Class modeling (part 2)

Fabrizio Maria Maggi
Institute of Computer Science
Supplementary Material about Java

Suggested:  http://www.learnjavaonline.org/

A more structured course:
https://www.udacity.com/course/java-programming-basics--ud282
https://www.udacity.com/course/Object-Oriented-Programming-in-Java

Other material:
https://www.sololearn.com/Course/Java
Software Development Methodology

Domain (Class) Model → Interaction Modelling → Application (Class) Model

Code Generation
Software Development Methodology

Domain Classes; Attributes; Relations

Domain (Class) Model

Interaction Modelling

Application (Class) Model

Code Generation

Domain (Class) Model → Interaction Modelling → Application (Class) Model
Domain (Class) Model
## Domain (Class) Model

<table>
<thead>
<tr>
<th>Domain Entities</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image-url" alt="Image of domain entities" /></td>
<td><img src="image-url" alt="Image of domain entities" /></td>
</tr>
</tbody>
</table>

---

5 Systems modelling – Fabrizio Maria Maggi
Domain (Class) Model

WHAT?
To answer this question, the domain model provides classes with attributes and relations among them.

- Operations are **not** specified
How to create Domain Models
Class Modelling

- **Classes**
  - A class describes a group of objects with the same properties (attributes), behavior (operations), kinds of relationships and semantics
  - Classes often appear as nouns in problem descriptions with users

- **Objects**
  - An object is a concept, abstraction or thing with identity that has a meaning for an application
  - An object is an instance of a class
Class Diagrams

- **Class**
  - UML notation: box with a class name

- **Object**
  - UML notation: box with an object name followed by a colon and a class name. The object name and the class name are both underlined
Exercise 1: Domain Entities?

- Each company has a name. A company consists of departments. Each department has a name and is located in one or more offices. Each office is in a certain address. Each department has a manager and a set of employees. Each employee has a name and a title.
Exercise 1: Domain Entities?

- Each company has a name. A company consists of departments. Each department has a name and is located in one or more offices. Each office is in a certain address. Each department has a manager and a set of employees. Each employee has a name and a title.
Attributes and Values

- **Attribute**
  - An attribute is a named property of a class that describes a value held by each object of the class.
  - UML notation: attributes are listed in the second compartment of the class box. Optional details, such as type and default value, may follow each attribute.

- **Value**
  - A value is a piece of data.
  - UML notation: values are listed in the second compartment of the object box.
Exercise 1: Attributes?

- Each company has a name. A company consists of departments. Each department has a name and is located in one or more offices. Each office is in a certain address. Each department has a manager and a set of employees. Each employee has a name and a title.
Exercise 1: Attributes?

- Each company has a name. A company consists of departments. Each department has a name and is located in one or more offices. Each office is in a certain address. Each department has a manager and a set of employees. Each employee has a name and a title.

Characteristics of Domain Entities
Operations and Methods

- **Operation**
  - An operation is a function or procedure that may be applied by or to objects in a class.
  - UML notation: operations are listed in the third compartment of the class box.

- **Method**
  - A method is the implementation of an operation for a class.
  - A domain model is static and does not contain behavioral information.
(Binary) Links and Associations

- **Link**
  - A link is a physical or conceptual connection among objects
  - UML notation: line between objects. A link can have a name (underlined)

- **Association**
  - An association is a description of a group of links with common structure and common semantics
  - UML notation: line between classes. An association can have a name (not underlined)
Multiplicity

- Specifies the number of instances of one class that may relate to a single instance of an associated class
- UML notation: specified at the end of the association lines
  - Examples: “1” (exactly one); “3..5” (three to five, inclusive); “*” (many, zero or more)
Multiplicity many-to-many

Person
- name

Company
- name

John : Person
- = GE
- name = "John"

Mary : Person
- = GE
- name = "Mary"

Sue : Person
- = GE, IBM
- name = "Sue"

Alice : Person
- = IBM
- name = "Alice"

Jeff : Person
- name = "Jeff"

GE : Company
- = John, Mary, Sue

IBM : Company
- = Sue, Alice
Multiplicity one-to-one

Country
name

HasCapital
1

CapitalCity
name

Canada : Country
= Ottawa
name = "Canada"

Ottawa : CapitalCity
= Canada
name = "Ottawa"

France : Country
= Paris
name = "France"

Paris : CapitalCity
= France
name = "Paris"

Senegal : Country
= Dakar
name = "Senegal"

Dakar : CapitalCity
= Senegal
name = "Dakar"
Association end names

- Association ends can be provided with a name as well as with a multiplicity
Association end names

- Association end names are necessary for associations between two objects of the same class. They can also distinguish multiple associations between a pair of classes.
- Association end names as pseudo attributes
Associations as References

Association end name
MagicDraw might not produce the Collection type declaration. For example, it might generate the code "Course takes;" instead of "Set<Course> takes;". If you are having this issue, you can go back to the class diagram, double click on the association end "takes", then select "Language Properties" and then "Container". There you can select the type java.util.Set<type> (or any other container type that you prefer).
A constraint is a condition involving model elements, such as objects, classes, attributes, links, associations.

- A constraint specifies limitations that implementers need to satisfy.
Constraints

<table>
<thead>
<tr>
<th>Collection Type</th>
<th>isOrdered</th>
<th>isUnique</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multiset, bag</td>
<td>false</td>
<td>false</td>
</tr>
<tr>
<td>Sequence, array</td>
<td>true</td>
<td>false</td>
</tr>
<tr>
<td>Set</td>
<td>false</td>
<td>true</td>
</tr>
<tr>
<td>Ordered set</td>
<td>true</td>
<td>true</td>
</tr>
</tbody>
</table>
Generalization and Inheritance

- **Generalization** is the relationship between a class (superclass) and one or more variations of the class (subclasses)
  - The superclass holds common attributes, operations and associations. The subclasses add specific attributes, operations and associations (each subclass is said to inherit the features of its superclass)
  - Simple generalization organizes classes into a hierarchy
  - There can be multiple levels of generalizations
  - A large arrowhead denotes generalization. The arrowhead points to the superclass

- **A generalization set name** is an enumerated attribute that indicates which aspect of an object is being abstracted by a particular generalization
Generalization and Inheritance

Diagram
- name

Figure
- penThickness
- color
- centerPosition
- penType
- move()
- select()
- rotate()
- display()

ZeroDimensional
- Point
  - display()

OneDimensional
- orientation
- scale()

TwoDimensional
- orientation
- fillType
- scale()
- fill()

Line
- endPoints
- display()

Arc
- radius
- startAngle
- arcAngle
- display()

Spline
- controlPoints
- display()

Polygon
- numOfSides
- vertices
- display()

Circle
- diameter
- display()
- rotate()
Packages

- You can fit a class model on a single page for many small and medium-sized problems
  - However it is often difficult to grasp the entirety of a large model.
- A package is a group of elements (classes, associations, and nested packages) with a common theme.
  - A package partitions a model making it easier to understand and manage,
- The UML notation for a package is a box with a tab:
  - The tab suggests the enclosed content, like a tabbed folder.
Exercise 1: Relations?

- Each company has a name. A company consists of departments. Each department has a name and is located in one or more offices. Each office is in a certain address. Each department has a manager and a set of employees. Each employee has a name and a title.
Exercise 2: Domain Entities?

- A school has a name and can have many students. A student is associated to a name, a student ID and a date of birth. Each student has to take a course but one student can take at most 6 courses. Each course has a title. For a course there is at least one student in the school who has taken the course. Each course has one professor. Each professor has a name and a room number. A professor can teach several courses.
Exercise 2: Attributes?

- A school has a name and can have many students. A student is associated to a name, a student ID and a date of birth. Each student has to take a course but one student can take at most 6 courses. Each course has a title. For a course there is at least one student in the school who has taken the course. Each course has one professor. Each professor has a name and a room number. A professor can teach several courses.
Exercise 2: Relations?

- A school has a name and can have many students. A student is associated to a name, a student ID and a date of birth. Each student has to take a course but one student can take at most 6 courses. Each course has a title. For a course there is at least one student in the school who has taken the course. Each course has one professor. Each professor has a name and a room number. A professor can teach several courses.
The cinema booking system should store seat bookings for multiple theatres. A theatre has a name and an address. Each theatre has seats arranged in rows. Customers can reserve seats and are given a row number and seat number. They may request bookings of several seats. Each booking is for a particular show (i.e., the screening of a given movie at a certain time). Shows are at an assigned date and time, and scheduled in a theatre where they are screened. The system stores the customers telephone numbers.
Exercise 3: Attributes?

- The cinema booking system should store seat bookings for multiple theatres. A theatre has a name and an address. Each theatre has seats arranged in rows. Customers can reserve seats and are given a row number and seat number. They may request bookings of several adjacent seats. Each booking is for a particular show (i.e., the screening of a given movie at a certain time). Shows are at an assigned date and time, and scheduled in a theatre where they are screened. The system stores the customers telephone numbers.
Exercise 3: Relations?

- The cinema booking system should store seat bookings for multiple theatres. A theatre has a name and an address. Each theatre has seats arranged in rows. Customers can reserve seats and are given a row number and seat number. They may request bookings of several adjacent seats. Each booking is for a particular show (i.e., the screening of a given movie at a certain time). Shows are at an assigned date and time, and scheduled in a theatre where they are screened. The system stores the customers telephone numbers.
Exercise 4: Domain Entities?

- A paper reviewing system has several conferences. Each conference has a title and a year and is managed by a chair and a list of committee members. Committee members and chairs must be assigned to one, but possibly more conferences. They have a name and an affiliation. A conference has several submitted papers, but a paper can be submitted to only one conference. A paper is assigned to 3 reviewers taken from the committee members. A paper can be accepted rejected or under review. We also know the paper titles and list of authors with their names and affiliations.
Exercise 4: Attributes?

- A paper reviewing system has several conferences. Each conference has a title and a year and is managed by a chair and a list of committee members. Committee members and chairs must be assigned to one, but possibly more conferences. They have a name and an affiliation. A conference has several submitted papers, but a paper can be submitted to only one conference. A paper is assigned to 3 reviewers taken from the committee members. A paper can be accepted rejected or under review. We also know the paper titles and list of authors with their names and affiliations.
Exercise 4: Relations?

- A paper reviewing system has several conferences. Each conference has a title and a year and is managed by a chair and a list of committee members. Committee members and chairs must be assigned to one, but possibly more conferences. They have a name and an affiliation. A conference has several submitted papers, but a paper can be submitted to only one conference. A paper is assigned to 3 reviewers taken from the committee members. A paper can be accepted rejected or under review. We also know the paper titles and list of authors with their names and affiliations.
A library system is used to manage physical items borrowings in different libraries. Each library has a name and a digital catalog of library items – that can be of type book, CD or DVD. Each item in the catalog is identified by an ID. Items are barcoded. The purpose of barcoding is to link the barcoded physical item to the electronic record in the catalog. The barcode contains the item's ID, title and type. The system also contains a list of borrowers with their names, addresses and phone numbers. Each borrower can borrow several library items. Each borrowing has a borrowing date and a returning date.
Exercise 5: Attributes?

- A library system is used to manage physical items borrowings in different libraries. Each library has a name and a digital catalog of library items – that can be of type book, CD or DVD. Each item in the catalog is identified by an ID. Items are barcoded. The purpose of barcoding is to link the barcoded physical item to the electronic record in the catalog. The barcode contains the item's ID, title and type. The system also contains a list of borrowers with their names, addresses and phone numbers. Each borrower can borrow several library items. Each borrowing has a borrowing date and a returning date.
Exercise 5: Relations?

- A library system is used to manage physical items borrowings in different libraries. Each library has a name and a digital catalog of library items – that can be of type book, CD or DVD. Each item in the catalog is identified by an ID. Items are barcoded. The purpose of barcoding is to link the barcoded physical item to the electronic record in the catalog. The barcode contains the item's ID, title and type. The system also contains a list of borrowers with their names, addresses and phone numbers. Each borrower can borrow several library items. Each borrowing has a borrowing date and a returning date.
Exercise 6: Domain Entities?

- The customers use banking application of a bank to manage current and saving accounts of customers. Each account has an account number and an opening date. We assume that a customer at least has a current account. No customer can have more than one account of the same type. The bank also has a manager who has administrator privileges. In the application, a customer or a manager is represented by a user profile that contains the name, date of birth, username, password and role. The role can be either customer or manager. A manager cannot have any account.
Exercise 6: Attributes?

- The customers use banking application of a bank to manage current and saving accounts of customers. Each account has an account number and an opening date. We assume that a customer at least has a current account. No customer can have more than one account of the same type. The bank also has a manager who has administrator privileges. In the application, a customer or a manager is represented by a user profile that contains the name, date of birth, username, password and role. The role can be either customer or manager. A manager cannot have any account.
Exercise 6: Relations?

- The customers use banking application of a bank to manage current and saving accounts of customers. Each account has an account number and an opening date. We assume that a customer at least has a current account. No customer can have more than one account of the same type. The bank also has a manager who has administrator privileges. In the application, a customer or a manager is represented by a user profile that contains the name, date of birth, username, password and role. The role can be either customer or manager. A manager cannot have any account.