

# UML: Διαγράμματα

# UML: Unified modelling language

- \* Γλώσσα μοντελοποίησης για ανάλυση και σχεδιασμό
- \* Παρέχει το συμβολισμό για ανάλυση και σχεδιασμό.
- \* Είναι γλώσσα συμβολισμού.
- \* Δεν είναι ολόκληρη μεθοδολογία.
- \* Αναπτύχθηκε από τους G. Booch, τον J. Rumbaugh και τον I. Jacobson.
- \* Όμως η γλώσσα UML περιέχει στοιχεία και από άλλες μεθοδολογίες

# Ιστορία

- \* **Πρώτη έκδοση:** Rational Software Corporation (σύμπραξη εταιρειών). Οκτώβρης 1995.
- \* **Δεύτερη και Τρίτη έκδοση:** Ιούλιος και Οκτώβριος 1996
- \* **Έκδοση 1.1:** Σεπτέμβρης 1997

# UML

## Γλώσσα μοντελοποίησης και όχι μεθοδολογία.

- \* Η μεθοδολογία αποτελείται από μια γλώσσα μοντελοποίησης και μια διαδικασία.
- \* Η UML ορίζει μόνο τη γλώσσα μοντελοποίησης και όχι τη διαδικασία.
- \* Η γλώσσα μοντελοποίησης βοηθάει στην περιγραφή του σχεδιασμού.
- \* Η διαδικασία ορίζει τον τρόπο που θα γίνει ο σχεδιασμός

# UML: εργαλεία

- \* Η UML ορίζει δύο εργαλεία:
  - \* Ένα συμβολισμό: Διαγράμματα της UML
  - \* Ένα μεταμοντέλο: ορισμός του συμβολισμού

# Διαδικασία

## Rational Unified Process

- \* Διαδικασία για την ανάπτυξη λογισμικού
- \* Προτείνεται από Booch, Rumbaugh και Jacobson (δημιουργούς UML).
- \* Ο κύκλος ζωής λογισμικού προτείνεται να είναι επαναληπτικός.
  - \* Η ανάπτυξη δηλαδή να προχωρεί σε μια σειρά επαναλήψεων μέχρι να εξελιχθεί το τελικό προϊόν.
- \* Η διαδικασία Rational Unified Process αποτελείται από ένα σύνολο οδηγιών σχετικά με τις τεχνικές και οργανωτικές απόψεις της ανάπτυξης λογισμικού.
- \* Θα ασχοληθούμε κυρίως με την Ανάλυση Απαιτήσεων και το Σχεδιασμό.

# Διαδικασία

## Rational Unified Process

- \* Η διαδικασία Rational Unified Process είναι δομημένη βάσει δύο διαστάσεων:
  - \* Χρόνος: Χωρισμός του κύκλου ζωής σε φάσεις και επαναλήψεις.
  - \* Τμήματα διαδικασίας: Καλά ορισμένες εργασίες.

# Rational Unified Process: ΧΡΟΝΟΣ

\* Η δόμηση ενός έργου σε σχέση με το χρόνο ακολουθεί τις εξής φάσεις που έχουν σχέση με το χρόνο:

1) **Έναρξη (Inception):**

Καθορίζει την προοπτική του έργου.

2) **Εκπόνηση μελέτης (Elaboration):**

των  
Σχεδιασμός των απαιτούμενων δραστηριοτήτων και βασικών χαρακτηριστικών.

3) **Κατασκευή (Construction):**

Ανάπτυξη του προϊόντος σε μια σειρά επαναλήψεων.

4) **Μετάβαση (Transition):**

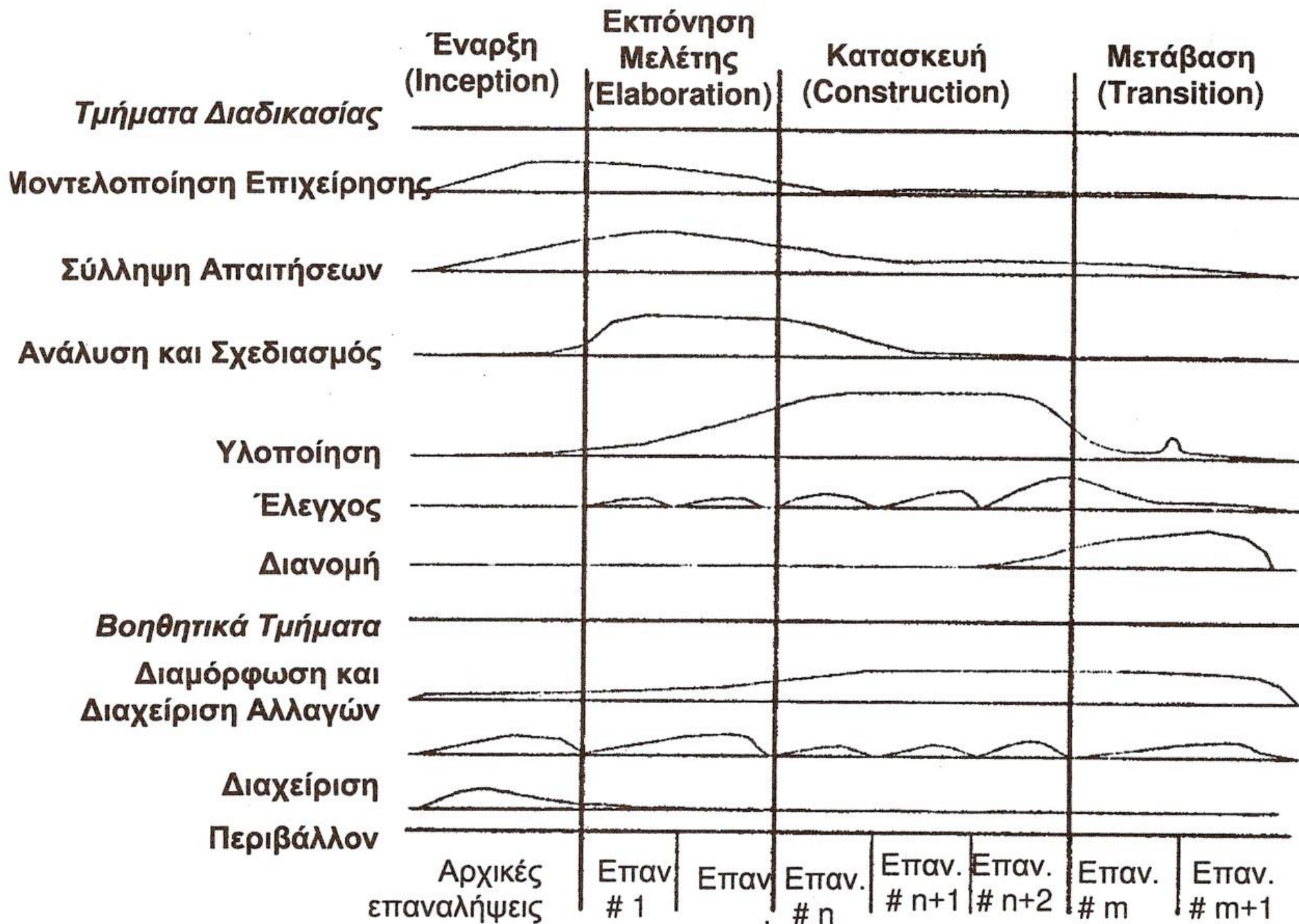
Προμήθεια του προϊόντος στους χρήστες (αξιολόγηση, διανομή, εκπαίδευση).



# Rational Unified Process: ΤΜΗΜΑΤΑ ΔΙΑΔΙΚΑΣΙΑΣ

- \* Η δόμηση έργου σύμφωνα με τη **διάσταση** των τμημάτων διαδικασίας περιλαμβάνει τις ακόλουθες δραστηριότητες:
  - 1) **Σύλληψη απαιτήσεων (Requirements capture):**  
Μια αφήγηση του τι πρέπει να κάνει το σύστημα.
  - 2) **Ανάλυση και σχεδιασμός (Analysis and design):**  
Μια περιγραφή του πως θα υλοποιηθεί το σύστημα.
  - 3) **Υλοποίηση (Implementation):**  
Η παραγωγή του κώδικα.
  - 4) **Έλεγχος (Test):**  
Η επαλήθευση του συστήματος.

# Φάσεις



# Διαγράμματα της UML

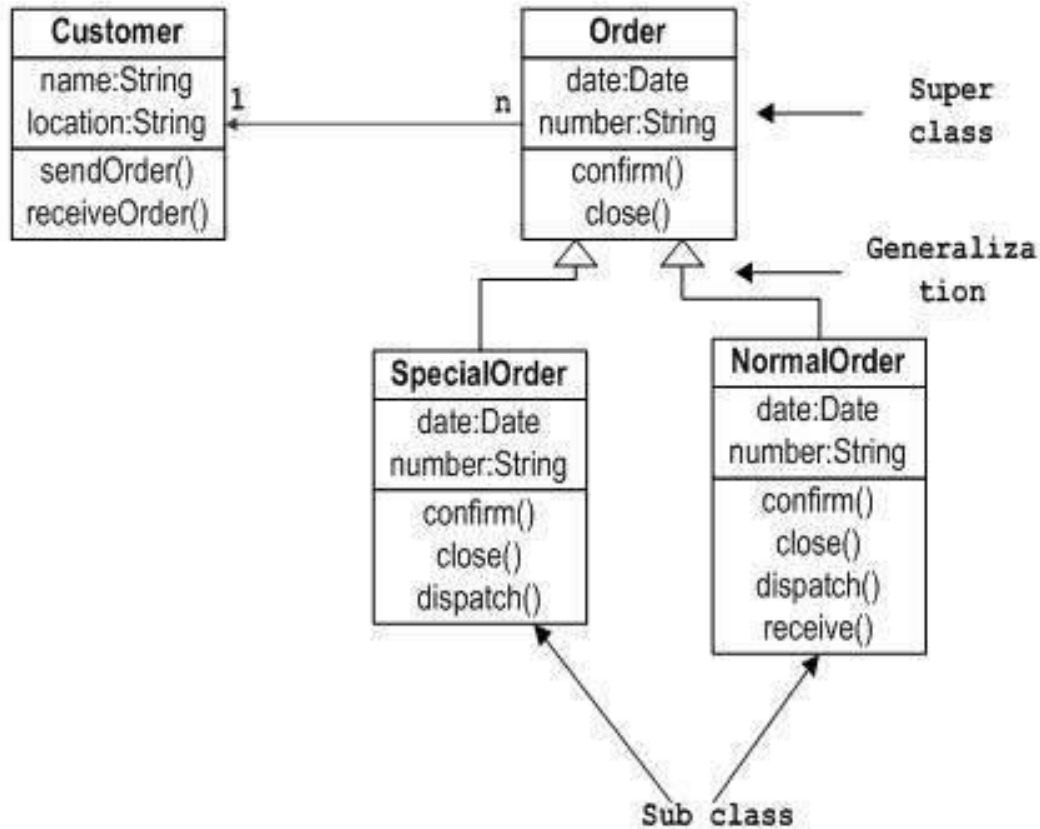
- \* Αναπαράσταση διαφορετικών τρόπων μοντελοποίησης:
  1. **Διαγράμματα κλάσεων (Class Diagrams):** Αναπαριστούν τη στατική δομή όσον αφορά στις κλάσεις και τις σχέσεις τους (αντικειμενοστρεφής βάση δεδομένων)
  2. **Διαγράμματα αντικειμένων (Object Diagrams):** Αναπαριστούν αντικείμενα και τις σχέσεις τους και αντιστοιχούν σε απλοποιημένα διαγράμματα συνεργασίας που δεν αναπαριστούν μετάδοση μηνυμάτων.
  3. **Διαγράμματα συνεργασίας (Collaboration Diagrams):** Η αναπαράσταση των αντικειμένων, συνδέσεων και αλληλεπιδράσεων.
  4. **Διαγράμματα σειράς (Sequence diagrams):** Χρονική αναπαράσταση των αντικειμένων και των αλληλεπιδράσεών τους.
  5. **Διαγράμματα δραστηριοτήτων (Activity diagrams):** Αναπαριστούν τη συμπεριφορά μιας λειτουργίας ως σύνολο ενεργειών.

# Διαγράμματα της UML

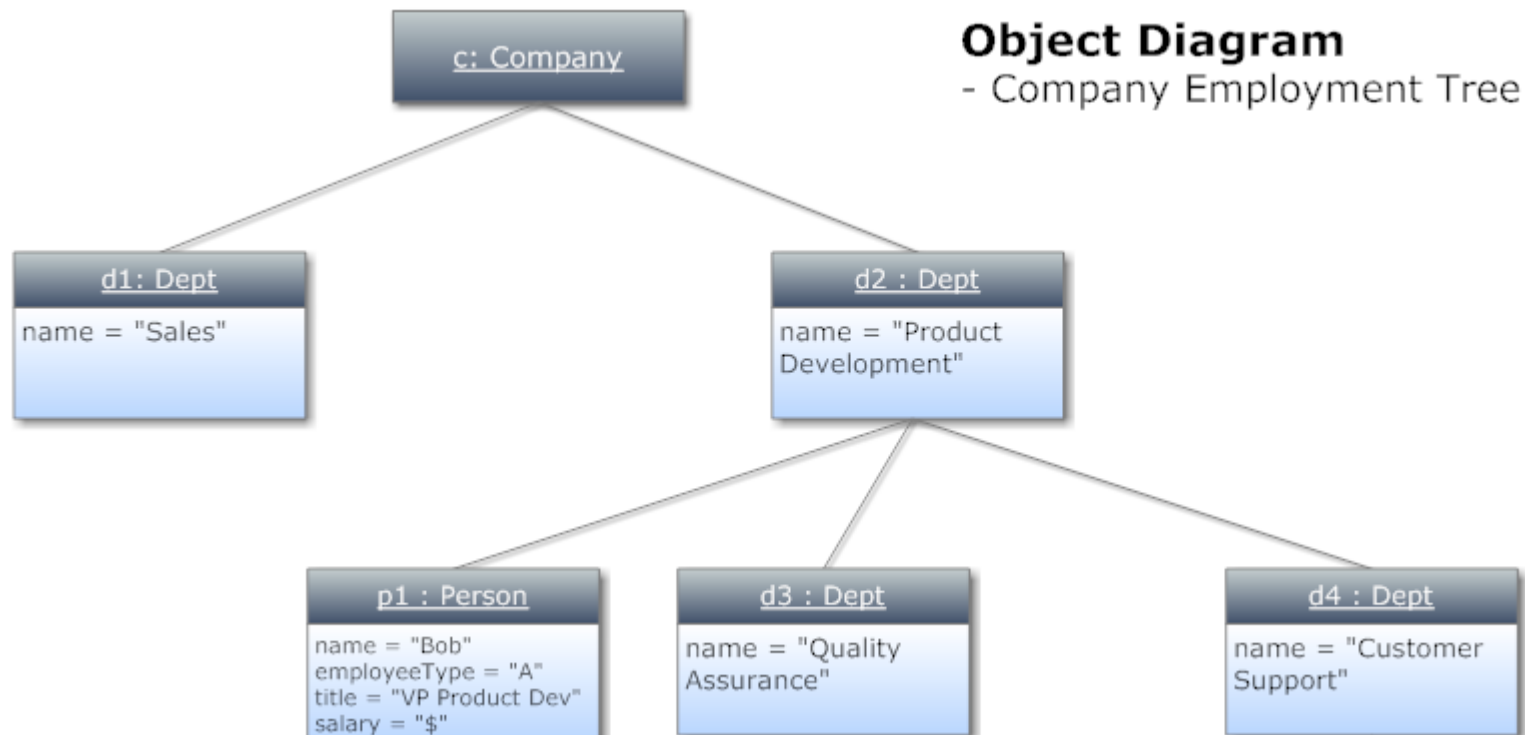
- 6. Διαγράμματα καταστάσεων (Statechart diagrams):**  
Αναπαριστούν τη συμπεριφορά της τάξης όσον αφορά στις καταστάσεις της.
- 7. Διαγράμματα εξαρτημάτων (Component diagrams):**  
Αναπαριστούν τα φυσικά εξαρτήματα μιας εφαρμογής.
- 8. Διαγράμματα διανομής (Deployment diagrams):**  
Αναπαριστούν τη διανομή των εξαρτημάτων σε συγκεκριμένα τεμάχια του hardware (υλικού).
- 9. Διαγράμματα περιπτώσεων χρήσης (Use case diagrams):**  
Αναπαριστούν τις λειτουργίες ενός συστήματος από την οπτική γωνία του χρήστη.

# Διαγράμματα κλάσεων

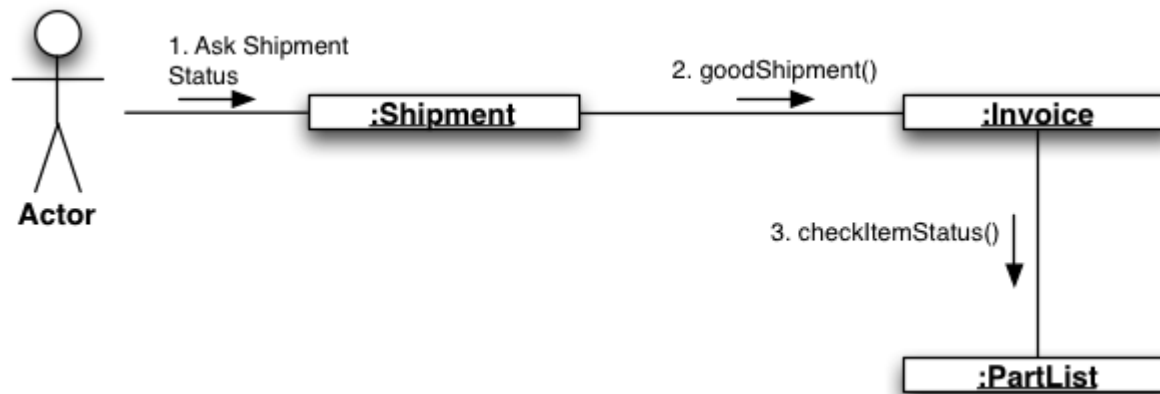
Sample Class Diagram



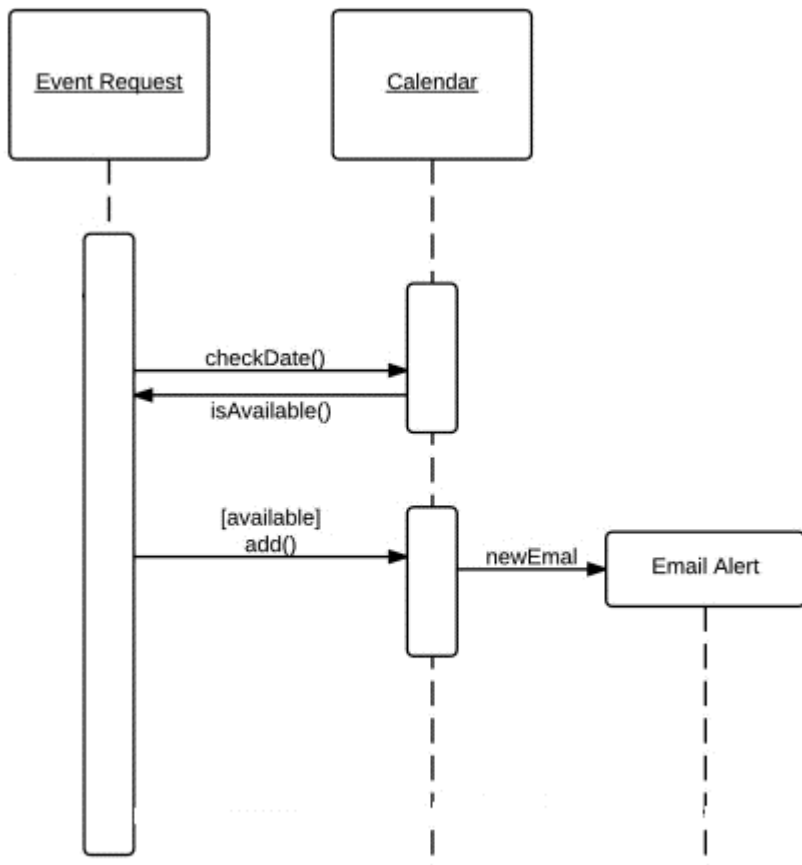
# Διαγράμματα αντικειμένων



# Διαγράμματα συνεργασίας

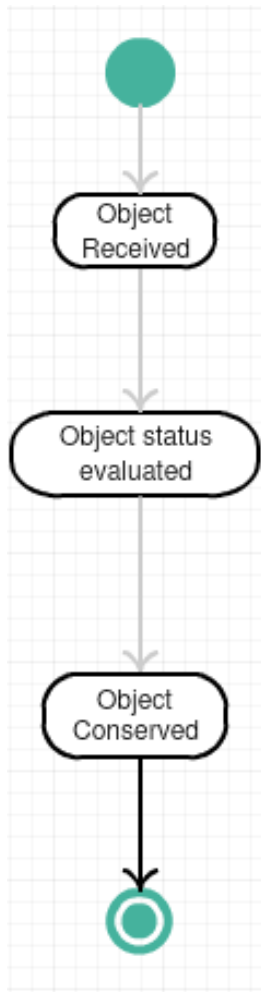


# Διαγράμματα σειράς

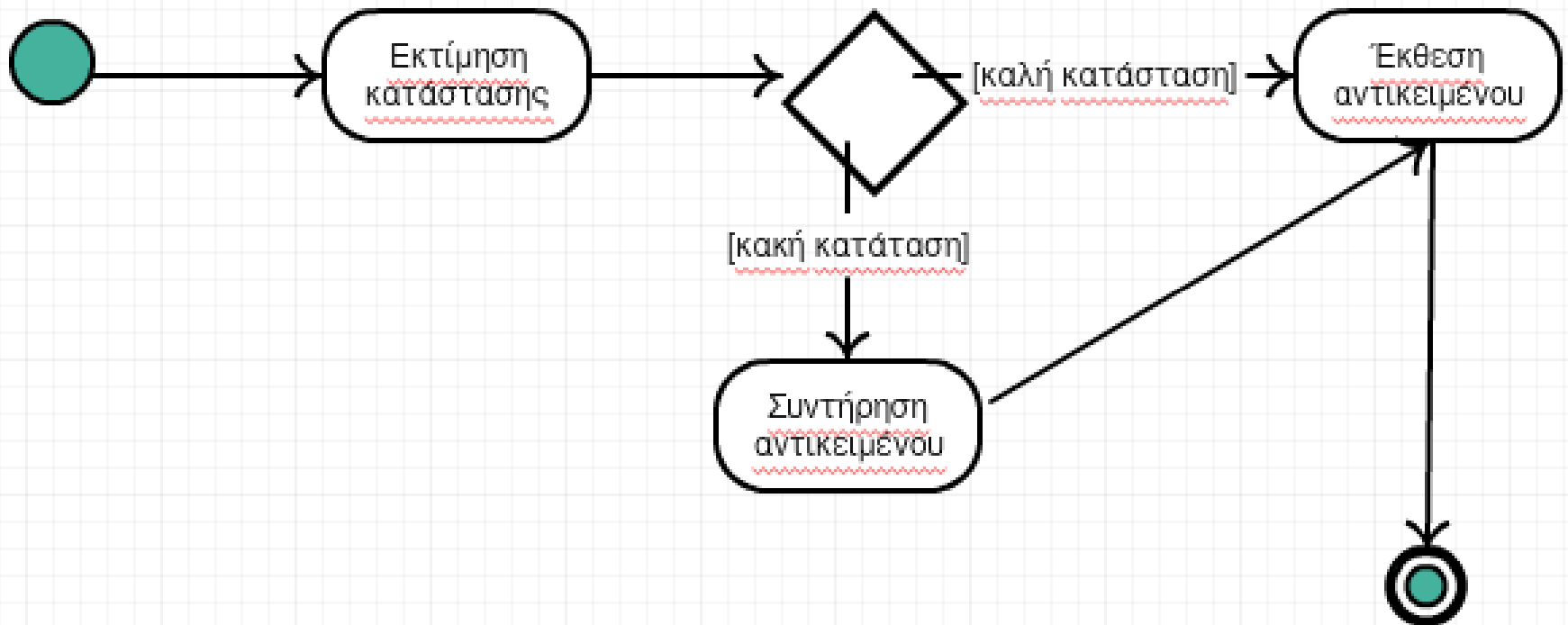




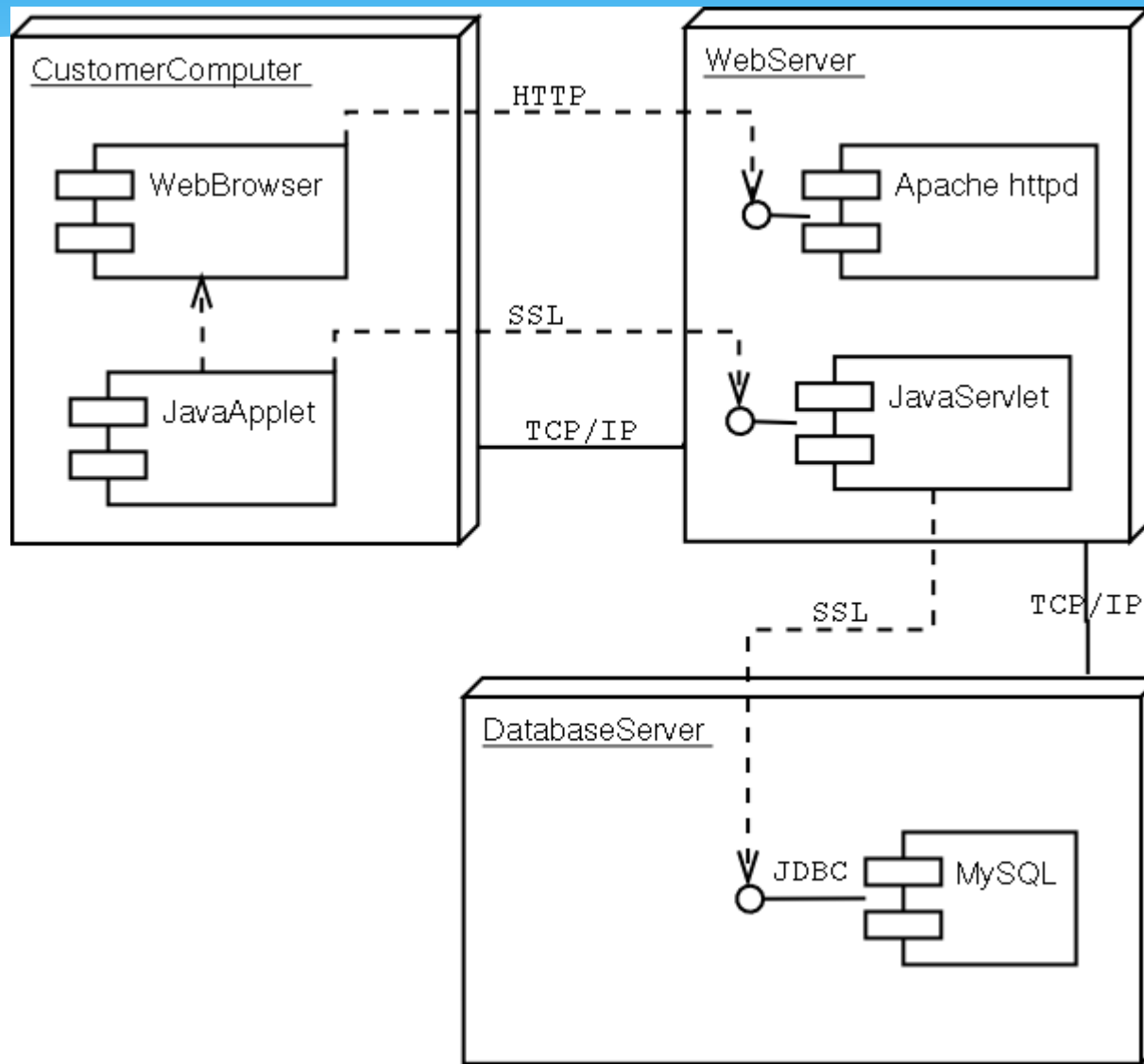
# Διαγράμματα καταστάσεων



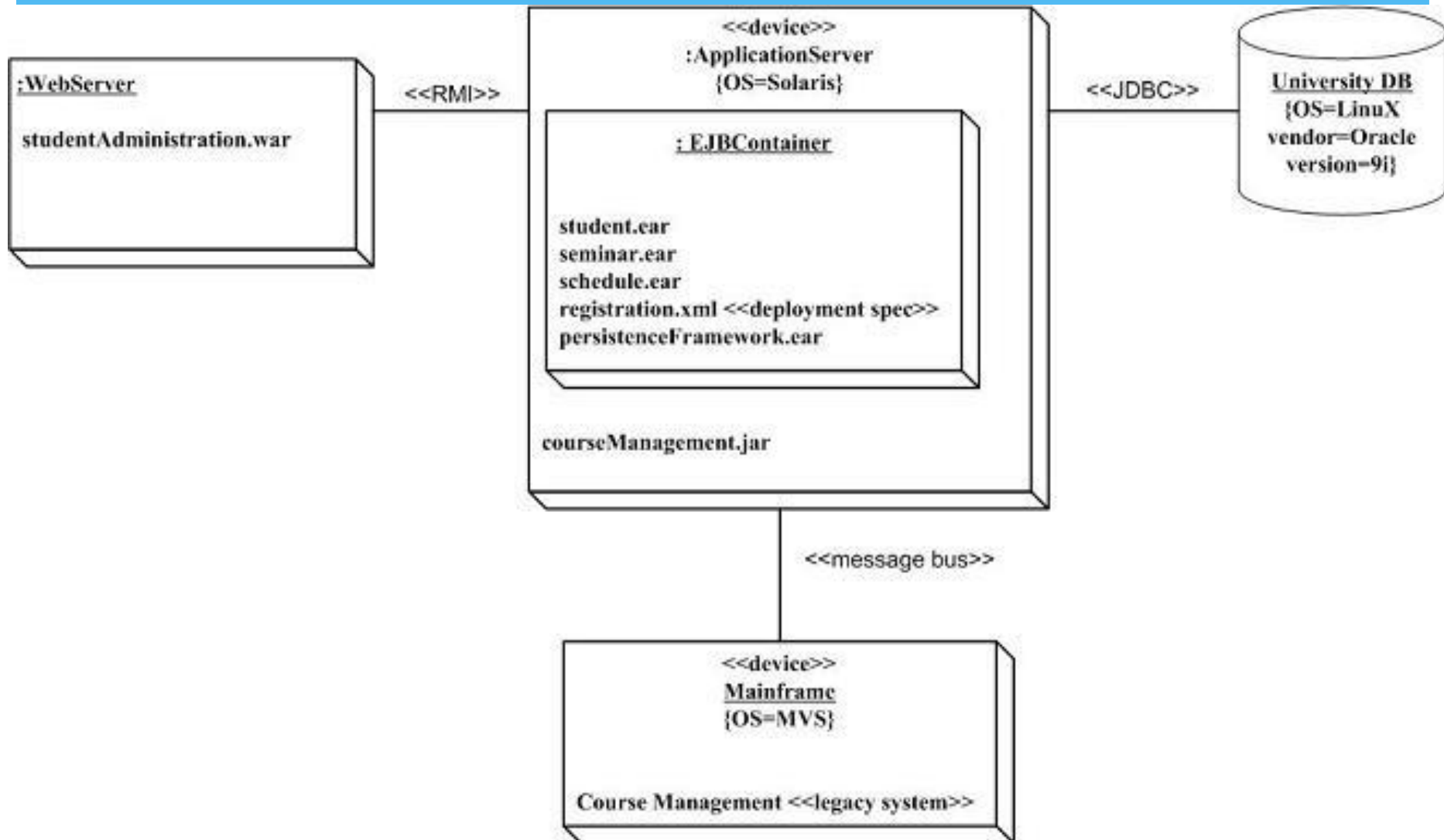
# Διαγράμματα δραστηριοτήτων



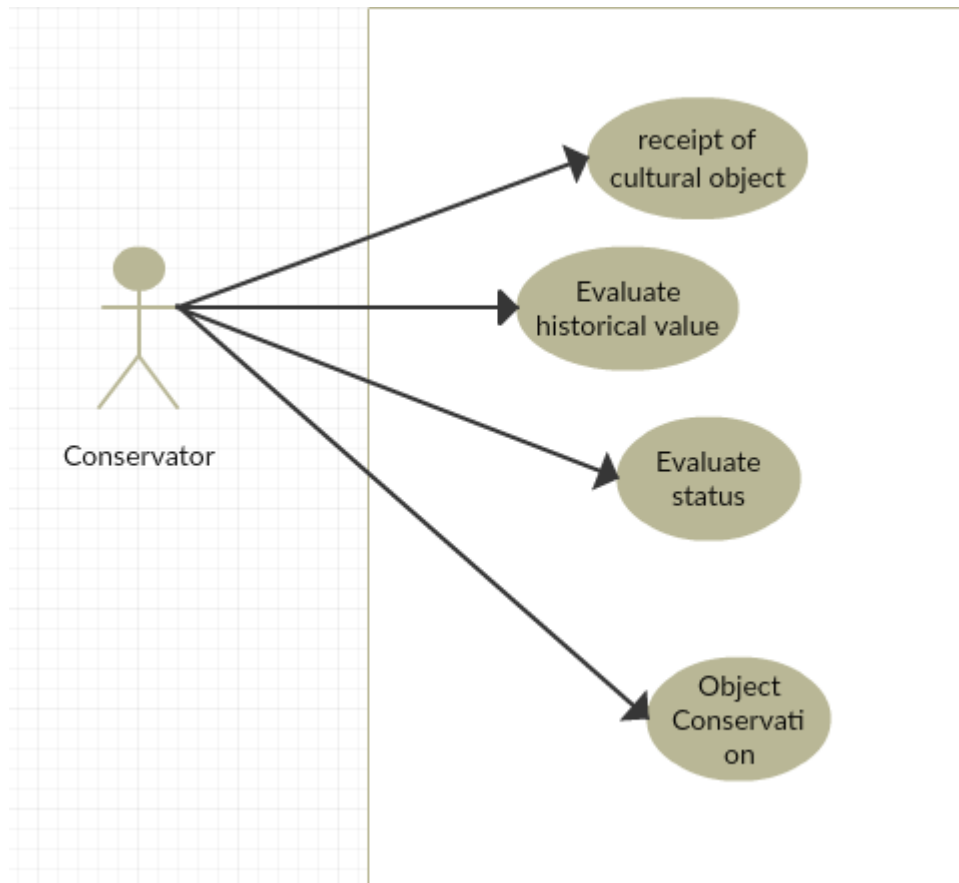
# Διαγράμματα εξαρτημάτων



# Διαγράμματα διανομής

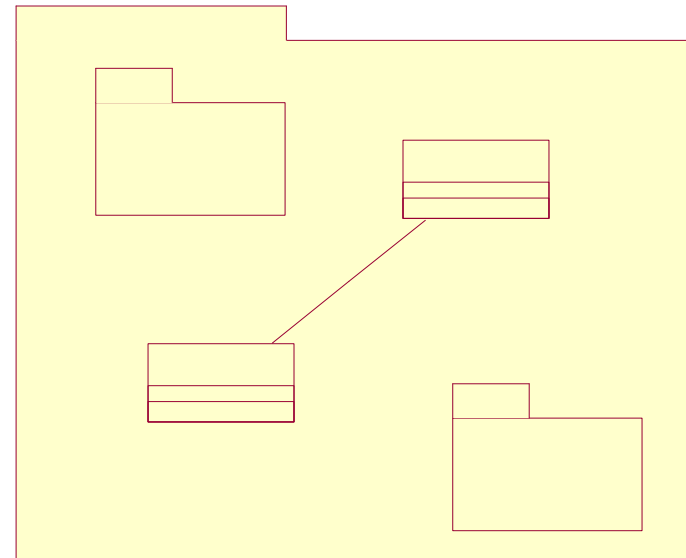
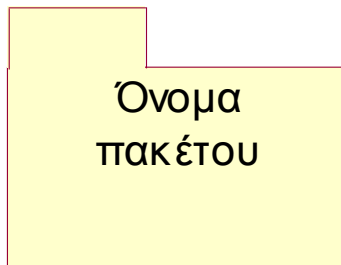


# Διαγράμματα περιπτώσεων χρήσης



# πακέτα

- \* Ομαδοποίηση γίνεται μέσω των πακέτων
- \* Λογική ομαδοποίηση για οικονομία χώρου και καλύτερη κατανόηση
- \* Ένα πακέτο μπορεί να περιέχει άλλο πακέτο κ.ο.κ
- \* Κάθε στοιχείο ανήκει σε ένα πακέτο
- \* Πακέτο Ρίζα...



# UML: Μηχανισμοί

- \* Στερεότυπα (Stereotypes): εξειδικεύουν τις κλάσεις
- \* Συσχετισμένες τιμές (Tagged values): επεκτείνουν τα χαρακτηριστικά των κλάσεων
- \* Σημειώσεις (Notes)
- \* Περιορισμούς (Constraints): επεκτείνουν τη σημασιολογία
- \* Εξαρτήσεις (Dependencies)