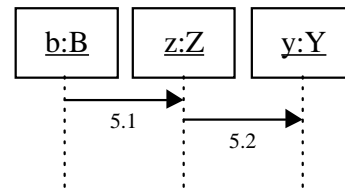
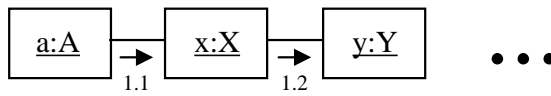
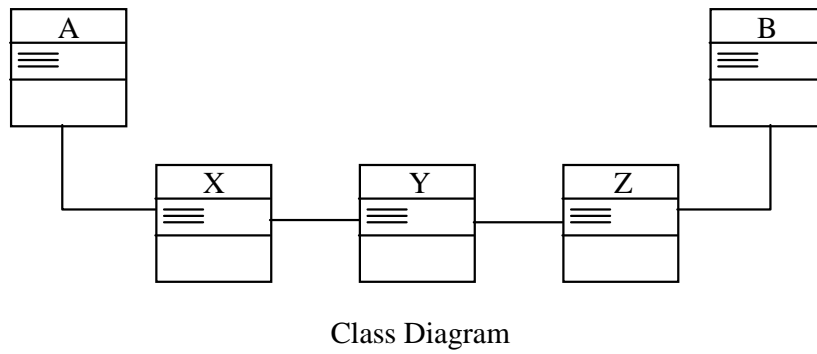
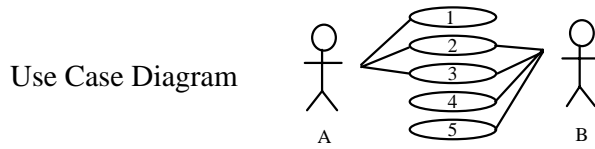


## Moving into Design

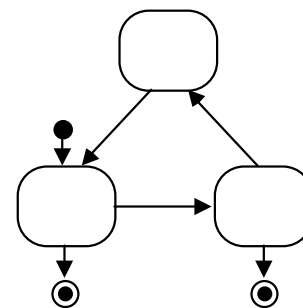
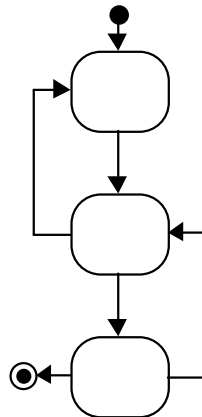
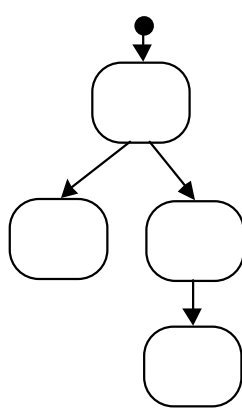
- UML for Analysis (Revisited)
- Design Versions of Analysis Diagrams
- UML for Design

# UML for Analysis (Revisited)



Collaboration Diagram for 1

Sequence Diagram for 5



Statechart for X

Statechart for Y

Statechart for Z

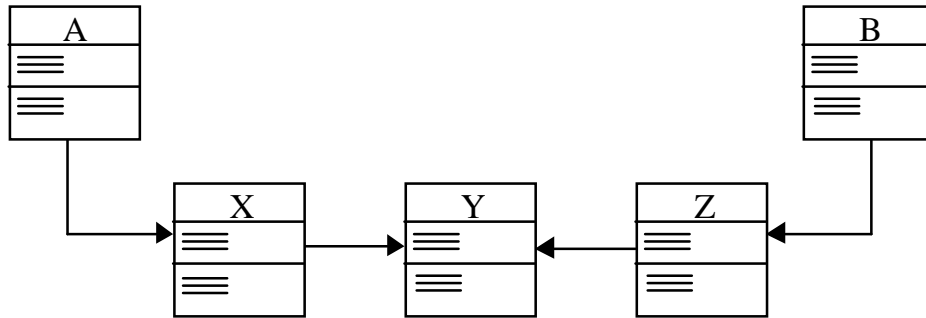
## Design Versions of Analysis Diagrams

- Design use case diagrams
  - Some modelers prefer to describe the technology-constrained dialog between the actor and the system
  - Some modelers prefer to include response time constraints, system management and configuration use cases
  
- Design class diagrams
  - Can include attribute and operation visibilities
    - \* “+” means public
    - \* “#” means protected
    - \* “-” means private
  - Class attributes and operations can be shown via underlines

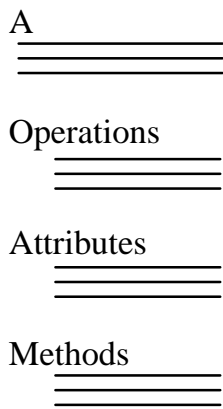
```
+size: area
#visible: boolean
+default-size: area
#maximum-size: area
-Xptr: XWindowPtr
```
  - Can show class dependencies instead of associations
  
- Design interaction diagram
  - Some modelers prefer to annotate the diagram with timing information

- It may be a good idea to stereotype or note the design version of the diagram as being <<design>> rather than <<analysis>>

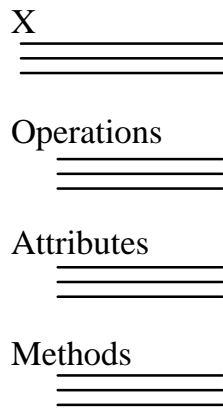
# UML for Design



Class Diagram

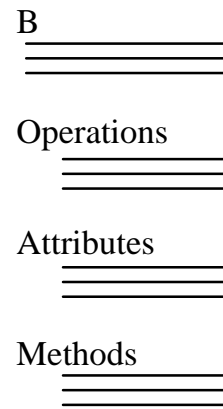


Detailed design  
for A



Detailed design  
for X

...



Detailed design  
for B

## Key Points

- Design-specific information can be added to the diagrams