Chapter 7
Domain Object Model

Domain Object Model (DOM)
- Domain
  - application area (a subsystem) of an enterprise information system
  - defines the scope of an application under development
- DOM is more than domain model in UP (class model for the domain)

Use cases and actors
- Use case represents a major piece of system functionality
- Actor
  - a role that somebody or something plays with regard to a use case
  - communicates with a use case (via «communicate» relationship) and expects from it some feedback – a value or observable result
- Multiple use case diagrams (multiple viewpoints) may be desirable:
  - no actors – use cases and their relationships
  - single actor’s viewpoint – all use cases for each important actor
- Use case diagrams visualize use cases, actors, and their relationships (but the usefulness of diagrams in a use case model are questionable)
- Use case specifications, which are text documents stored in a CASE tool repository, are the main power and benefit of use case modeling

Use case relationships
- «include» relationship - the execution of one use case enfolds (always) the functionality of the included use case
- «extend» relationship - the execution of one use case may need to be extended (sometimes) by the functionality of the extending use case

Use case diagram for CM
Alternative use case diagram for CM

Domain class model

Domain glossary

Classes and attributes

Class relationships
Summary

- **Domain Object Model (DOM)** is a model of one application area (domain) of an enterprise
- **Domain use case model** is a UML use case diagram and associated use case documents defined for a domain
- **Domain glossary** expands a business glossary by adding to it terms and definitions specific to the domain
- **Domain class model** defines classes within a domain and relationships between these classes
  - There are three main kinds of relationships: associations, aggregations, and generalizations