5

Chapter 5: Modeling Systems Requirements: Events and Things

Systems Analysis and Design in a Changing World, 3rd Edition

5

Learning Objectives

- Explain the many reasons for creating information system models
- Describe three types of models and list some specific models used for analysis and design
- Explain how events can be used to define system requirements
- Identify and analyze events to which a system responds
- Recognize that events trigger system activities or use cases

5

Learning Objectives (continued)

- Explain how the concept of things in the system also defines requirements
- Explain the similarities and the differences between data entities and objects
- Identify and analyze data entities and objects needed in the system
- Read, interpret, and create an entity-relationship diagram
- Read, interpret, and create a class diagram

5

Overview

- Document functional requirements by creating models
- Models created during analysis phase activity: *Define system requirements*
- Two concepts define system requirements in traditional approach and object-oriented approach
  - Events
  - Things

5

Models and Modeling

- Analyst describes information system requirements using a collection of models
- Complex systems require more than one type of model
- Models represent some aspect of the system being built
- Process of creating model helps analyst clarify and refine design
- Models assist communication with system users

5

Reasons for Modeling

FIGURE 5-2
Reasons for modeling.

Types of Models

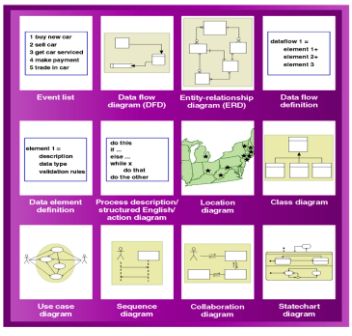
- Different types of models are used in information systems development
  - Mathematical - formulas that describe technical aspects of the system
  - Descriptive - narrative memos, reports, or lists that describe aspects of the system
  - Graphical - diagrams and schematic representations of some aspect of the system

Overview of Models Used in Analysis and Design

- Analysis phase activity named "define system requirements"
  - Logical models
  - Provide detail without regard to specific technology
- Design phase
  - Physical models
  - Provide technical details
  - Extend logical models

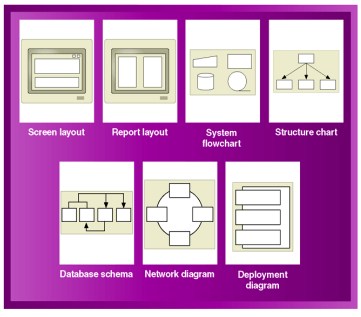
Models Used in Analysis

FIGURE 5-4
Models created during the analysis phase.



Models Used in Design

FIGURE 5-5
Some models created during the design phase.

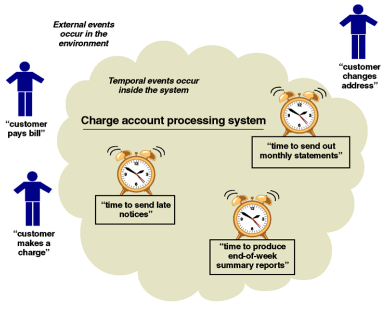


Events and System Requirements

- Events
  - Occurrences at a specific time and place
  - Trigger all system processing
- Requirement definition
  - Determine relevant events
    - External events first
    - Temporal events second
  - Decompose system into manageable units

Events Affecting a Charge Account Processing System

FIGURE 5-6
Events affecting a charge account processing system.



Types of Events

- External
  - Outside system
  - Initiated by external agent or actor
- Temporal
  - Occurs as result of reaching a point in time
  - Based on system deadlines
- State
  - Something inside system triggers processing need

External Event Checklist

FIGURE 5-7
External event checklist.

External Events to Look for Include:
√ External agent wants something resulting in a transaction
√ External agent wants some information
√ Data changed need to be updated
√ Management wants some information

Temporal Event Checklist

FIGURE 5-8
Temporal event checklist.

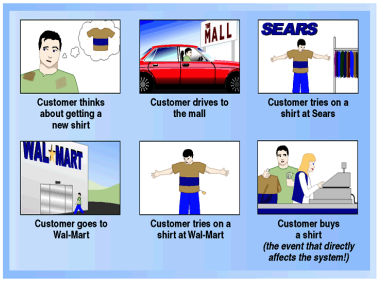
Temporal Events to Look for Include:
√ Internal outputs needed
  √ Management reports (summary or exception)
  √ Operational reports (detailed transactions)
  √ Internal statements and documents (including payroll)
√ External outputs needed
  √ Statements, status reports, bills, reminders

Identifying Events

- Can be difficult to determine
- Often confused with conditions and responses
- May be useful to trace a transaction's life cycle
- Certain events left to design phase
  - Systems controls to protect system integrity
  - Perfect technology assumption defers events

Sequence of Actions that Lead up to Only One Event Affecting the System

FIGURE 5-9
Sequence of actions that lead up to only one event affecting the system.



Sequence of "Transactions" for One Specific Customer Resulting in Many Events

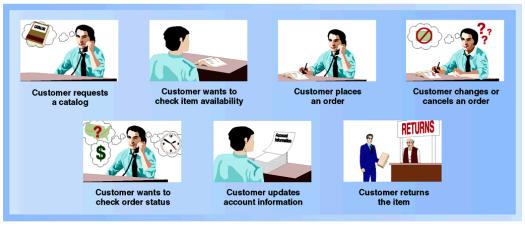
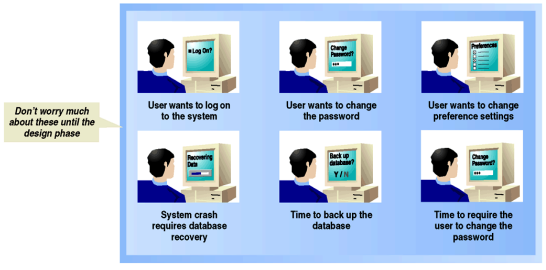


FIGURE 5-10
The sequence of "transactions" for one specific customer resulting in many events.

Events Deferred Until the Design Phase

FIGURE 5-11

Events deferred until the design phase.



Events in the RMO case

- Important external events involve customers
  - Customer checks item availability, customer places order, customer changes or cancels order
- Other external events involve departments
  - Shipping fulfills order, marketing sends promotion to customer, merchandising updates catalog
- Temporal events include periodic reports
  - Time to produce order summary reports, Time to produce fulfillment summary reports

Information about each Event in an Event Table

The event that causes the system to do something.

Source: For an external event, the external agent, or actor is the source of the data entering the system.

Response: What output (if any) is produced by the system?

Event	Trigger	Source	Activity/Use Case	Response	Destination
Customer wants to check item availability	Item inquiry	Customer	Look up item availability	Item availability details	Customer

Trigger: How does the system know the event occurred? For external events, this is data entering the system. For temporal events, it is a definition of the point in time that triggers the system processing.

Activity/Use Case: What does the system do when the event occurs?

Destination: What external agent gets the output produced?

FIGURE 5-14

Information about each event in an event table.

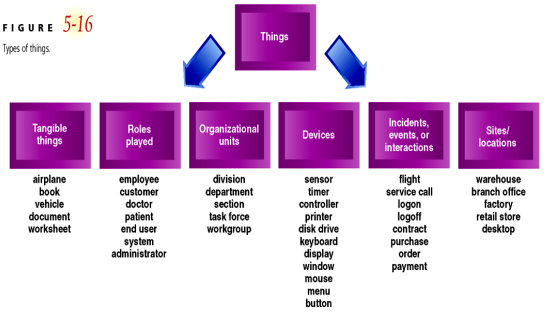
Things and System Requirements

- Define system requirements by understanding system information that needs to be stored
- Store information about things in the problem domain that people deal with when they do their work
- Analysts identify these types of things by considering each event in the event list
  - What things does the system need to know about and store information about?

Types of Things

FIGURE 5-16

Types of things.



Procedure for Developing an Initial List of Things

- Step 1: Using the event table and information about each event, identify all nouns about system
- Step 2: Using other information from existing systems, current procedures, and current reports or forms, add items or categories of information needed
- Step 3: Refine list and record assumptions or issues to explore

Characteristics of Things

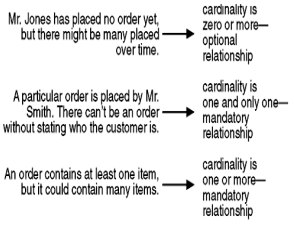
- Relationship
  - Naturally occurring association among specific things
  - Occur in two directions
  - Number of associations is cardinality or multiplicity
    - Binary, unary, ternary, n-ary
- Attribute
  - One specific piece of information about a thing

Relationships Naturally Occur Between Things

FIGURE 5-18
Relationships naturally occur among things.

Cardinality/Multiplicity of Relationships

FIGURE 5-19
Cardinality/multiplicity of relationships.



Attributes and Values

FIGURE 5-20
Attributes and values.

All customers have these attributes:	Each customer has a value for each attribute:		
Customer ID	101	102	103
First name	John	Mary	Bill
Last name	Smith	Jones	Casper
Home phone	555-9182	423-1298	874-1297
Work phone	555-3425	423-3419	874-8546

Data Entities

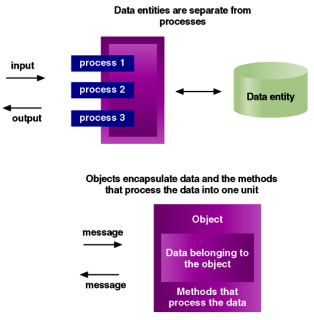
- Things system needs to store data about in traditional IS approach
- Modeled with entity-relationship diagram (ERD)
- Requirements model used to create the database design model for relational database

Objects

- Objects do the work in system and store information in object-oriented approach
- Objects have behaviors and attributes
  - Class: Type of thing
  - Object: Each specific thing
  - Methods: Behaviors of objects of the class
- Objects contain values for attributes and methods for operating on those attributes
- An object is encapsulated – a self-contained unit

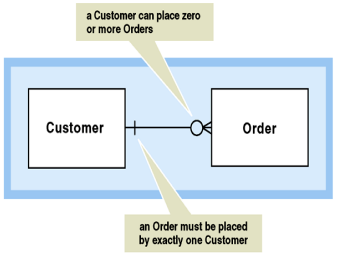
Data Entities Compared with Objects

FIGURE 5-21
Data entities compared with objects.



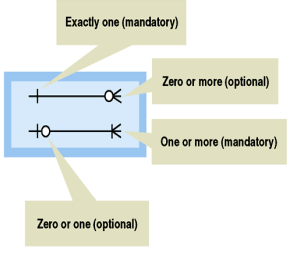
Simple Entity-relationship Diagram

FIGURE 5-22
A simple entity-relationship diagram.



Cardinality Symbols of Relationships

FIGURE 5-23
Cardinality symbols of relationships.



Expanded ERD with Attributes Shown

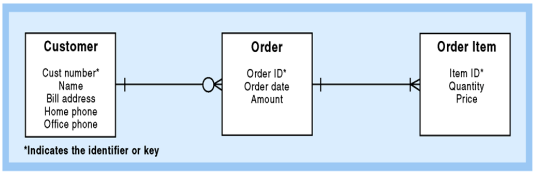
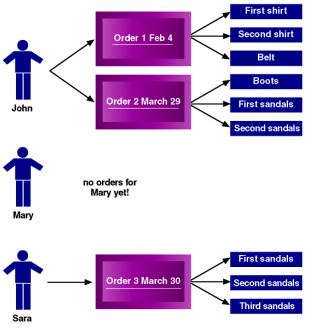


FIGURE 5-24
An expanded ERD with attributes shown.

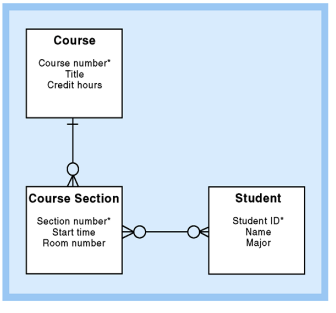
Customers, Orders, and Order Items

FIGURE 5-25
Customers, orders, and order items consistent with the expanded ERD.



University course enrollment ERD

FIGURE 5-26
A university course enrollment ERD with a many-to-many relationship.



Refined University course enrollment ERD

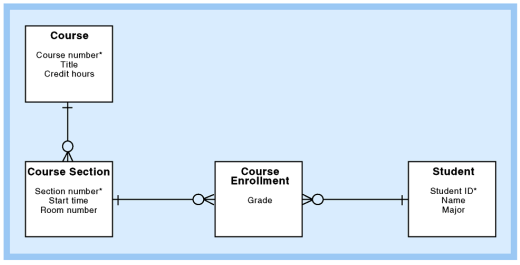


FIGURE 5-27
A refined university course enrollment ERD with an associative entity.

RMO Customer Support ERD

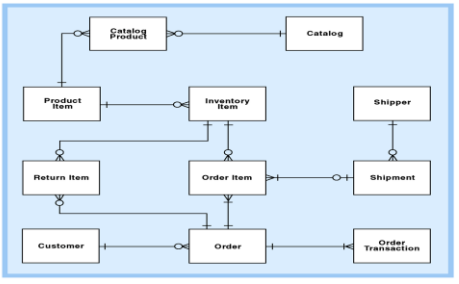


FIGURE 5-28
Rocky Mountain Outfitters customer support system entity-relationship diagram (ERD) without attributes.

The Class Diagram

- Models classes of objects instead of data entities
- Generalization/specialization hierarchies
  - General superclasses to specialized subclasses
  - Inheritance allows subclasses to share characteristics of their superclasses
- Aggregation (whole-part hierarchies)
  - Relates objects and its parts
  - Defines object in terms of its parts

A Generalization/Specialization Hierarchy for Motor Vehicles

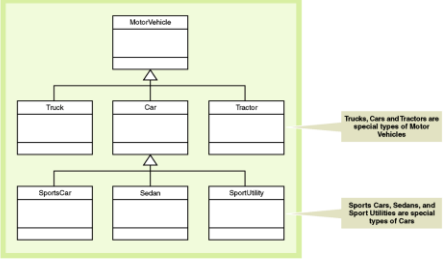
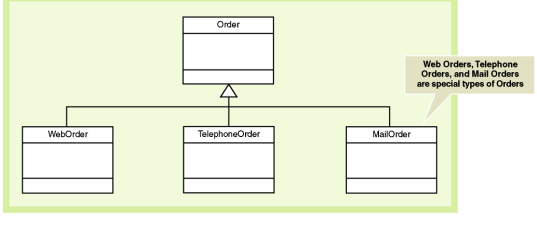


FIGURE 5-29
A generalization/specialization hierarchy for motor vehicles.

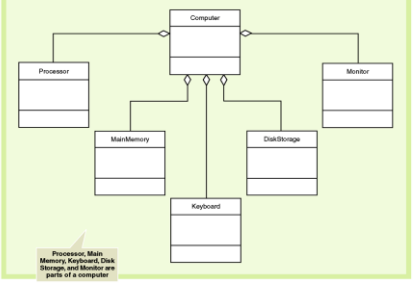
A Generalization/Specialization Hierarchy for Orders

FIGURE 5-30
A generalization/specialization hierarchy for orders.



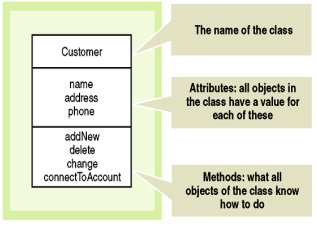
Aggregation or Whole-Part Relationships

FIGURE 5-31
Whole-part (aggregation) relationships between a computer and its parts.



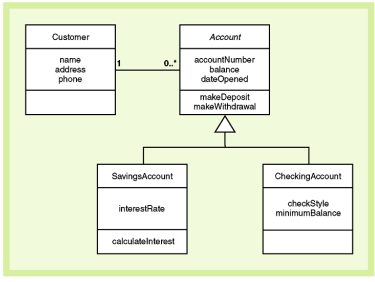
The Class Symbol for the Class Diagram

FIGURE 5-32
The class symbol with three sections for name, attributes, and methods.



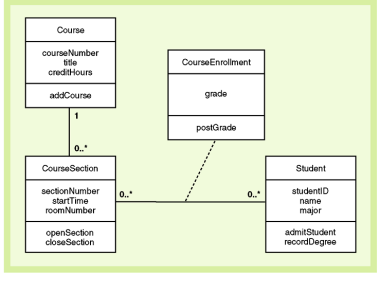
Bank Account System Class Diagram

FIGURE 5-33
A bank account system class diagram.



Enrollment Class Diagram with Association Class

FIGURE 5-34
University course enrollment class diagram with an association class.



RMO Class Diagram

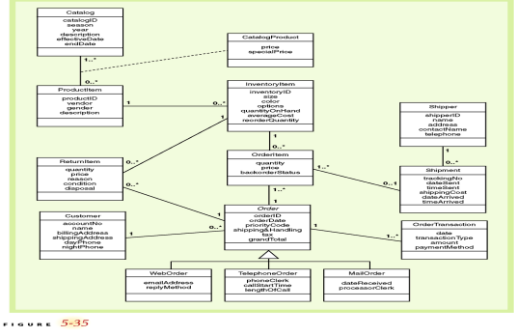
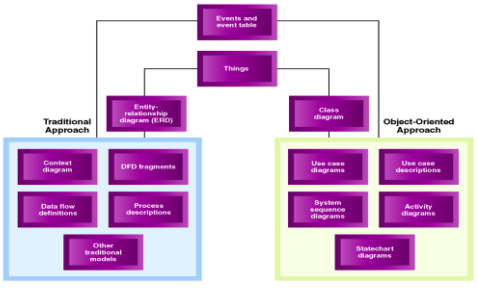


FIGURE 5-35
Rocky Mountain Outfitters class diagram.

Where You Are Headed

FIGURE 5-36
Requirements models for the traditional approach and the object-oriented approach.



Summary

- Analysis Phase: Define system requirements
- Models created to: further learning process, reduce complexity, communicate with team members, and document requirements
- Many types of models used:
  - Mathematical, descriptive, graphical
- Key early step in modeling to identify and list:
  - Events that require a response from system
  - Things users deal with in work environment

5

Summary (continued)

- Events are memorable, can be described, and occur at specific time and place
- External events occur outside system, triggered by someone interacting with system
- Temporal events occur at defined point in time, such as end of day or end of month
- State events based on internal system change
- Event table records event, trigger, source, activity or use case, response, and destination

5

Summary (continued)

- Things are what user deals with and system remembers, such as customer placing an order
- Traditional approach uses entity-relationship diagrams (ERD) for data entities, attributes of data entities, and relationships between entities
  - Things are shown as data entities
- Object-oriented approach uses class diagrams for classes, attributes, methods of class, and associations among classes
  - Things are shown as objects belonging to a class