

# 4η Διάλεξη

*UML – Διαγράμματα  
αλληλεπίδρασης*

# Διαγράμματα αλληλεπίδρασης

Τα διαγράμματα αλληλεπίδρασης είναι μοντέλα που περιγράφουν κάποιες ομάδες αντικειμένων

Μπορούν να χρησιμοποιηθούν για να δείξουν ένα σχέδιο για μια συγκεκριμένη λειτουργία ενός προγράμματος

Είναι δυναμικά διαγράμματα γιατί δείχνουν πως μεταλλάσσεται το σύστημά

# Διαγράμματα αλληλεπίδρασης

Αναπαριστούν συνήθως πώς διάφορα αντικείμενα ενός συστήματος αλληλεπιδρούν προκειμένου να ολοκληρώσουν μια λειτουργία ενός συστήματος

# Διαγράμματα αλληλεπίδρασης

Τα διαγράμματα αλληλεπίδρασης είναι δύο ειδών

- Διαγράμματα ακολουθίας (sequence diagrams)
- Διαγράμματα συνεργασίας (collaboration diagrams)



# ΤΕΧΝΟΛΟΓΙΑ ΛΟΓΙΣΜΙΚΟΥ Ι

κ. ΠΕΤΑΛΙΔΗΣ

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΤΕ



# Άδειες Χρήσης

- Το παρόν εκπαιδευτικό υλικό υπόκειται σε άδειες χρήσης Creative Commons.
- Για εκπαιδευτικό υλικό, όπως εικόνες, που υπόκειται σε άλλου τύπου άδειας χρήσης, η άδεια χρήσης αναφέρεται ρητώς.



# Χρηματοδότηση

- Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στα πλαίσια του εκπαιδευτικού έργου του διδάσκοντα.
- Το έργο «Ανοικτά Ακαδημαϊκά Μαθήματα στο ΤΕΙ Κεντρικής Μακεδονίας» έχει χρηματοδοτήσει μόνο τη αναδιαμόρφωση του εκπαιδευτικού υλικού.
- Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «Εκπαίδευση και Δια Βίου Μάθηση» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους.



# Παράδειγμα

Έστω ότι θέλετε να αναπαραστήσετε την ακόλουθη ροή των γεγονότων.

- Ένας πελάτης (ο Jim) δίνει την παραγγελία του μέσα από μια φόρμα
- Η παραγγελία μπορεί να έχει παραπάνω από ένα στοιχεία (π.χ. Έναν υπολογιστή και έναν εκτυπωτή)
- Αφού δημιουργηθεί η παραγγελία πρέπει να ετοιμαστεί κάθε στοιχείο της παραγγελίας για παράδοση.
- Για κάθε στοιχείο ελέγχουμε αν υπάρχει στην αποθήκη.
- Αν υπάρχει στην αποθήκη το παραδίδουμε και μειώνουμε αντίστοιχα το απόθεμα της αποθήκης.
- Αν δεν υπάρχει και χρειάζεται να το παραγγείλουμε



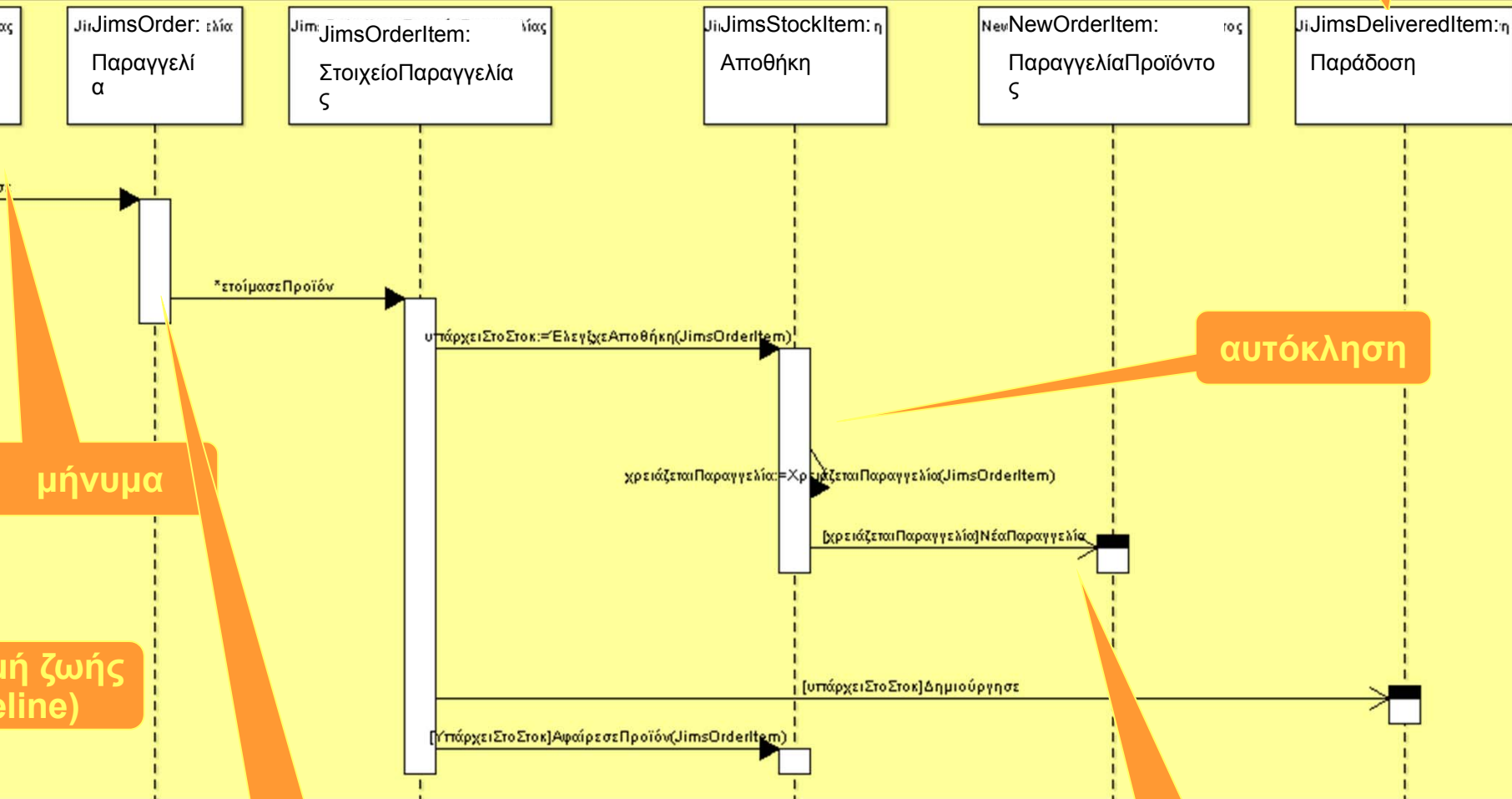
# Παράδειγμα

Μπορείτε να χρησιμοποιήσετε για την αναπαράσταση της ροής αυτών των γεγονότων το ακόλουθο διάγραμμα ακολουθίας

Σε αυτό το διάγραμμα κάθε παραλληλόγραμμο αφορά και ένα αντικείμενο

# Διαγράμματα ακολουθίας

Αντικείμενο



μήνυμα

αυτόκληση

μή ζωής  
(lifeline)

# Επεξήγηση του διαγράμματος

Το αντικείμενο `JimsOrderForm` στέλνει ένα μήνυμα `ετοίμασε()` στο αντικείμενο `JimsOrder`

Για κάθε στοιχείο παραγγελίας, το αντικείμενο `JimsOrder` στέλνει ένα μήνυμα `ετοίμασεΠροϊον()` στο αντικείμενο `JimsOrderItem`

Το αντικείμενο `JimsOrderItem` στέλνει ένα μήνυμα `έλεγεξεΑποθήκη()` στο αντικείμενο `JimsStockItem` και αποθηκεύει το αποτέλεσμα στη μεταβλητή `ΥπάρχειiΣτοΣτοκ`

Το αντικείμενο `JimsStockItem` στέλνει ένα μήνυμα στον εαυτό του `χρειάζεταιiΠαραγγελία()` και αποθηκεύει το αποτέλεσμα στη μεταβλητή `χρειάζεταιiΠαραγγελία`

Αν η μεταβλητή `χρειάζεταιiΠαραγγελία` είναι

# Διαγράμματα ακολουθίας

---

Στα διαγράμματα ακολουθίας μπορείτε να προσθέσετε και επεξηγηματικά σχόλια

Σχόλια

# Παράδειγμα

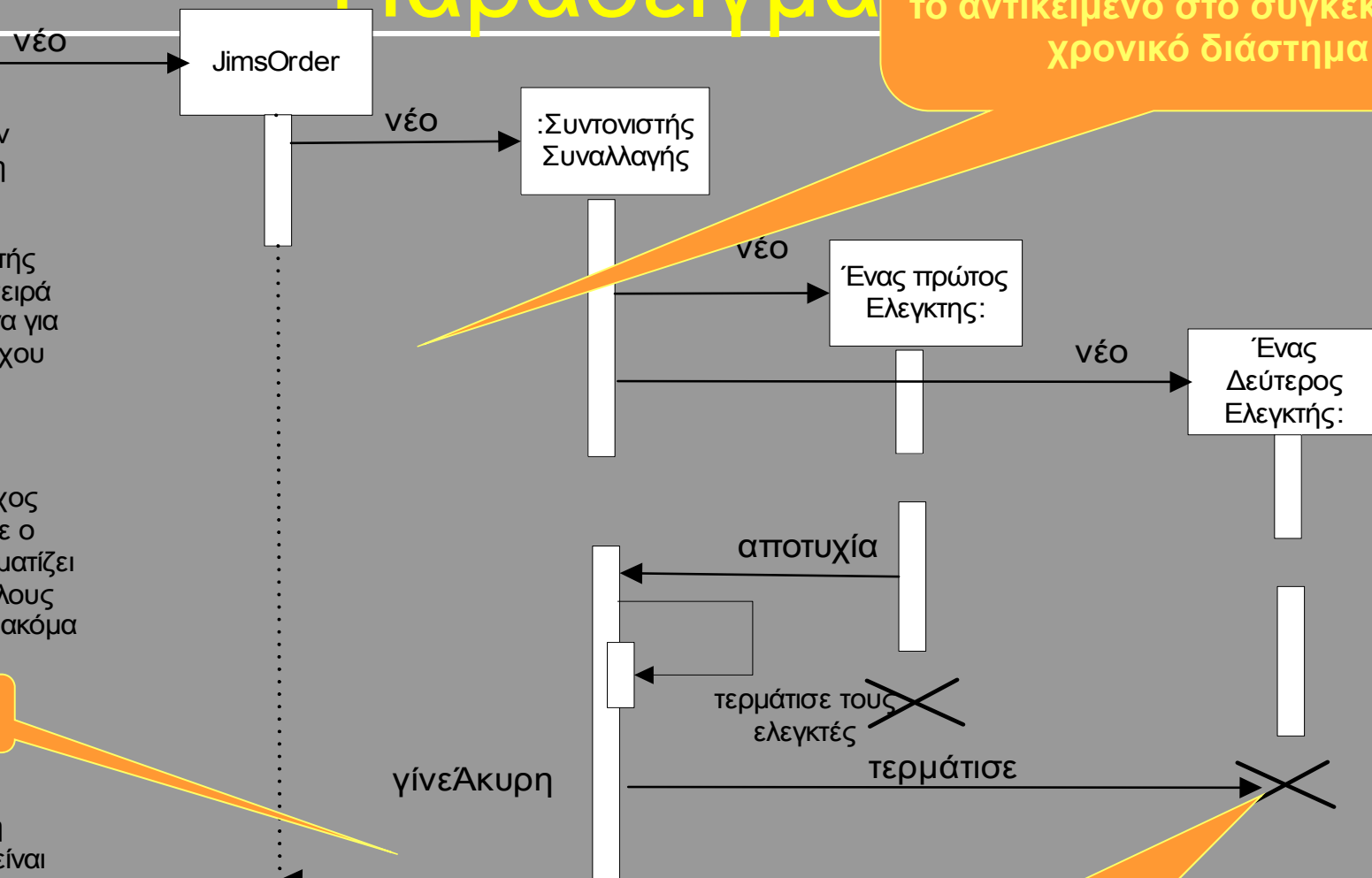
Χρησιμοποιήστε τη διακεκομμένη γραμμή όταν δεν έχετε να τονίσετε κάτι σημαντικό για το αντικείμενο στο συγκεκριμένο χρονικό διάστημα

ται  
ιγή  
εί έναν  
για τη  
των  
ον  
τονιστής  
για σειρά  
ές, ένα για  
ς ελέγχου

έλεγχος  
ει τότε ο  
ς τερματίζει  
ς άλλους  
ύνται ακόμα

οφή

ει στη  
πως είναι



γίνε Άκυρη

τερμάτισε

# Διαγράμματα ακολουθίας

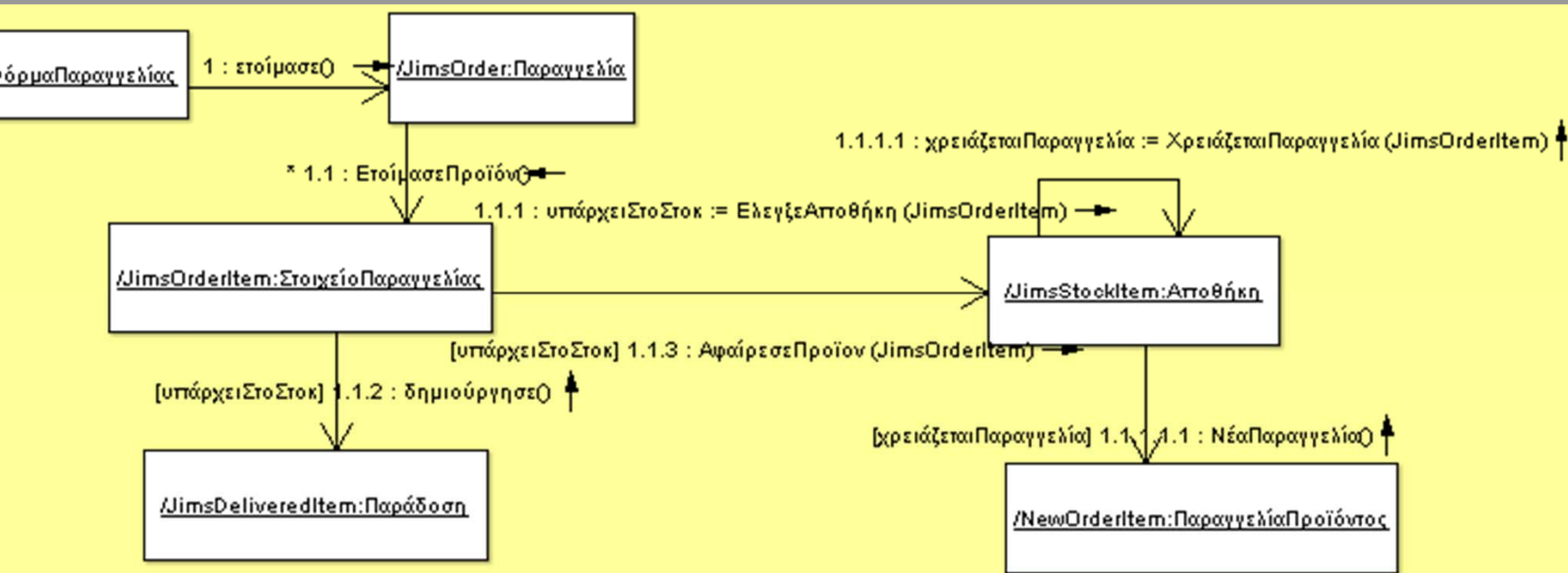
Στα διαγράμματα ακολουθίας  
προσπαθήστε να δείξετε μία  
συγκεκριμένη περίπτωση φορά

Αν υπάρχουν πολλές εναλλακτικές ροές  
κάντε και πολλά διαγράμματα

# Διαγράμματα συνεργασίας

Σε ένα διάγραμμα συνεργασίας τα αντικείμενα απεικονίζονται ως εικονίδια Όπως και σε ένα διάγραμμα ακολουθίας τα βέλη απεικονίζουν αποστολές μηνυμάτων στο πλαίσιο της αναπαράστασης μιας συγκεκριμένης λειτουργίας

# Διάγραμμα συνεργασίας





# Επεξήγηση διαγράμματος

Το αντικείμενο `JimOrderForm` στέλνει ένα μήνυμα `ετοίμασε ()` στο αντικείμενο `JimOrder`

Για κάθε στοιχείο παραγγελίας, το αντικείμενο `JimOrder` στέλνει ένα μήνυμα `ετοίμασεΠροϊον ()` στο αντικείμενο `JimOrderItem`

Το αντικείμενο `JimOrderItem` στέλνει ένα μήνυμα `έλεγξεΑποθήκη ()` στο αντικείμενο `JimStockItem` και αποθηκεύει το αποτέλεσμα στη μεταβλητή `ΥπάρχειΣτοΣτοκ`

Το αντικείμενο `JimStockItem` στέλνει ένα μήνυμα στον εαυτό του `χρειάζεταιΠαραγγελία ()` και αποθηκεύει το αποτέλεσμα στη μεταβλητή `χρειάζεταιΠαραγγελία`

Αν η μεταβλητή `χρειάζεταιΠαραγγελία` είναι

# Διαγράμματα συνεργασίας

Στα διαγράμματα συνεργασίας η αρίθμηση των μηνυμάτων δείχνει την ακολουθία τους

Αυτό μπορεί να κάνει πιο δύσκολη την παρακολούθηση της ακολουθίας σε σχέση με την παράθεση των μηνυμάτων από πάνω προς τα κάτω

Όμως μπορείτε να δείξετε πιο εύκολα πως συνδέονται τα αντικείμενα μεταξύ τους

# Διαγράμματα αλληλεπίδρασης

Η μέθοδος ονομασίας των αντικειμένων που εμφανίζεται στα διαγράμματα αλληλεπίδρασης ακολουθεί τη μορφή:

- Όνομα αντικειμένου: όνομα κλάσης

Πολλές φορές μπορεί να αναγράψουμε μόνο το ένα από τα δύο, δηλ.

- Όνομα αντικειμένου:

- :Όνομα κλάσης ή Όνομα κλάσης

# Διαγράμματα ακολουθίας - συνεργασίας

Τα δύο είδη διαγραμμάτων είναι  
ισοδύναμα

Απλά κάποιες φορές είναι πιο εύκολο  
(οπτικά) να αναπαραστήσουμε την  
πληροφορία είτε με τον ένα είτε με τον  
άλλο τρόπο

# Τι είδους μηνύματα υπάρχουν;

Τα μηνύματα από ένα αντικείμενο σε ένα άλλο  
τα διαγράμματα αλληλεπίδρασης μπορεί να  
γίνουν:

## – Απλά

- Ο έλεγχος περνάει από το ένα αντικείμενο στο άλλο, χωρίς να περιγράφονται άλλες λεπτομέρειες

## – Σύγχρονα

- Υλοποιούνται σαν μια κλήση συνάρτησης.

## – Ασύγχρονα

- Ο αποστολέας συνεχίζει να εκτελεί τις υπόλοιπες λειτουργίες του αφού στείλει το μήνυμα, χωρίς να

# Αναπαράσταση μηνυμάτων



Σύγχρονα



Ασύγχρονα



Απλά



Σύγχρονα με άμεση  
επιστροφή

# Πότε χρησιμοποιούνται τα διαγράμματα αλληλεπίδρασης

Όταν θέλετε να μελετήσετε τον τρόπο με τον οποίο συμπεριφέρεται ένας αριθμός αντικειμένων στο πλαίσιο της μελέτης μιας συγκεκριμένης λειτουργίας ενός συστήματος

Αν θέλετε να μελετήσετε τη συμπεριφορά ενός μόνο αντικειμένου χρησιμοποιήστε τους χάρτες κατάστασης που θα δούμε

παρακάτω