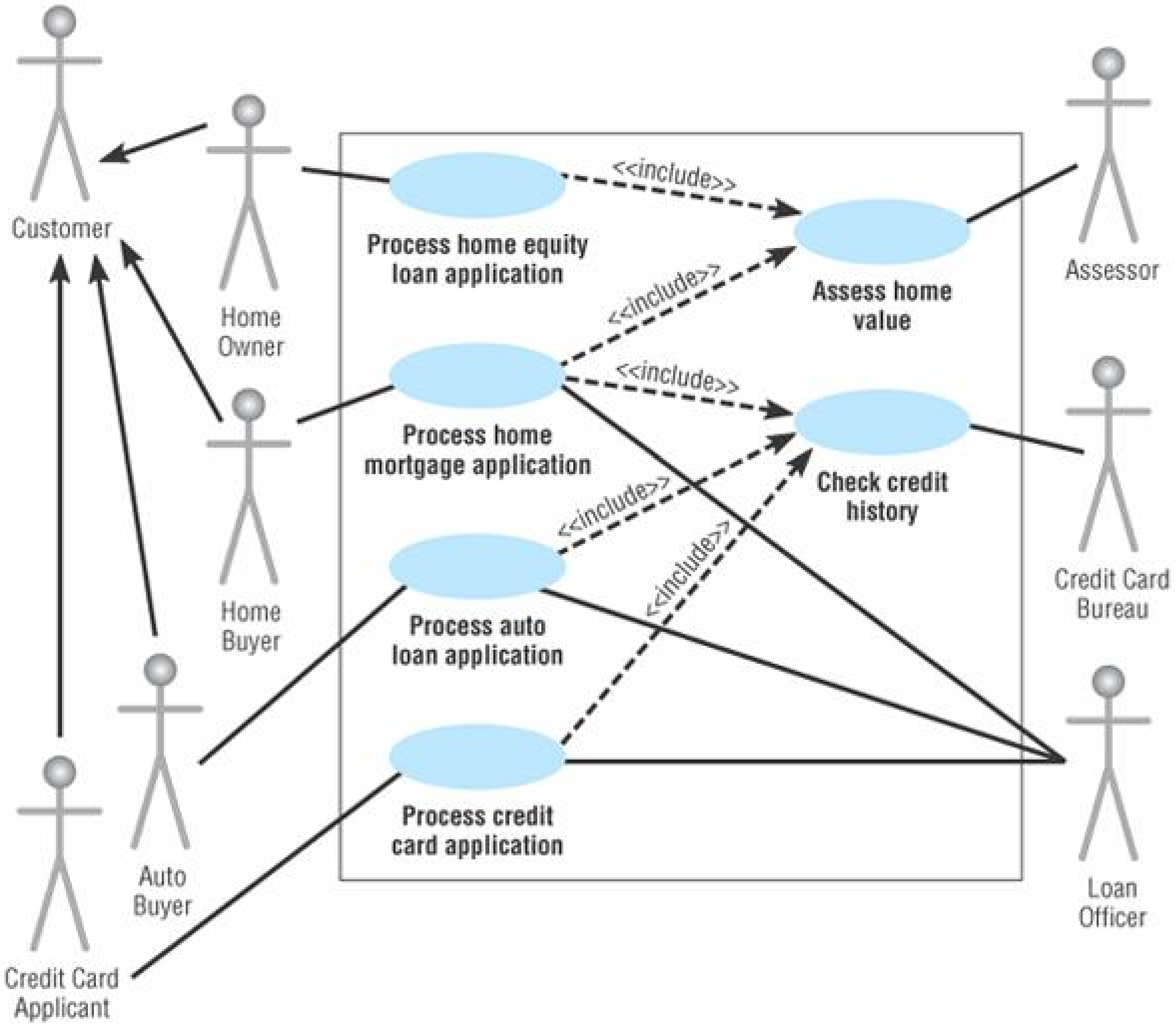
Include vs. Extend

- Use <<extend>> if you want to model an extension to, or a variation of, a complete use case that exists in its own right
- Use <<include>> if you want to factor the common behavior among two or more use cases into a single generalized use case

Actors can be grouped in generalization categories.

Figure 6.6

Examples of Generalized Use Cases and Include Relationships



Use Case examples

- Person spell checks a typed document
- Clerk prints a sales receipt for a video rental.
- Receptionist schedules an appointment.
- Advisor registers student for classes.

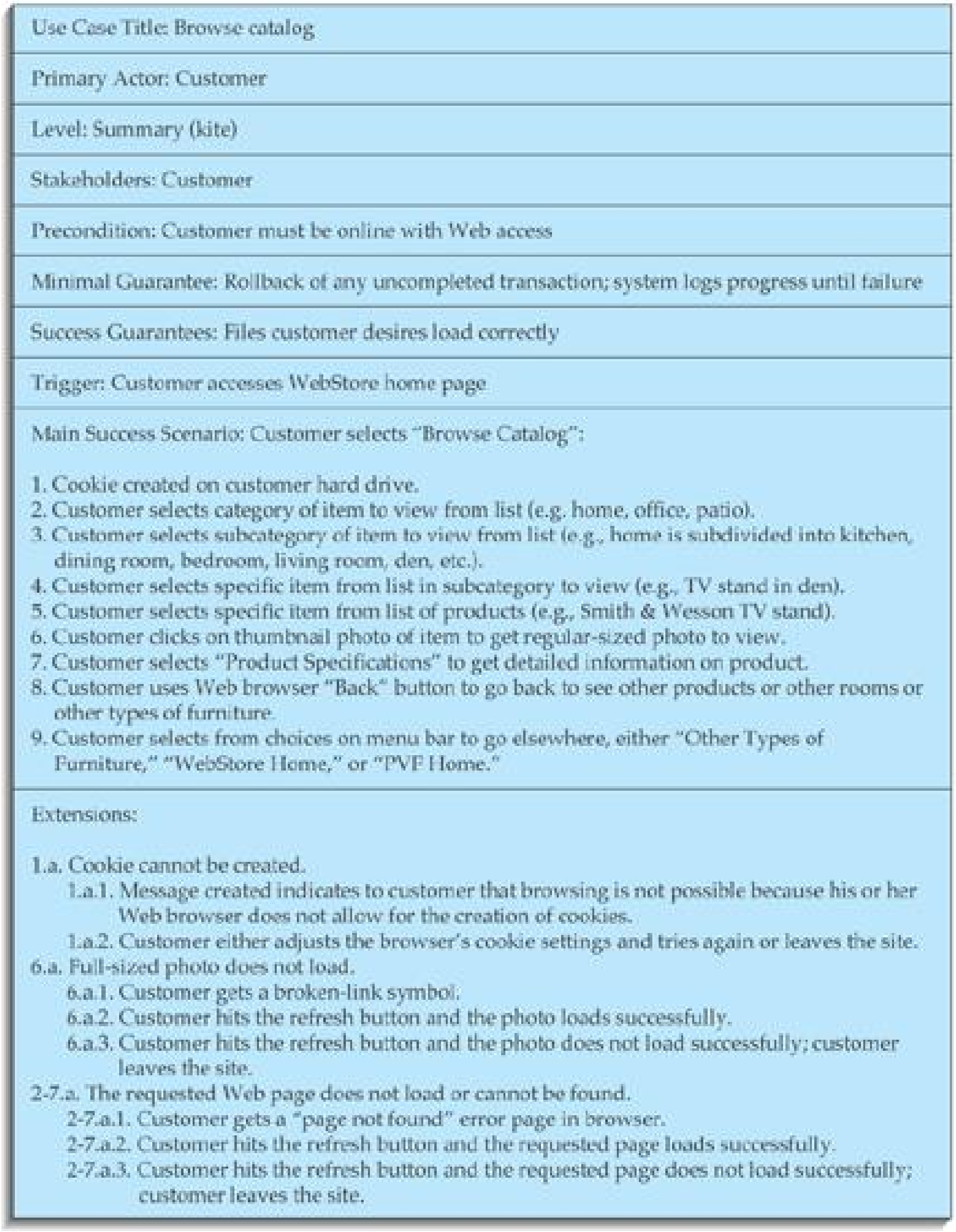
Written Use Cases

- Document containing detailed specifications for a use case
- Contents can be written as simple text or in a specified format

Sample Format for Written Use Case

- ***Title*** – descriptive name, matches name in use case diagram
- ***Primary actor*** – usually a user role
- ***Stakeholders*** – any group or individual with an interest in the function of the use case
- ***Precondition*** – conditions that must be satisfied in order to execute the use case
- ***Minimal guarantee*** – outputs that can be expected if the service attempt failed
- ***Success guarantee*** – outputs that can be expected if the service succeeds
- ***Trigger*** – an event or action that initiates the use case
- ***Main success scenario*** – description of sequence of interactions between actor and use case during the use case execution
- ***Extensions*** – detailed description of how errors are dealt with

Figure 6.9 Jim Woo's Completed Template for PVF's "Browse Catalog" Use Case



Use Case Title: Browse catalog

Primary Actor: Customer

Level: Summary (kite)

Stakeholders: Customer

Precondition: Customer must be online with Web access

Minimal Guarantee: Rollback of any uncompleted transaction; system logs progress until failure

Success Guarantees: Files customer desires load correctly

Trigger: Customer accesses WebStore home page

Main Success Scenario: Customer selects "Browse Catalog":

1. Cookie created on customer hard drive.
2. Customer selects category of item to view from list (e.g. home, office, patio).
3. Customer selects subcategory of item to view from list (e.g., home is subdivided into kitchen, dining room, bedroom, living room, den, etc.).
4. Customer selects specific item from list in subcategory to view (e.g., TV stand in den).
5. Customer selects specific item from list of products (e.g., Smith & Wesson TV stand).
6. Customer clicks on thumbnail photo of item to get regular-sized photo to view.
7. Customer selects "Product Specifications" to get detailed information on product.
8. Customer uses Web browser "Back" button to go back to see other products or other rooms or other types of furniture.
9. Customer selects from choices on menu bar to go elsewhere, either "Other Types of Furniture," "WebStore Home," or "PVF Home."

Extensions:

1.a. Cookie cannot be created.
  1.a.1. Message created indicates to customer that browsing is not possible because his or her Web browser does not allow for the creation of cookies.
  1.a.2. Customer either adjusts the browser's cookie settings and tries again or leaves the site.
6.a. Full-sized photo does not load.
  6.a.1. Customer gets a broken-link symbol.
  6.a.2. Customer hits the refresh button and the photo loads successfully.
  6.a.3. Customer hits the refresh button and the photo does not load successfully; customer leaves the site.
2-7.a. The requested Web page does not load or cannot be found.
  2-7.a.1. Customer gets a "page not found" error page in browser.
  2-7.a.2. Customer hits the refresh button and the requested page loads successfully.
  2-7.a.3. Customer hits the refresh button and the requested page does not load successfully; customer leaves the site.

A sample written use case

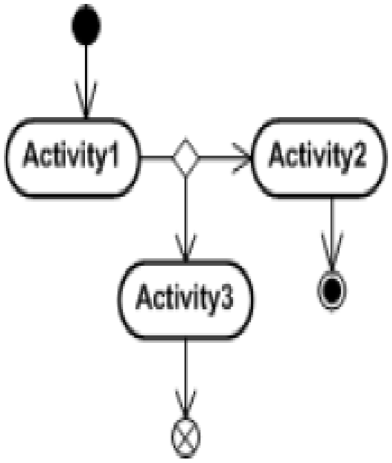
Activity Diagrams

- The purpose of the activity diagram is to model the procedural flow of actions that are part of a larger activity. In projects in which use cases are present, activity diagrams can model a specific use case at a more detailed level. However, activity diagrams can be used independently of use cases for modeling a business-level function.
- Because it models procedural flow, the activity diagram focuses on the action sequence of execution and the conditions that trigger or guard those actions.
- It "drills down" into the details of how a given business use case is implemented

[From Rational Edge: The Activity Diagram]

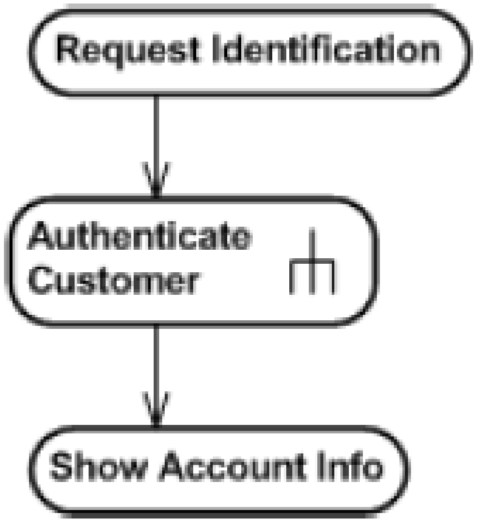
Starting and Stopping

- The solid circle indicates the beginning of the sequence of activities.
- The circle with an X represents an end of a "flow" but not the end of the entire use case. In other words, some subtask completes, but the entire use case is not yet complete.
- The "target" indicates that the entire use case is complete.



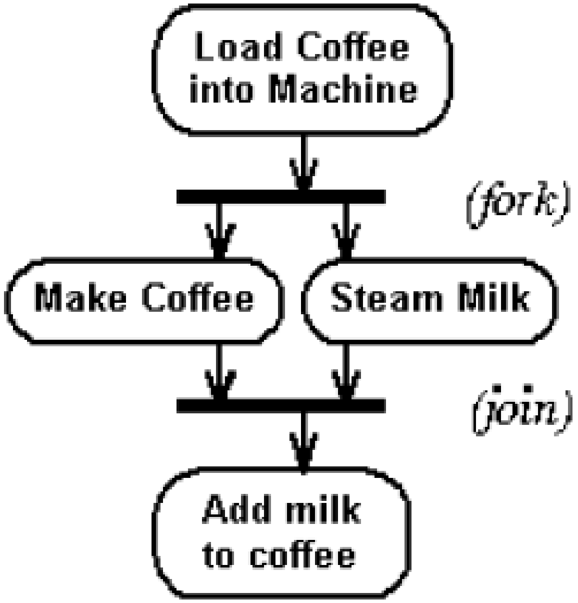
Sub-Activity

- The "rake" symbol indicates that the "activity" is complex enough to merit its own activity diagram. In use-case analysis, this is a "subcase"---a stand-alone activity that occurs in more than one use case but is not large enough to be a use case in its own right.

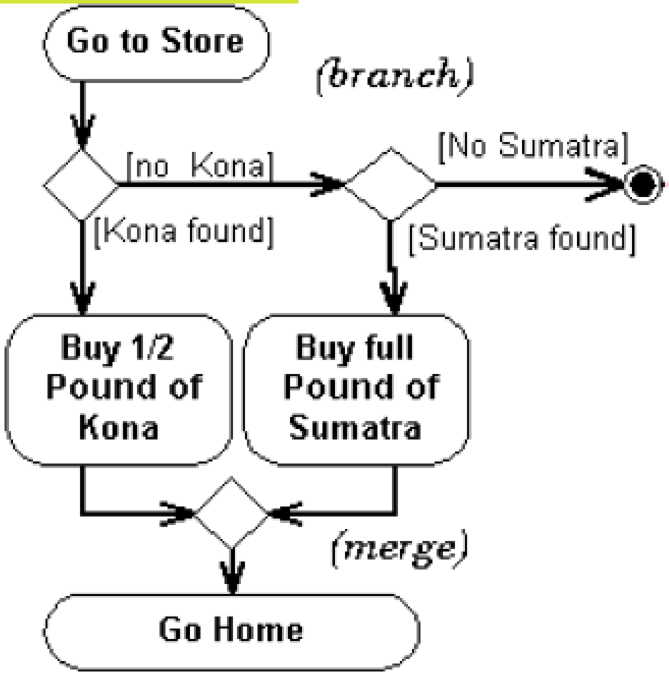


Synchronization

- Used either when several activities can go on in parallel or when the order in which a set of activities execute is immaterial. The heavy bar at the top is a *fork*. After the fork, all activities can (but are not required to) go on in parallel. Progress cannot continue past the bar on the bottom (the *join*) until all the activities that feed into the join complete.



Decision



- A decision activity, the guard (tests) labels the decision that was made. The diamond with outgoing arrows (the *branch*) specifies an OR operation, with a condition imposed by the guard. The diamond with incoming arrows (a *merge*) simply provides an end to the OR operation. A merge can occur without an associated branch if the diagram has multiple start states.

Swim Lanes

- Activities are arranged into vertical or horizontal zones delimited with lines. Each zone represents a broad area of responsibility, typically implemented by a set of classes or objects. For example, the swim lane labeled *accounting* could represent objects of several classes (Bookkeeper, Clerk, MailRoom, Accountant) working in concert to perform the single "cut paycheck“ activity.

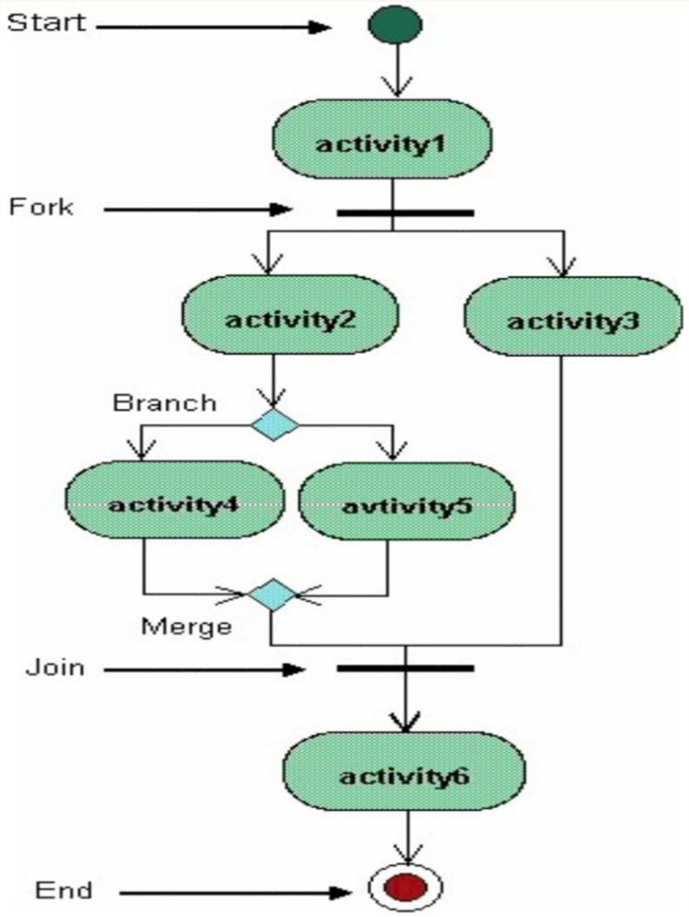
[From Holub Associates: UML Reference Card]

Activity Diagram Notation

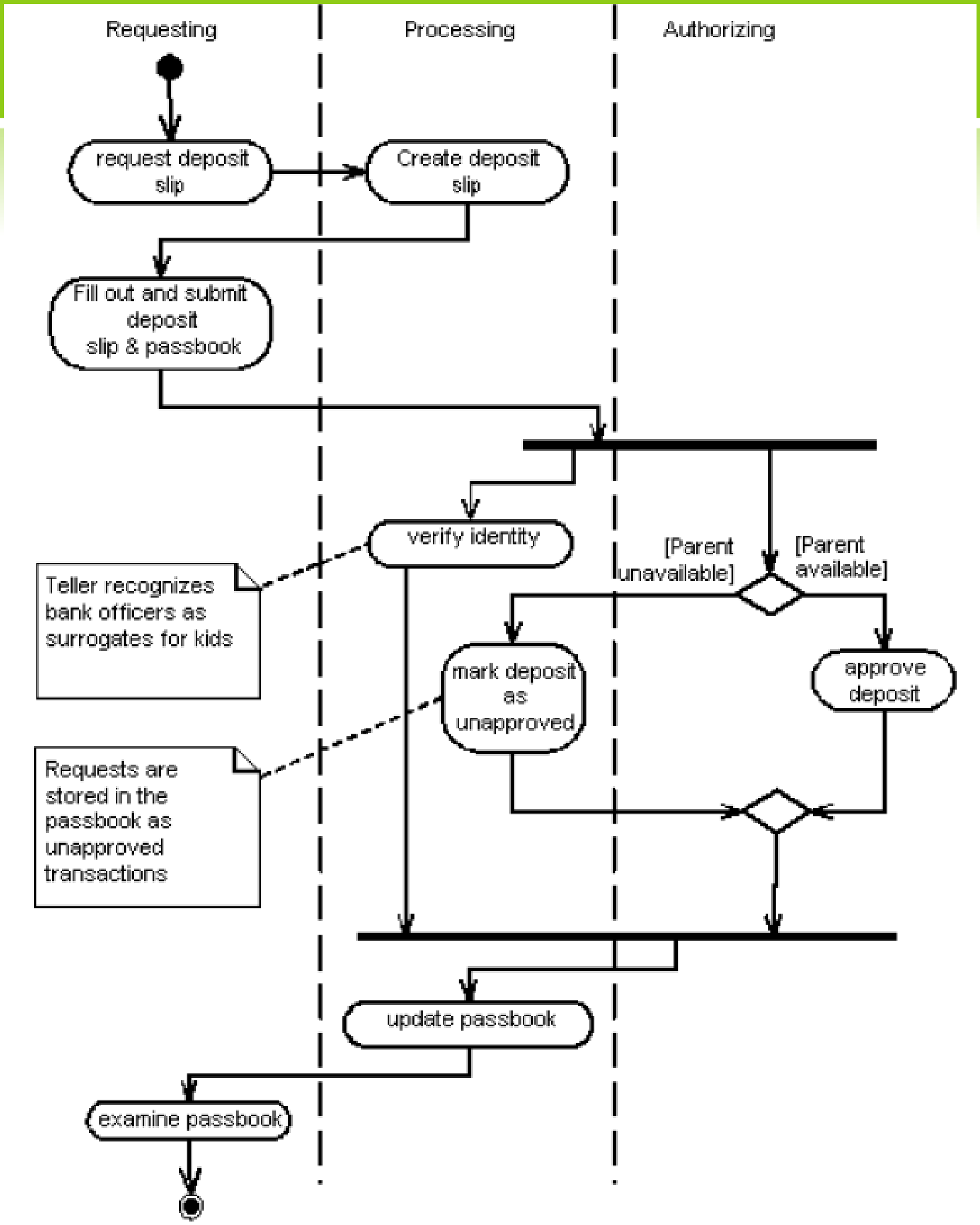
Notation	Name	Description
●	Start	The start of activities
◉	Stop	The end of activities
▭	Activity	Task performed. Steps in workflows.
◇	Decision	Decision to make
⊟	Object	Object to refer, e.g. documents. Affected by the workflow, change state as the workflow goes along.
▬	Synchronization bar	Upper : fork Lower : join ≥ 2 activities occur simultaneously

Example of Activity Diagram

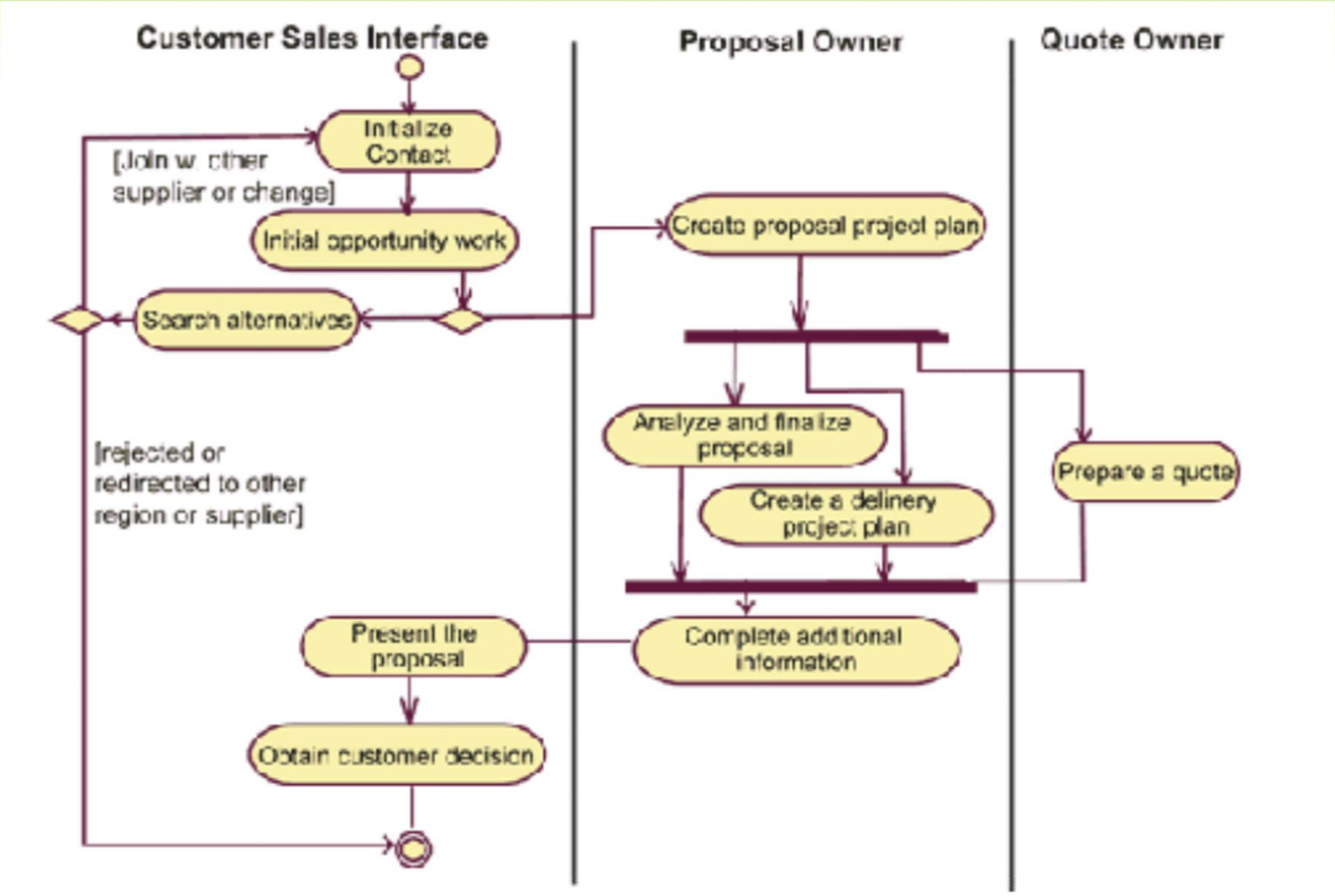
Normally used
With business use
Case model



Activity Diagram Example


Requesting
Processing
Authorizing
request deposit slip
Create deposit slip
Fill out and submit deposit slip & passbook
verify identity
[Parent unavailable]
[Parent available]
mark deposit as unapproved
approve deposit
Teller recognizes bank officers as surrogates for kids
Requests are stored in the passbook as unapproved transactions
update passbook
examine passbook

Activity Diagram Example

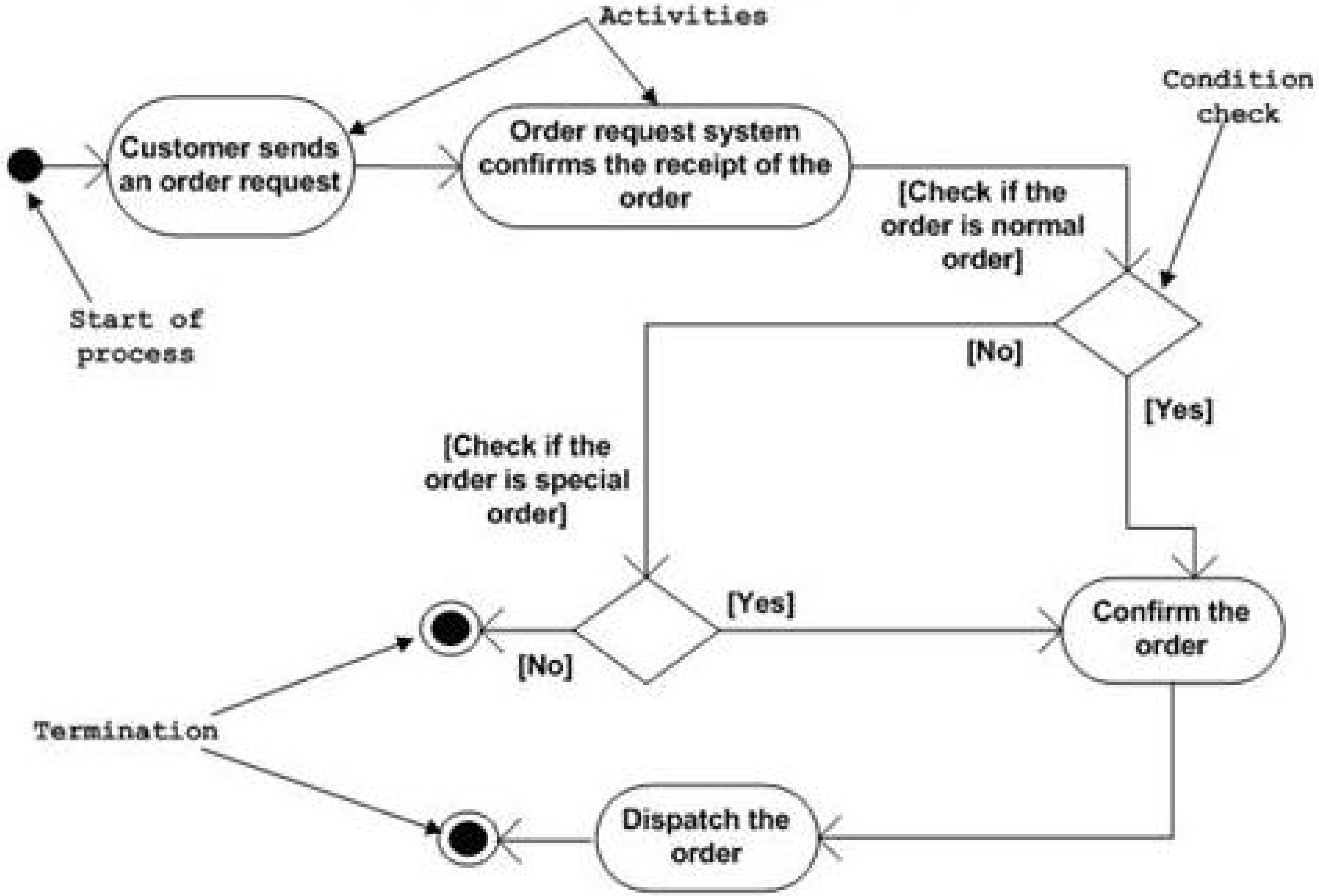

Customer Sales Interface
Proposal Owner
Quote Owner
Initialize Contact
[Join w. other supplier or change]
Initial opportunity work
Search alternatives
Create proposal project plan
Analyze and finalize proposal
Create a delivery project plan
Prepare a quote
[rejected or redirected to other region or supplier]
Present the proposal
Complete additional information
Obtain customer decision

ACTIVITY DIAGRAM FOR ORDER MANAGEMENT SYSTEM

- The four main activities in order management system:
  - Send order by the customer
  - Receipt of the order
  - Confirm order
  - Dispatch order
  - After receiving the order request condition checks are performed to check if it is normal or special order. After the type of order is identified dispatch activity is performed and that is marked as the termination of the process.

ACTIVITY DIAGRAM: ORDER MANAGEMENT SYSTEM

Activity diagram of an order management system


Activities
Condition check
Customer sends an order request
Order request system confirms the receipt of the order
[Check if the order is normal order]
Start of process
[No]
[Yes]
[Check if the order is special order]
[Yes]
Confirm the order
[No]
Termination
Dispatch the order

EXAMPLE: SCENARIO FOR ACTIVITY DIAGRAM

- Draw an activity diagram that represents the **making of a cup of Milo**
  - The initial two activities are Fill kettle with water, Find cup and they may be performed in parallel
  - When the Find cup are completed the activity place milo in cup can start
  - The kettle must have boiled and milo must have been placed in the cup before the activity Add water to cup can begin
  - If milk is required then activity Add milk should be performed
  - Lastly, stir the milo, water and milk(if any).

Recap

- After studying this chapter we learned to:
  - Understand how to structure requirements with use case diagrams.
  - Explain the basics of use case construction using UML standards.
  - Construct use case diagrams.
  - Write text-based use cases.
  - Construct Activity Diagrams