CSCI 4050/6050
Software Engineering

Events and Use cases

Dr. Eman Saleh   **eman.saleh@uga.edu**

Outline

Use Cases and Event Decomposition
Use Cases in the Ridgeline Mountain Outfitters Case

In part, based on slides of the book Systems analysis and design in a changing world by Satizinger and others

Introduction

- We will discuss wo techniques for identifying use cases are the user goal technique and the event decomposition technique

- The user goal technique or the user stories which begins by identifying end users called actors and asking what specific goals they have when interacting with the system. Discusses in previous lecture

- The event decomposition technique begins by identifying events that occur that require the system to respond.
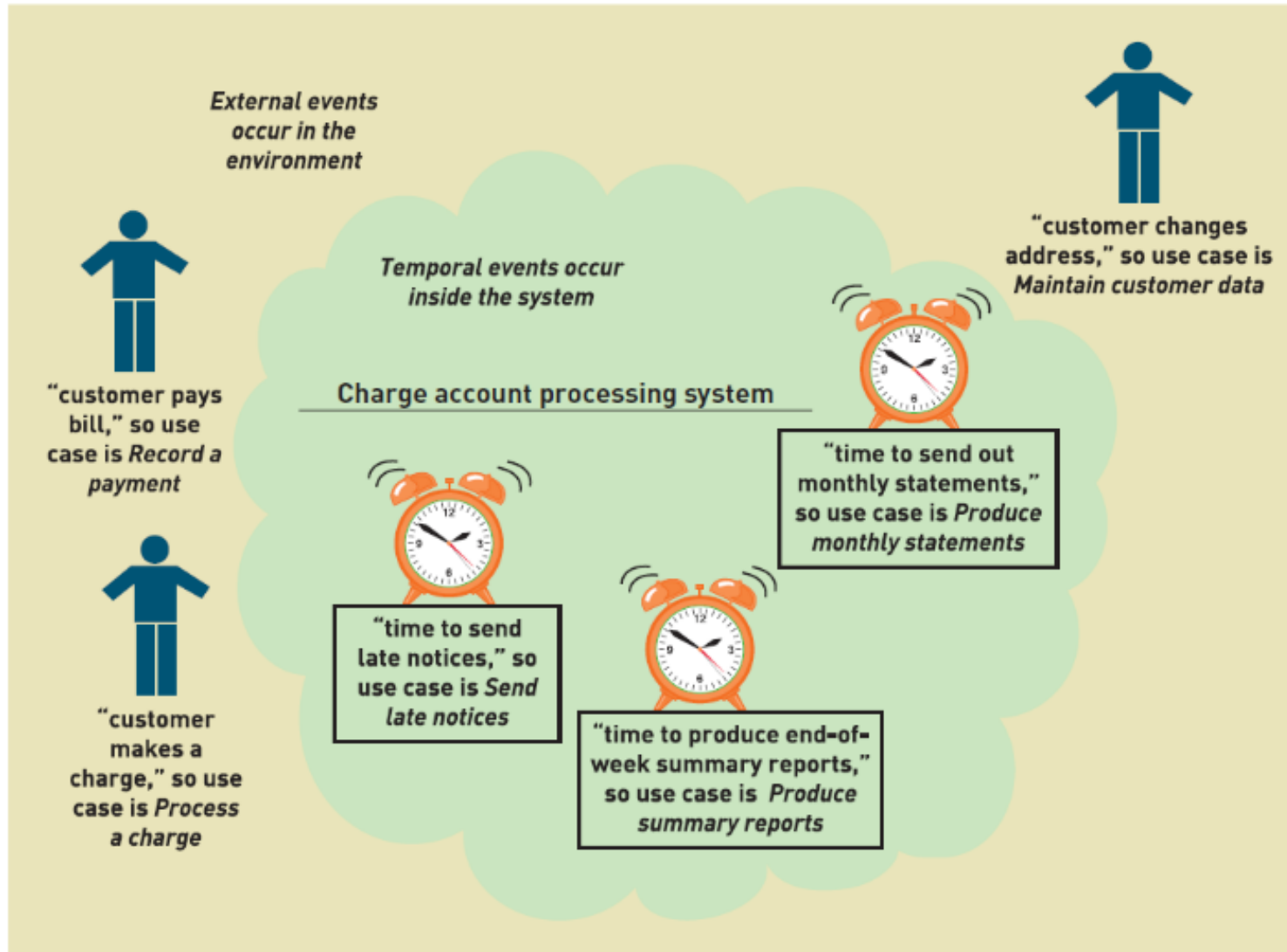
Events and Use Cases

External events occur in the environment

Temporal events occur inside the system

Charge account processing system

"customer pays bill," so use case is *Record a payment*

"customer makes a charge," so use case is *Process a charge*

"time to send late notices," so use case is *Send late notices*

"time to produce end-of-week summary reports," so use case is *Produce summary reports*

"time to send out monthly statements," so use case is *Produce monthly statements*

"customer changes address," so use case is *Maintain customer data*

Types of Events

- ## External Event
  - an event that occurs outside the system, usually initiated by an external agent or actor

- ## Temporal Event
  - an event that occurs as a result of reaching a point in time

- ## State Event/ internal
  - an event that occurs when something happens inside the system that triggers some process
  - reorder point is reached for inventory item

External Event Checklist

- External agent or actor wants something resulting in a transaction
  - Customer buys a product
- External agent or actor wants some information
  - Customer wants to know product details
- External data changed and needs to be updated
  - Customer has new address and phone
- Management wants some information
  - Sales manager wants update on production plans

Temporal Event Checklist

- Internal outputs needed at points in time
  - Management reports (summary or exception)
  - Operational reports (detailed transactions)
  - Internal statements and documents (including payroll)
- External outputs needed at points of time
  - Statements, status reports, bills, reminders

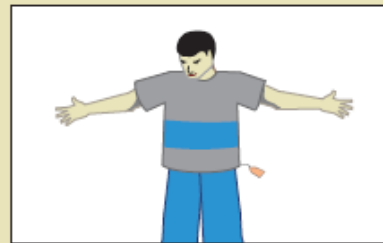# Finding the actual event that affects the system

Tracing a sequence of transactions resulting in many events

Customer requests a catalog

Customer wants to check item availability

Customer places an order

Customer changes or cancels an order

Customer wants to check order status

Customer updates account information

Customer returns the item

Figure from systems analysis and design in a changing world by Satzingers et. al.
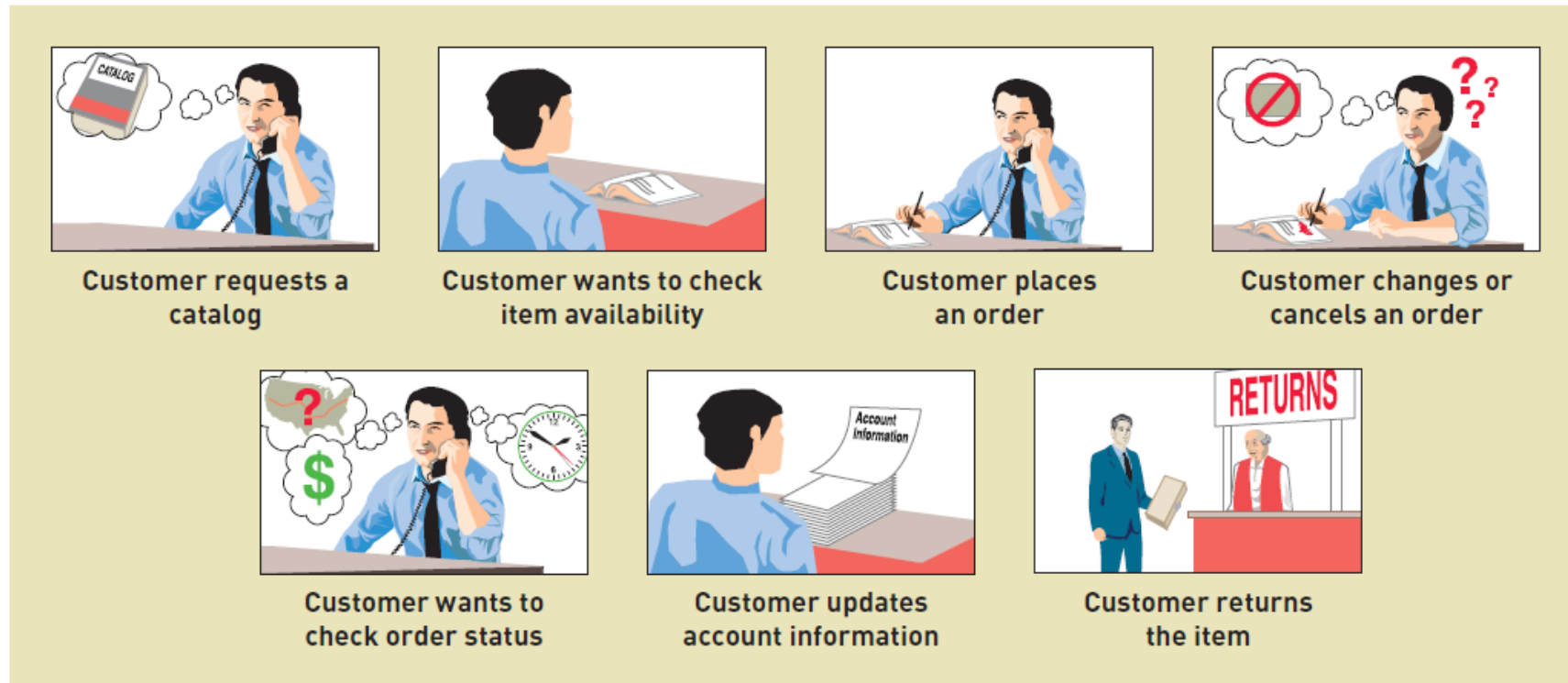
Perfect Technology Assumption

- Don't worry about functions built into system because of limits in technology and people. Wait until <u>design.</u>

Don't worry much about these until you are considering design issues

User wants to log on to the system

User wants to change the password

User wants to change preference settings

System crash requires database recovery

Time to back up the database

Time to require the user to change the password
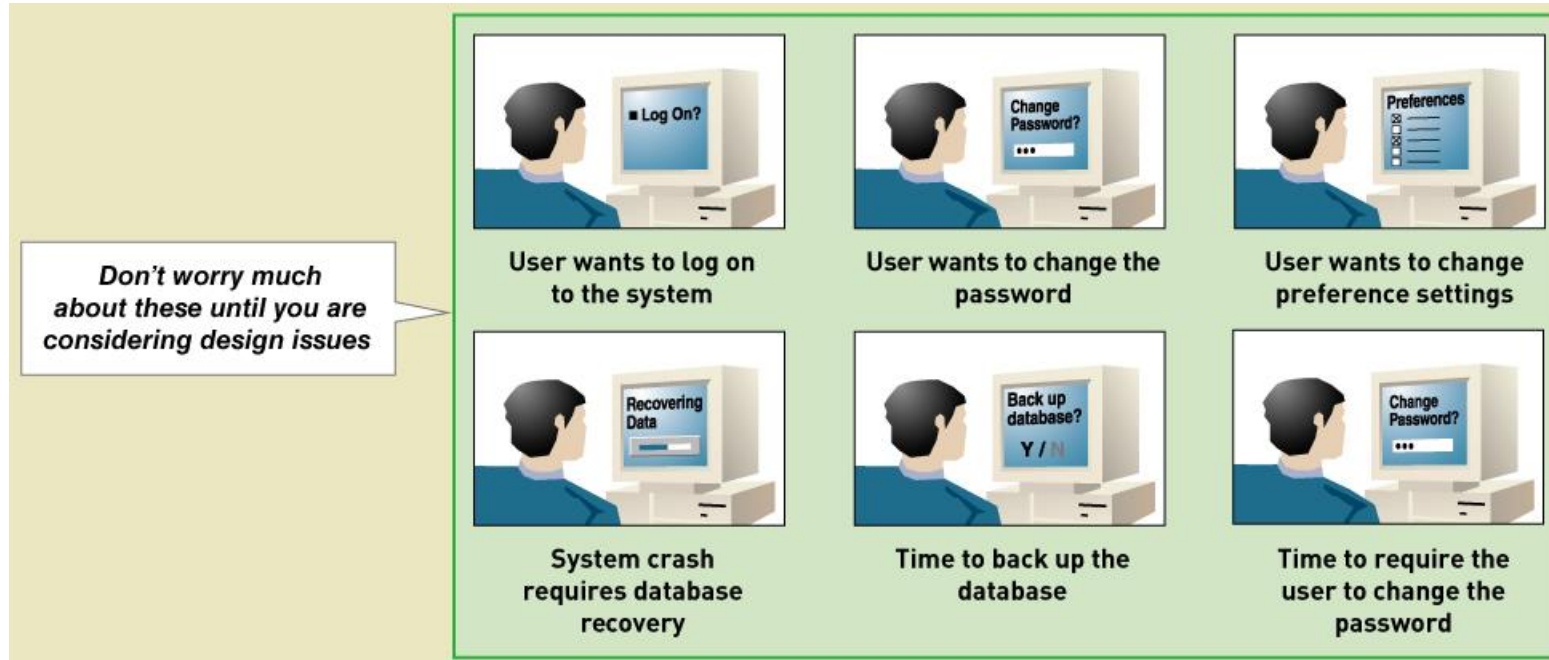
Most of these events involve **system controls**, which are checks or safety procedures put in place to protect the integrity of the system

External Events Check List

External events to look for include:
√ External agent wants something resulting in a transaction
√ External agent wants some information
√ Data changed and needs to be updated
√ Management wants some information

Temporal Events Check List

Temporal events to look for include:
√ Internal outputs needed
   √ Management reports (summary or exception)
   √ Operational reports (detailed transactions)
   √ Internal statements and documents (including payroll)
√ External outputs needed
   √ Statements, status reports, bills, reminders

Event Decomposition Technique: Specific Steps

1. Consider the external events in the system environment that require a response from the system by using the checklist shown in [slide 10](#)
2. For each external event, identify and name the use case that the system requires
3. Consider the temporal events that require a response from the system by using the checklist shown in [slide 11.](#)
4. For each temporal event, identify and name the use case that the system requires and then establish the point of time that will trigger the use case

Event Decomposition Technique: Specific Steps
(continued)

5. Consider the state events that the system might respond to, particularly if it is a real-time system in which devices or internal state changes trigger use cases.

6. For each state event, identify and name the use case that the system requires and then define the state change.

7. When events and use cases are defined, check to see if they are required by using the perfect technology assumption. Do not include events that involve such system controls as login, logout, change password, and backup or restore the database, as these are put in later.

Event Decomposition Technique: Benefits

- Events are broader than user goal: Capture temporal and state events
- Help decompose at the right level of analysis: an elementary business process (EBP)
- EBP is a fundamental business process performed by one person, in one place, in response to a business event
- Uses perfect technology assumption to make sure functions that support the users work are identified and not additional functions for security and system controls

Use Cases Decomposition

- You might decompose use cases by subsystems or by user actors

RMO CSMS Project Use Cases

CSMS Sales Subsystem	
Use cases	**Users/actors**
Search for item	Customer, customer service representative, store sales representative
View product comments and ratings	Customer, customer service representative, store sales representative
View accessory combinations	Customer, customer service representative, store sales representative
Fill shopping cart	Customer
Empty shopping cart	Customer
Check out shopping cart	Customer
Fill reserve cart	Customer
Empty reserve cart	Customer
Convert reserve cart	Customer
Create phone sale	Customer service representative
Create store sale	Store sales representative

RMO CSMS Project Use Cases

CSMS Order Fulfillment Subsystem	
Use cases	**Users/actors**
Ship items	Shipping
Manage shippers	Shipping
Create backorder	Shipping
Create item return	Shipping, customer
Look up order status	Shipping, customer, management
Track shipment	Shipping, customer, marketing
Rate and comment on product	Customer
Provide suggestion	Customer
Review suggestions	Management

RMO CSMS Project Use Cases

CSMS Customer Account Subsystem	
Use cases	**Users/actors**
Create/update customer account	Customer, customer service representative, store sales representative
Process account adjustment	Management
Send message	Customer
Browse messages	Customer
Request friend linkup	Customer
Reply to linkup request	Customer
Send/receive partner credits	Customer
View "mountain bucks"	Customer
Transfer "mountain bucks"	Customer

RMO CSMS Project Use Cases

| CSMS Marketing Subsystem	
Use cases	**Users/actors**
Add/update product information	Merchandising, marketing
Add/update promotion	Marketing
Add/update accessory package	Merchandising
Add/update business partner link	Marketing

CSMS Reporting Subsystem	
Use cases	**Users/actors**
Produce daily transaction summary report	Management
Produce sales history report	Management, marketing
Produce sales trends report	Marketing
Produce customer usage report	Marketing
Produce shipment history report	Management, shipping
Produce promotion impact report	Marketing
Produce promotional partner activity report	Management, marketing

Use Case Diagrams (revisited)

- Use case diagram— a UML model used to graphically show uses cases and their relationships to actors
- Recall UML is Unified Modeling Language, the standard for diagrams and terminology for developing information systems
- Actor is the UML name for a end user
- **Automation boundary**— the boundary between the computerized portion of the application and the users who operate the application
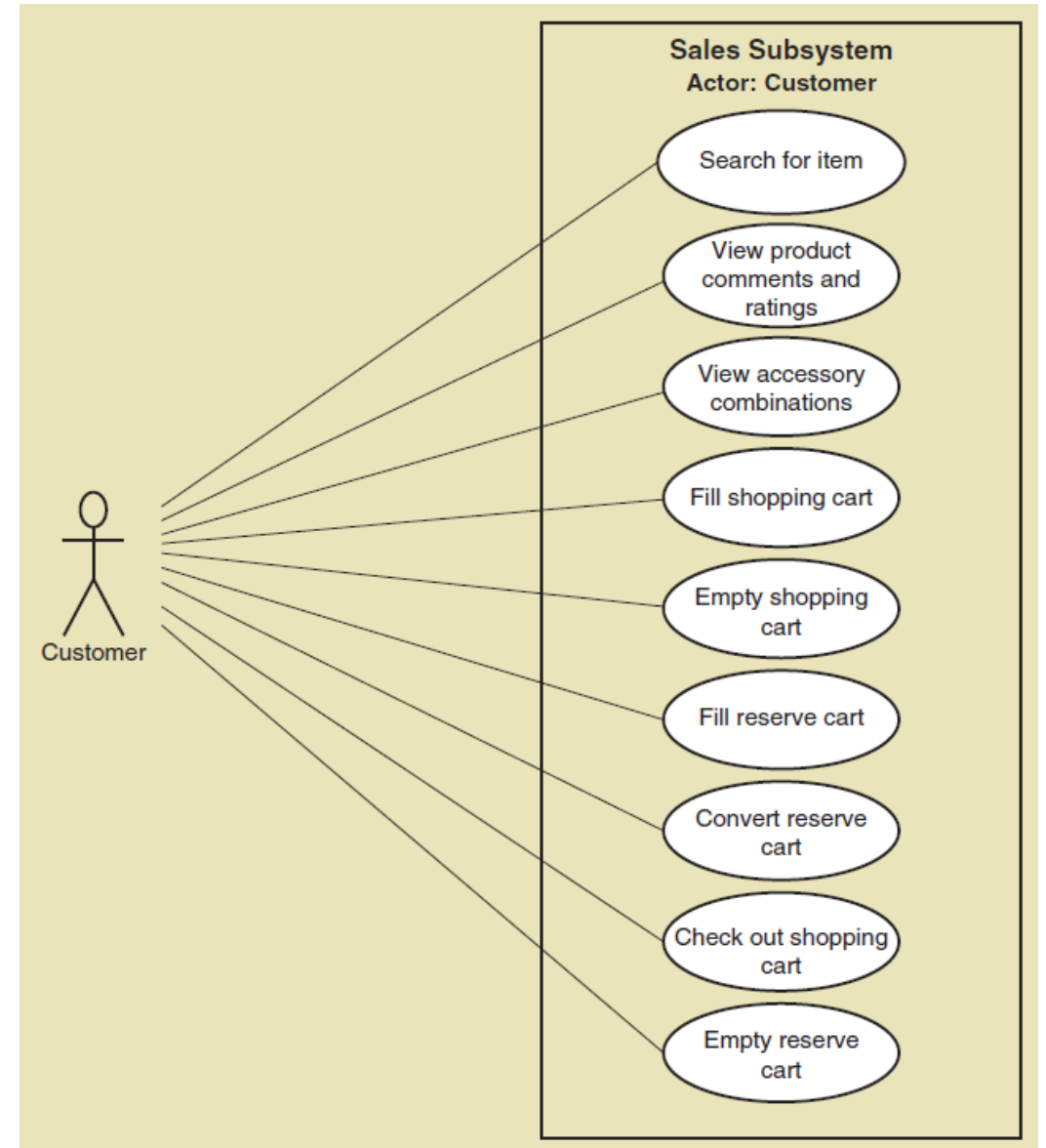
Use Case Diagrams
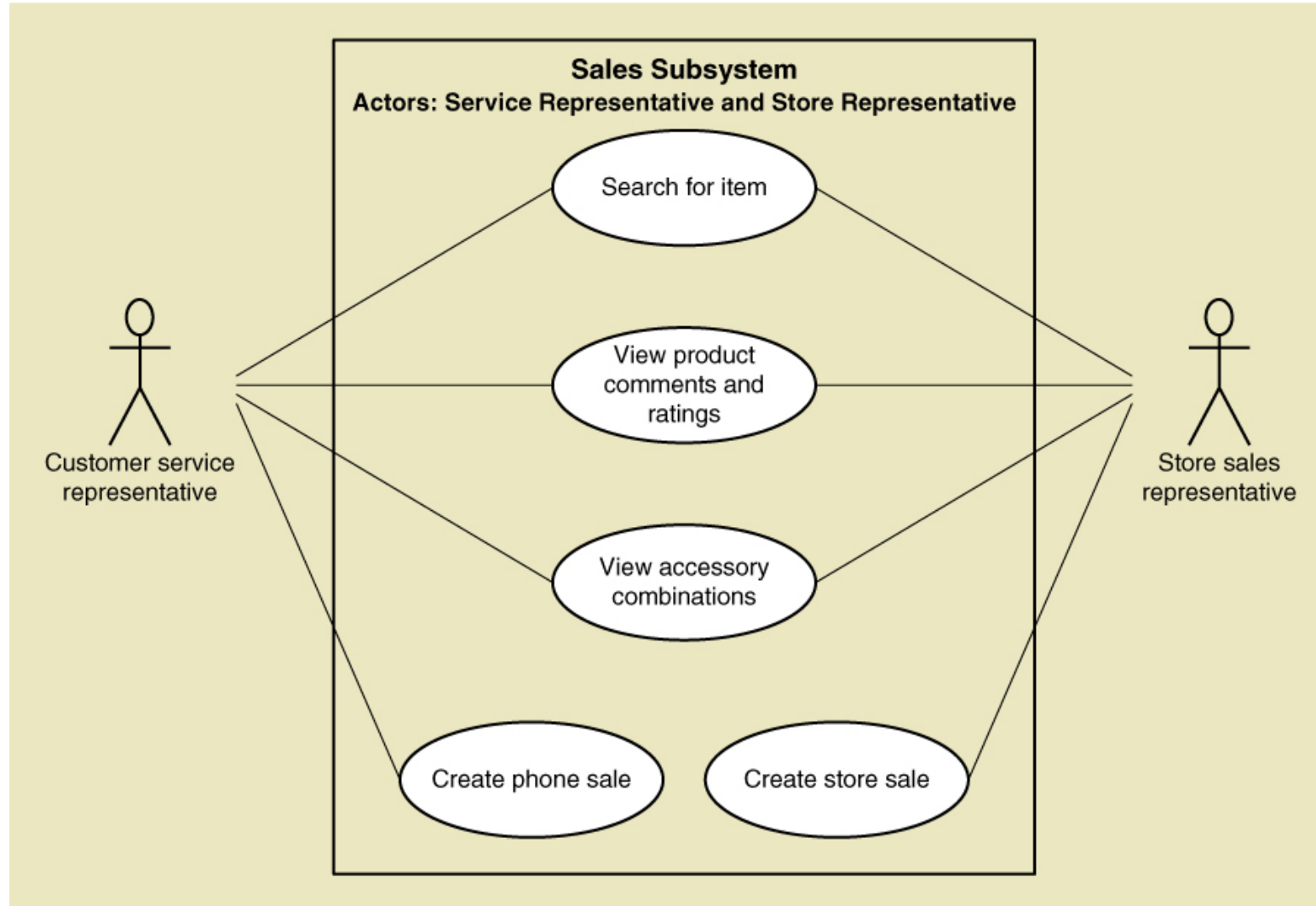
Draw for each subsystem
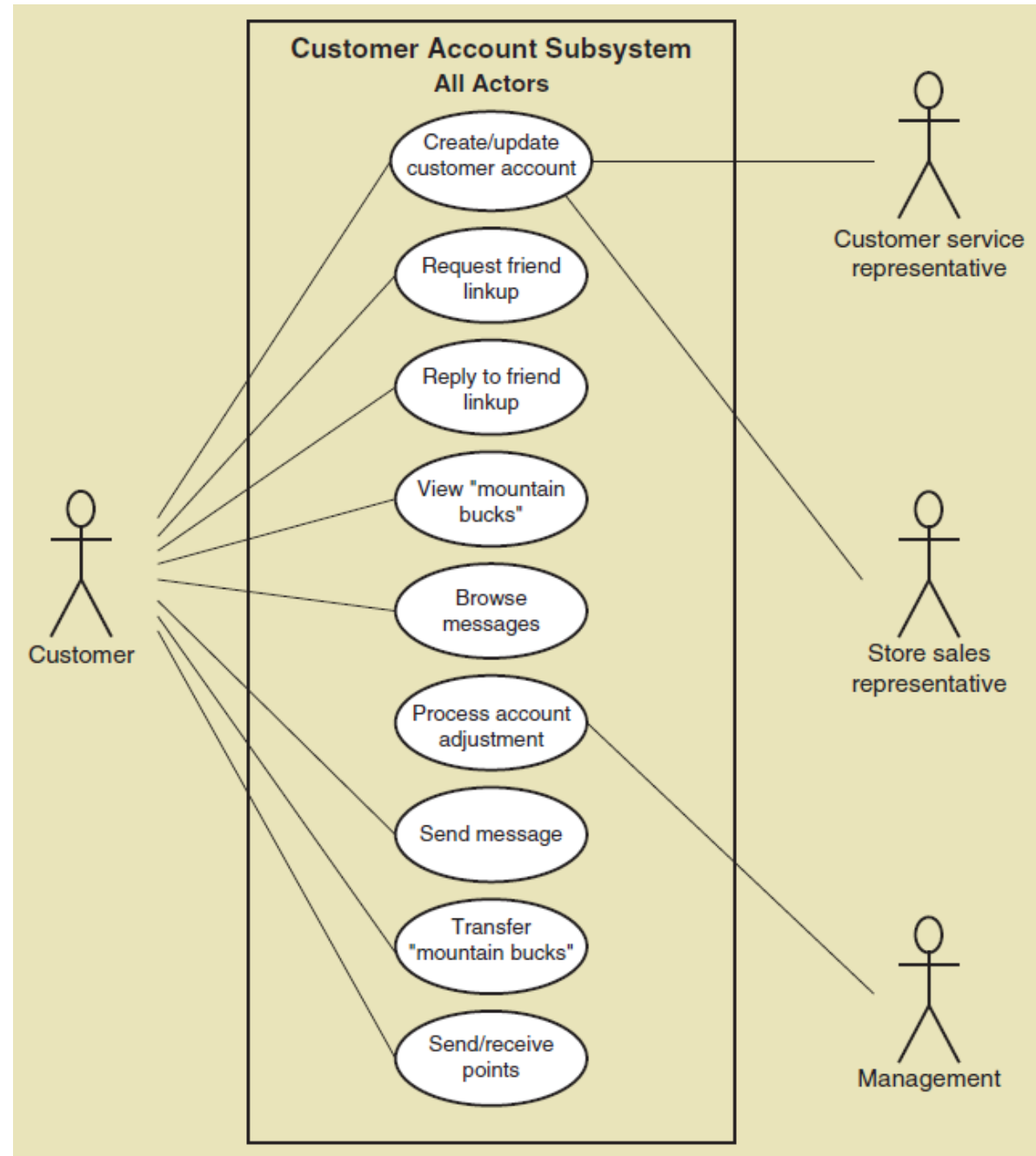
Use Case Diagrams

Draw for a single actor, such as customer

Use Case Diagrams
Draw for internal RMO actors

Use Case Diagrams: Steps

1. Identify all the stakeholders and users who would benefit by seeing a use case diagram

2. Determine what each stakeholder or user needs to review in a use case diagram: each subsystem, for each type of user, for use cases that are of interest

3. For each potential communication need, select the use cases and actors to show and draw the use case diagram. There are many software packages that can be used to draw use case diagrams

4. Carefully name each use case diagram and then note how and when the diagram should be used to review use cases with stakeholders and users

Summary

- WE focus on modeling functional requirements as a part of systems analysis
- Use cases are the functions identified, the activities the system carries out usually in response to a user request
- The user goal technique begins by identifying end users called actors and asking what specific goals they have when interacting with the system
- The event decomposition technique begins by identifying events that occur that require the system to respond.

Summary

- Three types of events include external, temporal, and state events
- Brief use case descriptions are written for use cases
- The use case diagram is the UML diagram used to show the use cases and the actors
- The use case diagram shows the actors, the automation boundary, the uses cases that involve each actor, and the <<includes>> relationship.
- A variety of use case diagrams are draw depending on the presentation needs of the analysis