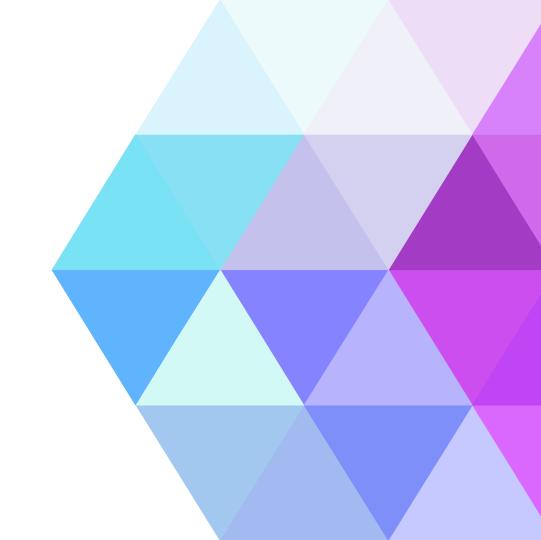# OBJECT ORIENTED ANALYSIS AND DESIGN

Lecture 01

Today's Outline

Administrative Stuff

Overview of 309

Introduction to Object Oriented Analysis & Design

Difference between SAD and OOAD

Office hours

Office 6 (CS building)

Office hours: 3:00 to 4:00 pm

(Monday, Thursday, Friday)

About the course

- Object Oriented Analysis & Design is an innovative way of thinking about problems using models organized in real world concepts.

- To study the fundamental construct of OOAD which combines both data structure and behavior in a single entity.

- To explain how a software design may be represented as a set of interacting objects that manage their own state and operations

- Various cases studies will be used throughout the course to demonstrate the concepts learnt.

- A strong in class participation from the students will be encouraged and required during the discussion on these case studies.

Pre-requisites /Knowledge assumed

Programming language concepts

Data structure concepts

We assume you have the skills to code in any programming language therefore you

can design, implement, test, debug, read, understand and document the programs.

Tentative Grading Policy

Assignments/Class Participation: 10 %

Quizzes: 10%

Mid Exam 1: 15 %

Mid Exam 2: 15 %

Final: 40%

Presentations + Projects: 10%

An advanced level project

Should provides interesting realistic problem,

Small 3-4 sized team

Emphasis on good OOAD methodology

Homework's and deliverables tied to the project

Course Material

You will have Presentations of each topic and ***reference books*** in PDF format will be available on slate.
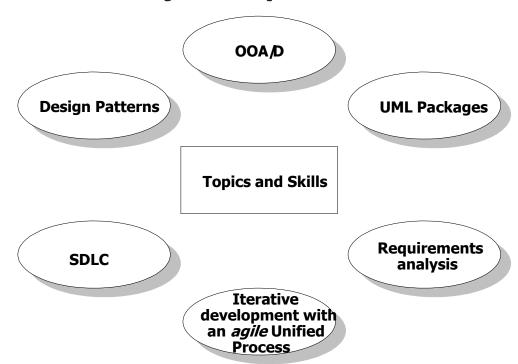
Text Books

1. *UML 2 Toolkit by Hans-Erik Eriksson, Magnus Penker, Brian Lyons, David Fado*
2. *Applying UML and Patterns 3rd Edition by Craig Larman*

Reference Books

1. *UML and the Unified Process, Practical object-oriented analysis and design by Jim Arlow, Ila Neustadt*
2. *The Unified Modeling Language Reference Manual, 2nd edition by James Rumbaugh, Ivar Jacobson and Grady Booch*
3. *UML Distilled, 2nd Edition by Martin Flower*

Hands-on labs will be part of the course.

Major Topics of the course

OOA/D

Design Patterns

UML Packages

Topics and Skills

SDLC

Requirements analysis

Iterative development with an *agile* Unified Process

Course Goals/Objectives

By the end of this course, you will be able to

Perform analysis on a given domain and come up with an Object Oriented Design (OOD).

Practice various techniques which are commonly used in analysis and design phases in the software industry.

Use Unified Modeling Language (UML) as a tool to demonstrate the analysis and design ideas

Analyze, design and implement practical systems of up to average complexity within a team and develop a software engineering mindset

Object-Oriented Analysis and Design (OOA/D)

- During OOA, the emphasis is on finding and describing the objects (or concepts) in the problem domain, i.e., domain objects.

- During OOD (or simply object design), the emphasis is on defining software objects and how they collaborate to fulfill the requirements.

- During OOP (OO Programming) or *Implementation*, design objects are implemented in a programming language.
  - Implementation is also known as *Coding* or *Construction*.

What is System Analysis and Design?

What is a system?

All activities that go into producing an information system solutions

Analysis and Design of the system?

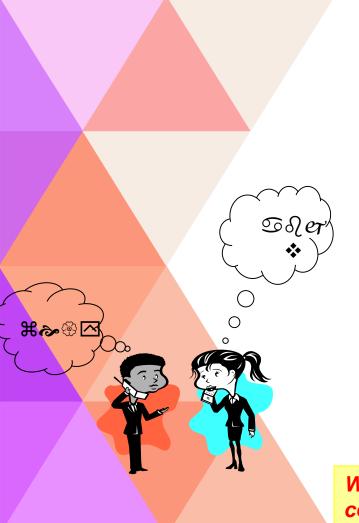Mainly deals with the system development activities

What is Software Analysis and Design?

What is a software?

A series of processes that, if followed, can lead to the development of an application.

Analysis and Design of the software?

Includes all activities, which help the transformation of requirement specification into implementation.

Why Object-Oriented?

*"The "software crises" came about when people realized the major problems in software development were … caused by **communication** difficulties and the management of **complexity"** [Budd]*

The Whorfian Hypothesis:

Human beings … are very much at the mercy of the particular language which has become the medium of expression for their society … the 'real world' is … built upon the language habits … We cut nature up, organize it into concepts, and ascribe significances as we do, largely because we are parties to an agreement to organize it in this way … and is codified in the patterns of our language.

What kind of language can alleviate difficulties with communication & complexity hopefully well?

Why Object-Oriented?
– Consider Human Growth & Concept Formation

Communication & complexity about the problem and the solution, all expressed in terms of *concepts* in a language!

But then, What is CONCEPT? [Martin & Odell]

Consider Human Growth & Concept Formation

Stage	concepts
infant	*the world is a buzzing confusion*
very young age	*"blue" "sky" (individual concepts)* *"blue sky" (more complex concept)* *hypothesis: humans possess an innate capacity for perception*
getting older	*increased meaning, precision, subtlety, ...* *the sky is blue only on cloudless days* *the sky is not really blue* *it only looks blue from our planet Earth* *because of atmospheric effects*

Concept formation: from chaos to order!
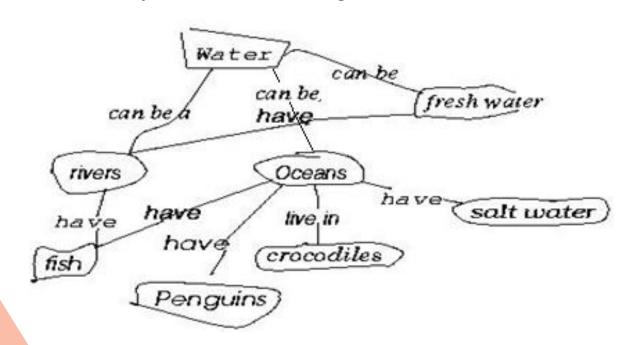
When given a list of concepts

(water, rivers, fish, fresh water, salt water, Oceans, Penguins, crocodiles)

You need to construct a concept diagram through which you understands the world and communicates meaning
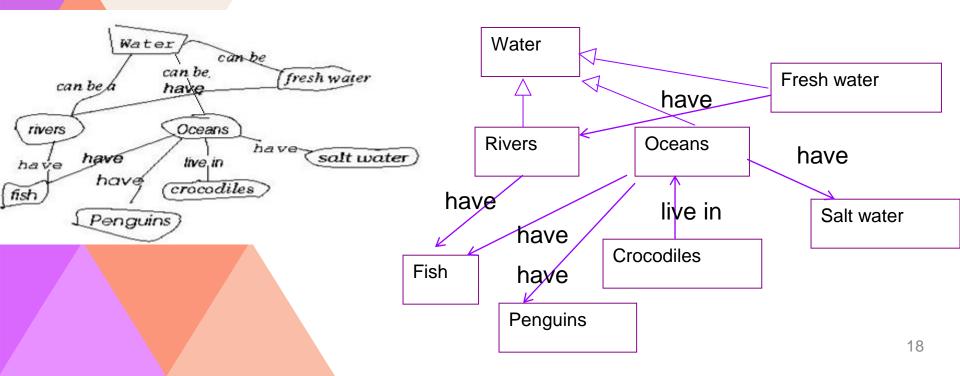
Why Object-Oriented?

- concepts and objects

So, concepts are needed to bring order ... into

What kind of language can be used to create this concept diagram, or your's mental image?

Why Object-Oriented?

What is a *model* and why?

- A model is a simplification of reality.

  E.g., a miniature bridge for a real bridge to be built
  - *Well...sort of….but not quite*
  - A model is *our* simplification of *our perception* of reality

    (that is, if it exists, otherwise it could be a mere illusion).

    communication is not about reality but about your/my/his/her perception of reality => validation and verification hard but needed

- A model is an *abstraction* of something for the purpose of *understanding*, be it the problem or a solution.

- To understand *why* a software system is needed, *what* it should do, and *how* it should do it.
- To communicate our understanding of why, what and how.
- To detect commonalities and differences in your perception, my perception, his perception and her perception of reality.
- To detect misunderstandings and miscommunications.

What is Object-Orientation?

- What is Object?

An object is any abstraction that models a single thing.
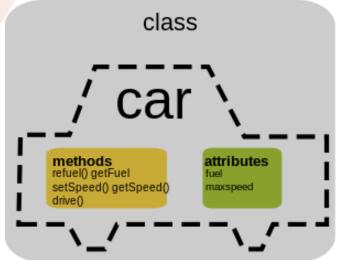
A representation of a specific entity in the real world.

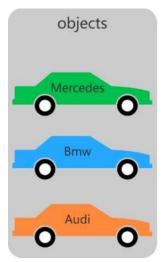A structure that has identity and properties and behavior.

May be tangible (physical entity) or intangible.

Examples: specific citizen, agency, job, location, order, etc.

It is an instance of a collective concept, i.e., a class.

What is Object-Oriented?

What is *CLASS*?

a collection of objects that share common properties, attributes, behavior, in general.

A collection of objects with the same data structure (attributes, state variables) and behavior (function/code/operations) in the solution space.

Classification

Grouping of common objects into a class

Instantiation.

The act of creating an instance.

Two Orthogonal Views of the Software

- There are two different ways to view software construction:
  - We can focus primarily on the function (Traditional System Development)

    OR
  - We can focus primarily on the data (Object Oriented System Development)

What is the Structure Analysis & Design (SAD)?

Functional view of the problem

Divide and Conquer

Traditional systems development technique that is time tested and easy to understand

Transform data into information, called as process centered technique.

Top Down Approach

Uses set of process models to describe a system graphically

Divide large, complex problem into smaller, more easily handled ones.

What is the Structure Analysis & Design?

Establish complete requirement documentation

Establish concrete requirement specification

Structured analysis is a set of techniques and graphical tools that allow the analyst to develop a new kind of system specification that are easily understandable to the user. Analysts work primarily with their wits, pencil and paper."

Improve Quality and reduce risk

Focus on reliability, flexibility & maintainability

Structure Analysis & Design

Advantages

- visual, so it is easier for users/programmers to understand
- Makes good use of graphical tools
- A mature technique
- Process-oriented approach is a natural way of thinking
- Flexible
- simple and easy to understand and implement

Disadvantages

- Not enough user-analyst interaction
- It depends on dividing system to sub systems but it is to decide when to stop decomposing.

What is Object Oriented Analysis & Design (OOAD)?

Object-Oriented analysis and design thoroughly represent complex relationships, as well as represent data and data processing with a consistent notation

Essential for robust, well designed software
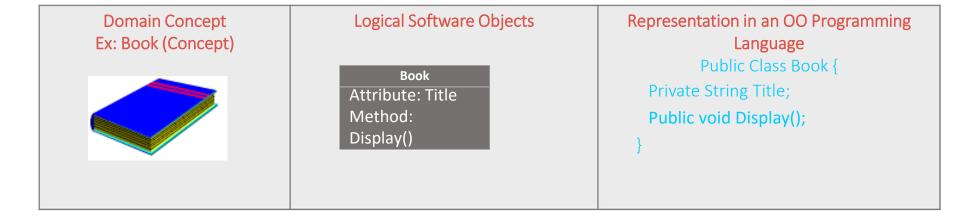
Emphasize on finding and describing objects

Blend analysis and design in evolutionary process

Deals with complexity inherent in real world

Object Oriented Analysis and Design

From Analysis to Implementation

Analysis	Design	Construction
Analysis Investigation of the problem	**Design** Logical Solution	**Construction** Code

Domain Concept Ex: Book (Concept)	Logical Software Objects	Representation in an OO Programming Language

	Book
	Attribute: Title
	Method:
	Display()
		Public Class Book {
		Private String Title;
		Public void Display();
		}

Similarities and difference between SAD and OOAD

Key Differences Between Structured and Object-Oriented Analysis and Design

	Structured	Object-Oriented
Methodology	SDLC	Iterative/Incremental
Focus	Processs	Objects
Risk	High	Low
Reuse	Low	High
Maturity	Mature and widespread	Emerging (1997)
Suitable for	Well-defined projects with stable user requirements	Risky large projects with changing user requirements

5

Case Study: A PAYROLL PROGRAM

- Consider a payroll program that process employee reocrds at a small manufacturing firm. The company has several classes of employees with particular payroll requirement and rules for processing each. The company has three types of employees:

- *Managers* receive a regular salary.

- *Office Workers* receive an hourly wage and are eligible for overtime after 40 hours.

- *Production Workers* are paid according to a piece rate.

You need to walk through traditional (SAD) and Object Oriented (OOAD) to highlight their differences.