



ICS 3105

OBJECT ORIENTED SOFTWARE ENGINEERING

CHAPTER 4

Unified Modeling Language



Learning Outcomes

- By the end of this chapter, the learner should be able to:
 - Describe history of UML.
 - Discuss features of UML.
 - Analyze advantages of UML.
 - List UML tools.



What is UML?

- The Unified Modeling Language (UML) is a **standard graphical language** for **modeling** object-oriented software.
- It was developed in the mid-1990s as a collaborative effort by James Rumbaugh, Grady Booch and Ivar Jacobson, each of whom had developed their own notation in the early 1990s.



UML

- The 'U' in UML stands for '**unified**', since its three developers **combined** the best features of the languages they had each previously developed.
- The custodian of the UML standard is the Object Management Group (OMG).



UML diagram types

- UML contains a variety of diagram types
 - Class diagrams, which describe classes and their relationships.
 - Interaction diagrams, which show the behavior of systems in terms of how objects interact with each other.
 - two types of interaction diagrams: sequence diagrams and communication diagrams.



UML Diagrams

- State diagrams and activity diagrams, which show how systems behave.
- Component and deployment diagrams, which show how the various components of systems are arranged logically and physically.



UML Features

- UML, is much more than just a set of notations for drawing diagrams and it has the following additional interesting features:
 1. The diagrams you create with UML are intended to be interconnected to form a unified model.



UML Features

2. UML has a **detailed semantics**, describing **mathematically** the meaning of many aspects of its notations.
3. UML has **extension mechanisms**, which allow software designers to represent concepts that are **not part of the core** of UML.



UML Features

4. UML has an associated textual language called Object Constraint Language(OCL) that allows you to formally state various facts about the elements of the diagrams.



UML Objective

- The objective of UML is to **assist** in software development.
- It is not a methodology, because it does not describe, in a step-by-step way, how to do things.



Why use a standard modeling language?

- Some developers have been successful at developing small software systems without the use of diagrams or other features of modeling languages.
- However, as their systems become larger and larger, such developers have an increasingly difficult time seeing the 'big picture' and are liable to create poor designs and take much longer in their work.



Why use a standard modeling language?

- Most systems are therefore **documented with the use of diagrams**.
- These diagrams provide **views** of structure and functionality that **would be difficult to grasp** by looking at code or textual descriptions alone.
- In other words, diagrams provide **abstraction**.



Why use a standard modeling language?

- The object-oriented paradigm uses modeling throughout.
- A model is a set of UML diagrams that represent one or more aspects of the software product to be developed.
- UML is the tool that is used to represent (model) the target software product.



Why use a standard modeling language?

- A model goes beyond a mere set of diagrams.
- A model captures an interrelated set of information about the system: a diagram is simply one view of that information.



Why use a standard modeling language?

- Several diagrams can present the same information in slightly different ways, either with different notations or with different levels of detail.



Why use a standard modeling language?

- A software engineer can delete an element from a diagram, and keep it in the model; if he/she delete an element from the model it should **disappear** from all diagrams.



Why use a standard modeling language?

- A model can lead software engineers to have insights about the system; they can analyze the model (manually or using tools) to discover problems and other properties of it.
- Simple diagrams generated from the model can also help communicate with clients and users.



Why use a standard modeling language?

- However, it is **up to the modeler** to generate these easy-to-understand views.
- Employing UML, a well-defined standard modeling language, adds additional advantages.



Advantages of UML

- 1) Since it is a standard notation, **everybody who looks at the model will be able to interpret it the same way.**
- 2) There is a **wide variety of tools available** to build UML models and to enable simulation, animation and/or generation of code for all or parts of a system.



Tools for creating UML models

- IBM Rational Software
- Together by Borland
- Objects By Design
- ObjectPlant: a good basic shareware tool for the Macintosh.
- Argo UML: an open source shareware project run by Tigris.



End of chapter 4