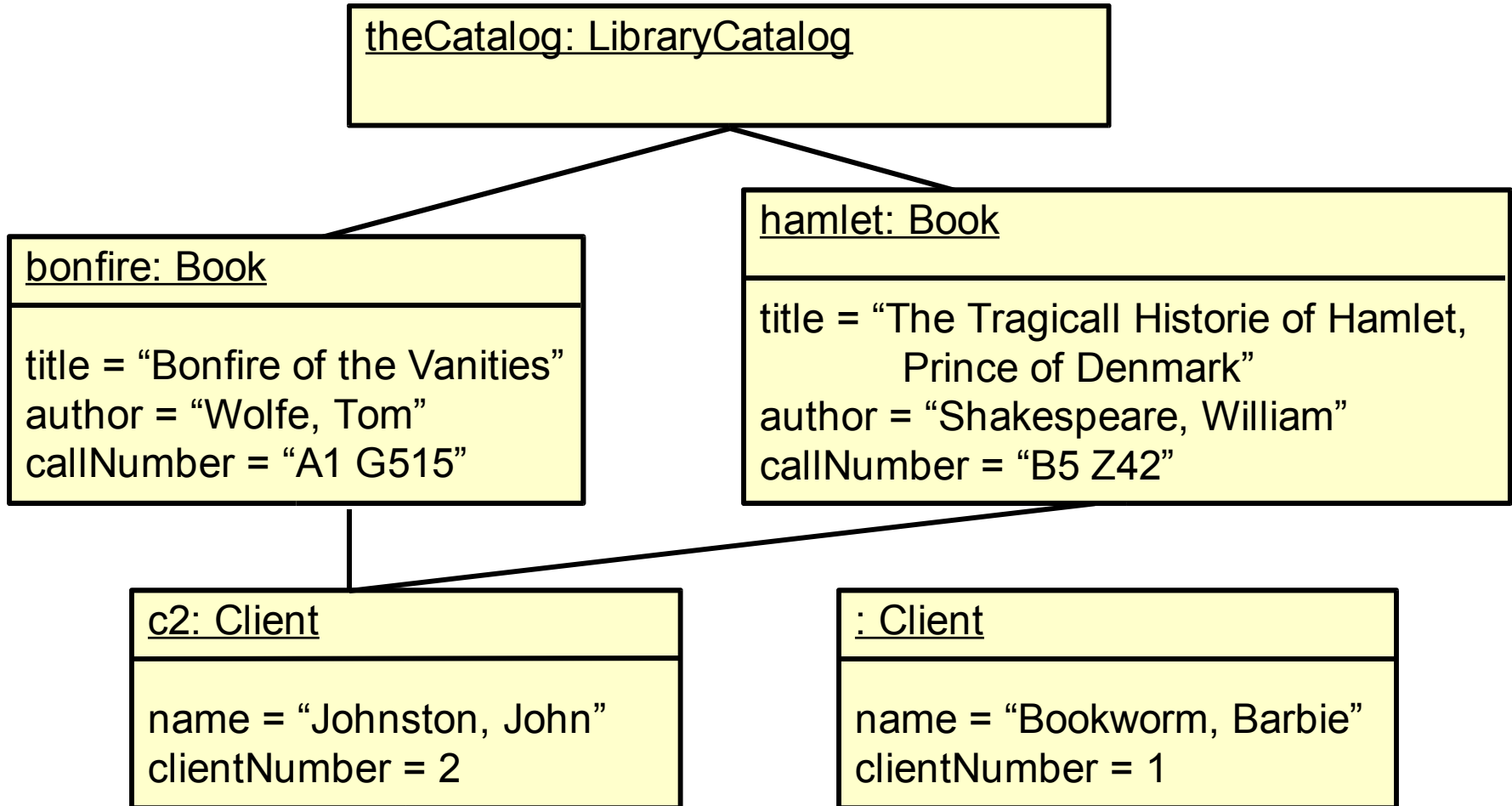


# Analysis Example: Library

- Library
  - has books
  - a book has an author and a title
  - has users
  - has a catalog
  - users are stored in a file
  - book has call number
- Also...
  - videos
  - audio CDs
  - users check in books, check out books
  - user can put hold on books

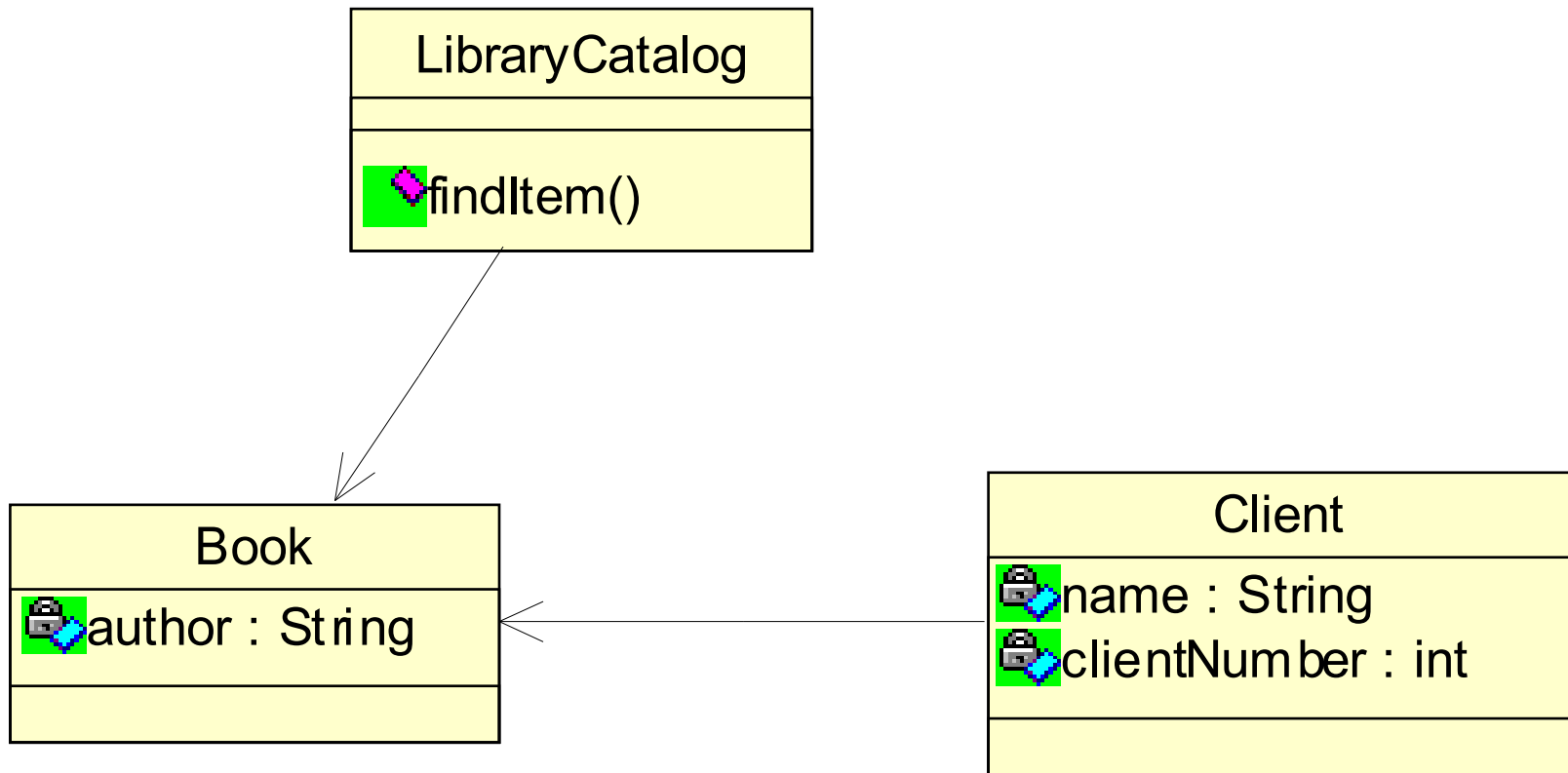
# Object Diagram



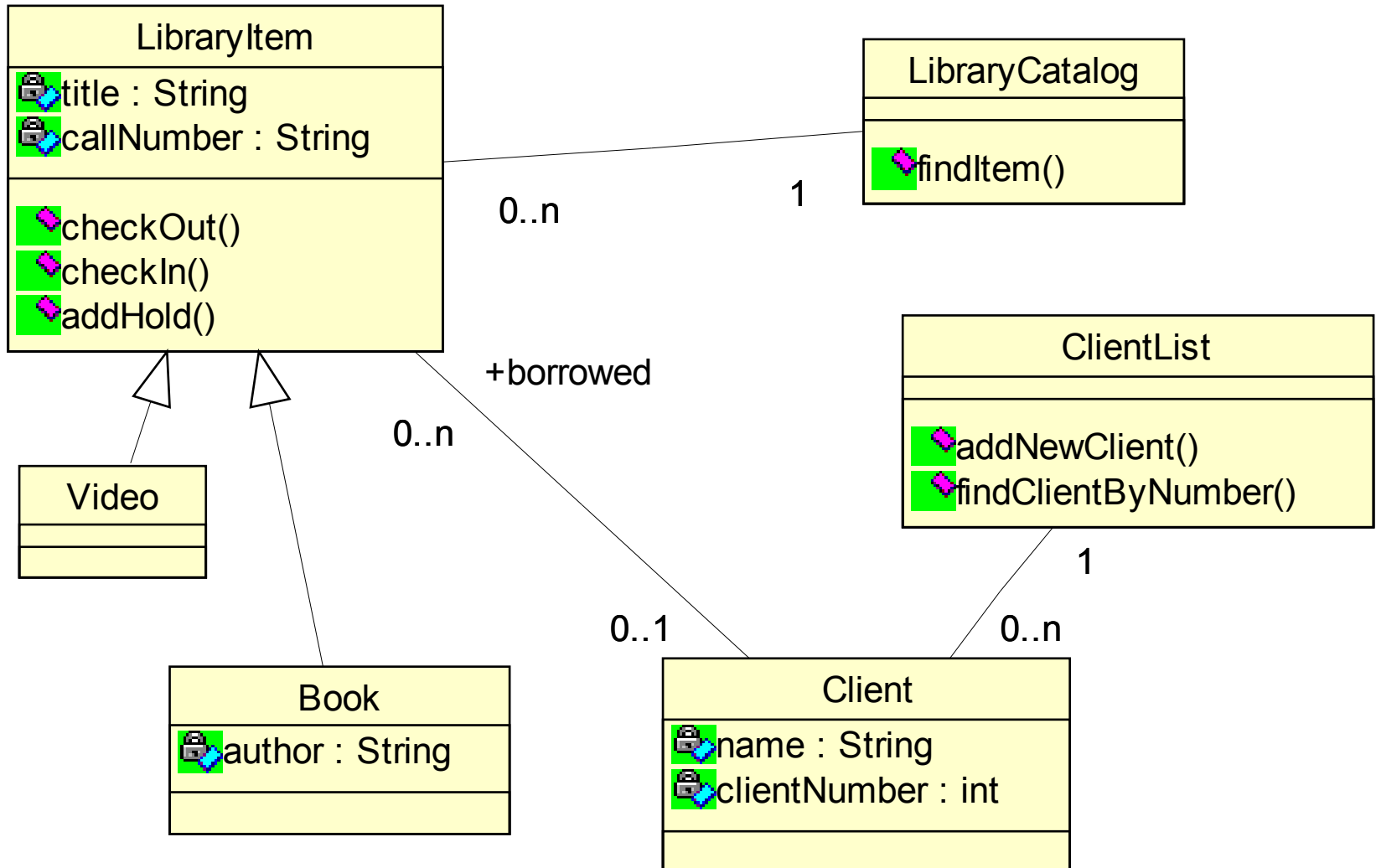
# Elements of UML Object Diagrams

- Objects
  - name: optional  
    anObject
  - class name: don't forget the colon before it  
    : Object
  - attributes (in Java: fields)  
    name = value
- Links
  - may be named
  - in Java: references
- Object diagram shows a set of objects at a given time
  - completeness impossible
- Later: interaction diagrams
  - more complex object diagrams

# Class Diagram



# Library Class Diagram – Refined



# Elements of UML Class Diagrams

- Classes
  - name
  - attributes (in Java: fields)
    - visibility, name, type
  - operations (in Java: methods)
    - visibility, name, return and parameter types
- Associations
  - in Java: references
  - in C++: pointers or references
  - at each end:
    - role name
    - multiplicity (0..1, 1, \*, 1..\*, and others)
- Generalizations
  - in Java and C++: inheritances

# Purpose of Class Diagrams

- Most important UML diagram
- Class diagrams can completely specify a system
- Used both in analysis and design
- Two different purposes:
  - conceptual model
    - used in analysis
    - shows little detail
    - real world
  - implementation model
    - used in design
    - shows detail
    - adapted to programming language, platform
- Can easily be mapped to code (and back)