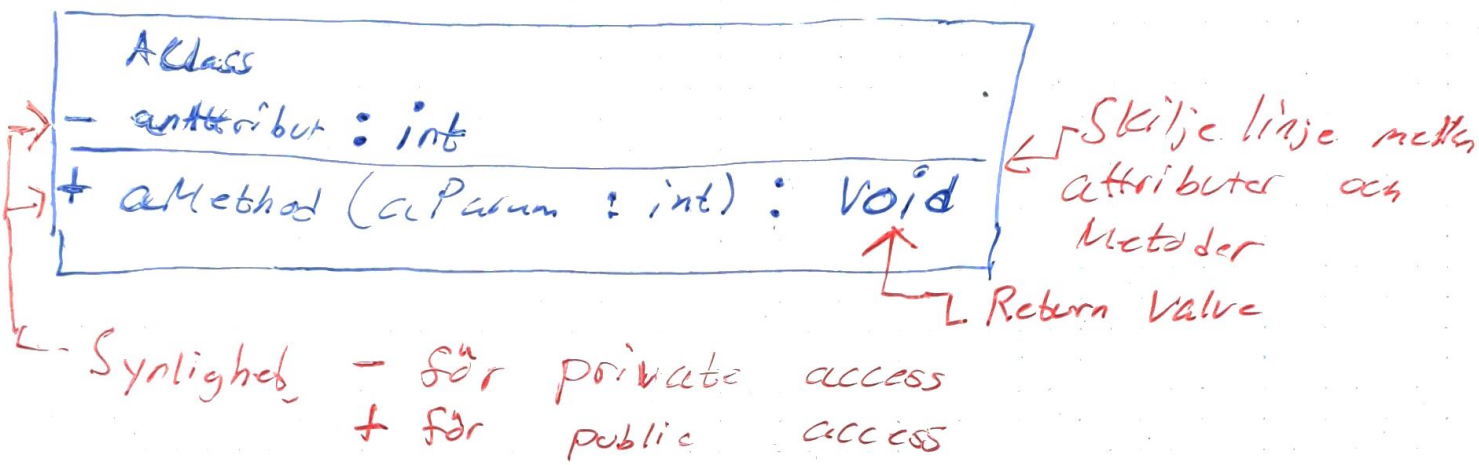
OD - Ch 5. (F4)

5.1 UML

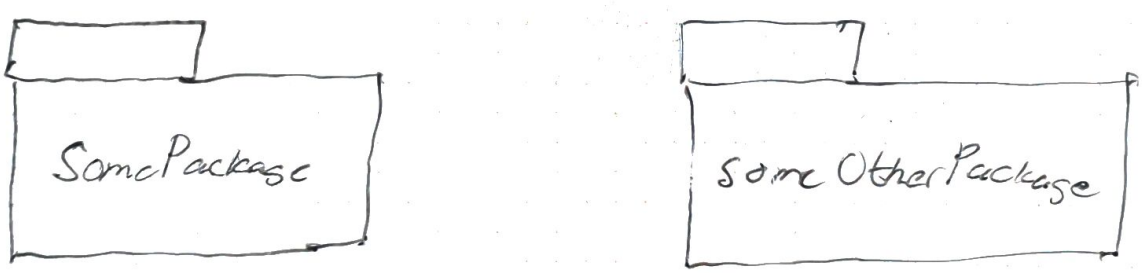
Package diagram, Communication Diagram.

Class and sequence diagram.

Class Diagram:



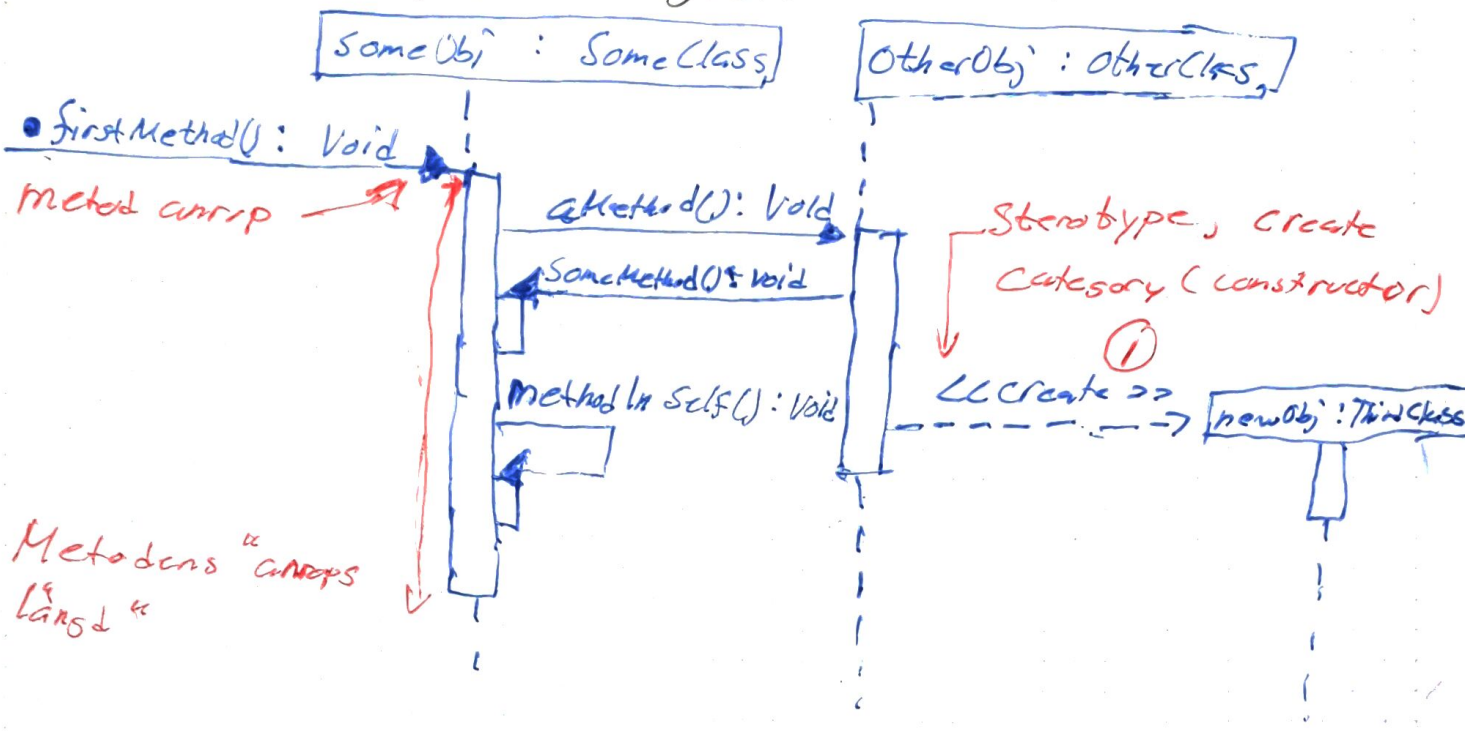
Package Diagram:



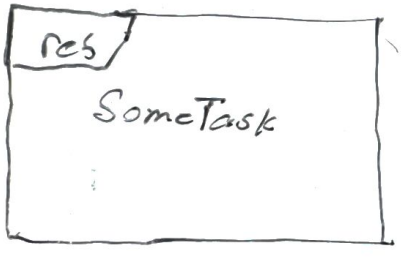
Grupperingar tex Java package för Java Packages

Säger inget om beroenden.

Sequence Diagram:



1 << static >> Statisk metod

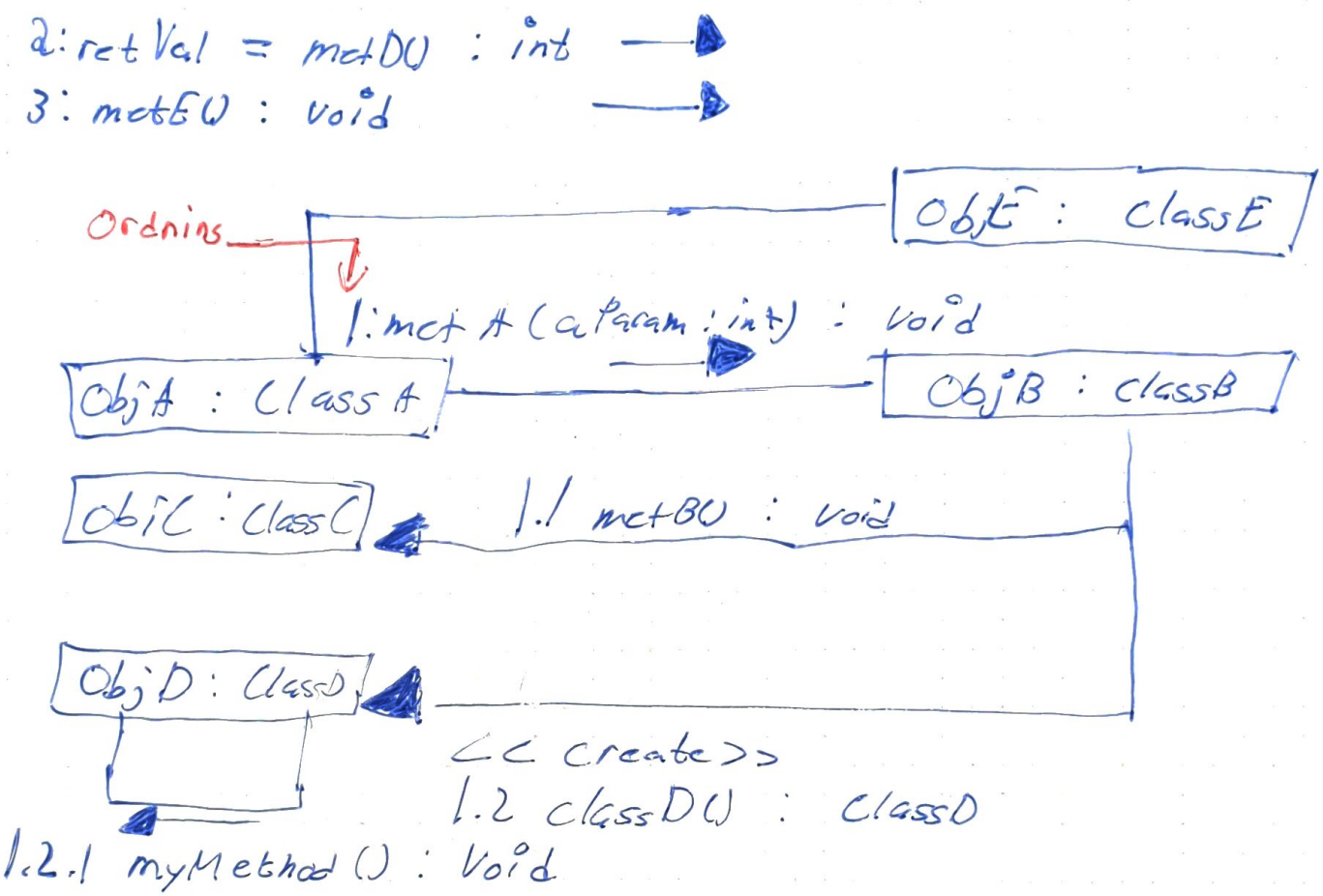


Flera metod anrop, om det blir otydligt med alla eller pilar bakom.

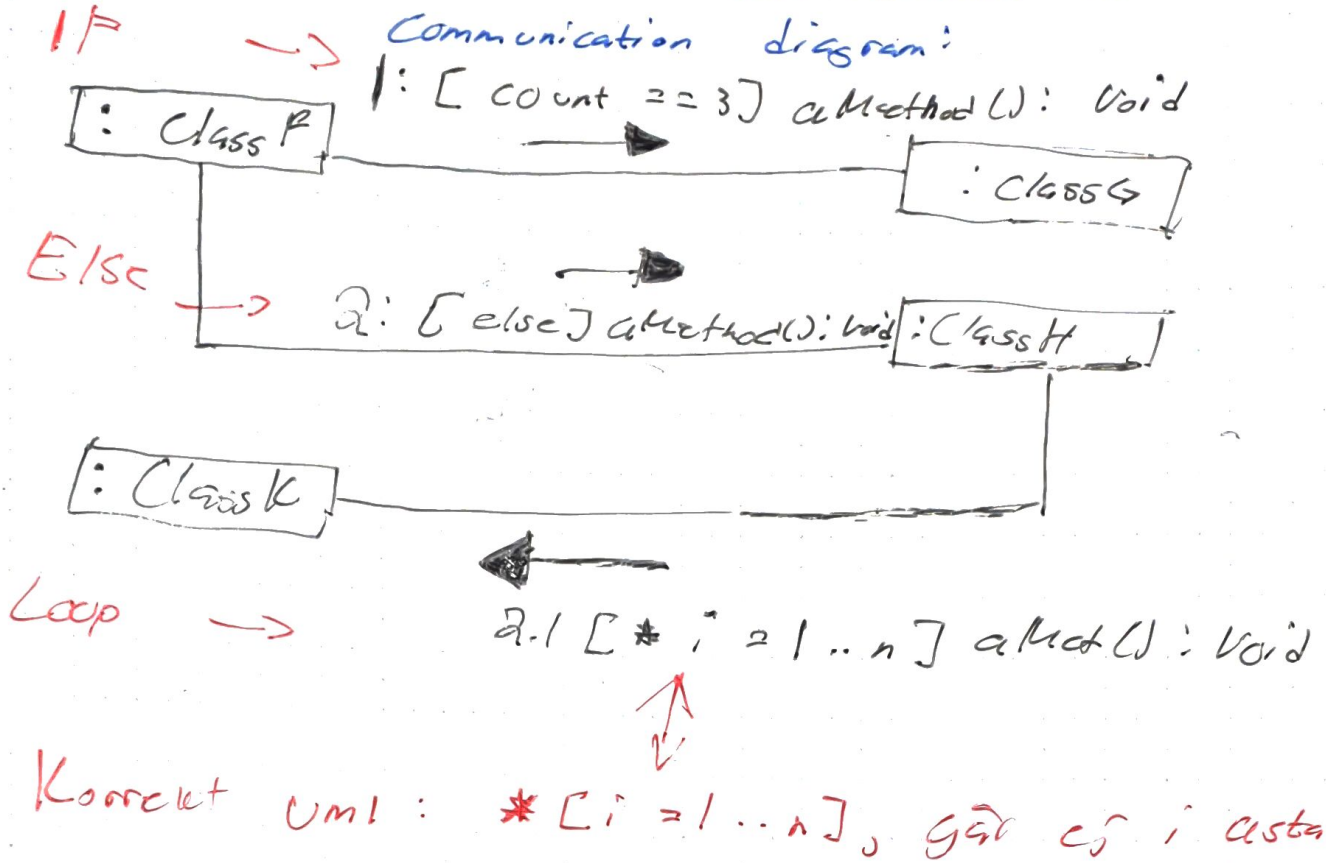
Communication Diagram.

Har samma syfte som ett sequense diagram, illustrera flödet av meddelanden mellan objekt. Saknar tidsaxel.

Kan läggas till både vertikalt och horisontellt.



Conditional Call and iteration in Communication diagram:



Korrekt uml: *[i = 1..n], går ej i astah.

5.2 - Design Concepts

Encapsulation:

Menar att man gömmer irrelevanta interna detaljer.

- Public
- Private

Poängen är att man kan ändra internt utan att ändra publika interface, (API).

API, så liten som möjligt.

Cohesion:-- (Sammanhållning)

Mäter hur väl definierad en klass är och hur dess tasks passar ihop. Klassen ska bara känna till den abstraktion och utföra uppgifter som är relaterade till den.

Ex Class Employee har metod getAllEmployees() (dvs låg cohesion)

Alltid sträva efter högre cohesion

Low coupling: (Koppling)

Mäter antalet beroenden finns inget mått, eller vad det gör, men inga beroenden som ej behövs. Låg coupling = Bra.

Kolla efter "spider in the webb" [Fig 5.13]