Tim Pengajar IF2250

IF2250 – Rekayasa Perangkat Lunak

OO Methods (Bag 1)

SEMESTER II TAHUN AJARAN 2023/2024

KNOWLEDGE & SOFTWARE ENGINEERING

Object Oriented methodologies

A. The Rumbaugh OMT
B. The Booch methodology
C. Jacobson's methodologies

A. Rumbaugh's Object Modeling Technique (OMT)

- A method for analysis, design and implementation by an object-oriented technique.
- fast and intuitive approach for identifying and modeling all objects making up a system.
- Class attributes, methods, inheritance and association can be expressed easily.
- Dynamic behavior of objects can be described using the OMT dynamic model.
- Detailed specification of state transitions and their descriptions within a system

Four phases of OMT (can be performed iteratively)

1. ***Analysis***: objects, dynamic and functional models
2. ***System Design***: basic architecture of the system.
3. ***Object Design***: static, dynamic and functional models of objects.
4. ***Implementation***: reusable, extendible and robust code.

Three different parts of OMT modeling

1. An **object model** - object model & data dictionary
2. A **dynamic model** - state diagrams & event flow diagrams
3. A **functional model** - data flow & constraints

I. Object Model

- structure of objects in a system.
- Identity, relationships to other objects, attributes and operations.
- Object diagram
  - Classes interconnected by association lines
  - **Classes** - a set of individual objects
  - **Association lines** - relationship among classes (i.e., objects of one class to objects of another class)

2. Dynamic Model

- States, transitions, events and actions
- OMT state transition diagram - network of states and events

3. Functional Model

- DFD - (Data Flow Diagram)
- Shows flow of data between different processes in a business.
- Simple and intuitive method for describing business processes without focusing on the details of computer systems.

B. The Booch Methodology

- Widely used OO method
- Uses the object paradigm
- Covers the design and analysis phase of an OO system
- Criticized for his large set of symbols

Diagrams of Booch method

- **Class diagrams -** describe roles and responsibilities of objects
- **Object diagrams -** describe the desired behavior of the system in terms of scenarios
- **State transition diagrams -** state of a class based on a stimulus
- **Module diagrams -** to map out where each class & object should be declared
- **Process diagrams -** to determine to which processor to allocate a process
- **Interaction diagrams -** describes behavior of the system in terms of scenarios

Booch method prescribes:

- Macro Development Process
  - Controlling framework for the micro process.
  - Primary concern-technical management of the system.
  - Each macro process has its own micro development process
- Micro Development Process

  Steps:

  - Identify classes & objects
  - Identify class & objects semantics
  - Identify class & object relationship
  - Identify class & objects interface and implementation

Steps for macro development process

1. Conceptualization
2. Analysis & Development of the model
3. Design or create the system architecture
4. Evolution or implementation
5. Maintenance

C. Jacobson Methodologies

- Use Cases.
- Object Oriented Software Engineering (OOSE).
- Objectory is built models
  - Use case model
  - Domain object model
  - Analysis object model
  - Implementation model
  - Test model
- Object Oriented Business Engineering (OOBE)
- OOBE is object modeling at the enterprise level.
  - Analysis phase
  - Design and Implementation phase
  - Testing phase
    - E.g. Unit testing, integration and system testing.

References

- **Object-oriented Modeling and Design** by James Rumbaugh, Michael Blaha, William Premerlani, Frederick Eddy, and WilliamLorensen,, Prentice Hall, 1991
- **Object-Oriented Analysis and Design with Applications** 3rd Edition by Grady Booch (Author), Robert A. Maksimchuk (Author), Michael W. Engle (Author), Bobbi J. Young (Author), Jim Conallen (Author), Kelli A. Houston (Author), Addison-Wesley, 2007
- **Object-Oriented Software Engineering: A Use Case Driven Approach** 1st edition by Ivar Jacobson, Magnus Christerson, Patrik Jonsson & Gunnar Overgaard. Addison-Wesley, 1992