Class dan Objek

Ali Tarmuji
Email: alitarmuji@gmail.com
ID YM: alitarmuji

Pokok Bahasan

- Konsep OO
- Class
- Hubungan Class dan Object

Konsep OO

- Object adalah:
  - Definisi Informal : sebuah object adalah representasi dari sebuah entitas, baik fisik, konseptual maupun software.
    - Entitas fisik misalnya : orang, mobil dan lain-lain
    - Entitas konseptual misalnya : proses kimia atau algoritma
    - Entitas software misalnya : linked list

Konsep OO

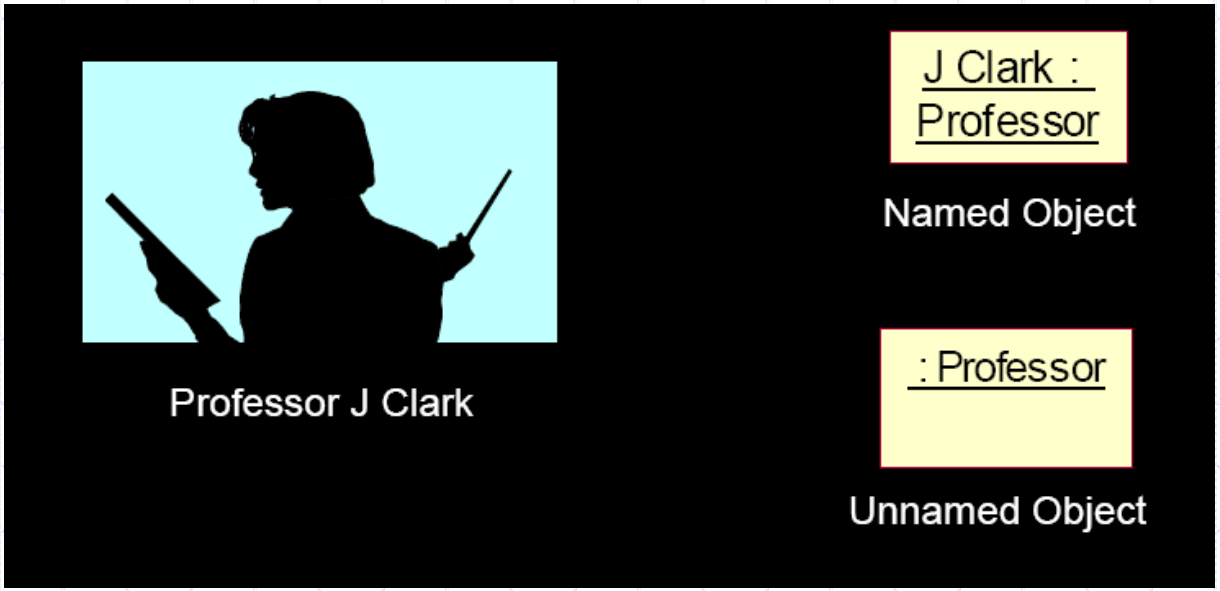
- Object adalah:
  - Definisi Formal : sebuah object adalah sebuah entitas dengan boundary yang terdefinisi dengan baik dan identitas yang menengkapsulasi state dan behaviour.
    - State : direpresentasikan oleh atribut dan relationship
    - Behaviour : direpresentasikan oleh operasi, method dan state machine

Konsep OO

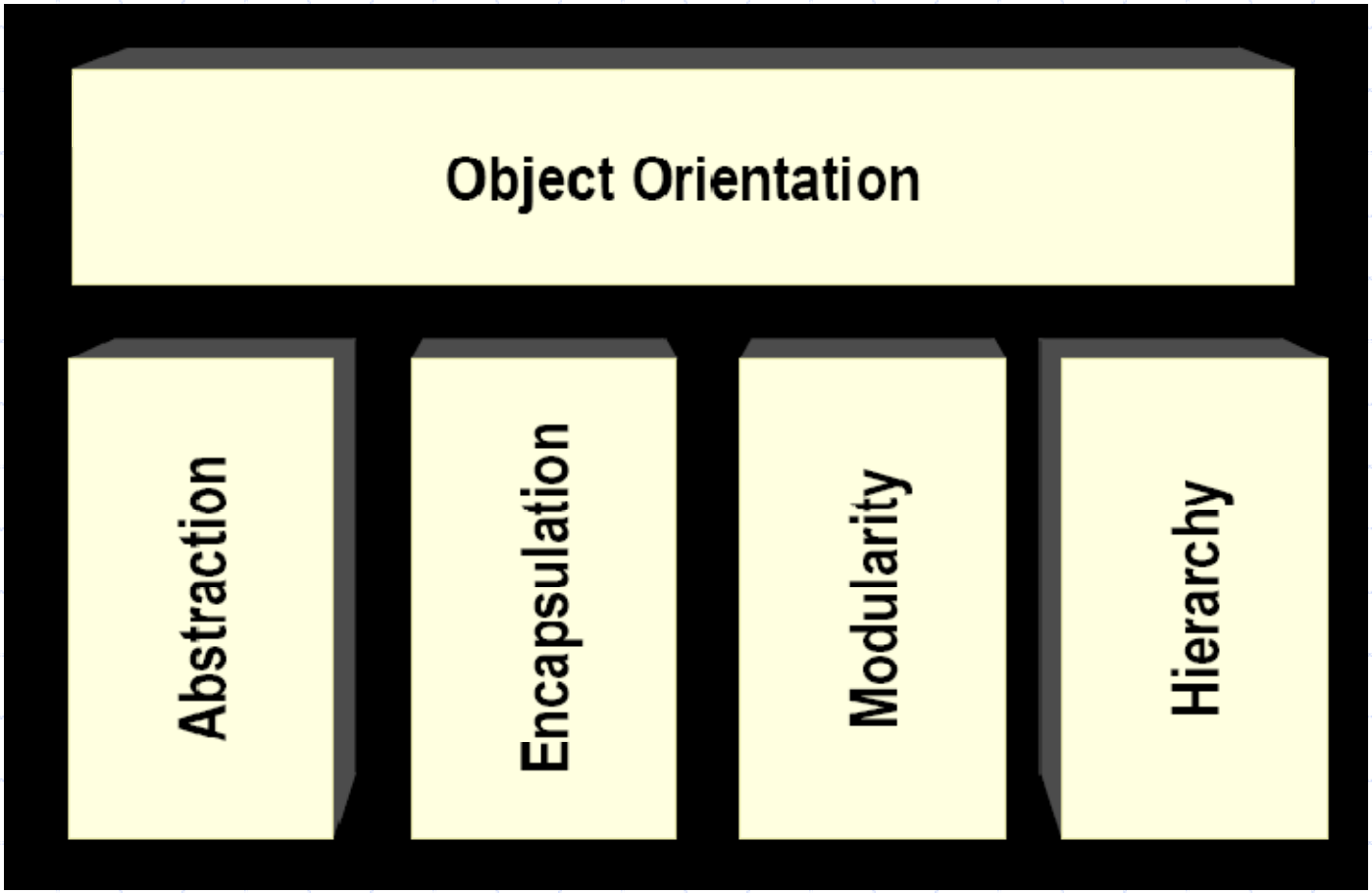
- State
  - State dari sebuah object adalah kondisi yang mungkin dialami oleh object
  - Secara normal, state object berubah setiap waktu
- Behaviour
  - Behaviour menentukan bagaimana sebuah object beraksi dan bereaksi
  - Behaviour yang tampak dari sebuah object dimodelkan oleh sekumpulan pesan(message) yang bisa direspon atau operasi-operasi yang bisa dijalankan oleh sebuah object

Representasi Object Dalam UML

- Sebuah Object direpresentasikan sebagai kotak dengan nama yang bergaris bawah



Prinsip dasar OO

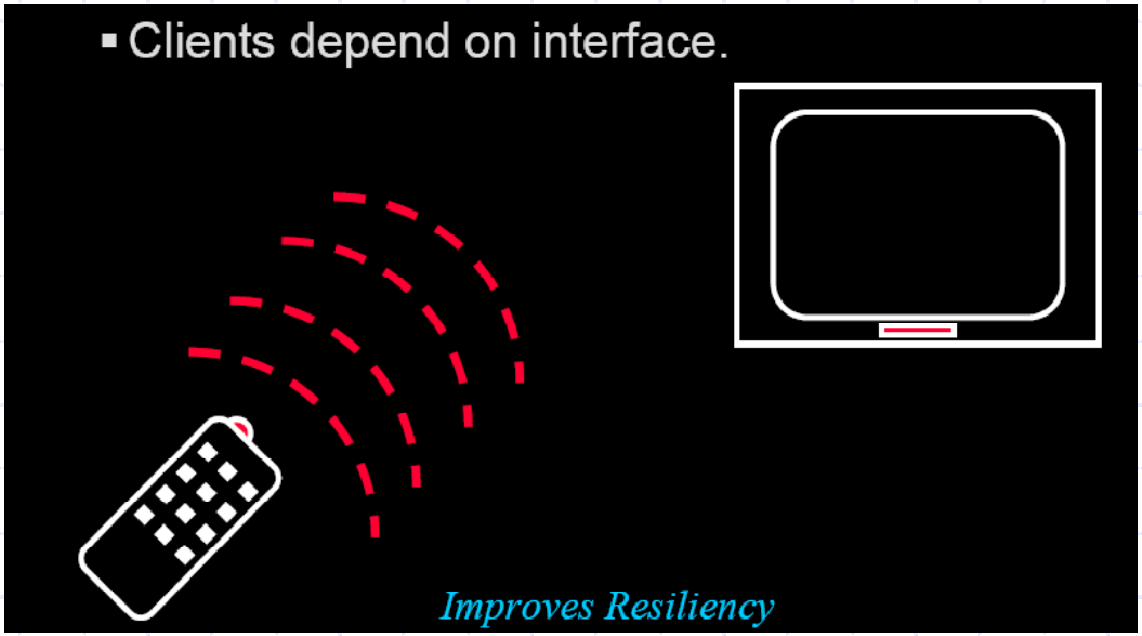


Abstraction

- Abstraction adalah karakteristik dasar dari sebuah entitas yang membedakan entitas tersebut dari entitas yang lain
- Abstraction mendefinisikan batasan dalam pandangan viewer
- Abstraction bukanlah pembuktian nyata,hanya menunjukkan intisari/pokok dari sesuatu

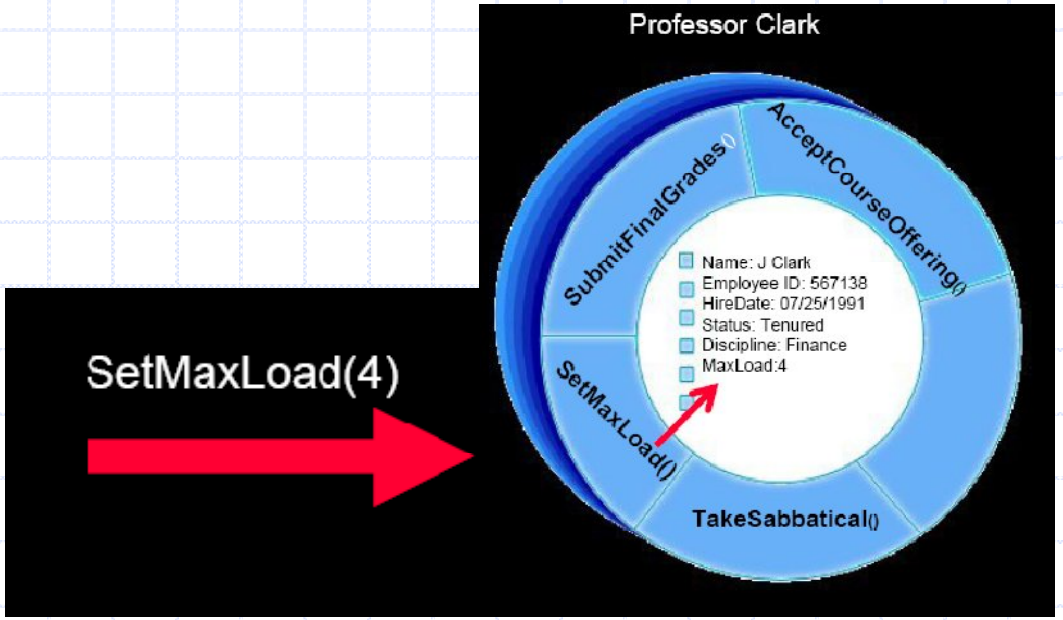
Encapsulation

- Encapsulation adalah menyembunyikan implementasi dari client, sehingga client hanya tergantung pada interface



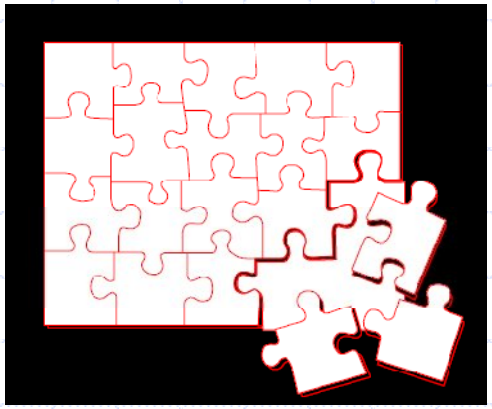
Ilustrasi Encapsulation

- Seorang Professor bisa megajar 4 class pada semester depan

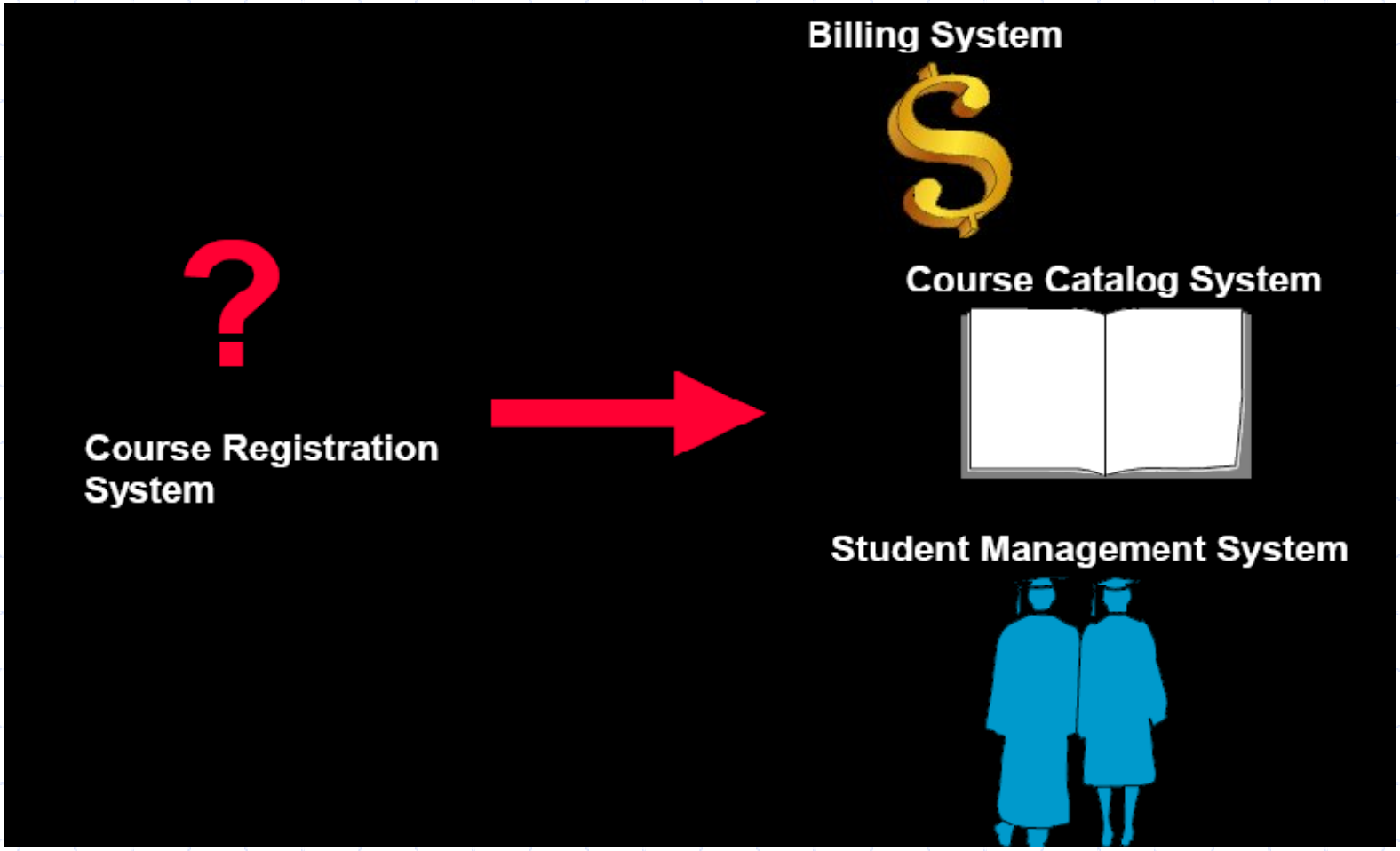


Modularity

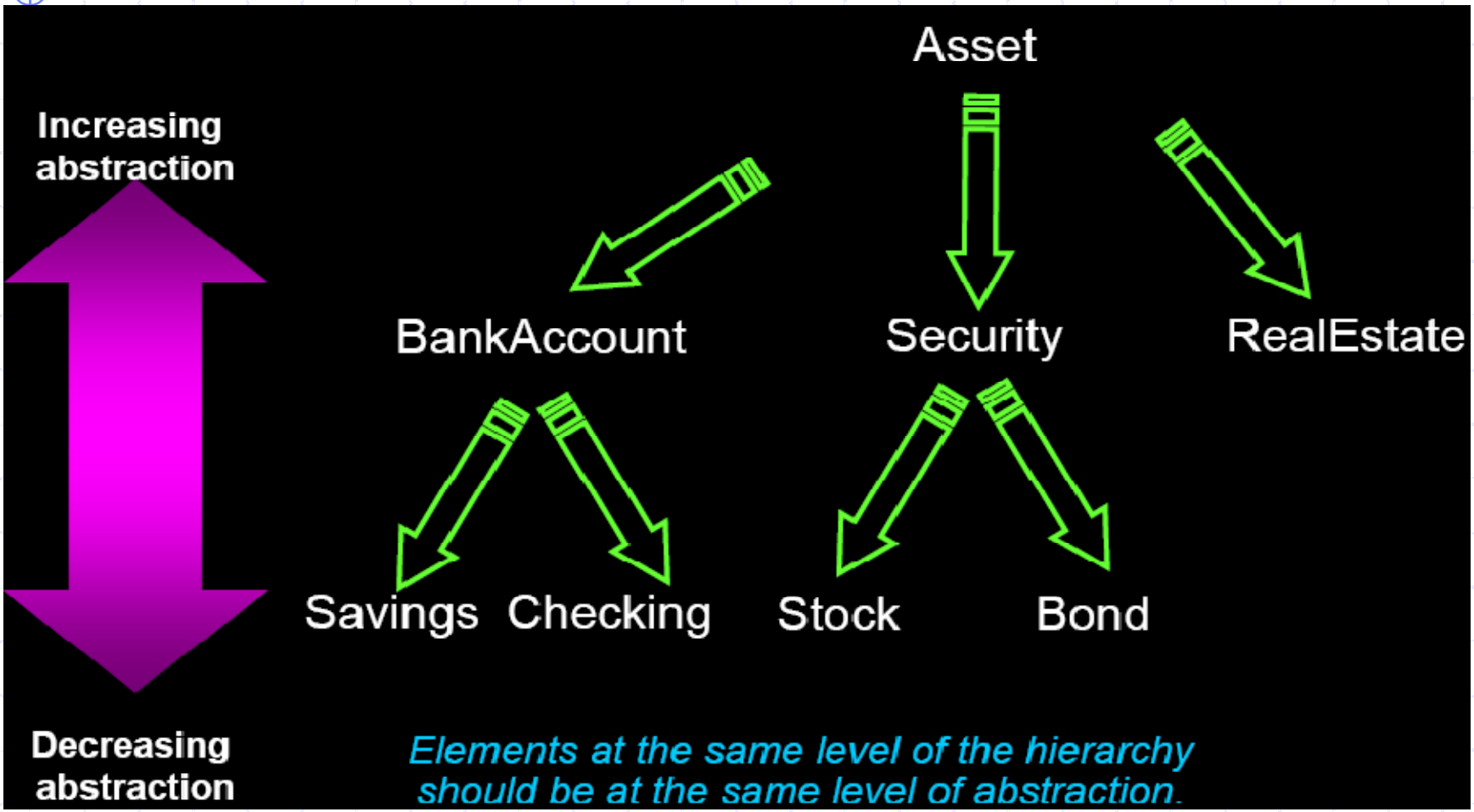
- Adalah pemecahan sesuatu yang kompleksmenjadi bagian-bagian yang mudah diatur
- Modularity membantu orang dalam memahami sesuatu yang kompleks



Contoh Modularity



Hierarchy


Asset
Increasing abstraction
BankAccount
Security
RealEstate
Savings
Checking
Stock
Bond
Decreasing abstraction
Elements at the same level of the hierarchy should be at the same level of abstraction.

Class

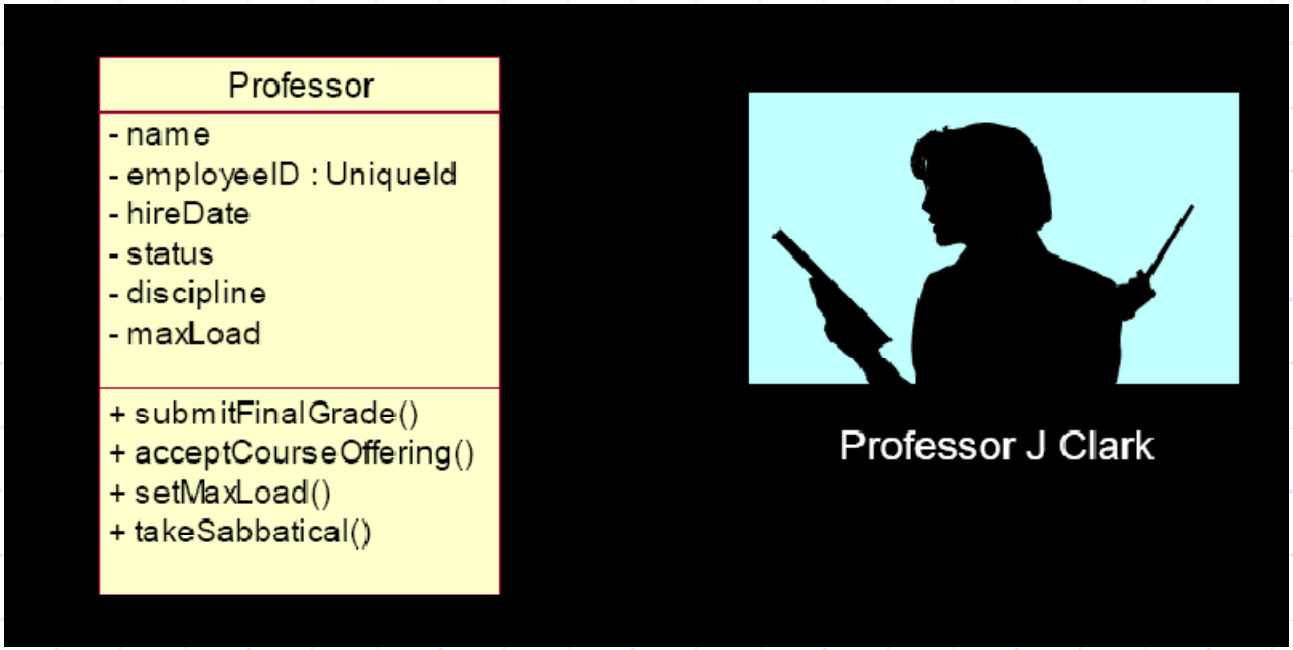
- Class adalah :
  - Sekumpulan object yang berbagi atribut umum dan behaviour secara umum.
  - Sekumpulan object yang memiliki struktur data dan behaviour yang sama
  - Blue print atau definisi sebuah object
- Instance adalah sebuah object yang dibuat oleh sebuah class
- Instantiation adalah pembuatan instance

Class

- Specialization adalah pendefinisian sebuah class sebagai pendetilan class yang lain
- Subclass adalah sebuah yang didefinisikan dalam rangka specialization superclass menggunakan inheritance
- Superclass adalah sebuah class yang bertugas menurunkan sifat(inheritance) dalam sebuah hirarki class
- Inheritance adalah penduplikasian atribut dan behaviour superclass ke subclassnya.

Representasi Class dalam UML

- Sebuah class direprsentasikan dengan kotak dengan pembagi



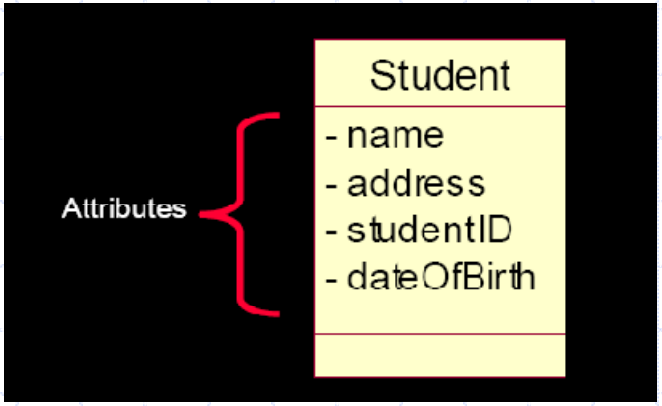
Hubungan antara Class dengan Object

- Sebuah Class merupakan definisi abstract dari sebuah object. Class mendefinisikan struktur dan behaviour dari masing-masing object di dalam sebuah class. Class bertugas sebagai template untuk pembuatan class.
- Object dikelompokkan dalam class

Contoh hubungan class dengan object

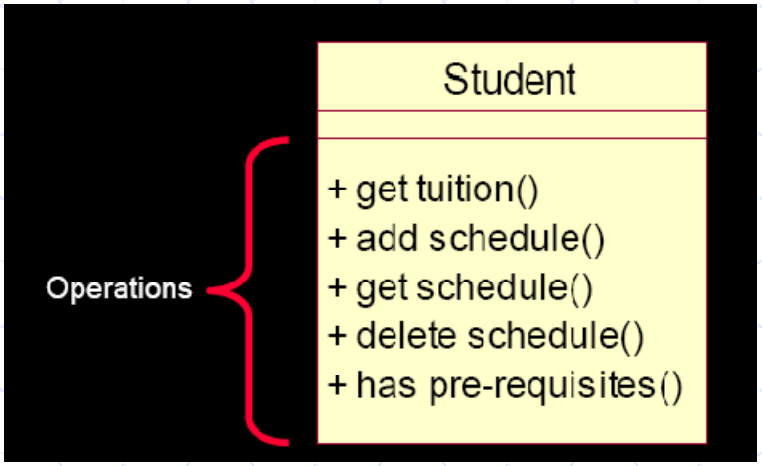
Atribut

- Adalah property class yang memiliki nama, dimana property itu menggambarkan range nilai
- Sebuah class bisa memiliki beberapa atribut atau tidak sama sekali



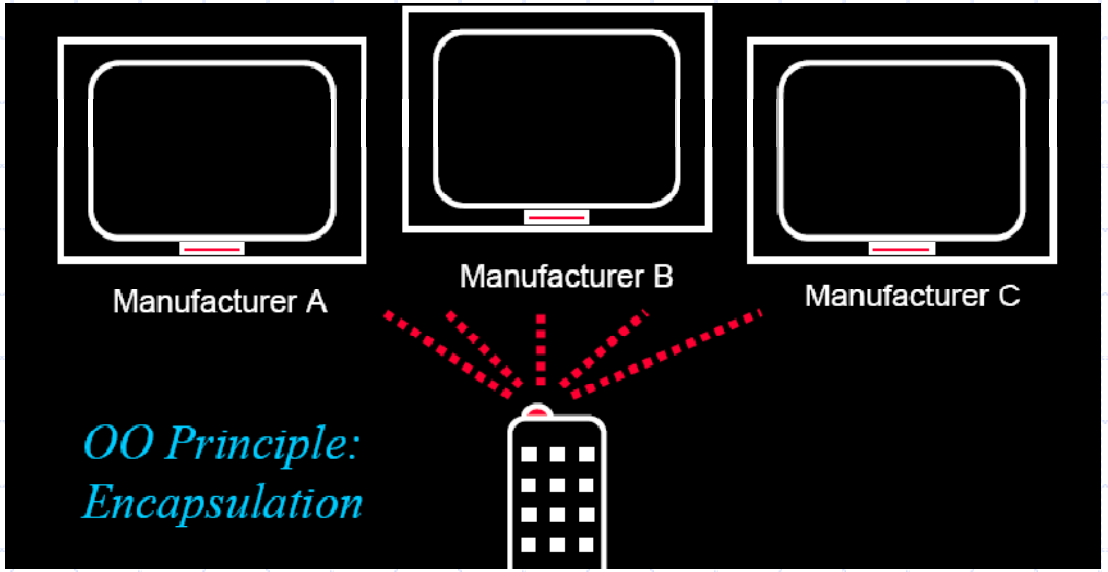
Operation

- Operation adalah implementasi dari sebuah service yang dapat direques dari object class untuk menghasilkan behaviour

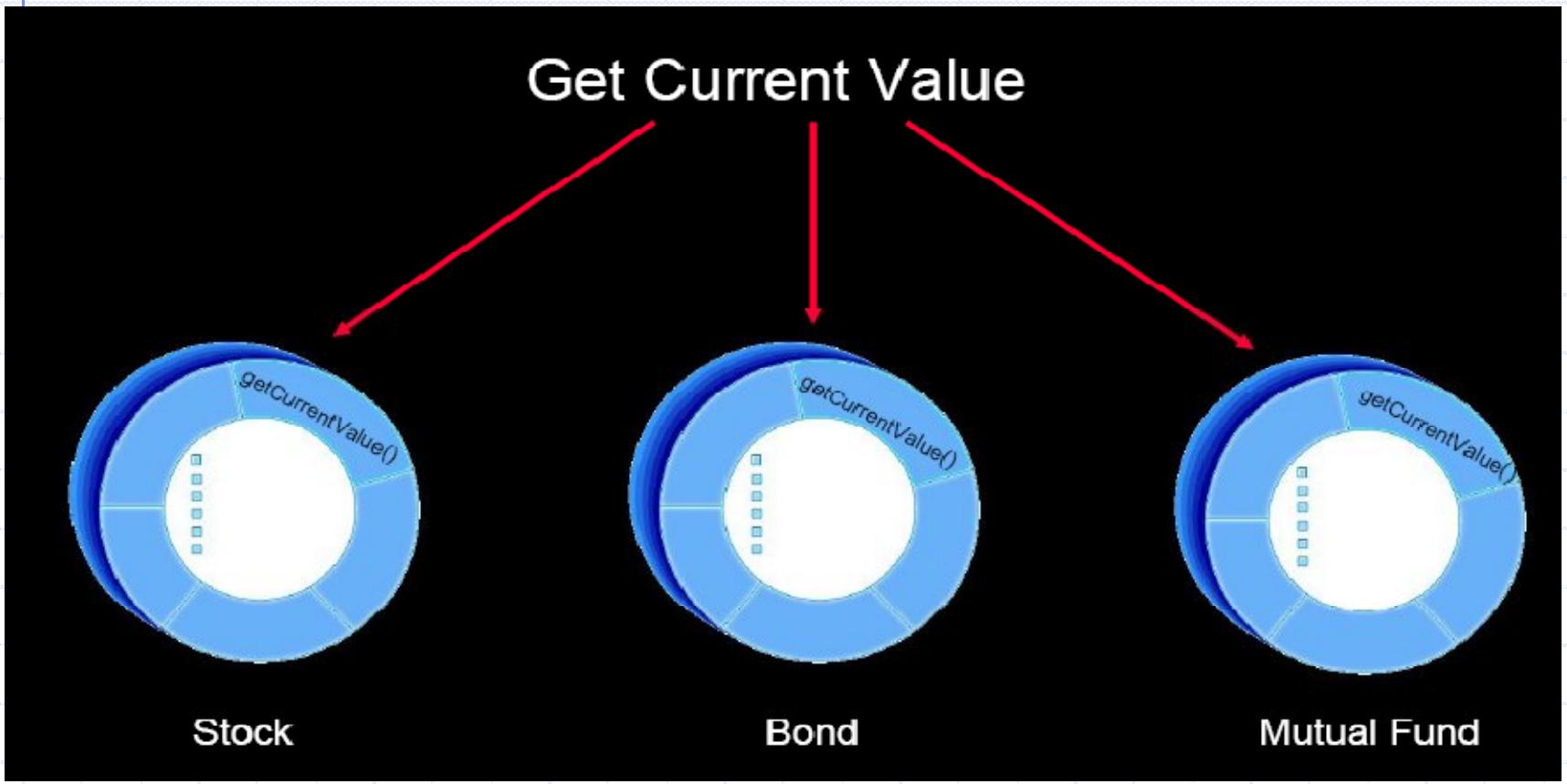


Polymorphisme

- Polymorphisme adalah kemampuan untuk menyembunyikan implementasi-implementasi yang berbeda didalam sebuah interface tunggal.

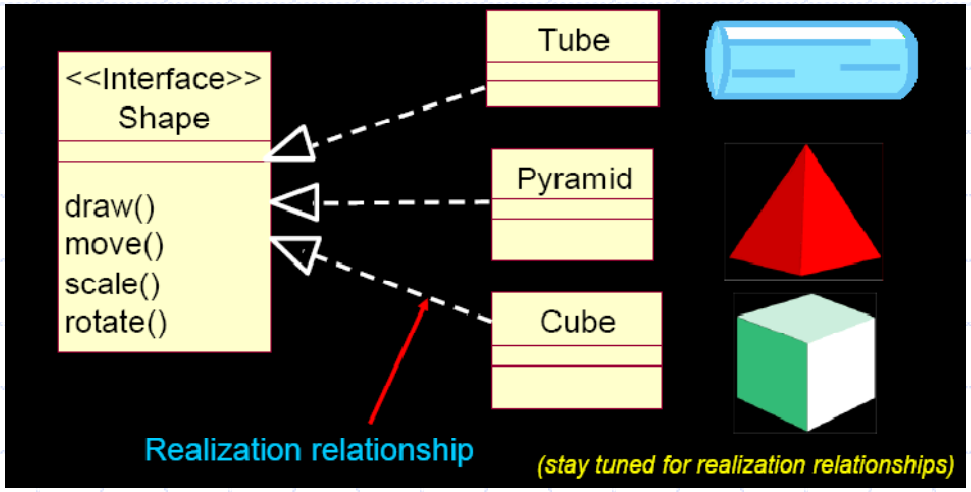


Contoh Polymorphisme

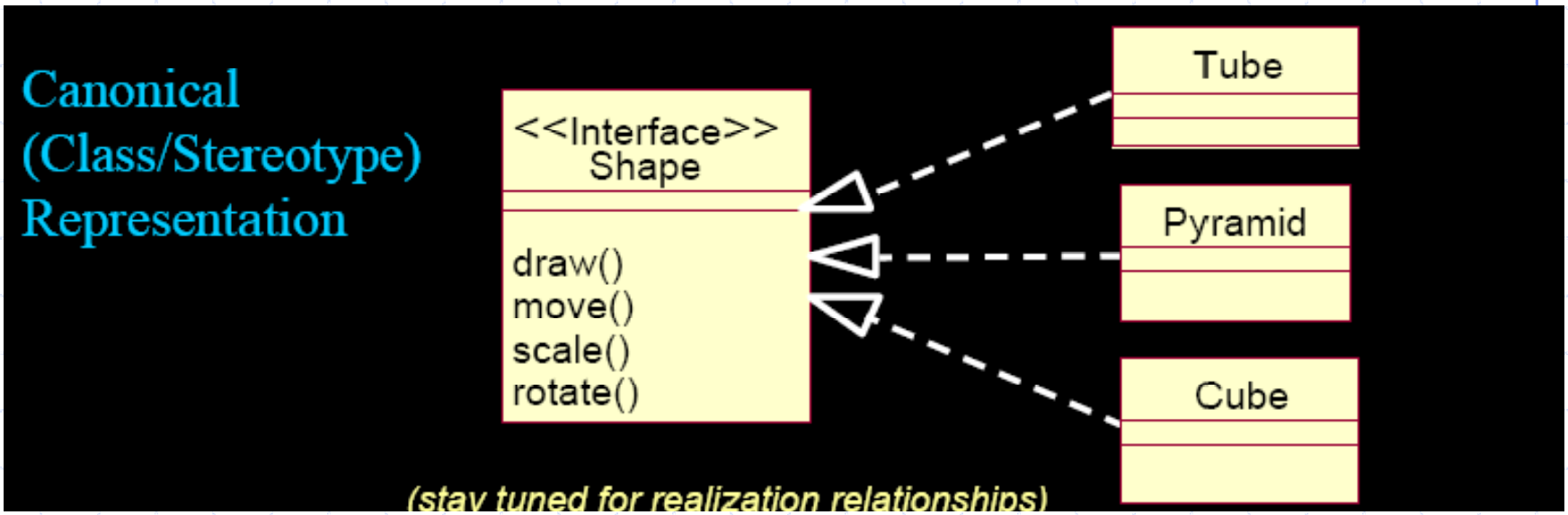


Interface

- Interface adalah pewujudan dari polymorphisme

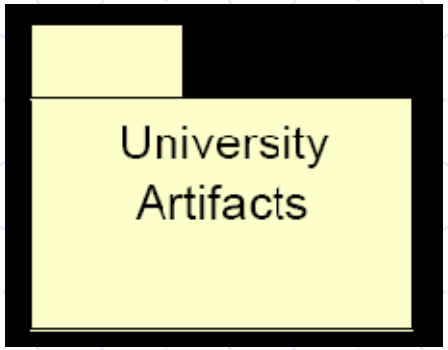


Representasi Interface dalam UML



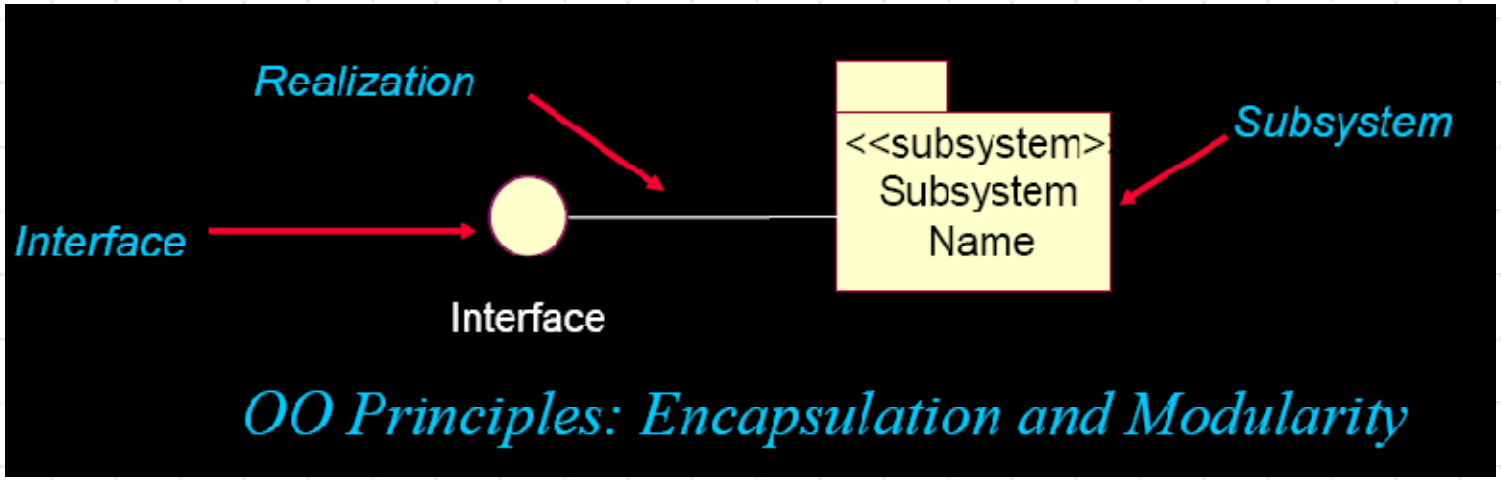
Package

- Package adalahmekanisme untuk menyusun elemen-elemen menjadi kelompok-kelompok.



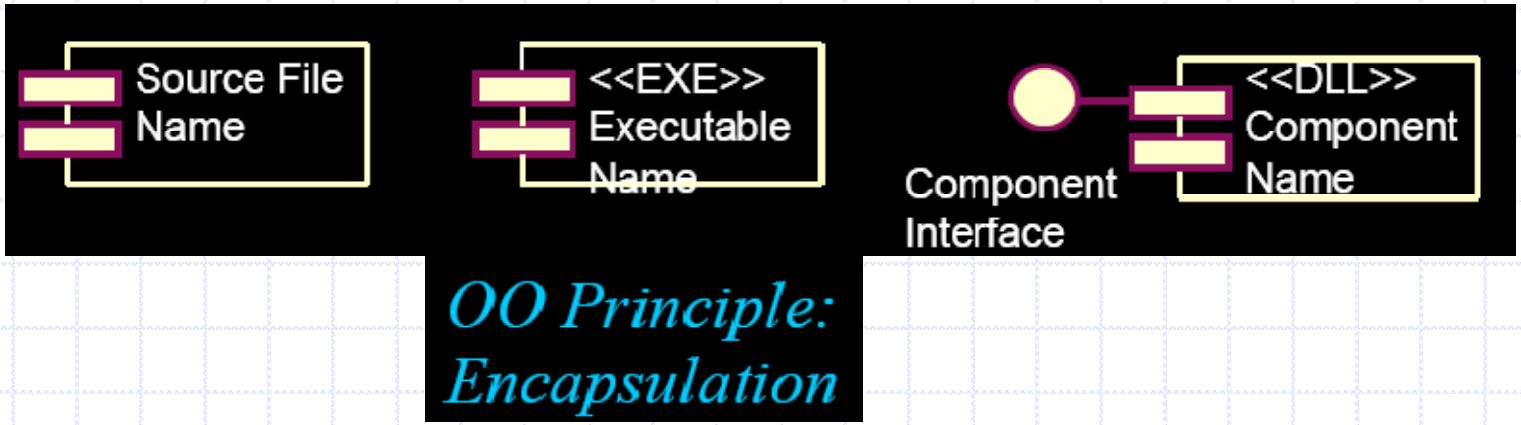
Subsystem

- Subsystem adalah kombinasi dari package dan class
- Subsystem merealisasikan satu atau lebih interface, dimana interface sebagai pendefinisi perilakunya.



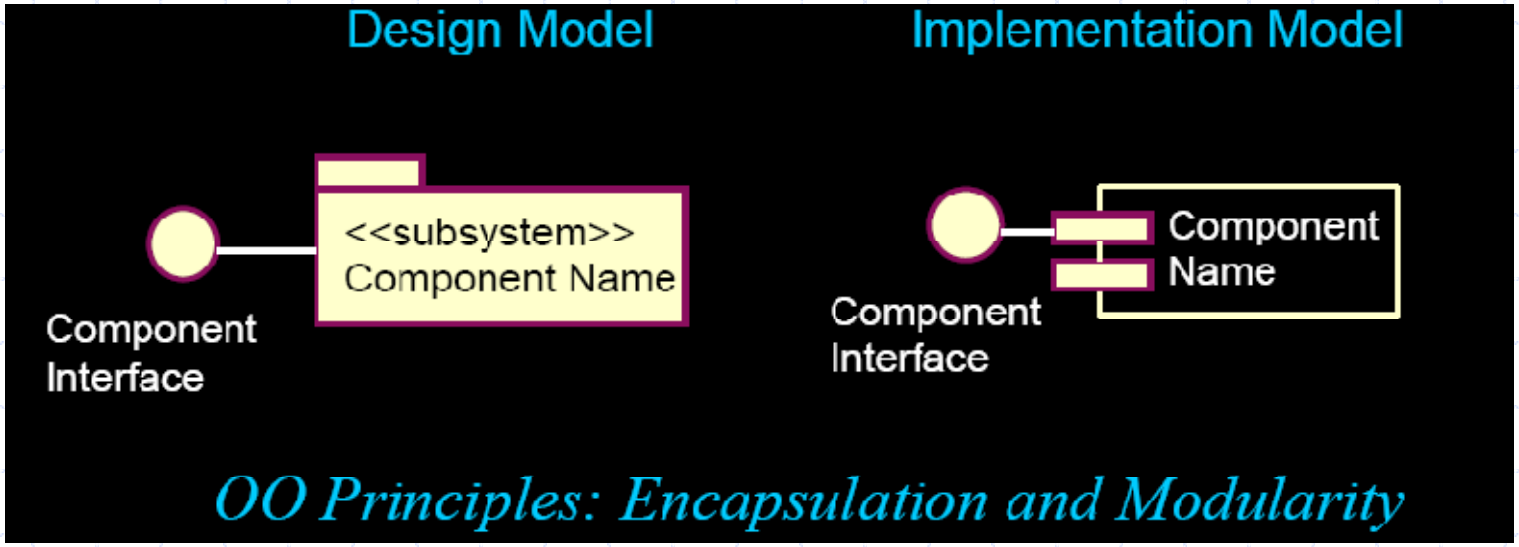
Component

- Component adalah bagian system yang dapat di-replace dan hampir independent. Component ini melaksanakan fungsi yang jelas dalam suatu arsitektur.
- Sebuah component bisa berupa:
  - Sebuah component source code
  - Sebuah component run time
  - Sebuah component executable



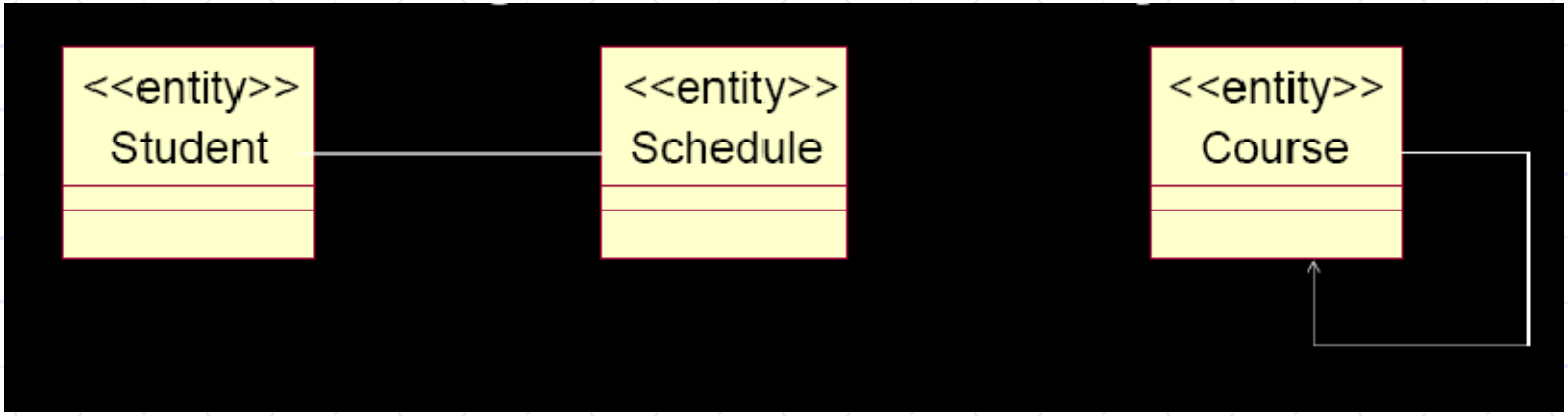
Subsystem dan component

- Component adalah realisasi phisic dari sebuah abstraksi dalam desain
- Subsystem dapat digunakan untuk merepresentasikan component dalam sebuah desain



Association

- Association adalah hubungan semantic antara dua atau lebih classifier yang menetapkan hubungan antar instance
- Association adalah hubungan structural yang menetapkan bahwa suatu object terhubung dengan object lain



Multiplicity

- Multiplicity adalah jumlah instance dari sebuh class yang berhubungan dengan satu instance class lain
- Untuk masing-masing association , ada dua keputusan multiplicity yang harus dibuat. Contoh:
  - Untuk masing-masing instance professor, ada beberapa course yang bisa ditawarkan
  - Untuk masing-masing instance penawaran course, mungkin ada nol atau satu professor sebagai pengajarnya

Penanda multiplicity

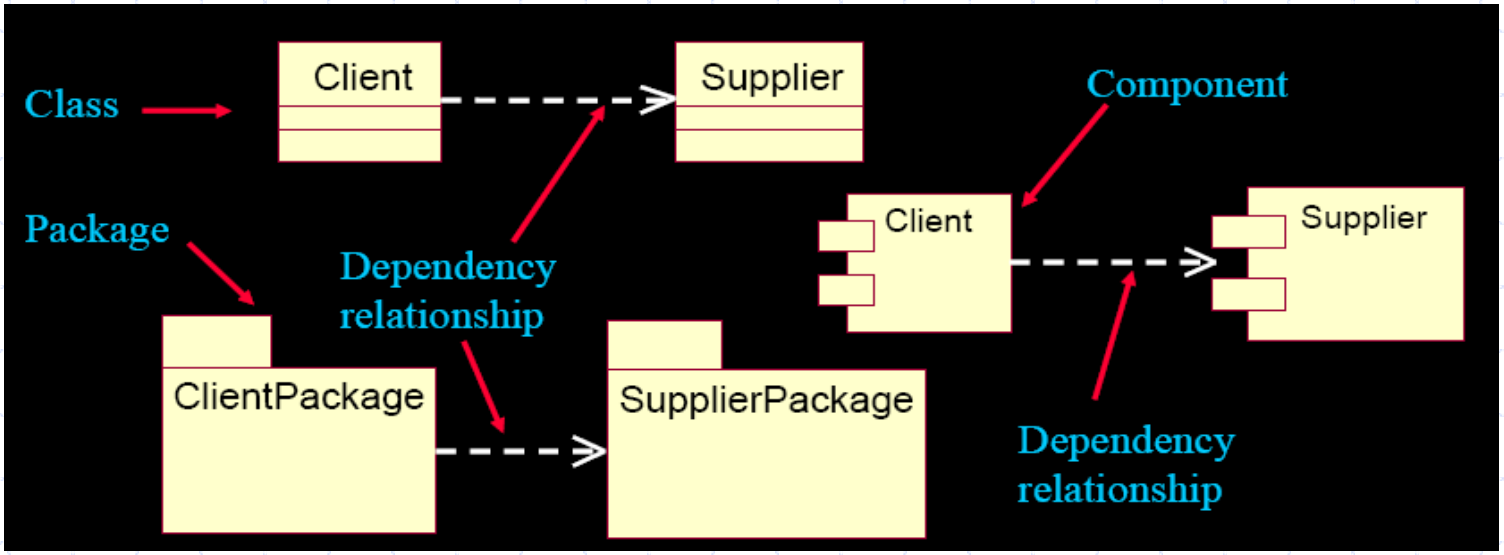
Unspecified	
Exactly one	1
Zero or more (many, unlimited)	0..*
	*
One or more	1..*
Zero or one (optional scalar role)	0..1
Specified range	2..4
Multiple, disjoint ranges	2, 4..6

Aggregation

- Sebuah aggregation adalah bentuk khusus association yang memodelkan hubungan whole-part antara sebuah aggregation(aggregation) dengan bagiannya.

Relationship : Dependency

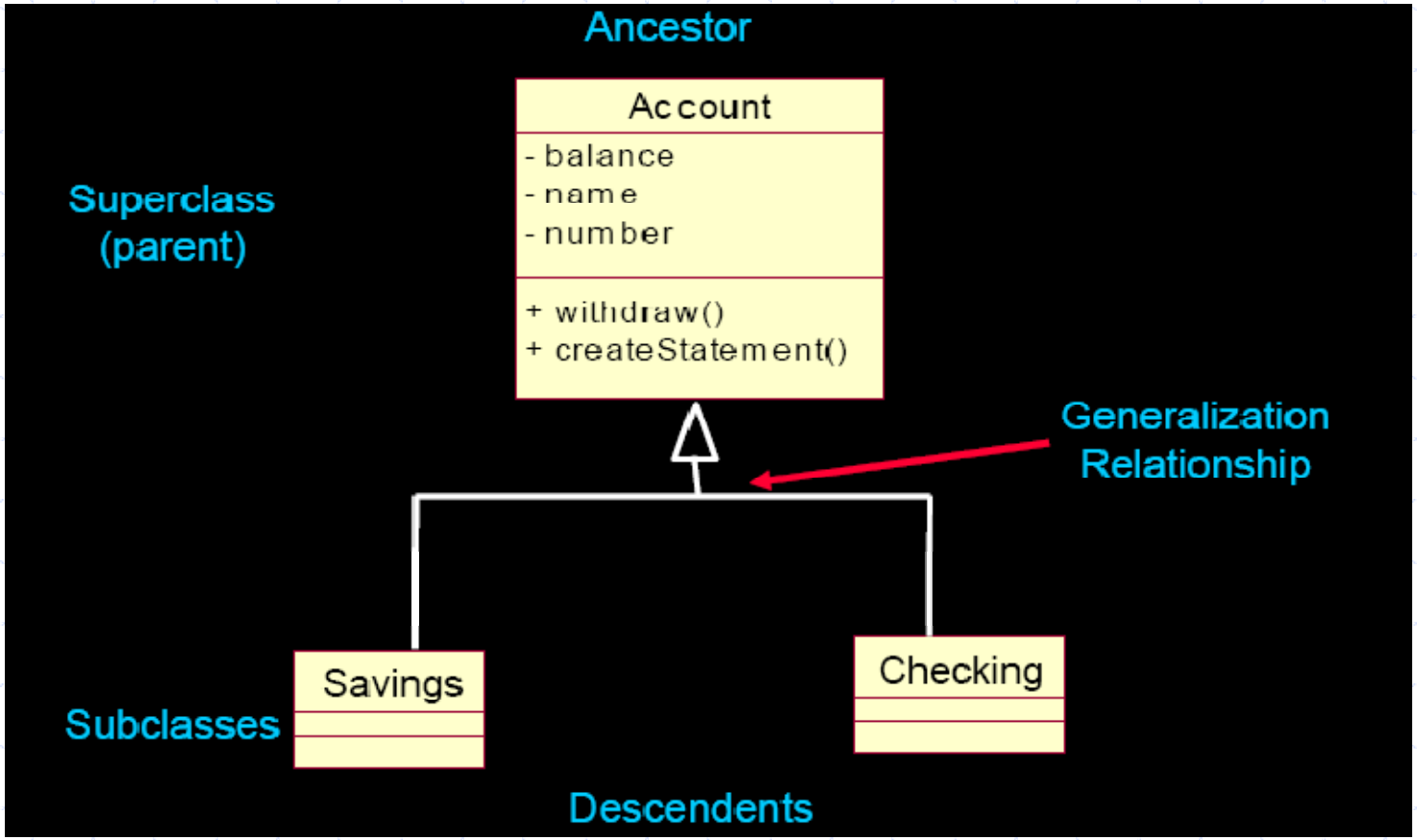
- Dependency adalah hubungan antara dua elemen dimana jika sebuah elemen mengalami perubahan akan menyebabkan perubahan pada elemen yang lain



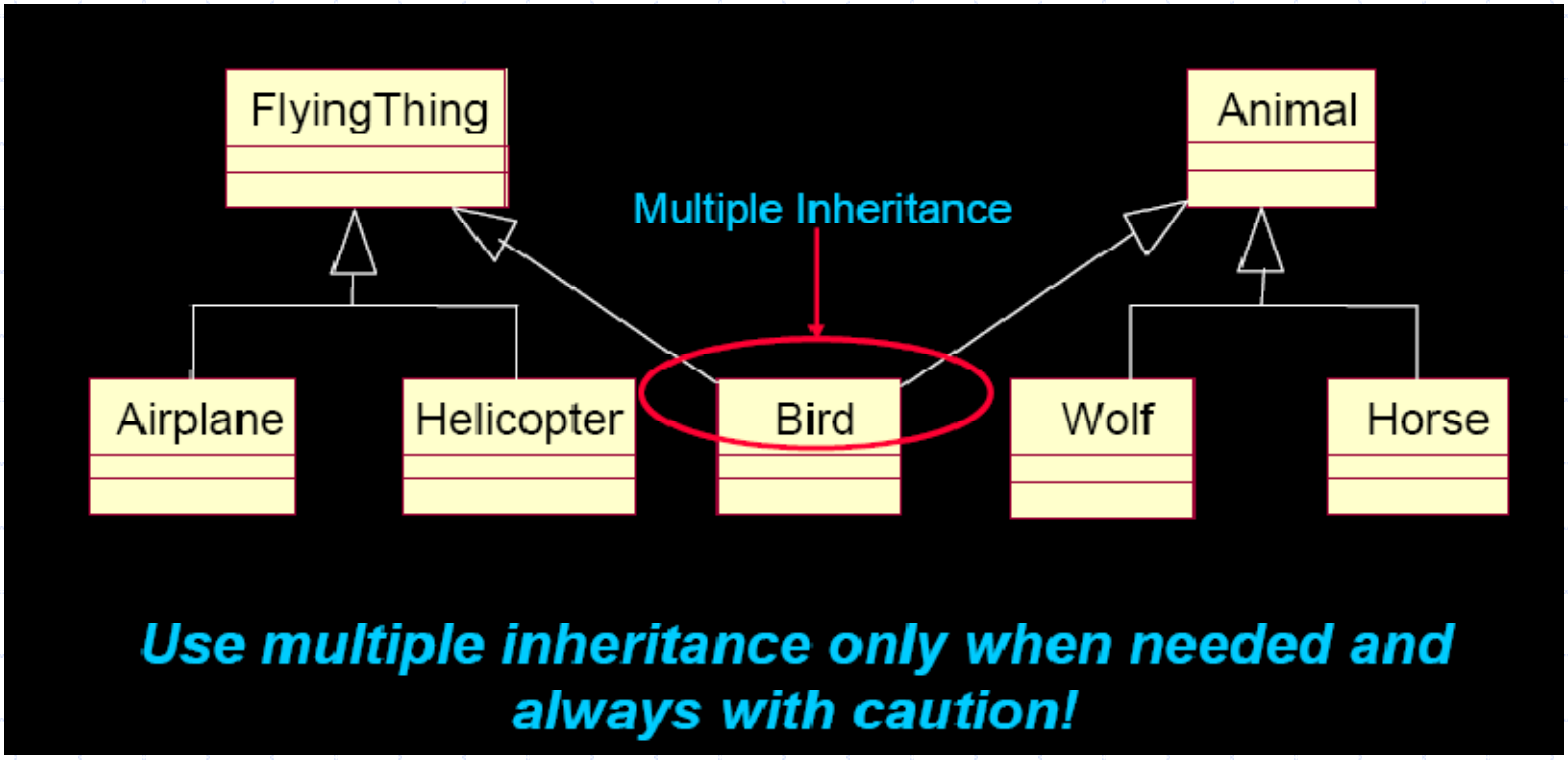
Generalization

- Generalization adalah hubungan diantara class-class dimana suatu class membagi struktur dan atau behaviour dengan class yang lain
- Mendefinisikan hirarki abstraksi dimana sebuah subclass mewarisi sifat dari satu atau lebih superclass → single inheritance, multiple inheritance

Contoh Single Inheritance



Contoh Multiple Inheritance

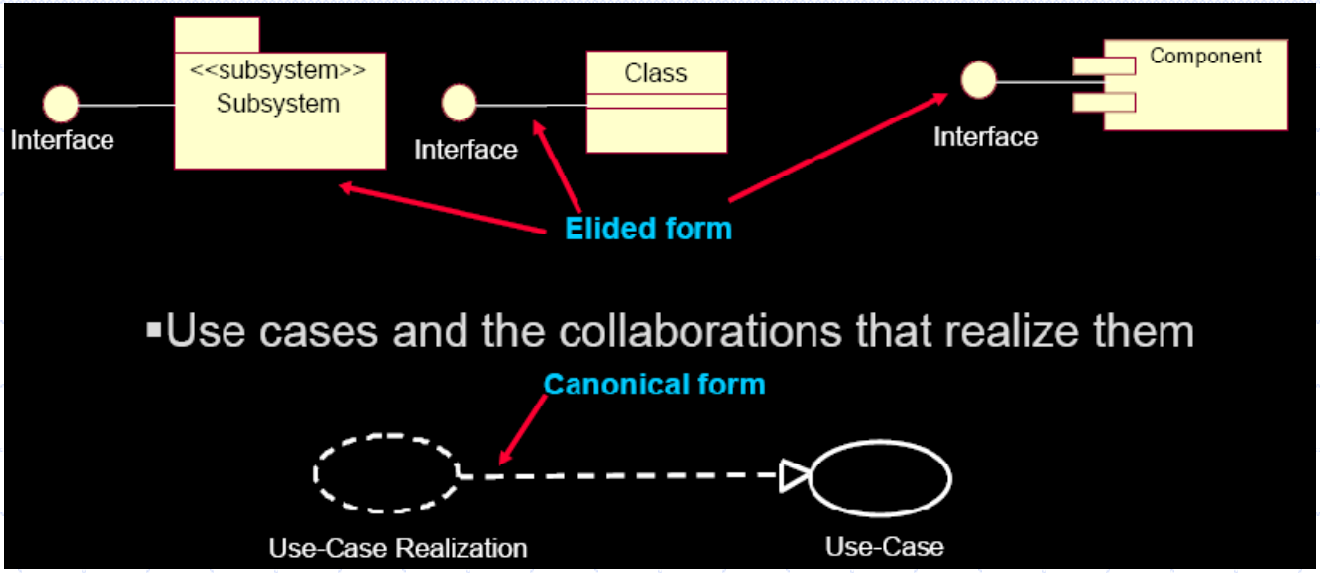


Hal-hal yang diwariskan

- Sebuah subclass mewarisi atribut,operation dan relationship superclassnya.
- Sebuah subclass bisa :
  - Menambah atribut, operation dan relationship
  - Mendefinisikan ulang operation-operation
- Atribut, operation, dan relationship umum diperlihatkan pada level tertinggi didalam hirarki

Realization

- Sebuah classifier bertugas sesuai dengan perjanjian yang disetujui classifier lain.
- Realization dapat ditemui antara interface dan classifier yang merealisasikannya.



Stereotype

- Stereotype mendefinisikan elemen model baru dalam model elemen yang lain.