



PXL – IT

42TIN280 Software Analysis - System & System Context – Domain Model Cheat sheet

Week 05 – semester 01

Luc Doumen

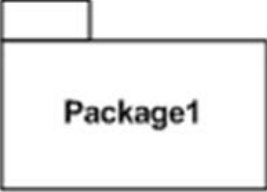
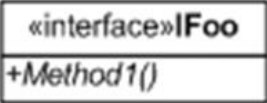
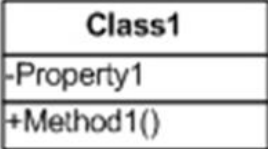







Nathalie Fuchs

**DE HOGESCHOOL
MET HET NETWERK**

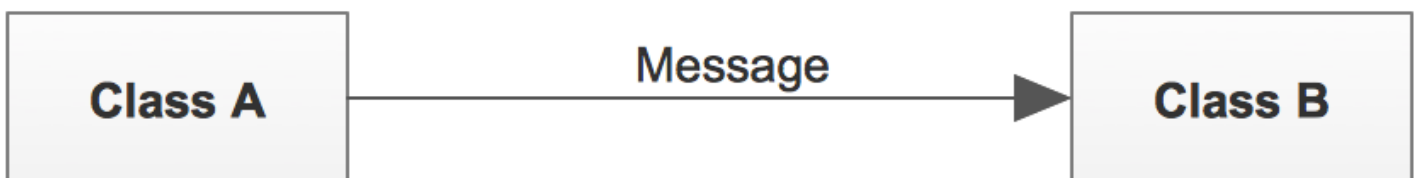
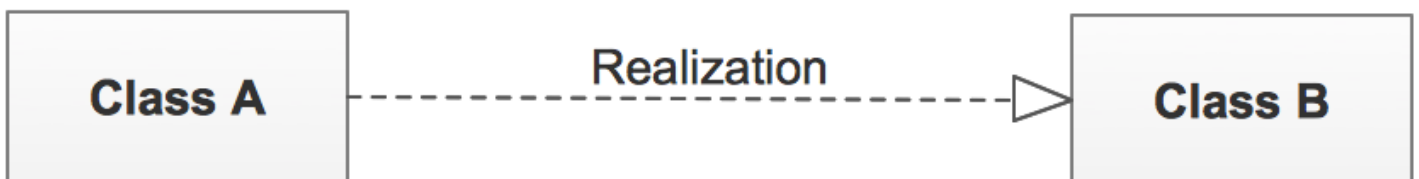
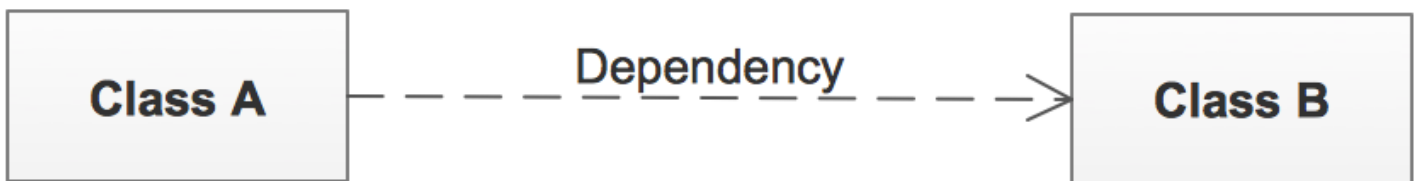
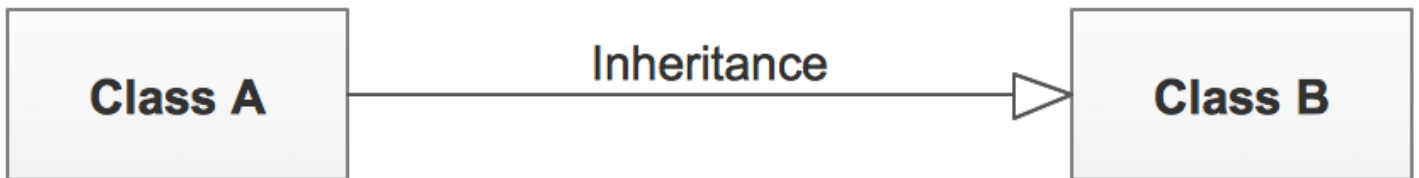
Hogeschool PXL – Elfde-Liniestraat 24 – B-3500 Haasrode
www.pxl.be - www.pxl.be/facebook



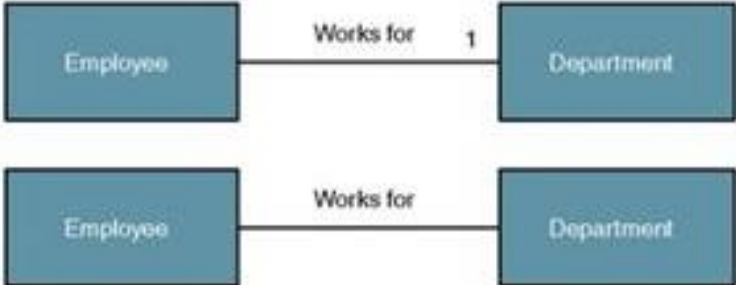


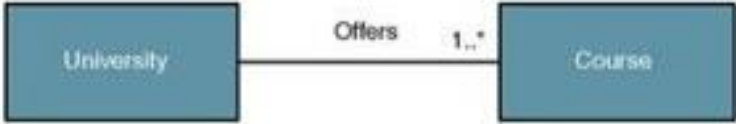
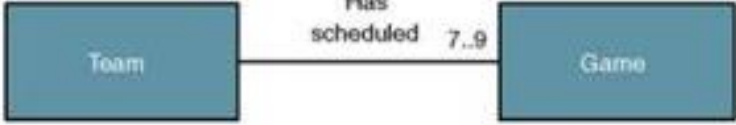
UML Domain modeling (1)

Shape	Description
	Package A collection of interfaces and classes.
	Interface Microsoft guidelines specify that interfaces should start with I. This graphic can also sometimes be used as an abstract class.
	Class Properties or attributes sit at the top, methods or operations at the bottom. + indicates public and # indicates protected.
These are both typically drawn vertically:	
	Inheritance - B inherits from A. "is-a" relationship.
	Generalization - B implements A,
	Association - A and B call each other
	One way Association. A can call B's properties/methods, but not visa versa.
	Aggregation A "has-a" instance of B. B can survive if A is disposed.
	Composition A has an instance of B, B cannot exist without A.
	A note Some descriptive text attached to any item.

UML Domain modeling (2)



UML Domain modeling (3)

Multiplicity	UML Multiplicity Notation	Association with Multiplicity	Association Meaning
Exactly 1	1 or <i>leave blank</i>	 <pre> graph LR Employee[Employee] --- "Works for" Department[Department] style Department stroke-width:2px </pre>	An employee works for one and only one department.
Zero or 1	0..1	 <pre> graph LR Employee[Employee] --- "Has" Spouse[Spouse] style Spouse stroke-width:2px </pre>	An employee has either one or no spouse.
Zero or more	0..* or *	 <pre> graph LR Customer[Customer] --- "Makes" Payment[Payment] style Payment stroke-width:2px </pre>	A customer can make no payment up to many payments.
1 or more	1..*	 <pre> graph LR University[University] --- "Offers" Course[Course] style Course stroke-width:2px </pre>	A university offers at least 1 course up to many courses.
Specific range	7..9	 <pre> graph LR Team[Team] --- "Has scheduled" Game[Game] style Game stroke-width:2px </pre>	A team has either 7, 8, or 9 games scheduled

UML Domain modeling – General steps

1. Prepare problem statement for the system being developed
2. Identify concepts (these are the classes & objects)
3. Develop a common vocabulary, dictionary, glossary
 - a) Make an alphabetic list
 - b) Count the occurrences
 - c) Make a glossary of terms ➔ domain classes
 - d) Create a first domain class diagram
4. Identify associations between concepts
5. Assign attributes to the concepts
6. Check for multiplicities and indicate in domain model
7. Iterate and refine the model