

Object Oriented Methodologies: Requirements Analysis

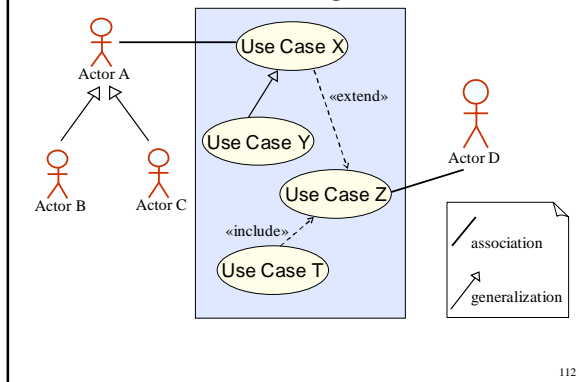
110

General Approach

- Requirements (Use Cases and Class/Resp. Collaboration)
- Determination of classes and objects
- Attributes and Operations are defined
- Hierarchies and structures to organize classes
- Relations among objects are defined
- Interaction among the objects are modeled
- All the models are refined through scenarios and use cases

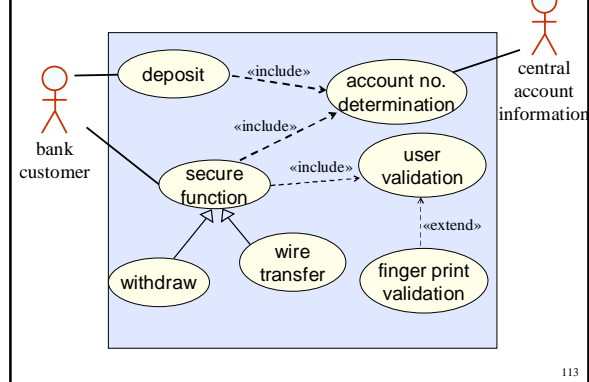
111

Use Case Diagrams



112

Example Use Case Diagram



113

Use Case Analysis notes

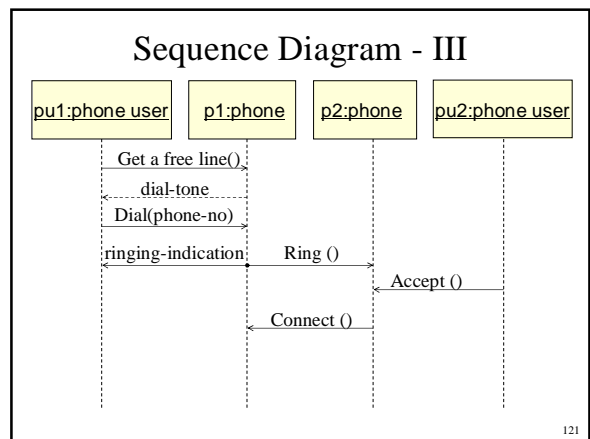
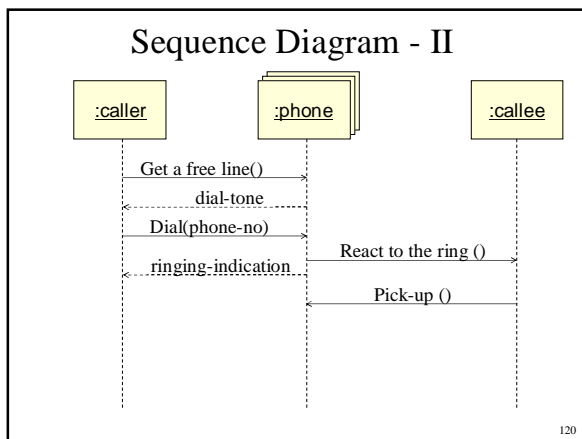
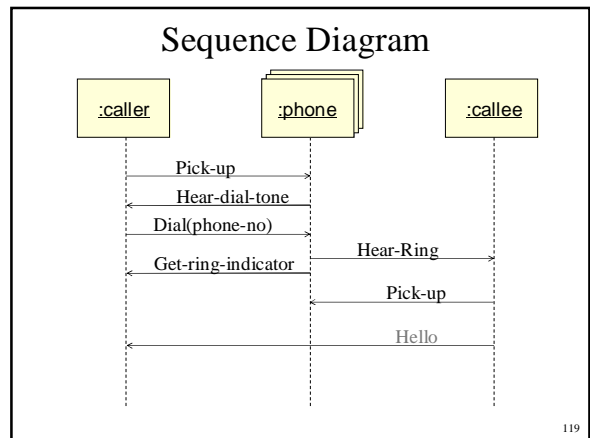
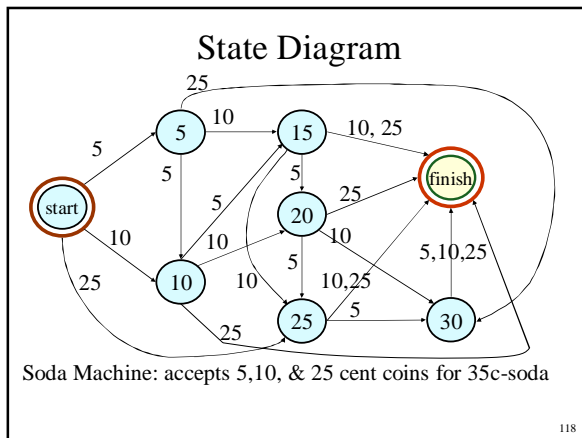
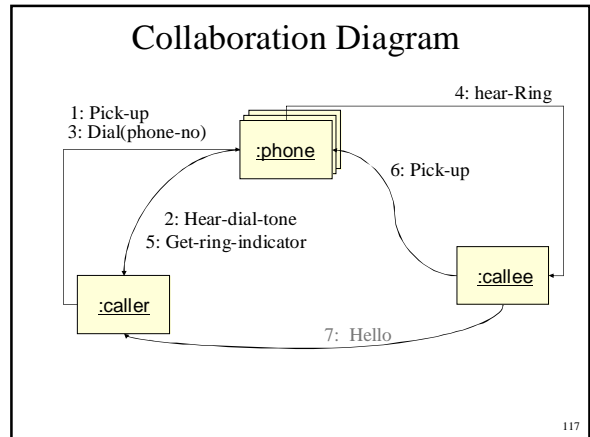
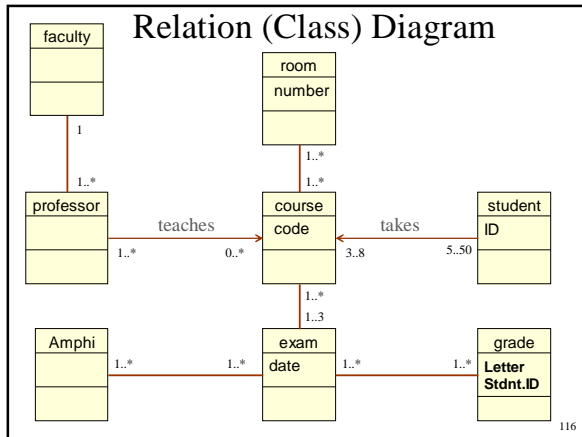
- One Use Case Diagram per capability
- One use case per system function
- Each use case explained through at least one scenario
- Each use case (or scenario) specified through one interaction diagram

114

Class / Responsibility / Collaboration

Class name :	
Class Type (tool, property, role, event ...) :	
Class Characteristics (tangible, atomic ...) :	
Responsibilities	Collaborating classes

115

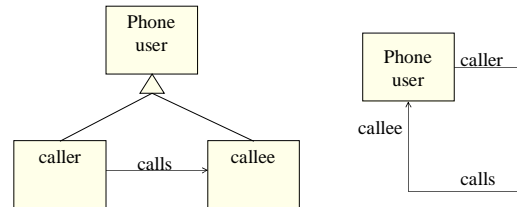


Message Numbering

- Hierarchical Numbering: 1.1, 1.5.2 ...
- Concurrent Messages: 2.a, 2.b ...
- Conditional messages: Guard: [a<8] 2.2
- No indication of reception time in Collaboration Diagrams.
- Different synchronization types are available

122

Class Diagram: inheritance/association



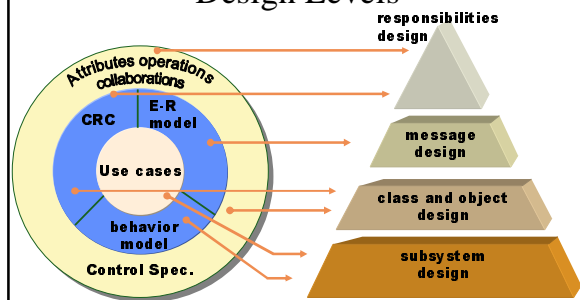
123

From Analysis to Design

- Similar Media are used
- The details introduced in the models, start matching the question: 'how'
- Classes are emphasized in Analysis, Objects in Design

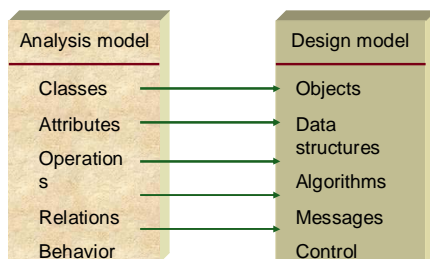
124

Design Levels



125

Transition to Design



126