Describing an Object Oriented System

UML

Unified Modeling Language

- Standard Notation for Object Oriented Systems
- There have been many different approaches to describe a Object Oriented System
  - UML attempted to combine these approaches -Hence the term “Unified”

Unified Modeling Language

- Uses diagram (pictorial) notation for documenting classes, objects, and packages

➡ All these items make up an Object Oriented System

UML Diagrams

- Structure Diagrams - Show the static architecture of the system.
- Behavior Diagrams - Show the behavior of a system or the system in process.
- Interaction Diagrams - Show the methods, interactions, and activities of the objects.

UML Structure Diagrams

They can be:

- Class Diagrams: Show classes, their fields and methods
- Composite Structure Diagrams: Used to have a means of representing the details of a class
- Component Diagrams: Software routines that satisfy certain functional requirements specified by interfaces - show the details of the components

UML Structure Diagrams

- Deployment Diagrams: Shows the assignment of executable files on the computing elements and the communication between entities
- Object Diagrams: Shows how objects are related and used at run-time.
- Package Diagrams: Class can be grouped into packages. These diagrams show packages and dependencies among them. Will a change in one package affect another package(s)?

An Example

UML Class Diagram