ΑΝΤΙΚΕΙΜΕΝΟΣΤΡΑΦΗΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ ΙΙ

Κωδικός Θ: ΤΠΕ00Λ, Κωδικός Ε: ΤΠΕ10Λ
Ώρες (Θ - ΑΠ - Ε): 2 - 1 - 2
Προαπαιτούμενα: Αντικειμενοστραφής Προγραμματισμός

Software Modeling
UML

ΘΕΩΡΙΑ & ΕΡΓΑΣΤΗΡΙΟ 2

Σημερινό μάθημα

- Software Modeling Languages
- UML
  - Use Case Diagrams
  - Class Diagrams
  - Sequence Diagrams
- UML tools
  - Visual Paradigm
- Ασκήσεις

Modelling Languages Definition

Modelling Language

- A modeling language is any artificial language that can be used to express information or knowledge or systems in a structure that is defined by a consistent set of rules. The rules are used for interpretation of the meaning of components in the structure.

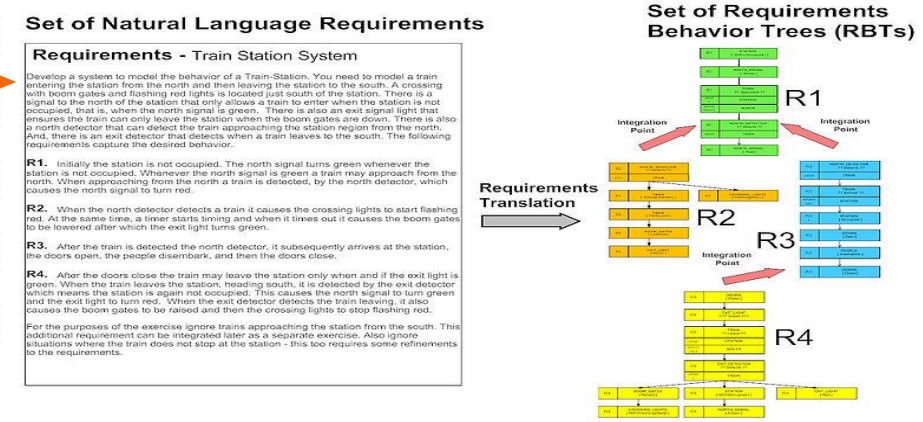
A modeling language can be graphical or textual.

- Graphical modeling languages use a diagram technique with named symbols that represent concepts and lines that connect the symbols and represent relationships and various other graphical annotation to represent constraints.
- Textual modeling languages typically use standardized keywords accompanied by parameters to make computer-interpretable expressions.

Modeling Languages: Graphical

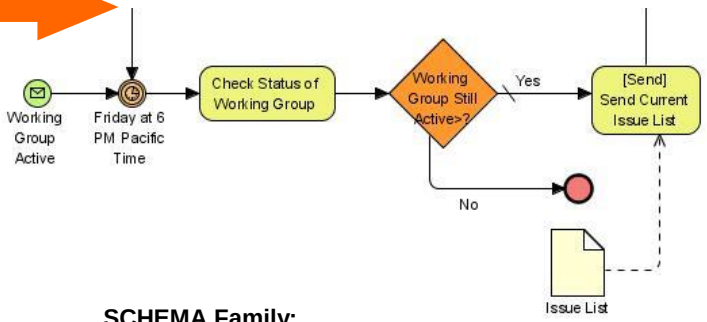
Behavior Trees

are a formal, graphical modelling language used primarily in systems and software engineering.



Business Process Modeling Notation

(BPMN, and the XML form BPML) is an example of a Process Modeling language.

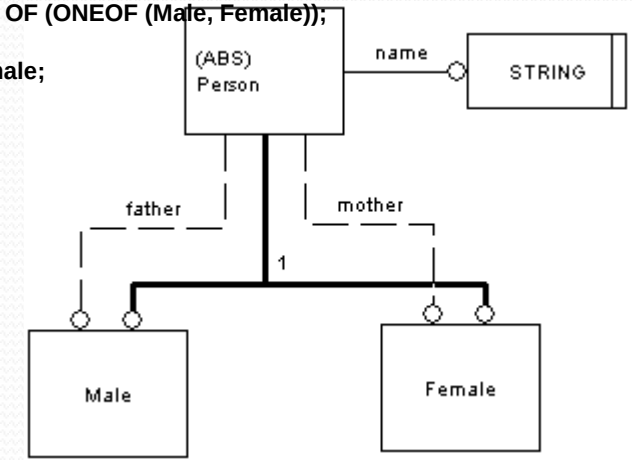


EXPRESS and EXPRESS-G

(ISO 10303-11) is an international standard general-purpose data modeling language.

Extended Enterprise Modeling Language (EEML) is commonly used for business process modeling across a number of layers.

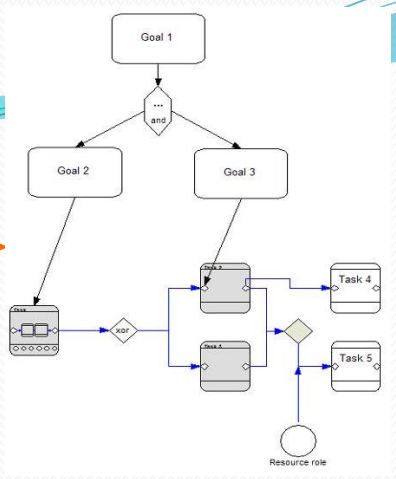
```
SCHEMA Family;
ENTITY Person
  ABSTRACT SUPERTYPE OF (ONEOF (Male, Female));
    name: STRING;
    mother: OPTIONAL Female;
    father: OPTIONAL Male;
END_ENTITY;

ENTITY Female
  SUBTYPE OF (Person);
END_ENTITY;

ENTITY Male
  SUBTYPE of (Person);
END_ENTITY;

END_SCHEMA;
```



Modeling Languages: Graphical

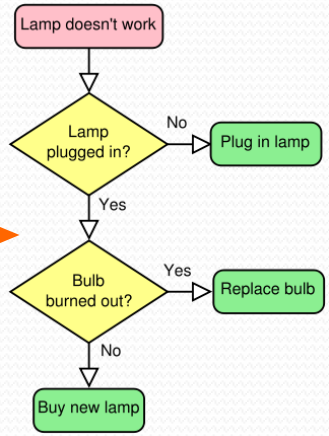
Extended Enterprise Modeling Language (EEML)

is commonly used for business process modeling across a number of layers.



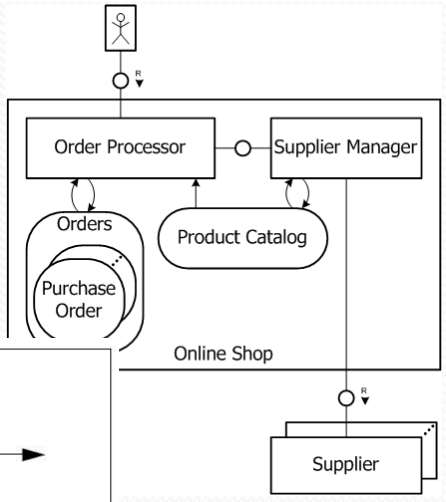
Flowchart

is a schematic representation of an algorithm or a stepwise process,



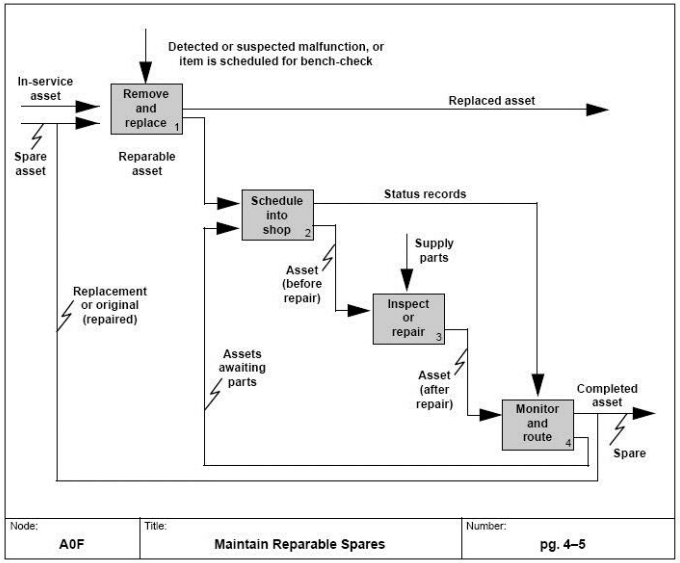
Fundamental Modeling Concepts (FMC)

modeling language for software-intensive systems.



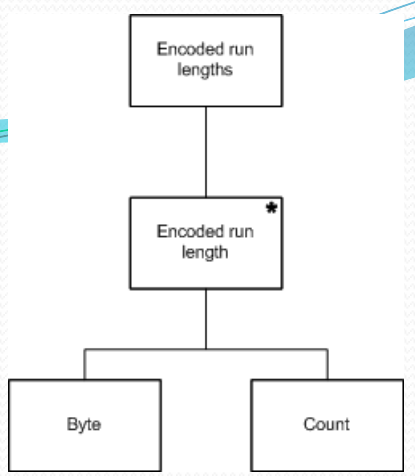
IDEF

is a family of modeling languages, which include IDEF0 for functional modeling, IDEF1X for information modeling, IDEF3 for business process modelling, IDEF4 for Object-Oriented Design and IDEF5 for modeling ontologies.

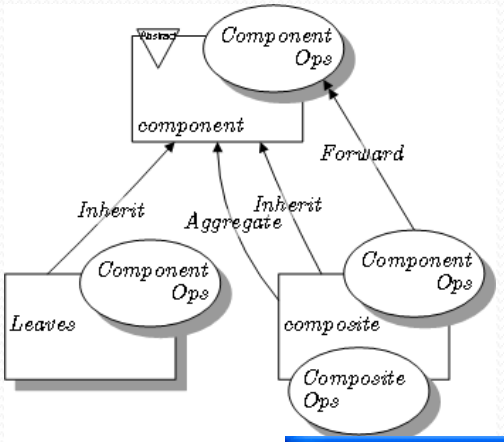


Modeling Languages: Graphical

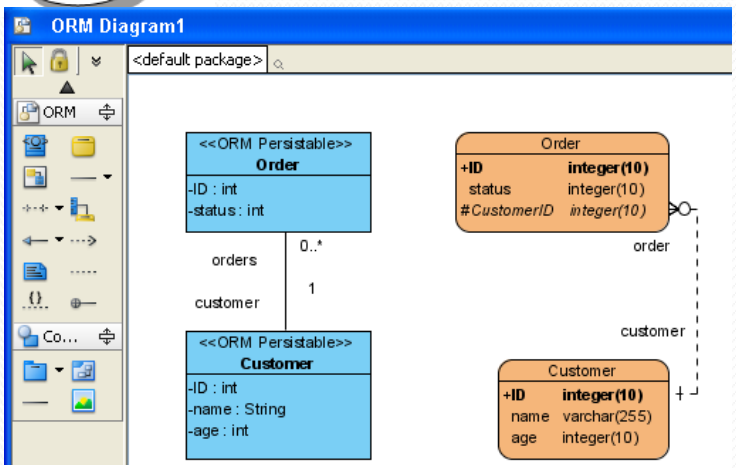
Jackson Structured Programming
(JSP) is a method for structured programming based on correspondences between data stream structure and program structure



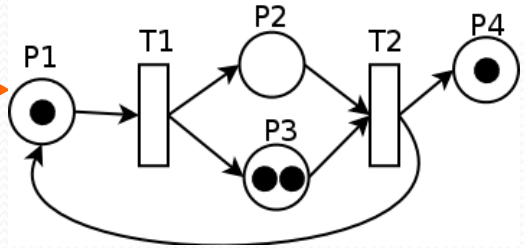
LePUS3
is an object-oriented visual Design Description Language and a formal specification language that is suitable primarily for modelling large object-oriented (Java, C++, C#) programs and design patterns.



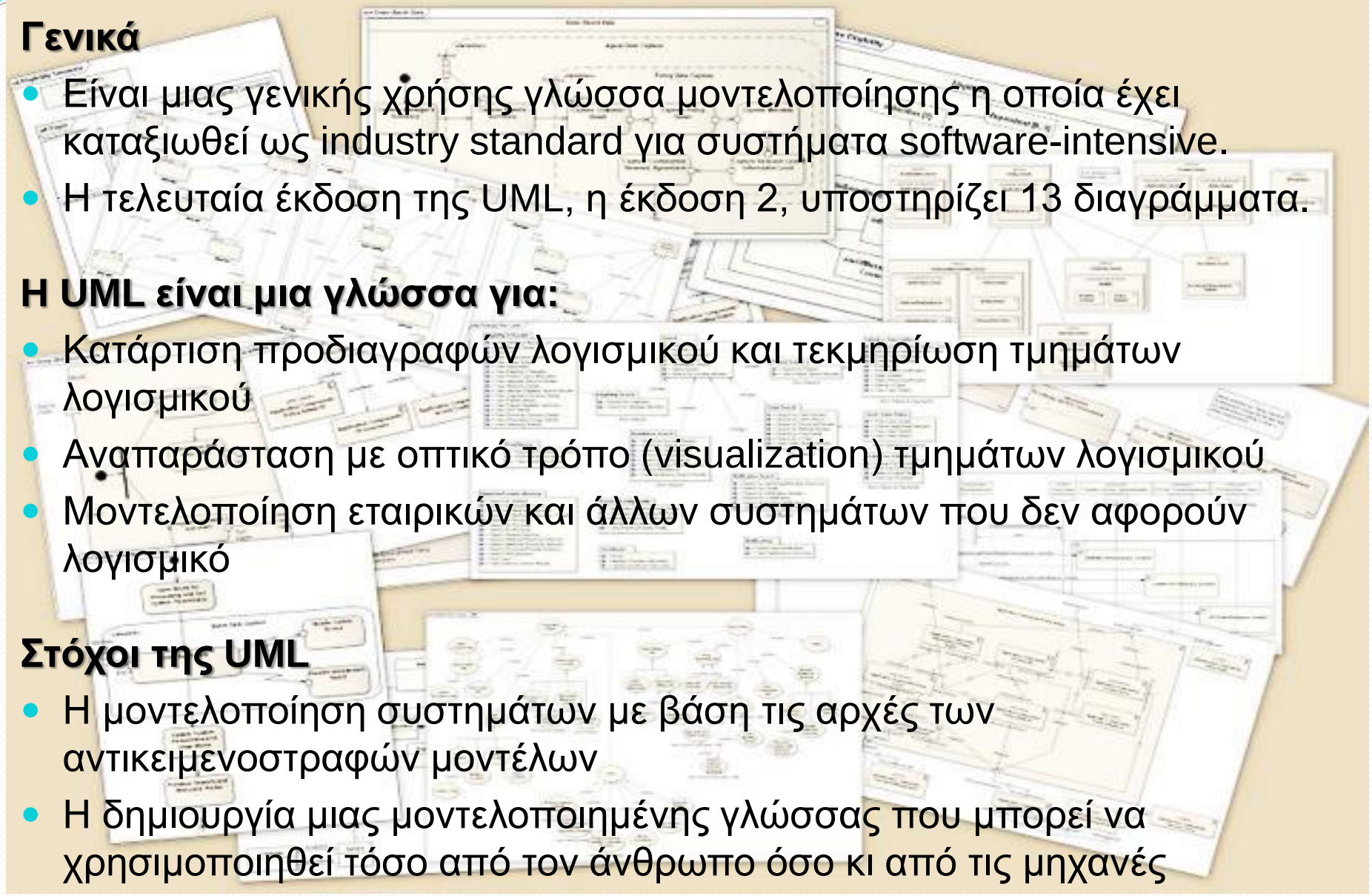
Object Role Modeling
(ORM) in the field of software engineering is a method for conceptual modeling, and can be used as a tool for information and rules analysis.



Petri nets
The simplicity of its basic user interface easily enabled extensive tool support over the years, particularly in the areas of model checking, graphically-oriented simulation, and software verification.



Modeling Languages: UML (Unified Modeling Language)



Γενικά

- Είναι μιας γενικής χρήσης γλώσσα μοντελοποίησης η οποία έχει καταξιωθεί ως industry standard για συστήματα software-intensive.
- Η τελευταία έκδοση της UML, η έκδοση 2, υποστηρίζει 13 διαγράμματα.

Η UML είναι μια γλώσσα για:

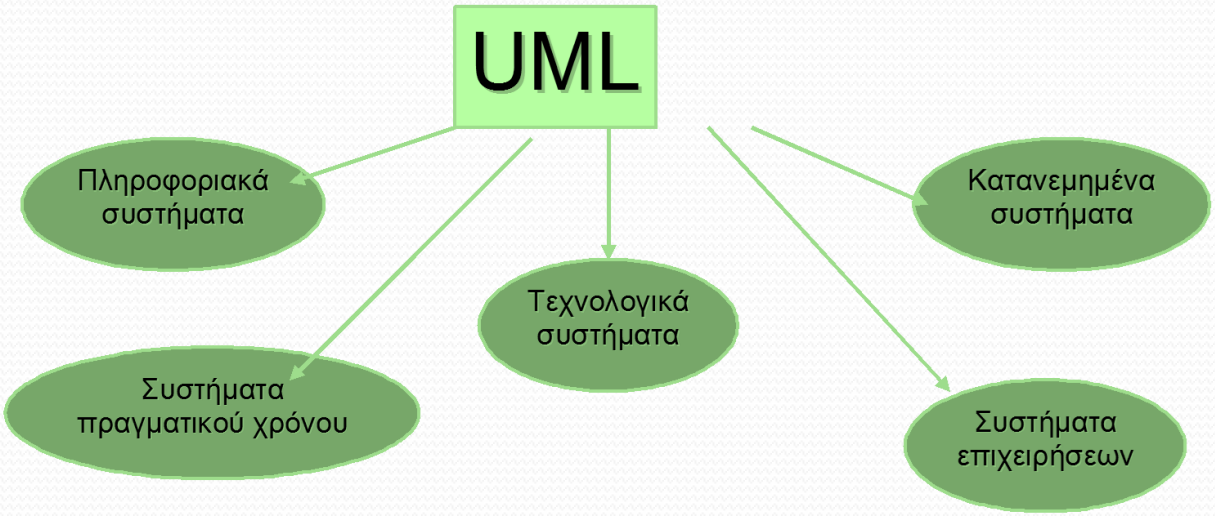
- Κατάρτιση προδιαγραφών λογισμικού και τεκμηρίωση τμημάτων λογισμικού
- Αναπαράσταση με οπτικό τρόπο (visualization) τμημάτων λογισμικού
- Μοντελοποίηση εταιρικών και άλλων συστημάτων που δεν αφορούν λογισμικό

Στόχοι της UML

- Η μοντελοποίηση συστημάτων με βάση τις αρχές των αντικειμενοστραφών μοντέλων
- Η δημιουργία μιας μοντελοποιημένης γλώσσας που μπορεί να χρησιμοποιηθεί τόσο από τον άνθρωπο όσο κι από τις μηχανές

Modeling Languages: UML (Unified Modeling Language)

Η χρήση της UML

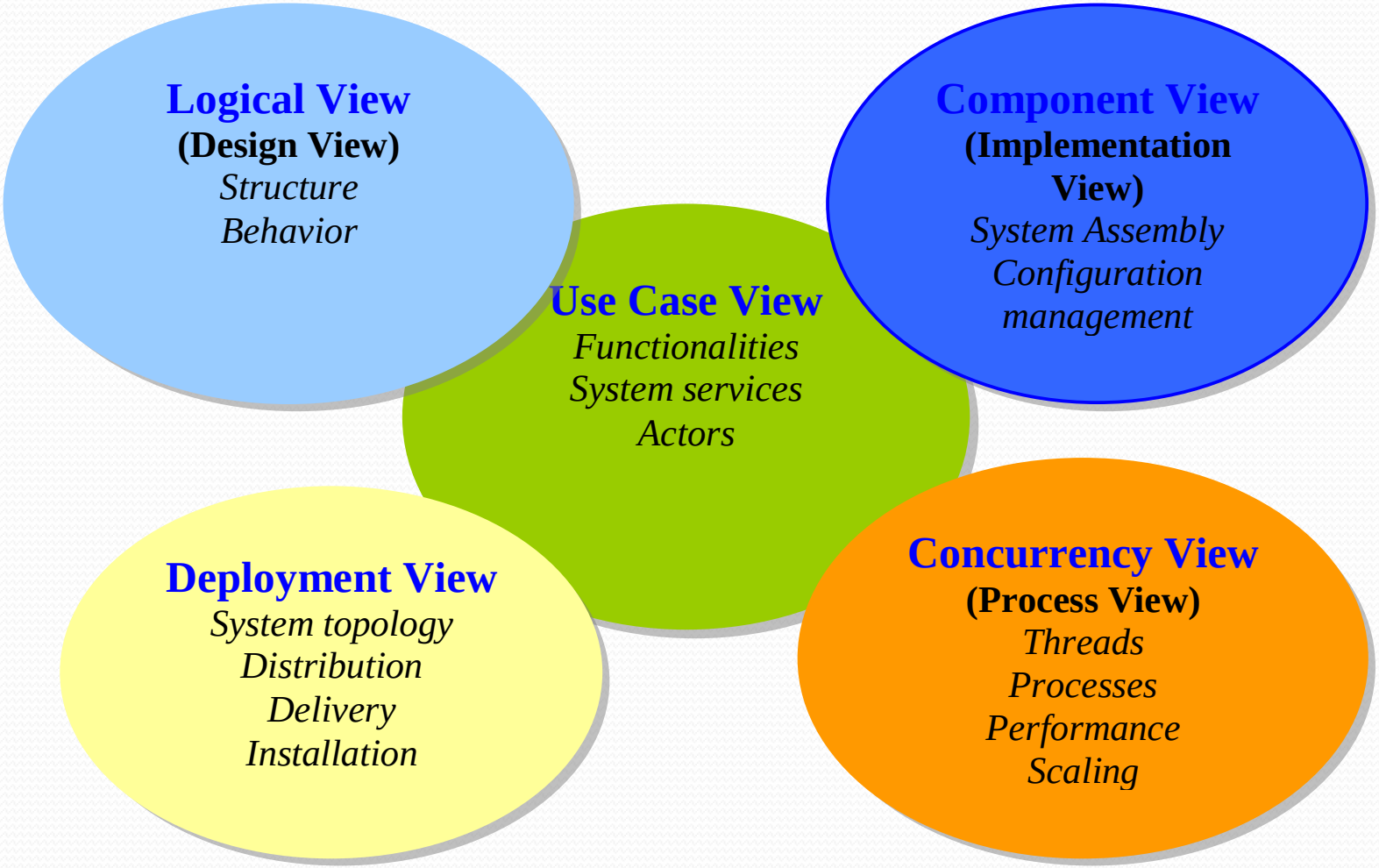


Τμήματα της UML

- Όψεις: Δείχνουν διαφορετικά χαρακτηριστικά του συστήματος που μοντελοποιούνται. Μια όψη αποτελείται από ένα σύνολο διαγραμμάτων
- Διαγράμματα: Περιγράφουν τα περιεχόμενα μιας όψης. Υπάρχουν εννέα διαφορετικά διαγράμματα που χρησιμοποιούνται σε συνδυασμό
- Στοιχεία μοντέλου: Είναι οι έννοιες που χρησιμοποιούνται στα διαγράμματα για να αναπαραστήσουν τις κλάσεις, τα αντικείμενα και τις μεταξύ τους συσχετίσεις

Modeling Languages: UML (Unified Modeling Language)

4 + 1 Όψεις της UML



Modeling Languages: UML (Unified Modeling Language)

Logical View

- **Επιδεικνύει** πως σχεδιάζεται η λειτουργικότητα του συστήματος.
- **Χρησιμοποιεί** διαγράμματα *class* και *object* για να αναπαραστήσει την στατική δομή του συστήματος.
- **Χρησιμοποιεί** διαγράμματα *state*, *sequence*, *collaboration*, και *activity* για να αναπαραστήσει την δυναμική συμπεριφορά του συστήματος.
- **Χρησιμοποιείται** από designers and developers.

Concurrency View

- **Αναφέρεται** σε θέματα επικοινωνίας και συγχρονισμού συστημάτων concurrent.
- **Χρησιμοποιεί / Περιγράφεται** από διαγράμματα *state*, *sequence*, *collaboration*, *activity*, *deployment*, και *component*.
- **Χρησιμοποιείται** από developers και system integrators.

Use Case View

- **Επιδεικνύει** την λειτουργικότητα του συστήματος όπως αυτή εννοείται από εξωτερικούς actors.
- Οι Actors μπορεί να είναι χρήστες ή άλλα συστήματα.
- **Χρησιμοποιεί / Περιγράφεται** από διαγράμματα *use case* και *activity*.
- Είναι η κεντρική προσέγγιση / σκοπιά (view) η οποία κατευθύνει την υλοποίηση τον υπόλοιπων.
- **Χρησιμοποιείται** από customers, designers, developers, testers.

Deployment View.

- **Επιδεικνύει** την ανάπτυξη (deployment) ενός συστήματος μέσα στην αρχιτεκτονική σχεδίαση του με computers και devices.
- **Περιγράφεται** από διαγράμματα *deployment* .
- **Χρησιμοποιείται** από developers, system integrators, και testers.

Component View

- **Επιδεικνύει** την οργάνωση των κομματιών (component) κώδικα και την αλληλοεξάρτιση τους.
- **Χρησιμοποιεί / Περιγράφεται** από διαγράμματα *component*.
- **Χρησιμοποιείται** από developers.

Modeling Languages: UML (Unified Modeling Language)

Τα διαγράμματα της UML

Structure Diagrams:

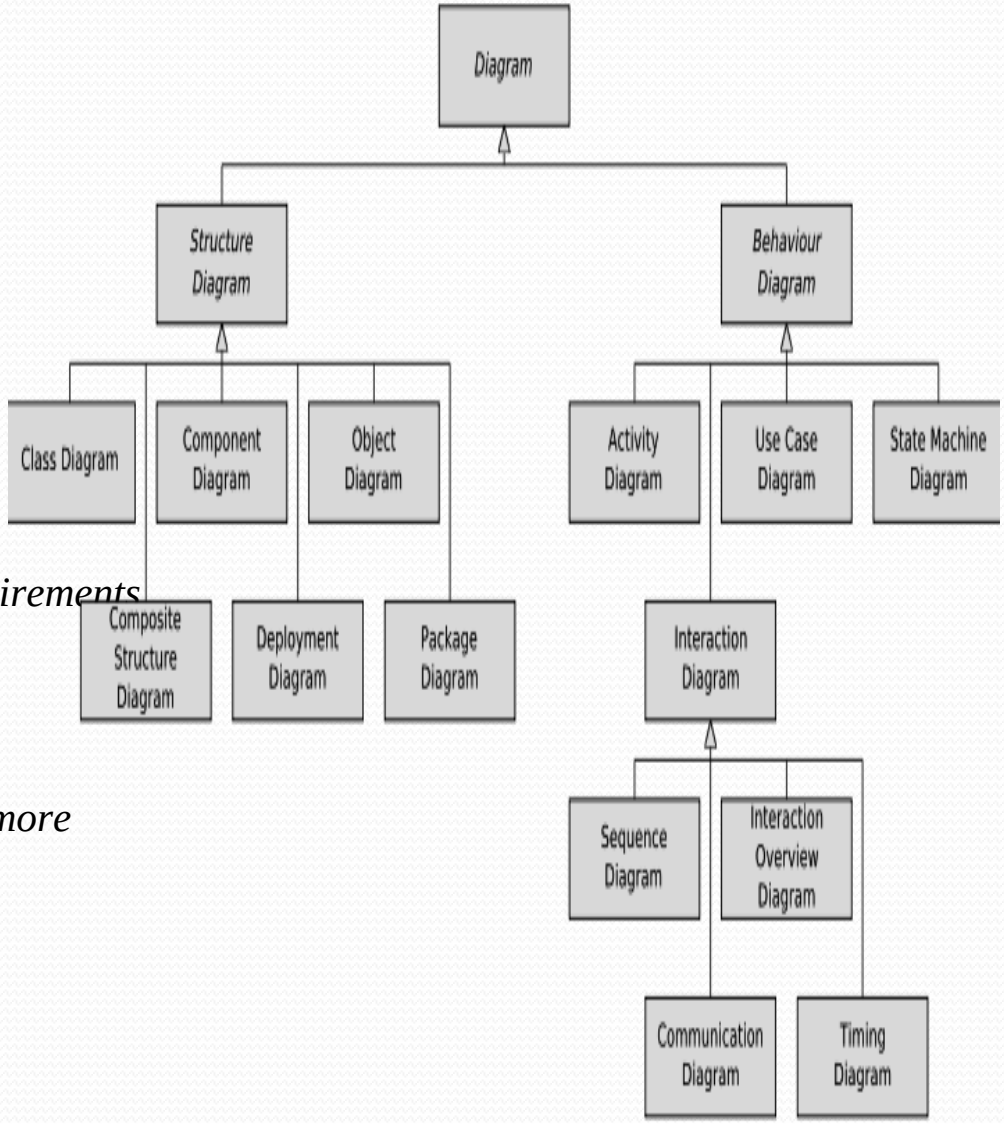
- Class Diagram,
- Object Diagram,
- Component Diagram,
- Composite Structure Diagram,
- Package Diagram,
- Deployment Diagram.

Behaviour Diagrams:

- Use Case Diagram
  (used by some methodologies during requirements gathering);
- Activity Diagram,
- State Machine Diagram.

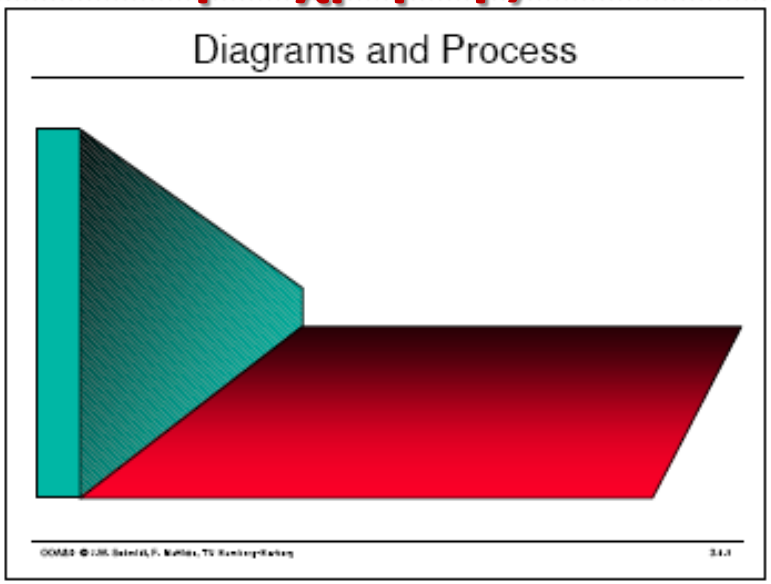
Interaction Diagrams: *all derived from the more general Behaviour Diagram,*

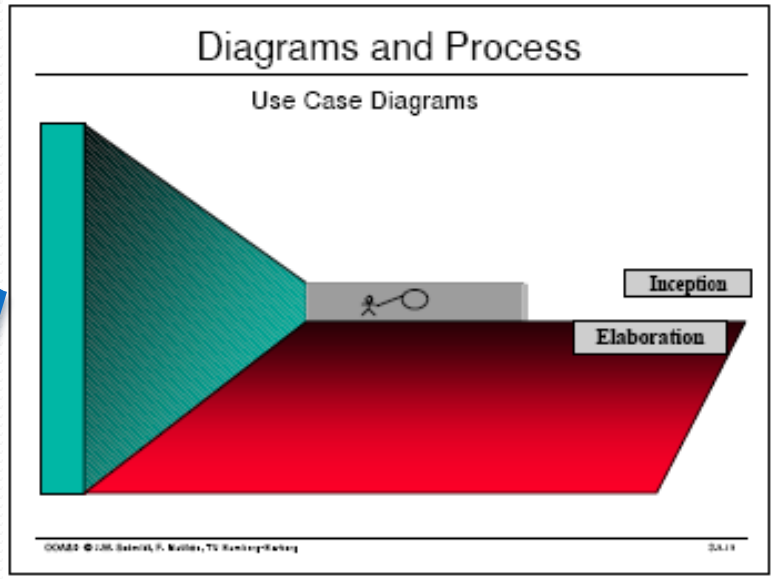
- *Sequence Diagram,*
- *Communication Diagram,*
- *Timing Diagram,*
- *Interaction Overview Diagram.*

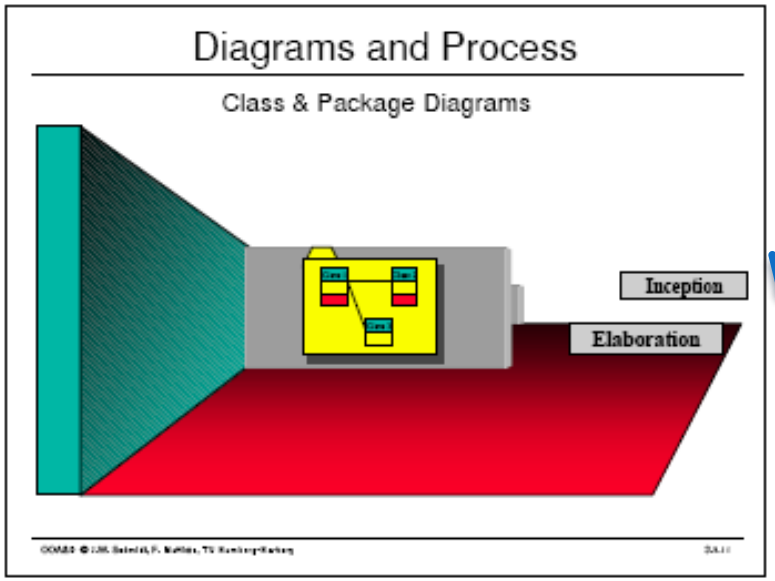


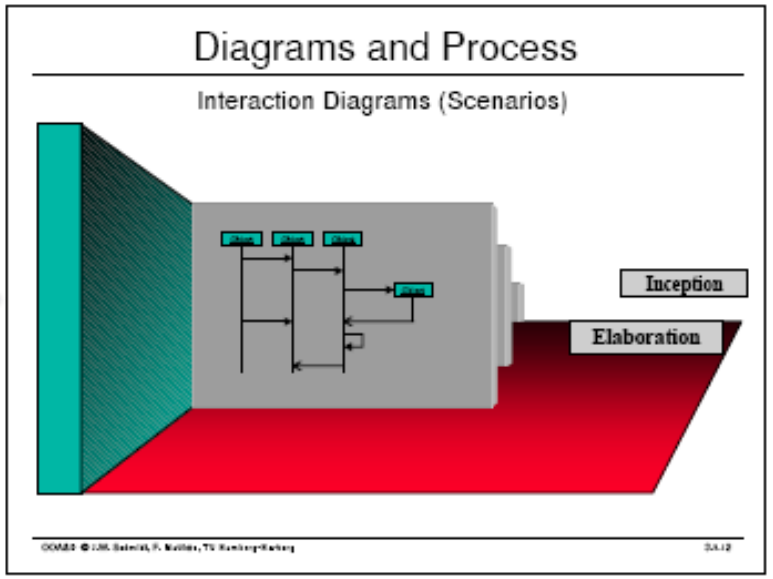
Modeling Languages: UML (Unified Modeling Language)

Μια σειρά χρήσης των διαγραμμάτων



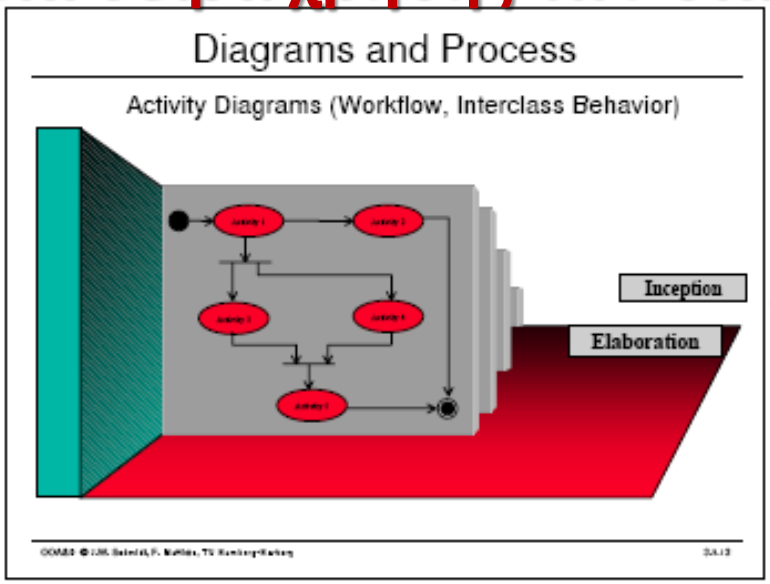


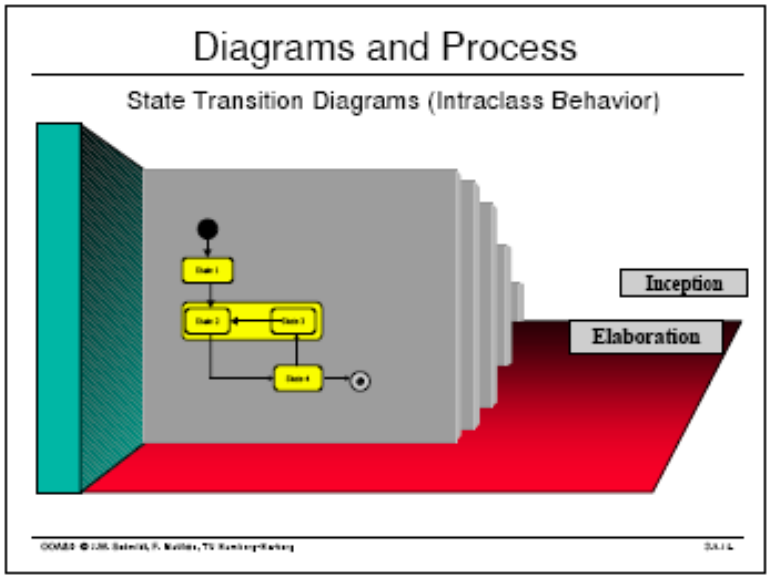


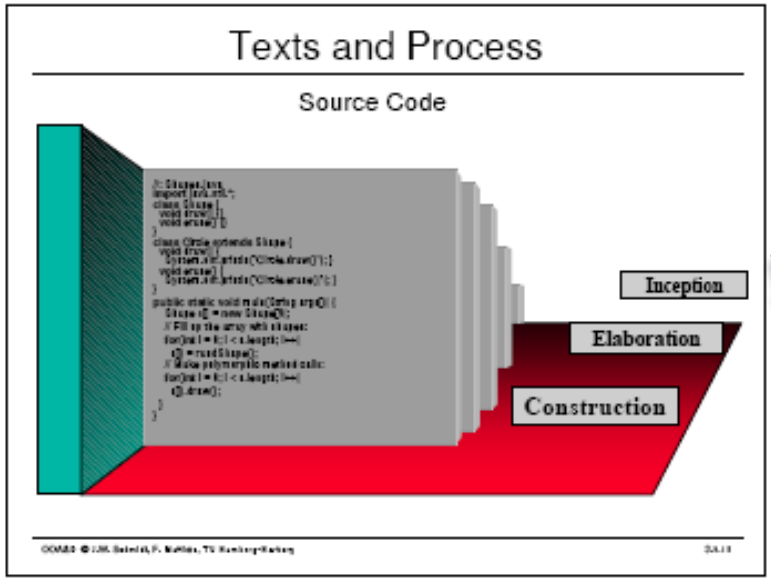


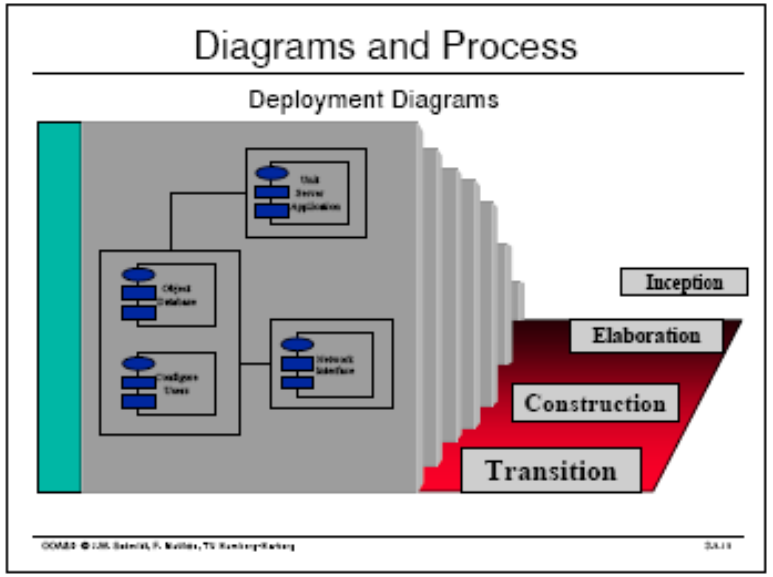
Modeling Languages: UML (Unified Modeling Language)

Μια σειρά χρήσης των διαγραμμάτων









Modeling Languages: UML (Unified Modeling Language)

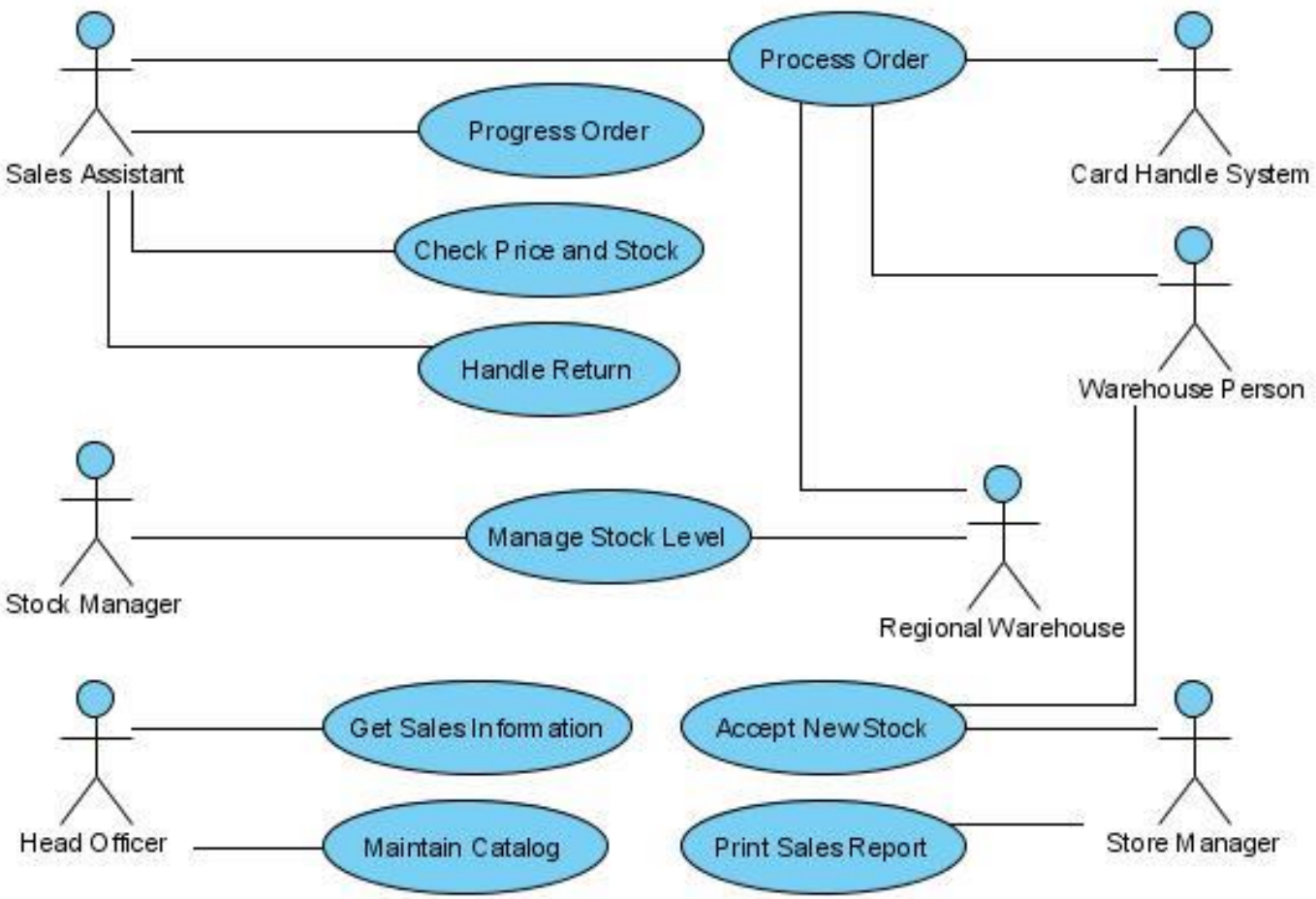
Use Case diagram, Class Diagram and Sequence Diagram

Στο μάθημα θα ασχοληθούμε με 3 από τα σπουδαιότερα διαγράμματα

- Το διάγραμμα Use Case
- Το διάγραμμα Class
- Το διάγραμμα Sequence

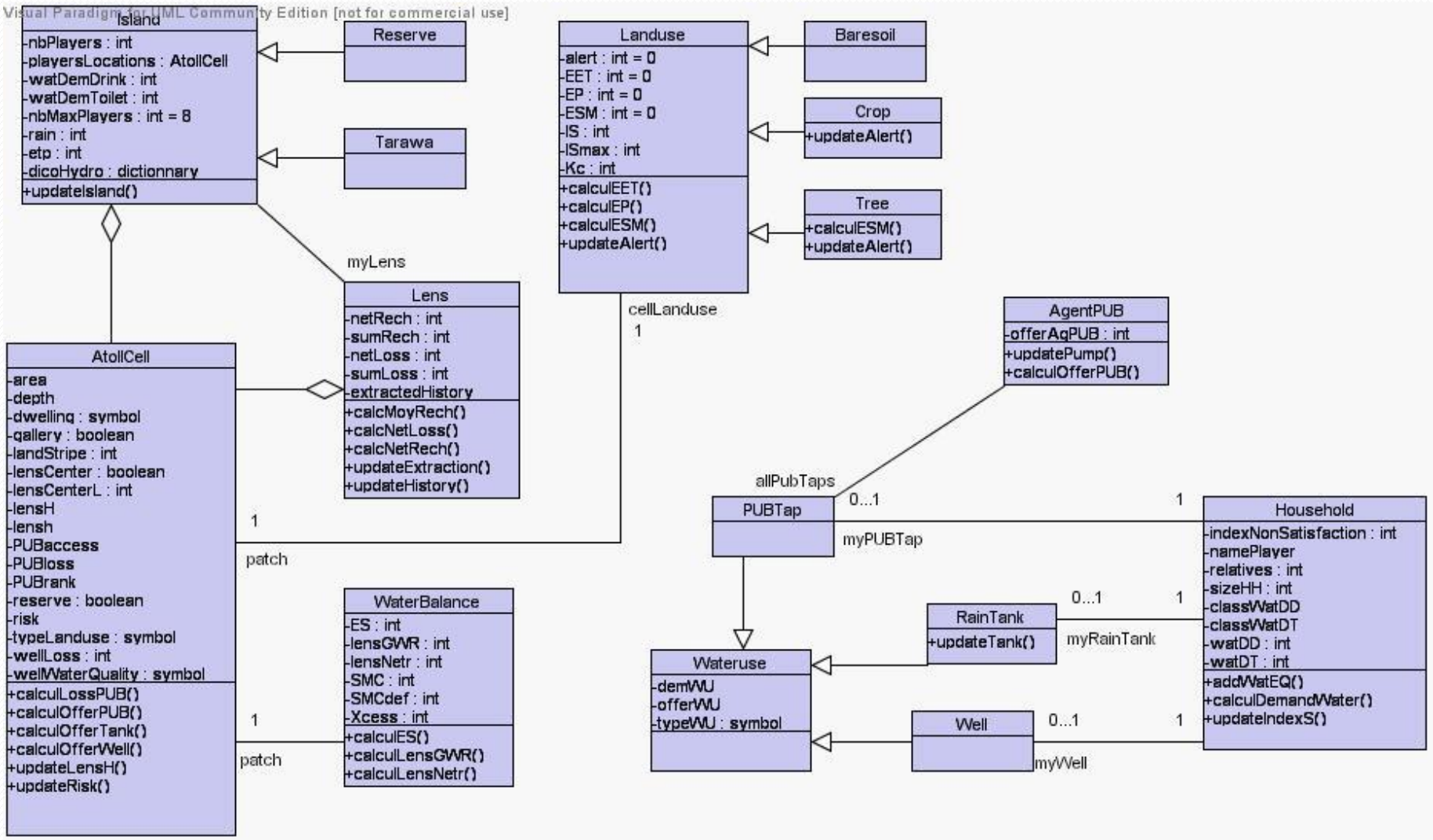
Modeling Languages: UML (Unified Modeling Language)

Use Case Diagram in Visual Paradigm



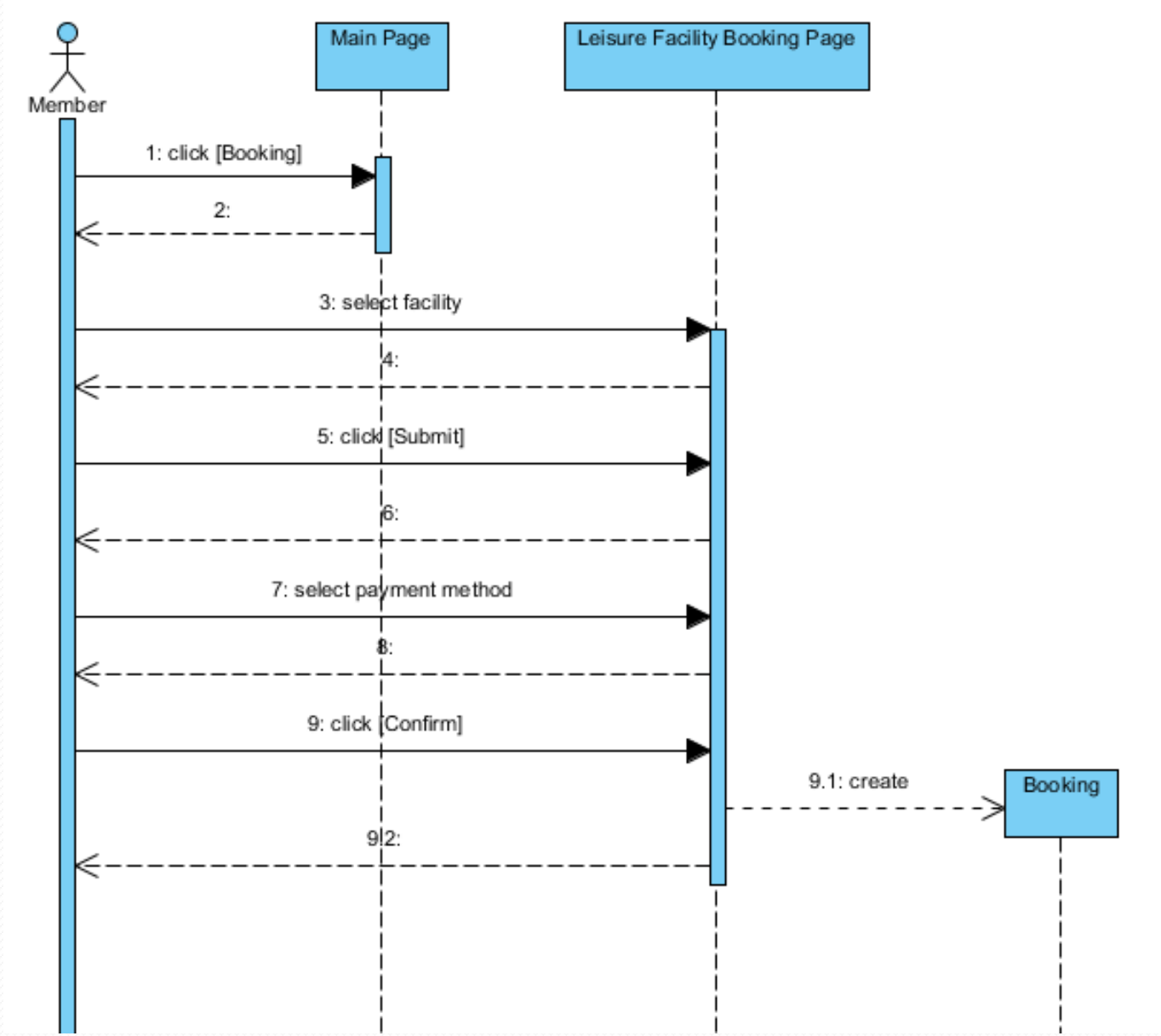
Modeling Languages: UML (Unified Modeling Language)

Class Diagram in Visual Paradigm



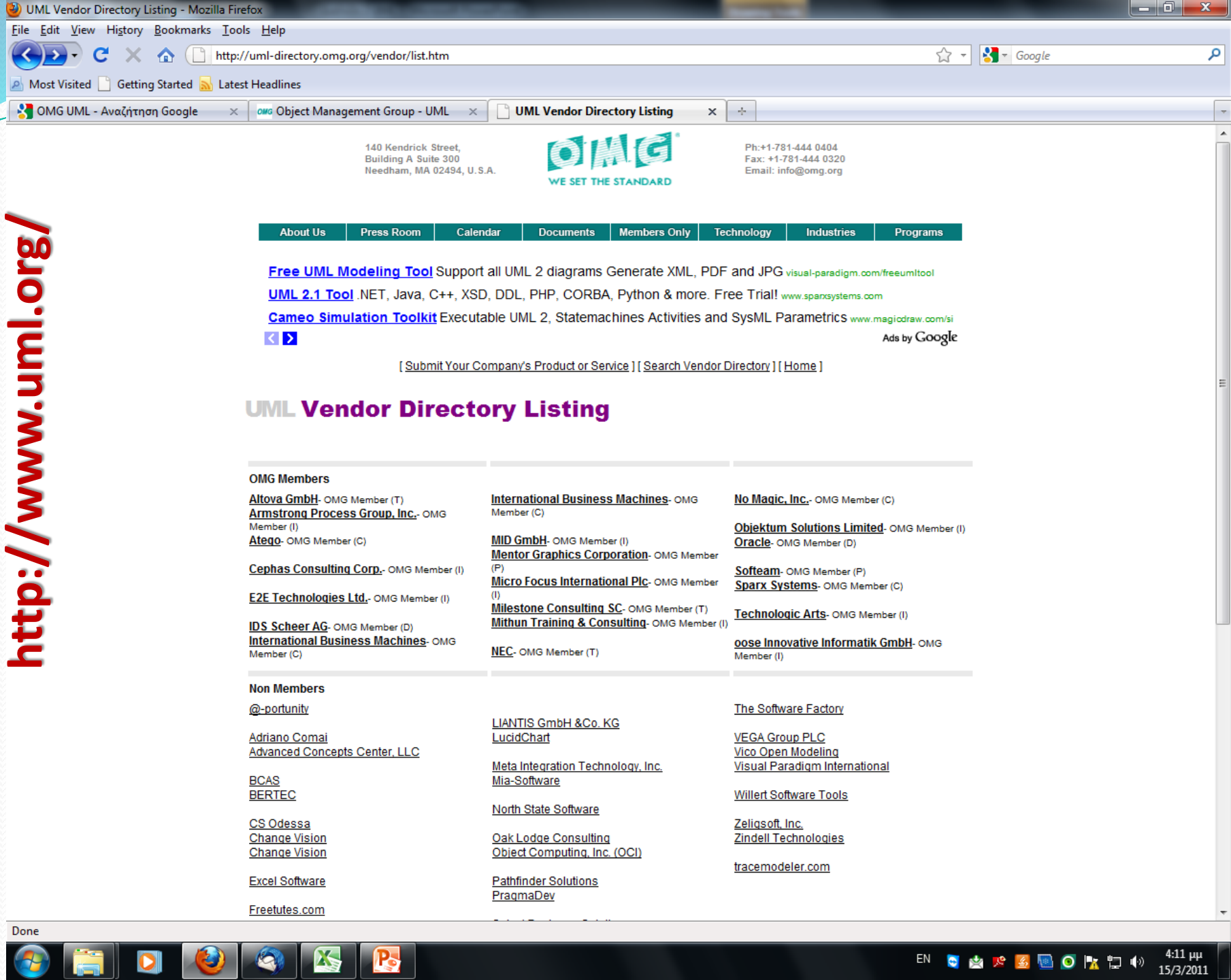
Modeling Languages: UML (Unified Modeling Language)

Sequence Diagram in Visual Paradigm



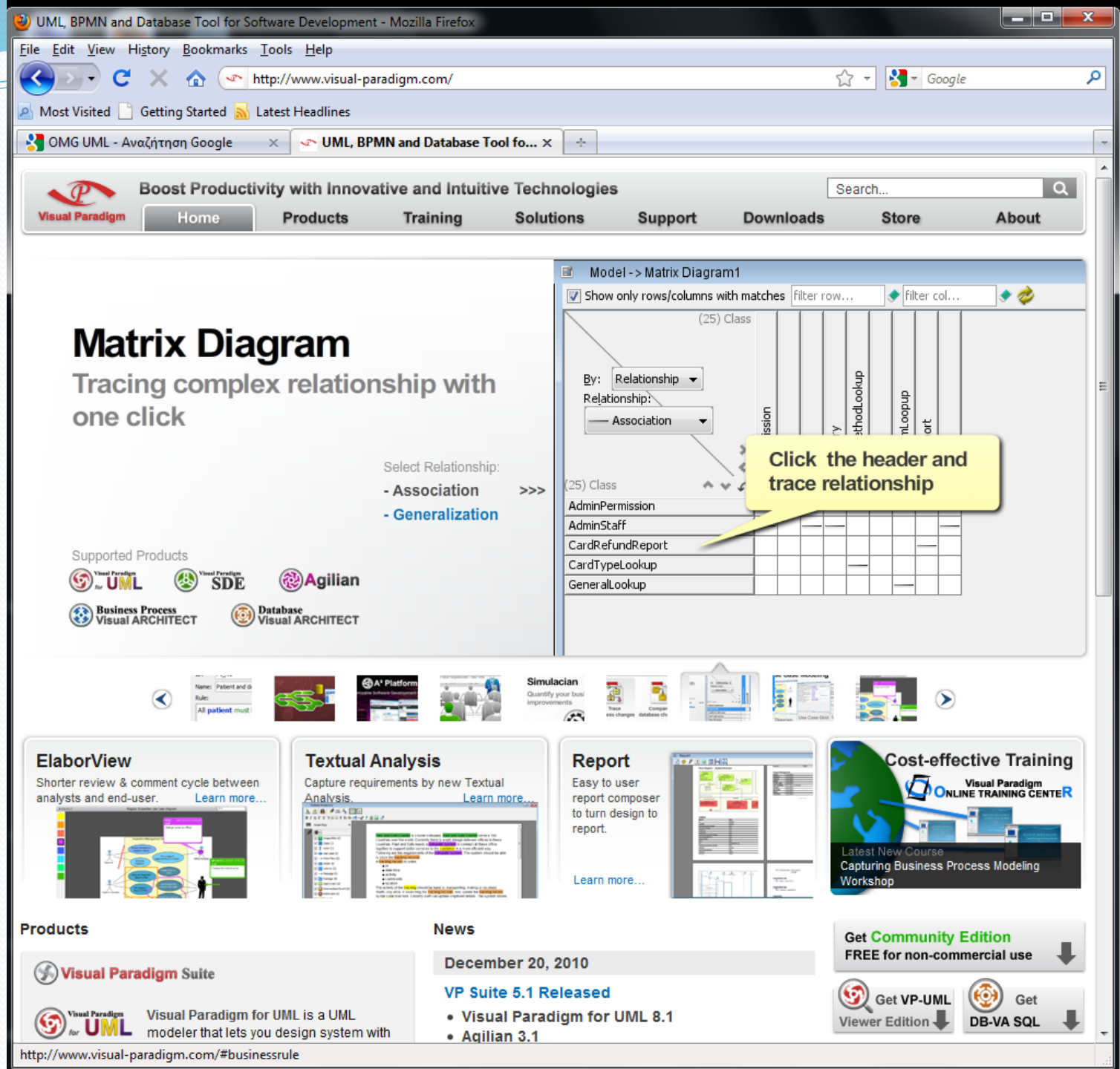
UML Modeling Tools

http://www.uml.org/



Free UML Modeling Tools

http://www.visual-paradigm.com/



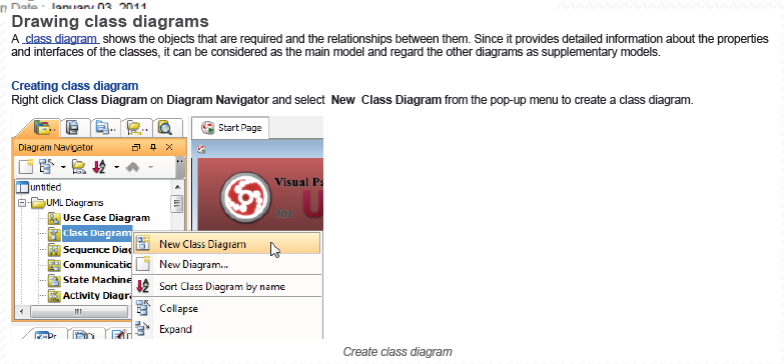
Free UML Modeling Tools:

Visual Paradigm

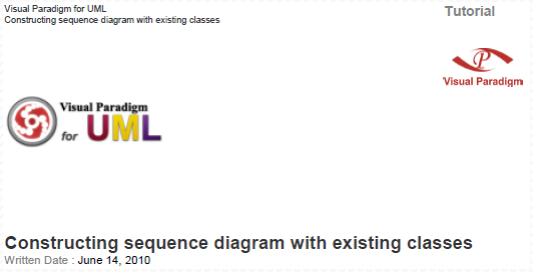
Ανοίξτε τα παρακάτω tutorials της Visual Paradigm

- Use Case tutorial

- Class tutorial

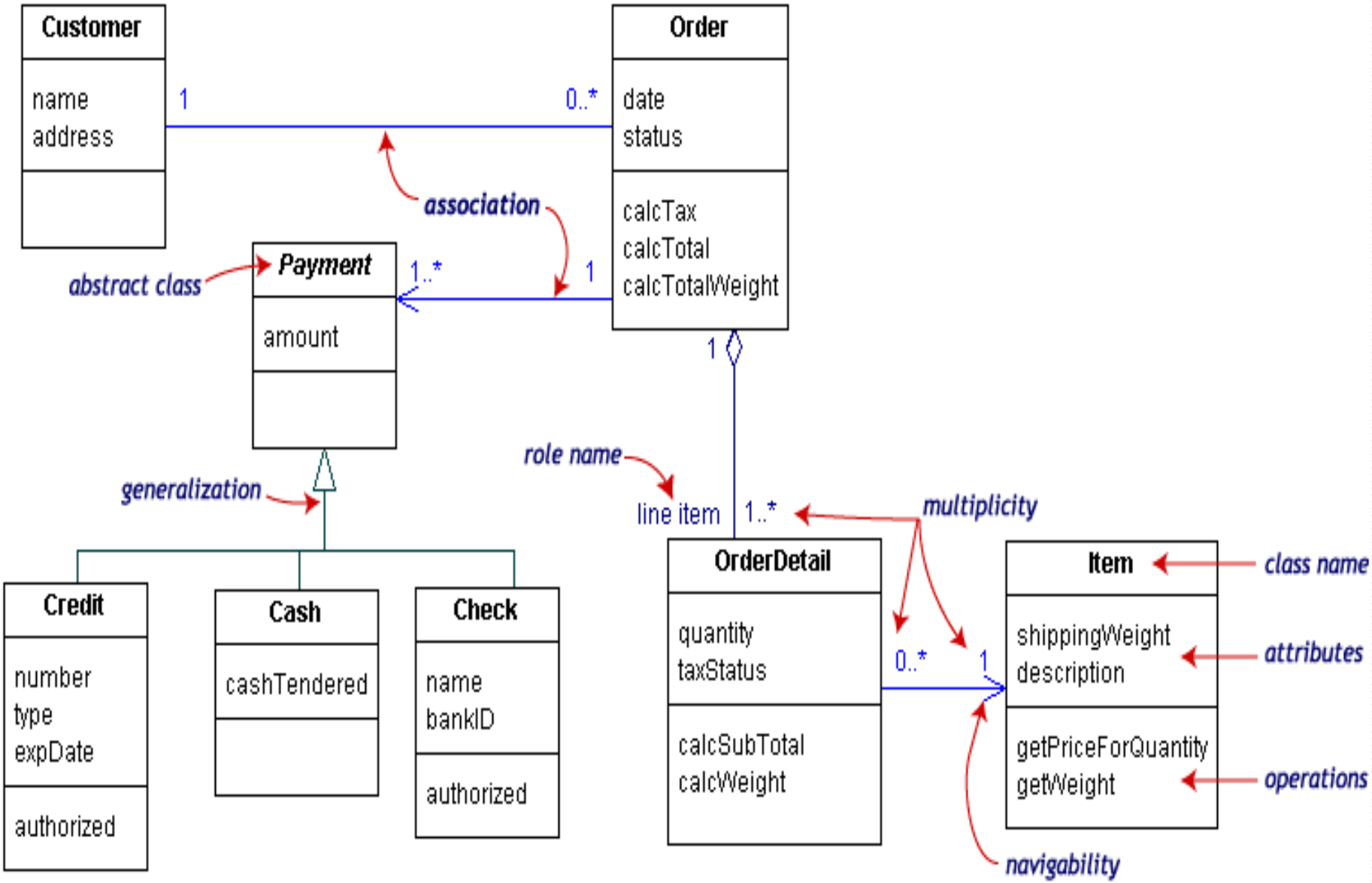


- Sequence tutorial



For a full list of tutorials visit:
http://www.visual-paradigm.com/product/vpuml/tutorials.jsp

Ασκήσεις


Customer
name
address
1
0..*
Order
date
status
association
calcTax
calcTotal
calcTotalWeight
abstract class
Payment
amount
1..*
1
1
role name
line item
1..*
multiplicity
generalization
Credit
number
type
expDate
authorized
Cash
cashTendered
Check
name
bankID
authorized
OrderDetail
quantity
taxStatus
calcSubTotal
calcWeight
0..*
1
Item
shippingWeight
description
getPriceForQuantity
getWeight
class name
attributes
operations
navigability

Ασκήσεις(προαιρετικές)

Σύνοψη Μαθήματος

Βιβλιογραφία

- JAVA Tutorials

  (http://java.sun.com/docs/books/tutorial/)

Επόμενο μάθημα