Tutorial 5: UML Class Diagram

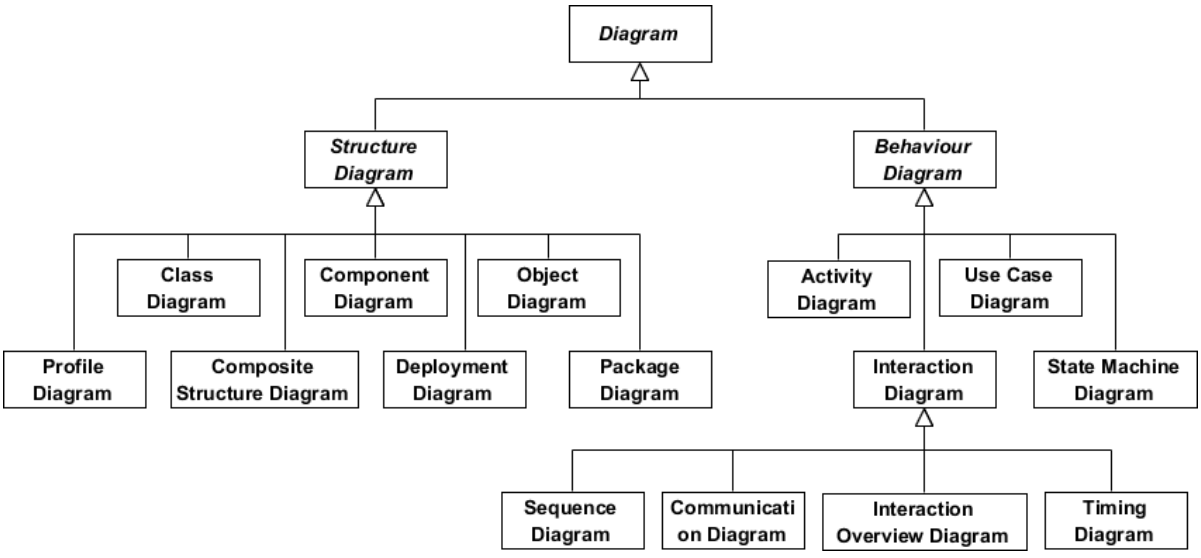
QUICK REVIEW

UML, short for Unified Modelling Language, is a standardized modelling language consisting of an integrated set of diagrams, developed to help system and software developers for specifying, visualizing, constructing, and documenting the artifacts of software systems, as well as for business modelling and other non-software systems. The UML represents a collection of best engineering practices that have proven successful in the modelling of large and complex systems. The UML is a very important part of developing object-oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects. Using the UML helps project teams communicate, explore potential designs, and validate the architectural design of the software.

The first thing to notice about the UML is that there are a lot of different diagrams (models) to get used to. The reason for this is that it is possible to look at a system from many different viewpoints. A software development will have many stakeholders playing a part. For Example:

- Analysts
- Designers
- Coders
- Testers
- QA
- The Customer
- Technical Authors

All of these people are interested in different aspects of the system, and each of them require a different level of detail. For example, a coder needs to understand the design of the system and be able to convert the design to a low-level code. By contrast, a technical writer is interested in the behaviour of the system as a whole, and needs to understand how the product functions. The UML attempts to provide a language so expressive that all stakeholders can benefit from at least one UML diagram.



Structure diagrams show the static structure of the system and its parts on different abstraction and implementation levels and how they are related to each other. The elements in a structure diagram represent the meaningful concepts of a system, and may include abstract, real world and implementation concepts. There are seven types of structure diagram as shown in the figure.

Behaviour diagrams show the dynamic behaviour of the objects in a system, which can be described as a series of changes to the system over time. There are seven types of behaviour diagrams as shown in the figure.

LAB PRACTICES

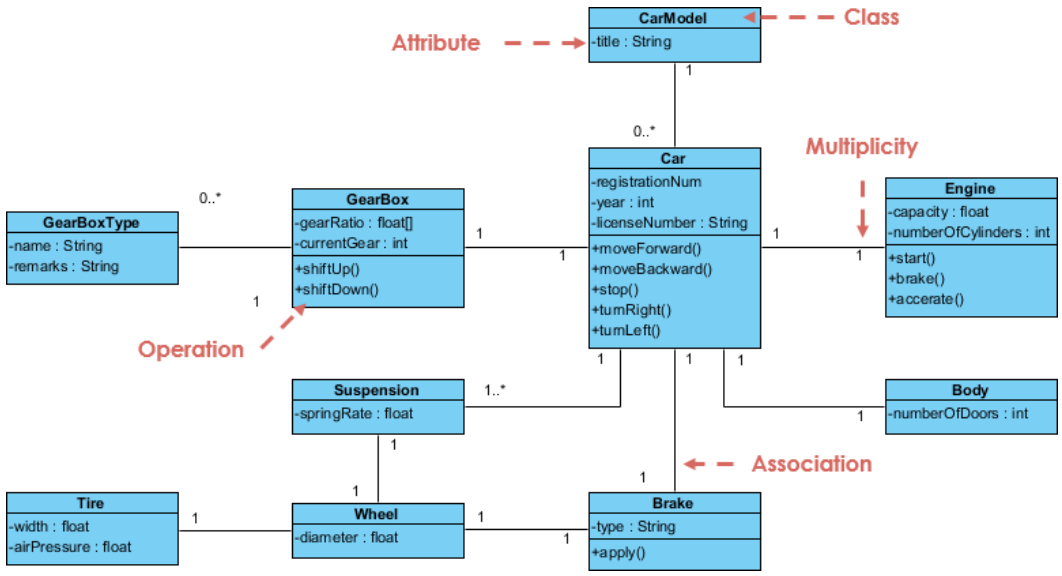
T5.1. Astah Professional

Astah Professional is a system design tool that supports UML (Unified Modelling Language) 2.x (partly), UML1.4, Flowchart, Data Flow Diagram, ER Diagram, CRUD, Requirement Diagram, Requirement Table, Traceability Map and Mind Map.
There are online resources for learning UML and Astah Professional:

- Astah Pro & UML User Guide
- UML History and Myths (Video)
- Class Diagrams in Astah
- UML Class Diagram – Basic (Video)

T5.1.1. Car Class warm-up ~ 10 min.

A car consists of different structural components, such as the engine, body, suspension, gearbox, etc. Each component in turn contains its own attributes and operations. For example, the engine has its capacity, and it can be started or stopped. We can represent the car with a class diagram. The figure below shows a simplified structural model of a car in a class diagram.
As a warm-up, use Astah Professional to create the class diagram shown in the figure below.



T5.2. Class Diagram

The class diagram is the main building block of object-oriented modelling. It is used for general conceptual modelling of the structure of the application, and for detailed modelling translating the models into programming code. Class diagrams can also be used for data modelling. The classes in a class diagram represent both the main elements, interactions in the application, and the classes to be programmed.
In the diagram, classes are represented with boxes that contain three compartments:

- The top compartment contains the name of the class.
- The middle compartment contains the attributes of the class.
- The bottom compartment contains the operations the class can execute.

In the design of a system, a number of classes are identified and grouped together in a class diagram that helps to determine the static relations between them. With detailed modelling, the classes of the conceptual design are often split into a number of subclasses.

T5.2.1. University

~ 20 min.

Read the raw description below, and draw a class diagram for a glossary based on the given information.

In a university there are different classrooms, offices, and departments.

- A department has a name and it contains many offices.
- Offices and classrooms have a number ID, and a classroom has a number of seats.

A person working at the university has a unique ID and can be a professor or an employee.

- A professor can be a full, associate, or assistant professor and he/she is enrolled in one department.
- Every Employee works in an office

You may follow the steps below:

1. Read, and underline nouns – nouns are candidates for classes (but also for attributes).
2. Draw classes.
3. Define attributes for classes.
4. Define relationships (look for relationships in verbs).
5. Define multiplicity.

T5.2.2. Airline

~ 20 min.

Read the raw description below, and draw a class diagram to model a system for management of flights and pilots.

An airline operates flights.

- Each airline has an ID.
- Each flight has an ID, a departure airport, and an arrival airport.
  - An airport has a unique identifier.
- Each flight has a pilot and a co-pilot, and it uses an aircraft of a certain type.
- A flight has also a departure time and an arrival time.

An airline owns a set of aircrafts of different types.

- An aircraft can be in a working state or it can be under repair, and in a particular moment an aircraft can be landed or airborne.

A company has a set of pilots.

- each pilot has an experience level: 1 is minimum, 3 is maximum.
- A type of aircraft may need a particular number of pilots, with different roles (illustrated in the table below). There must be at least one captain and one co-pilot, and a captain must have a level 3.

Aircraft Descriptor	Captain	Co-Pilot	Navigator
B737	Yes, experience level 3	Yes, experience level 2	No
A318	Yes, experience level 3	Yes, experience level 2	No
B747	Yes, experience level 3	Yes, experience level 3	Yes, experience level 2