

Object-Oriented Analysis

OOA consists of 3 *parallel* steps:

1. Use-case modeling

- Identify actors and use cases.
- Draw *use-case diagram* and *scenarios*.

2. Class modeling

- Determine classes, their attributes, and their interrelationships .
- Draw *class diagram*.

3. Dynamic modeling

- Determine the actions performed by or to each class.
- Draw *state diagram*.

Class Modeling

Objectives:

1. Determine the classes and their attributes.
2. Draw **class diagram**.
3. Determine the interrelationships between the classes.

Methods:

- [noun identification technique](#)
- [class, responsibility, collaboration \(CRC\) cards method](#)

Object-Oriented Analysis & Design using UML

- What kind of things are classes:
 - Tangible or 'real-world' things: book, copy, course
 - Roles: library member, student, director of studies
 - Events: arrival, leaving, request
 - Interactions: meetings, intersection
- The first two are much more common as sources of objects and classes than the last two.

Class Production Exercise

Sample System Requirements

Books and journals The library contains books and journals. It may have several copies of a given book. Some of the books are for short term loans only. All other books may be borrowed by any library member for three weeks. Members of the library can normally borrow up to six items at a time, but members of staff may borrow up to 12 items at one time. Only members of staff may borrow journals.

Borrowing The system must keep track of when books and journals are borrowed and returns enforcing the rules described above.

Noun Identification

1. Identify noun and noun phrases in a concise system requirements statement.
2. Discard those that are inappropriate as classes.

Sample System Requirements



Books and journals The library contains books and journals. It may have several copies of a given book. Some of the books are for short term loans only. All other books may be borrowed by any library member for three weeks. Members of the library can normally borrow up to six items at a time, but members of staff may borrow up to 12 items at one time. Only members of staff may borrow journals.

Borrowing The system must keep track of when books and journals are borrowed and returns enforcing the rules described above.

Object-Oriented Analysis & Design using UML

- Tips:

- you could experiment with keeping two lists one for strong candidates and one for weaker ones.
- The above list is not exhaustive so as you gain experience you can customise the list.

Noun Identification

Which kinds of things are good classes?

- **tangible or real-world things:** book, copy, course
- **roles:** library member, student, director of studies

Notes:

- The classes must provide **all** the **behavior** required by the system.
- The classes represent **enduring objects** in the real world.

Other less common classes:

- **events** and **operations** with states, behaviors, and identities:
arrival, leaving, request
- **interactions:** meeting, intersection

Note: Events, operations, and interactions may help to identify relationships between classes.

Object-Oriented Analysis & Design using UML

- Discard reasons:
 - redundant - the same class is given more than one name
 - vague - ambiguous noun (needs clarifying)
 - event/operation - something is done, Q; has an instance of the event or operation have state, behavior & identity. If not discard it.
 - Meta-language - where the noun is used in a definition e.g. requirements, system
 - Not within system scope - not inside the system
 - attribute - a simple aspect such as Name of a member (thus belongs to another class)

Noun Identification

Which nouns are inappropriate as classes?

- **redundant:** the same class is given more than one name
- **vague:** noun is ambiguous
- **stateless events or operations:** something done to, by, or in the system
- **meta-language:** part of the language of modeling, not part of the system
- **outside system scope:** refer to something outside of system
- **attribute:** simple thing that has no interesting behavior

Examples: Nouns that are discarded.

- library (outside system scope)
- short term loan (an event)
- member of library (redundant, same as library member)
- week (measures time, not a *thing*)
- item (vague)
- time (outside system scope)
- system (meta-language)
- rule (meta-language)

Noun Identification

Example: Nouns that are retained (first cut list of probably classes):

- book
- journal
- copy (of book)
- library member
- member of staff

Noun Identification

Schach introduces a 3-stage process:

1. Concise Problem Definition.

Define the product as briefly and concisely as possible, preferably in one sentence.

2. Informal Strategy.

Use constraints to express informal strategy, preferably in one paragraph.

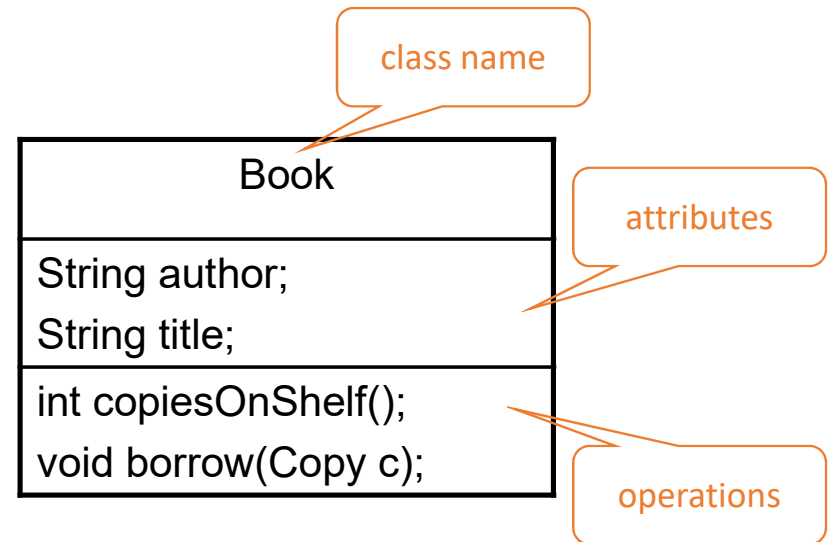
3. Formalize the Strategy.

Identify the nouns in the informal strategy. Use these nouns as candidate classes.

Attributes and Operations

- A class has attributes and operations.
- Attributes are data contained in a class.
- Operations (i.e., methods) defined the way classes interact (by sending messages).
- During OOA, focus more on attributes.

Example:



Associations Between Classes

Identify the relations or associations between classes.

Class A and class B are associated if

- a class A object sends a message to a class B object,
- a class A object creates a class B object,
- a class A object has a class B object as a component (composition),
- a class A object receives a message with a class B object as an argument,
- etc.

That is, there is something going on **between** the classes.

Object-Oriented Analysis & Design using UML

- Associations:
 - As classes correspond to nouns, associations correspond to verbs
 - Associations express the relationship between classes
 - Just as there are instances of classes (*objects*) there are instances of associations (*links* in UML)

Object-Oriented Analysis & Design using UML

- Association examples:
 - 'a library member borrows a book'
 - 'a person works for a company'
 - 'an account generates a statement'
 - 'a customer orders a product'
 - 'a tutor teaches a student'

Object-Oriented Analysis & Design using UML

- **Multiplicities:**
 - Is the number of objects of one class that relate to a single object of an associated class.
 - You can specify:
 - an exact number
 - a range of numbers
 - an arbitrary, unspecified number

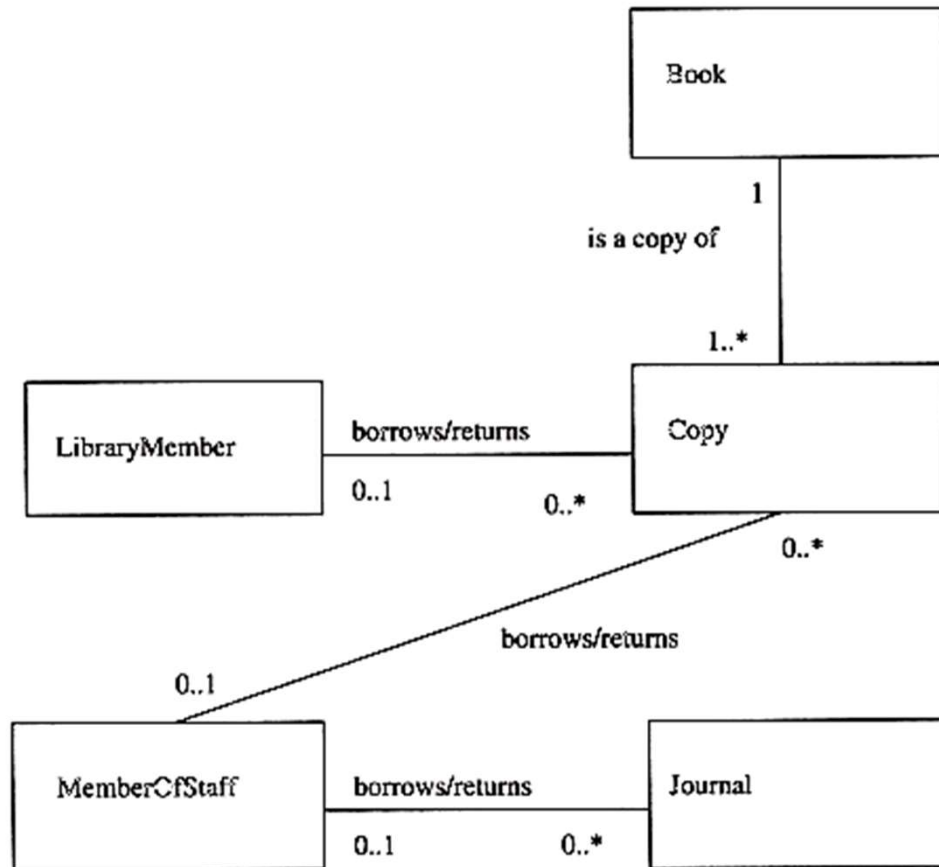
Associations Between Classes

Example:

- is a copy of
- borrows/returns

Multiplicities

- 1 one
- 0..1 0 or 1
- 0..* 0 or many
- 1..* 1 or many



Associations Between Classes

- MemberOfStaff shares all the associations of LibraryMember.
- MemberOfStaff is a special kind of LibraryMember.

