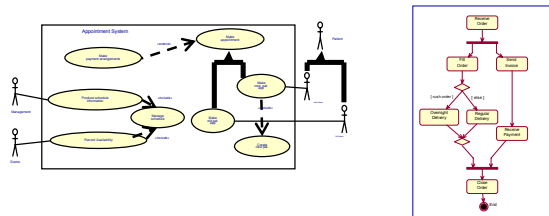


HY351:
Ανάλυση και Σχεδίαση Πληροφοριακών Συστημάτων
Information Systems Analysis and Design



Μοντελοποίηση Λειτουργιών (Functional Modeling)



Γιάννης Τζιτζίκας

Διάλεξη : 8
Ημερομηνία :
Θέμα :

Διάρθρωση

- Μοντελοποίηση Λειτουργιών (Functional Modeling)
- Μοντελοποίηση Λειτουργιών με τη UML
 - Περιπτώσεις Χρήσης (Use Cases)
 - Διαγράμματα Περιπτώσεων Χρήσης
 - Διαγράμματα Δραστηριοτήτων (Activity Diagrams)



Τι είναι η Μοντελοποίηση Λειτουργιών;

Ως κύριο στόχο έχει την περιγραφή:

- των επιχειρηματικών επεξεργασιών (business processes), και
- της αλληλεπίδρασης του Πληροφοριακού Συστήματος με το περιβάλλον του



Πως μοντελοποιούμε τις λειτουργίες στην Αντικειμενοστρεφή Ανάλυση και Σχεδίαση;

Συνήθως χρησιμοποιούμε δύο τύπους διαγραμμάτων της UML:

- **Διαγράμματα Δραστηριοτήτων (Activity Diagrams)**
 - Επιτρέπουν την μοντελοποίηση ροών εργασιών (του οργανισμού ή του συστήματος)
- **Περιπτώσεις Χρήσης και Διαγράμματα Περιπτώσεων Χρήσης**
 - Επιτρέπουν την μοντελοποίηση των βασικών αλληλεπιδράσεων μεταξύ του πληροφοριακού συστήματος και του περιβάλλοντος του.

Παρατηρήσεις:

- Μπορούμε να τα χρησιμοποιήσουμε για να περιγράψουμε το υπάρχον (χειρογραφικό ή μηχανογραφημένο) ή το προς κατασκευή πληροφοριακό σύστημα.

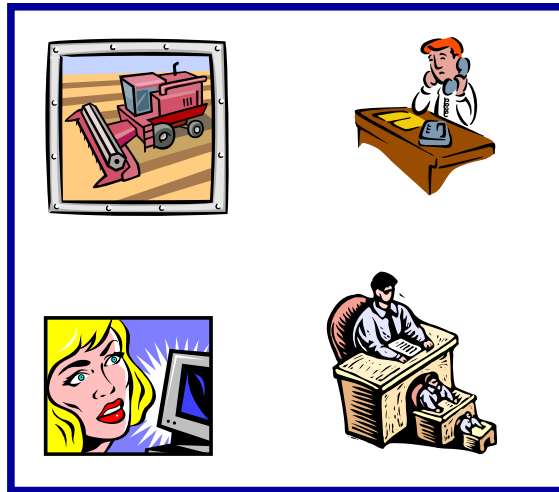


Τι διαφέρουν τα Διαγράμματα Δραστηριοτήτων από τα Διαγράμματα Περιπτώσεων Χρήσης;

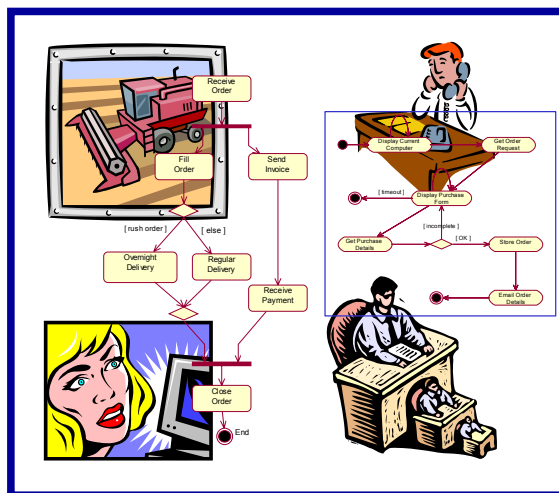
- **Activity Diagrams**
 - Τα χρησιμοποιούμε κυρίως για να μοντελοποιήσουμε τις «εσωτερικές» λειτουργίες του συστήματος
 - Π.χ. για να περιγράψουμε πως λειτουργεί ο οργανισμός, ή πως λειτουργεί εσωτερικά μια μέθοδος μιας κλάσης
- **Use Cases & Use Case Diagrams**
 - Τα χρησιμοποιούμε για να μοντελοποιήσουμε τις «εξωτερικές» λειτουργίες (τις άμεσα εμφανείς) του συστήματος
 - Π.χ. πως αλληλεπιδρά με τους χρήστες του συστήματος ή με άλλα πληροφοριακά συστήματα



Ένας οργανισμός



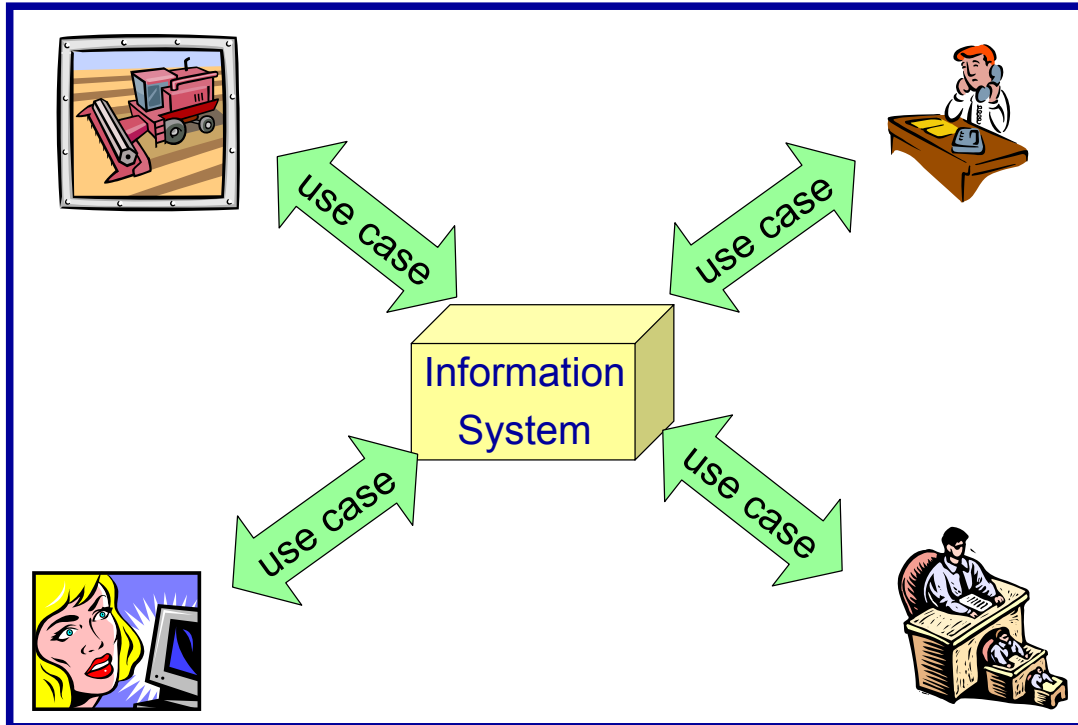
Τα Διαγράμματα Δραστηριοτήτων επιτρέπουν την μοντελοποίηση των επιχειρηματικών διαδικασιών του οργανισμού



Οργανισμός



Οι *Περιπτώσεις Χρήσης* μοντελοποιούν την αλληλεπίδραση μεταξύ το Πληροφοριακού Συστήματος και του Περιβάλλοντος του



Περιπτώσεις Χρήσης Use Cases



Τι μπορούμε να μοντελοποιήσουμε με Περιπτώσεις Χρήσης;

- Μοντελοποιούν την εξωτερική λειτουργική όψη του συστήματος
 - Πως αλληλεπιδρά με το περιβάλλον του (π.χ. με τους χρήστες
 - Περιγράφουν τι μπορούν να κάνουν οι χρήστες και πως ανταποκρίνεται το σύστημα
- Πλέον αποτελούν βασικό εργαλείο για την ανάπτυξη και προγραμματισμό πληροφοριακών συστημάτων.



Τι είναι μια Περίπτωση Χρήσης (Use Case);

Περίπτωση Χρήσης (Use Case) =

ένα σύνολο σεναρίων για την επίτευξη ενός **σκοπού** του χρήστη

- **Σενάριο** = μια ακολουθία βημάτων που περιγράφουν την αλληλεπίδραση μεταξύ χρήστη και συστήματος

Παράδειγμα Σεναρίου

Ο πελάτης πλοηγείται στον κατάλογο και προσθέτει τα επιθυμητά προϊόντα στο καλάθι αγοράς. Αν θελήσει να πληρώσει τότε περιγράφει τον επιθυμητό τρόπο αποστολής, δίνει τα στοιχεία της πιστωτικής του κάρτας και επιβεβαιώνει την αίτηση αγοράς. Το σύστημα ελέγχει τα στοιχεία της κάρτας και επιβεβαιώνει την αγορά (αμέσως και με την αποστολή η-μηνύματος επιβεβαίωσης).



Περιεχόμενο και μορφότυπος μια Περίπτωσης Χρήσης

- Μια Περ. Χ. συνήθως περιλαμβάνει:
 - Ένα συνηθισμένο σενάριο (όπου όλα πάνε καλά) (all-goes-well scenario)
 - Διάφορες εναλλακτικές ροές
- Ο μορφότυπος (format) μιας Περίπτωσης Χρήσης
 - Ακολουθία αριθμημένων βημάτων
 - Εναλλακτικές (alternatives) ως παραλλαγές αυτής της ακολουθίας
 - Προαιρετικό: προσθήκη μιας γραμμής με προσυνθήκες (preconditions), ήτοι συνθήκες που πρέπει να είναι αληθείς για να ξεκινήσει η Περίπτωση Χρήσης
- **Πόσο λεπτομερής πρέπει να είναι μια Περίπτωση Χρήσης;**
 - **Όσο μεγαλύτερο είναι το ρίσκο τόσο πιο αναλυτικοί πρέπει να είμαστε**



Παράδειγμα: Περίπτωση Χρήσης Αγορά Προϊόντος

Αγορά Προϊόντος

1. Ο Πελάτης πλοηγείται στον κατάλογο και επιλέγει τα προϊόντα που επιθυμεί να αγοράσει
2. Ο Πελάτης επιλέγει τη λειτουργία «Παραγγελία»
3. Ο Πελάτης συμπληρώνει τα στοιχεία αποστολής (δνση, παράδοση σε 24 ώρες, παράδοση σε 3 ημέρες)
4. Το Σύστημα του παρουσιάζει αναλυτικά την τιμή της παραγγελίας (περιλαμβανομένων των εξόδων αποστολής)
5. Ο Πελάτης συμπληρώνει τα στοιχεία της πιστωτικής του κάρτας
6. Το Σύστημα ελέγχει τα στοιχεία της κάρτας (εξουσιοδότηση, πιστωτικό όριο, ...)
7. Το Σύστημα επιβεβαιώνει την πώληση αμέσως
8. Το Σύστημα στέλνει ένα η-μήνυμα επιβεβαίωσης στον Πελάτη

Εναλλακτική: Authorization Failure

Στο βήμα 6, το Σύστημα αποτυγχάνει να εγκρίνει την αγορά μέσω πιστωτικής.

Ο Πελάτης μπορεί να ξαναδώσει τα στοιχεία της πιστωτικής του κάρτας και να ξαναπροσπαθήσει

Εναλλακτική: Μόνιμος (Σταθερός) Πελάτης

3α. Το σύστημα παρουσιάζει τη διεύθυνση του και τα 4 τελευταία ψηφία της πιστωτικής του

3β. Ο Πελάτης μπορεί να δεχθεί ή να επικαλύψει αυτά τα στοιχεία

Επιστροφή στο βήμα 8 του αρχικού σεναρίου



Παράδειγμα

Buy a Product

1. Customer browses through catalog and selects items to buy
2. Customer goes to check out
3. Customer fills in shipping information (address, next-day or 3-day delivery)
4. System presents full pricing information, including shipping
5. Customer fills in credit card information
6. System authorizes purchase
7. System confirms sale immediately
8. System sends confirming email to customer

Alternative: Authorization Failure

At step 6, system fails to authorize credit purchase
Allow customer to re-enter credit card information and re-try

Alternative: Regular Customer

- 3a. System displays current shipping information, pricing information, and last 4 digits of credit card information
 - 3b. Customer may accept or override these details
- Return to primary scenario at step 6



Τύποι Περιπτώσεων Χρήσης

ΕΠΟΠΤΙΚΕΣ/ ΒΑΣΙΚΕΣ (Overview/Essential):

- Υψηλού επιπέδου, που συνήθως συντάσσονται στην αρχή του έργου. Περιγράφουν τα ελάχιστα που χρειάζεται για να κατανοήσουμε την λειτουργικότητα του συστήματος. Είναι ανεξάρτητες υλοποίησης
 - (αλλιώς λέγονται Business Use Cases ή Overview Use Cases)

Λεπτομερείς/Πραγματικές (Detailed/Real)

- Εκλεπτύνουν και περιγράφουν ακριβώς τις εποπτικές. Περιγράφουν συγκεκριμένα βήματα (π.χ. η γραμματέας προσθέτει ένα .pdf έγγραφο στο σύστημα και μετά ..)
 - (αλλιώς λέγονται System Use Cases)



Elements of a Use-Case Description

Use Case Name:	ID:	Importance Level:
Primary Actor:	Use Case Type:	
Stakeholders and Interests:		
Brief Description:		
Trigger:		
Relationships: (Association, Include, Extend, Generalization)		
Normal Flow of Events:		
Subflows:		
Alternate/Exceptional Flows:		

Taken from Dennis et al. 2005



Παράδειγμα: Κλείσιμο Ιατρικού Ραντεβού (1/3)

Όνομα Περίπτωσης Χρήσης: **Κλείσιμο Ραντεβού** Αριθμός: **2** Σπουδαιότητα: **Υψηλή**

Κύριος Δράστης: **Ασθενής** Τύπος Περ. Χρήσης: **Αναλυτική/Ουσιώδης**

Εμπλεκόμενοι και Στόχοι:

Ασθενής: θέλει να ορίσει, αλλάξει ή ακυρώσει ένα ιατρικό ραντεβού

Σύντομη Περιγραφή: **Ο ασθενής θέλει να ορίσει, αλλάξει ή ακυρώσει ένα ιατρικό ραντεβού**

Ερέθισμα (συμβάν ενεργοποίησης): **Ο ασθενής συνδέεται και επιλέγει PANTEBOY**

Τύπος ερεθίσματος: **Εξωτερικό**

Σχέσεις:

Συσχέτιση (association)	: Ασθενής
Περιλαμβάνει (include)	: «Διευθέτηση Τρόπου Πληρωμής»
Επεκτείνει (extend)	: «Δημιουργία Νέου Ασθενούς»
Εξειδικεύει (generalization)	:



Παράδειγμα: Κλείσιμο Ιατρικού Ραντεβού (2/3)

Φυσιολογική ροή γεγονότων:

1. Ο ασθενής συνδέεται με τον ιστόχωρο
2. Ο ασθενής πληκτρολογεί το όνομα του και τον αριθμό μητρώου του
3. Το σύστημα επιβεβαιώνει ότι ο ασθενής είναι καταγεγραμμένος στη βάση
4. Το σύστημα εκκινεί την περίπτωση χρήσης «Διευθέτηση Τρόπου Πληρωμής»
5. Ο ασθενής επιλέγει τι θέλει να κάνει:
αν θέλει νέο ραντεβού, δες P1
αν θέλει ακύρωση ραντεβού, δες P2
αν θέλει αλλαγή ραντεβού δες P3
6. Το σύστημα εμφανίζει τα αποτελέσματα του 5 στον χρήστη

Συνιστώσες ροές γεγονότων

P1: Νέο ραντεβού

1. Το σύστημα ρωτά τον χρήστη για την επιθυμητή ημερομηνία και ώρα
2. Το σύστημα προγραμματίζει το νέο ραντεβού λαμβάνοντας υπόψη τις επιθυμίες του ασθενή και τις διαθέσιμες ημερομηνίες



Παράδειγμα: Κλείσιμο Ιατρικού Ραντεβού (3/3)

P2: Ακύρωση ραντεβού

1. Το σύστημα ρωτά τον χρήστη την ημερομηνία του προς ακύρωση ραντεβού
2. Το σύστημα βρίσκει και διαγράφει την αντίστοιχη εγγραφή από το πρόγραμμα των ραντεβού

P3: Αλλαγή ραντεβού

1. Το σύστημα εκτελεί το P2
2. Το σύστημα εκτελεί το P1

Εναλλακτικές Ροές - Εξαιρέσεις

3α: Το σύστημα εκτελεί την περίπτωση χρήσης «**Δημιουργία Νέου Ασθενούς**»

P1.2α1: Το σύστημα προτείνει εναλλακτικές ημερομηνίες/ώρες

P1.2α2: Ο ασθενής επιλέγει μια από αυτές ή αποφασίζει να μην κλείσει ραντεβού



Οδηγίες για τη σύνταξη Περιπτώσεων Χρήσης (Guidelines for Creating Use Case Descriptions)

- *Write each step in “Subject-Verb-Object” form*
- *Clarify who is the initiator in each step*
- *Describe the steps as if you were an independent external observer*
- *Write at the same level of abstraction*
- *Ensure that there is a sensible set of steps, like:*
 - *the primary actor initiates the dialog and sends data to the system*
 - *the system checks the validity of the data/request*
 - *the system processes the request*
 - *the system returns to the actor the results of the processing*
- *Apply KISS principle liberally*
- *Write repeating instructions after the set of steps to be repeated.*

Adapted from Dennis et al. 2005



Διαγράμματα Περιπτώσεων Χρήσης Use Case Diagrams



Διαγράμματα Περιπτώσεων Χρήσης Use Case Diagrams

Σκοπός:

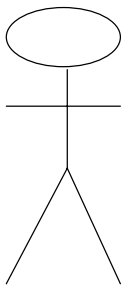
- Οπτικοποίηση πολλών Περιπτώσεων Χρήσης

Ένα Διάγραμμα Περιπτώσεων Χρήσης (Use Case Diagram) αποτελείται από:

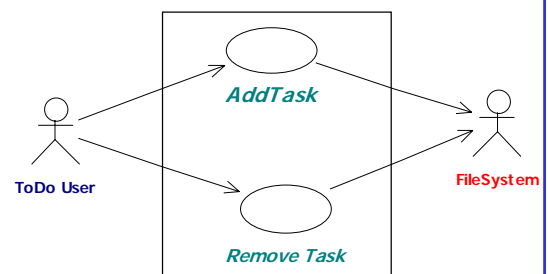
- Ένα πλαίσιο που υποδηλώνει το σύστημα
- Το πλαίσιο περιέχει ένα οβάλ για κάθε Περίπτωση Χρήσης
- Τα οβάλ των Περιπτώσεων Χρήσης μπορούν να συνδέονται με σχέσεις εγκλεισμού (*include*), γενίκευσης (*isA*) και επέκτασης (*extend*)
- Έξω από το πλαίσιο υπάρχουν σύμβολα που υποδηλώνουν τους **Δράστες (Actors)**, που είναι οι χρήστες (άνθρωποι ή άλλα συστήματα) του συστήματος οι οποίοι συνδέονται με ακμές με τα οβάλ των Περιπτώσεων Χρήσης στις οποίες μπορούν να λάβουν μέρος.
 - Οι δράστες μπορεί να συνδέονται με ακμές τύπου isA.

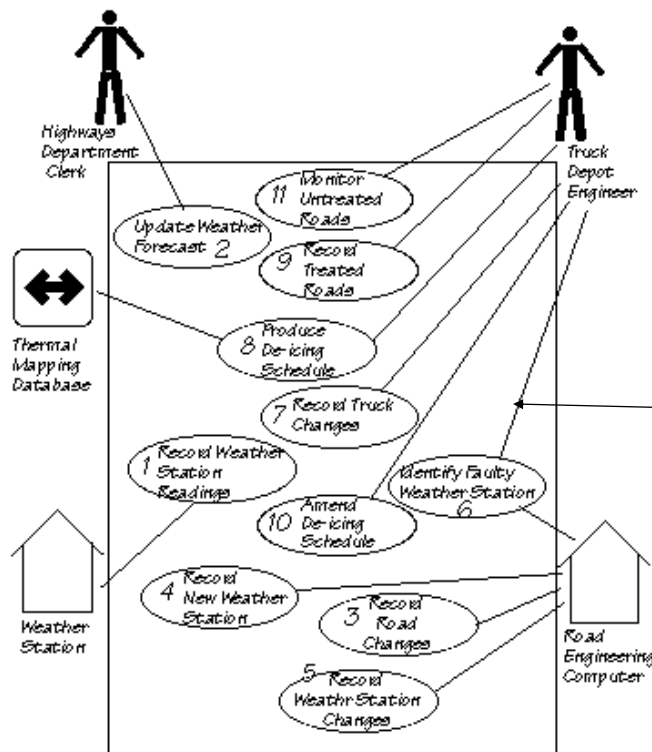


Use Case Diagrams > Actors Δράστες (Actors)



- Οι Δράστες «εκτελούν» τις Περιπτώσεις Χρήσης
- Δράστης = ρόλος που ένας χρήστης μπορεί να έχει σε σχέση με το σύστημα
 - Ένας χρήστης μπορεί να ενσαρκώσει πολλούς ρόλους
 - Οι δράστες δεν είναι πάντα άνθρωποι:
 - Μπορεί να είναι ένα εξωτερικό σύστημα που χρειάζεται κάποιες πληροφορίες από το εν λόγω.





Το πλαίσιο οριοθετεί το σύστημα
(Boundary of the system)

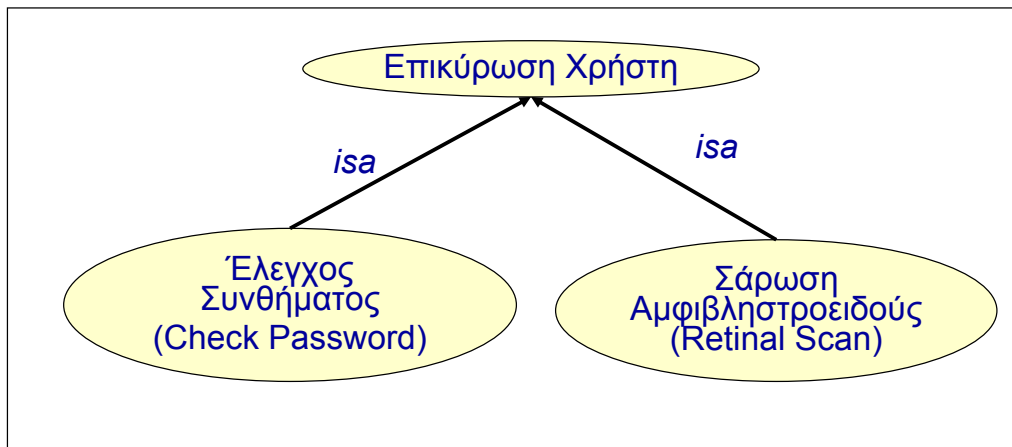
Taken from Volere Requirements Specification Document

Συσχετίσεις μεταξύ των περιπτώσεων χρήσης Use Case Diagrams > UC Relationships

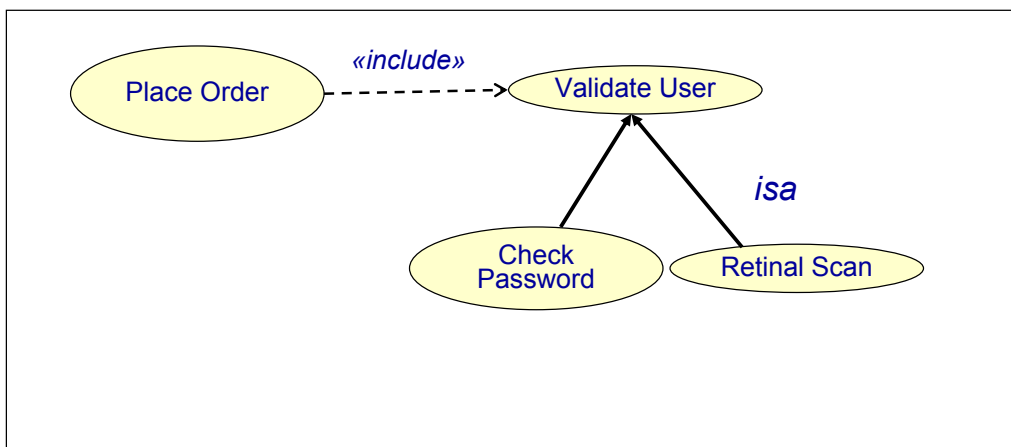
- **Include** \subseteq **UC** \times **UC**
 - Για αποφυγή επαναλήψεων (άρα για να αποφύγουμε το copy-paste)
- **ISA** \subseteq **UC** \times **UC**
 - Η ειδική ΠΧ μπορεί να υπερκαλύψει (override) οποιοδήποτε τμήμα της γενικής ΠΧ. Παρά ταύτα η ειδική πρέπει να αφορά στον γενικό στόχο του χρήστη στον οποίο αφορά και η γενική.
- **extend** \subseteq **UC** \times **UC**
 - Μοιάζει με την ISA. Η εκτεταμένη UC μπορεί να έχει επιπλέον συμπεριφορά αλλά μόνο σε δηλωμένα «σημεία επέκτασης» (“**extension points**”) της βασικής ΠΧ.



Παράδειγμα Γενίκευσης Example: Generalization

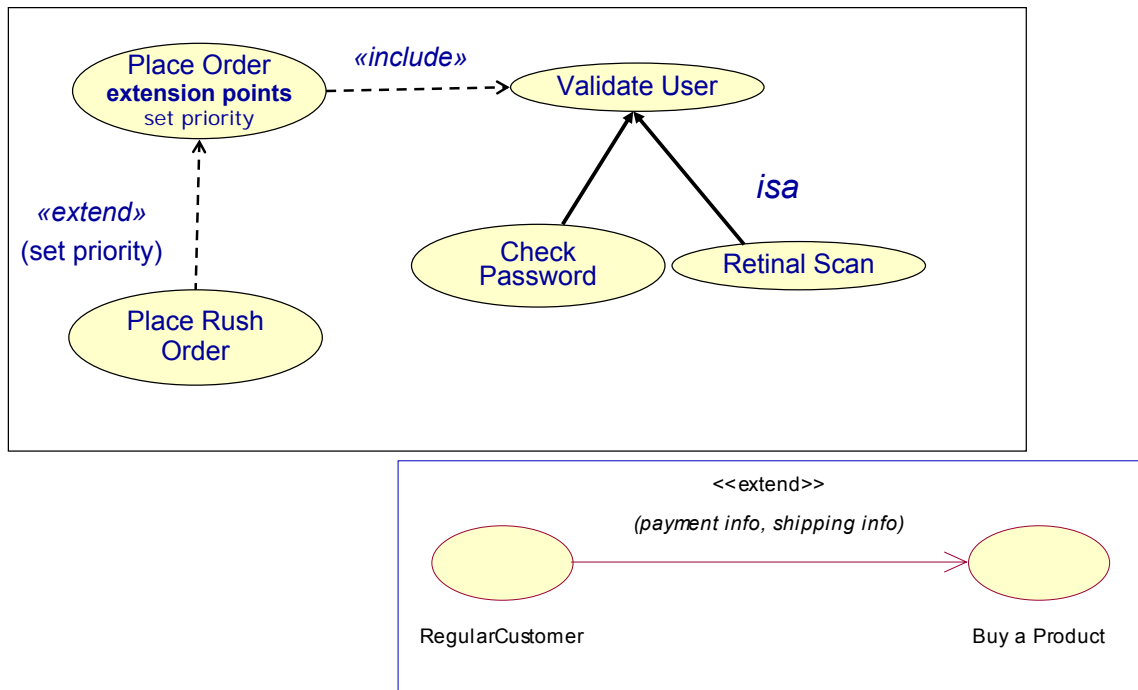


Παράδειγμα Γενίκευσης και Εγκλεισμού Example: Generalization + Include

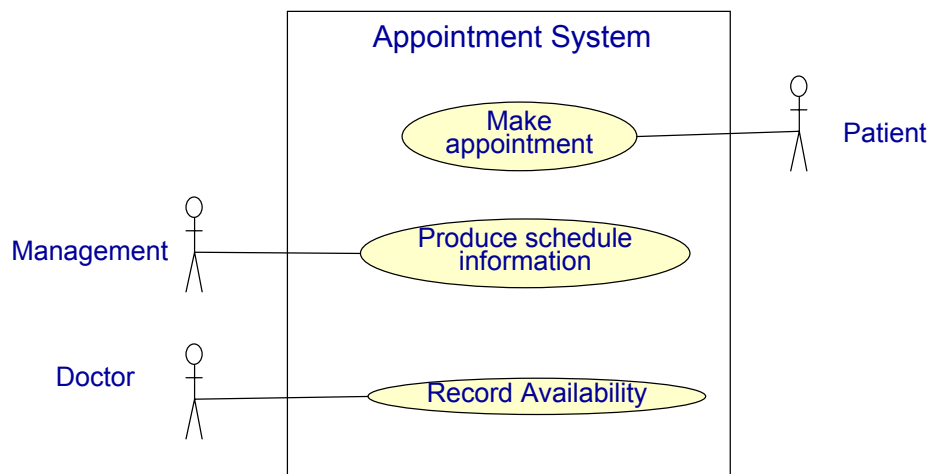




Παράδειγμα Γενίκευσης, Εγκλεισμού και Επέκτασης Example: Generalization + Include + Extend

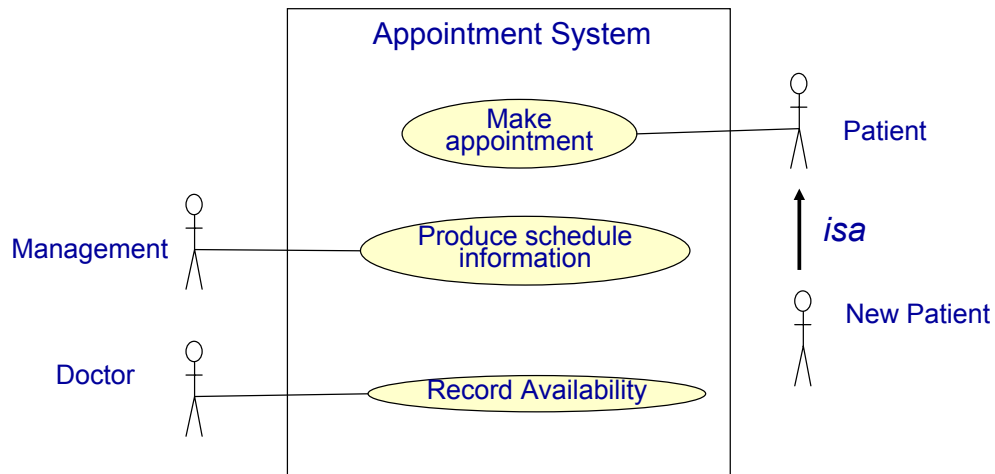


Παράδειγμα διαγράμματος που μοντελοποιεί τις λειτουργίες ενός συστήματος κράτησης ραντεβού

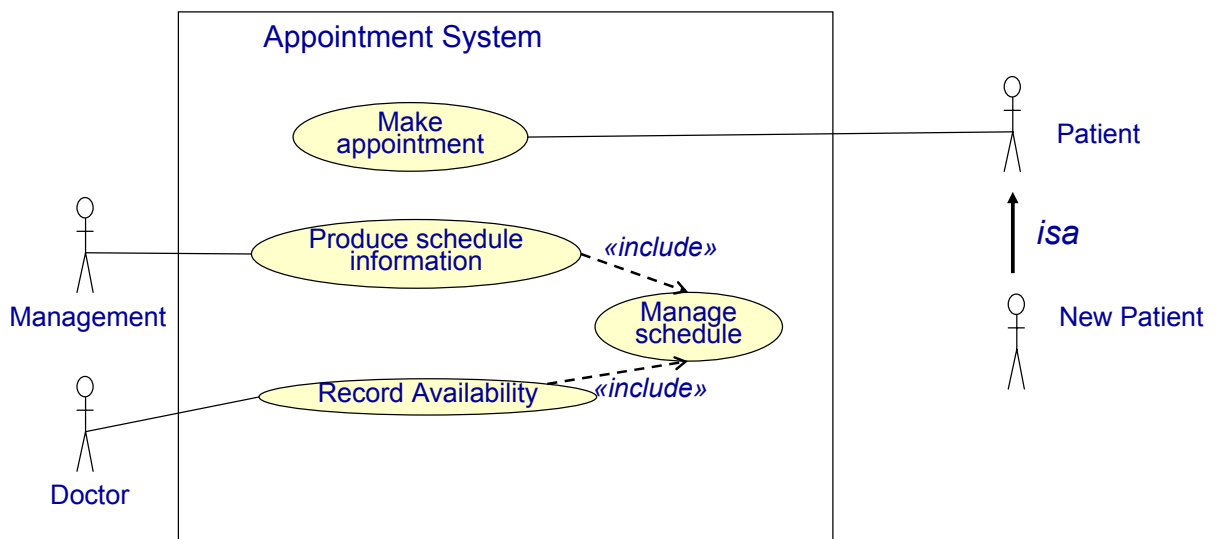




Εξειδίκευση Δραστήων (Specializing an actor)

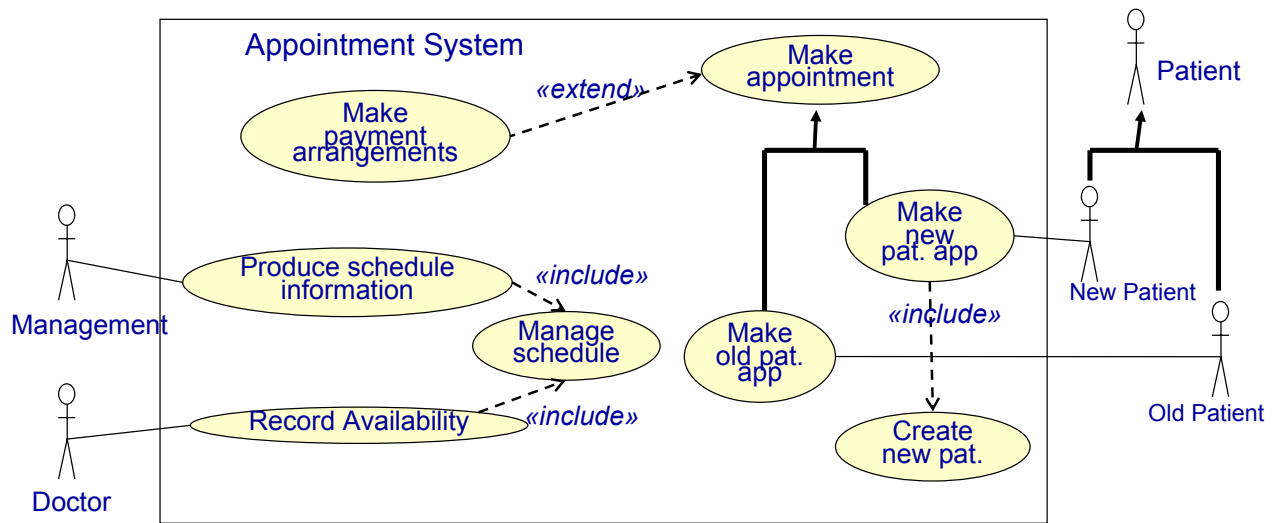


Use Case Diagram for an Appointment System Προσθήκη «include»





Use Case Diagram for an Appointment System Προσθήκη «Isa» και «extend»



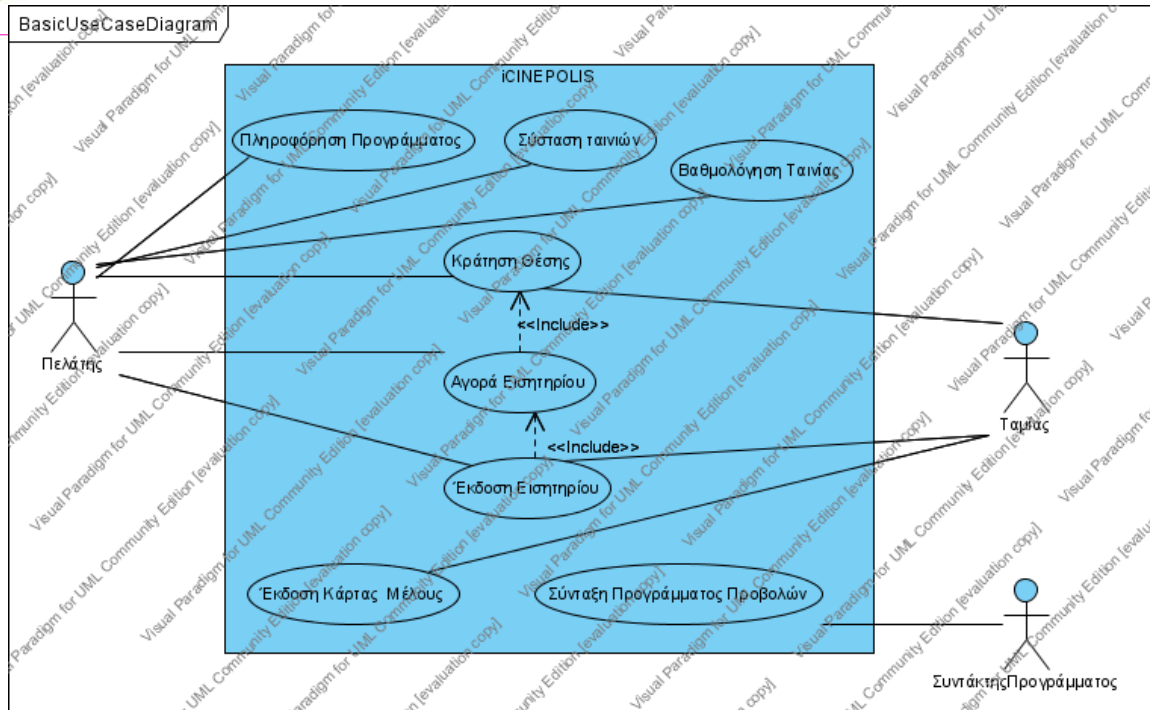
Παράδειγμα: Πληροφοριακό Σύστημα Κινηματογράφου

Στόχος:

- Αύξηση των κερδών μέσω μείωσης των λειτουργικών εξόδων και αύξησης των πελατών. Η αύξηση των πελατών αναμένεται να επιτευχθεί με την παροχή καλύτερων υπηρεσιών ενημέρωσης και εξυπηρέτησης πελατών. Το τελευταίο περιλαμβάνει κάρτα πελατών. Η μείωση των λειτουργικών εξόδων αναμένεται να επιτευχθεί με μείωση του προσωπικού για την έκδοση εισιτηρίων. Επίσης αναμένεται μείωση του χρόνου αναμονής στα εκδοτήρια εισιτηρίων και καλύτερη ενημέρωση πελατών για το πρόγραμμα προβολών μέσω email/SMS



Πληροφοριακό Σύστημα Κινηματογράφου



Μεθοδολογικά Ζητήματα

- Μπορούμε να ξεκινήσουμε από το Use Case Diagram και κατόπιν να περιγράψουμε κάθε Use Case
- Μετά μπορούμε να αναθεωρήσουμε και να βελτιώσουμε το UCD
- Πόσες Περιπτώσεις Χρήσης να ορίσουμε?
 - Π.χ. για ένα έργο 10 ανθρωποχρόνων \approx 12 Περιπτώσεις Χρήσης



Βήματα Σύνταξης Περιπτώσεων Χρήσης (Steps in Writing Use Cases)

A/ Identify the major UCs

- 1 Review the Activity Diagram (in case it is available)
- 2 Identify the system's boundaries
- 3 List the primary actors and their goals
- 4 Identify and write the major use cases
- 5 Carefully review use cases

B/ Expand the major UCs

- 6 Choose one major use case to expand
- 7 Fill in details on the use-case template
- 8 Fill in the steps of the normal flow of events
- 9 Normalize the size of each step
- 10 Describe alternate or exceptional flows
- 11 Simplify and organize as necessary

C/ Confirm the major UCs

- 12 Review the current set (check semantics, involve the users)
- 13 Iterate the entire set of steps

D/ Create the UC diagram

- 14 Start with system boundary
- 15 Place elements in order to be easy to read
- 16 Place actors on the diagram, connect them with UCs



Μερικά Πλεονεκτήματα των Περιπτώσεων Χρήσης

- Αντί της μοντελοποίησης και της κατανόησης ολόκληρου του οργανισμού, προσπαθούμε να καταλάβουμε κάθε Περίπτωση Χρήσης ξεχωριστά
- Από τις Περιπτώσεις Χρήσης μπορούμε να εκτιμήσουμε το χρόνο και την προσπάθεια που θα απαιτηθεί για το σχεδιασμό και υλοποίηση του συστήματος
- Ο Διευθυντής του έργου (Project manager) μπορεί να ελέγξει την πρόοδο του έργου παρατηρώντας την πρόοδο της κάθε Περίπτωσης Χρήσης



Διαγράμματα Δραστηριοτήτων Activity Diagrams



Τι μπορούμε να μοντελοποιήσουμε με Διαγράμματα Δραστηριοτήτων;

- Μπορούν να χρησιμοποιηθούν για να μοντελοποιήσουν οποιαδήποτε διαδικασία
 - Από επιχειρηματικές υψηλού επιπέδου, έως τη ροή ελέγχου μιας Περίπτωσης Χρήσης, ή τη ροή ελέγχου μιας συγκεκριμένης μεθόδου μιας κλάσης
- Μοιάζουν με *διαγράμματα ροής (flow charts)* αλλά έχουν την επιπλέον δυνατότητα να αναπαριστούν **παραλληλία (parallelism)**, **σύμπτωση (concurrency)** και **σύνθετες αποφάσεις (complex decisions)**.
- Συνδυάζουν ιδέες από διάφορες τεχνικές όπως:
 - event diagrams, SDL state modeling, workflow modeling, Petri Nets



Δραστηριότητες και Διαγράμματα Δραστηριοτήτων

Activities and Activity Diagrams

Activity (or activity state)

- Είναι μια κατάσταση πράττειν (state of doing something): είτε στον πραγματικό κόσμο (π.χ. πλήκτριση ενός γράμματος) ή στον υπολογιστή (π.χ. εκτέλεση μιας ρουτίνας λογισμικού)
- Μπορεί να παριστάνει μια χειρονακτική ή μια αυτοματοποιημένη δραστηριότητα
- Κάθε δραστηριότητα έχει ένα όνομα
 - Παραδείγματα ονομάτων: Λήψη Παραγγελίας, Κράτηση Ραντεβού, κλπ.

Activities vs Actions

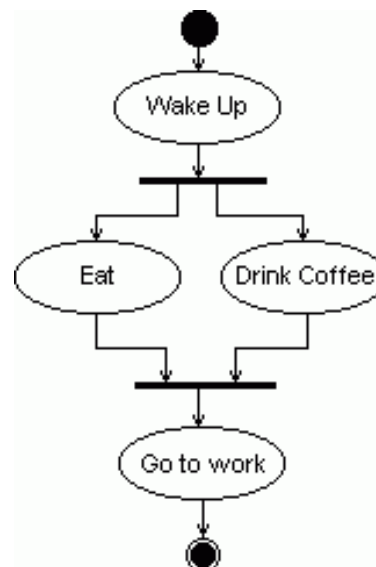
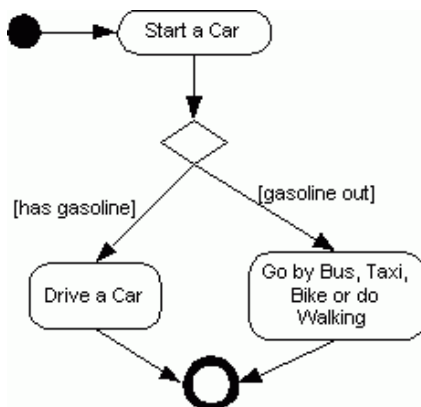
- Actions: άτομα (non decomposable)
- Activities: μπορούν να αναλυθούν σε άλλες activities ή/και actions.

Activity Diagram

- Περιγράφει την ακολουθία των δραστηριοτήτων και υποστηρίζει συνθήκες και παραλληλία



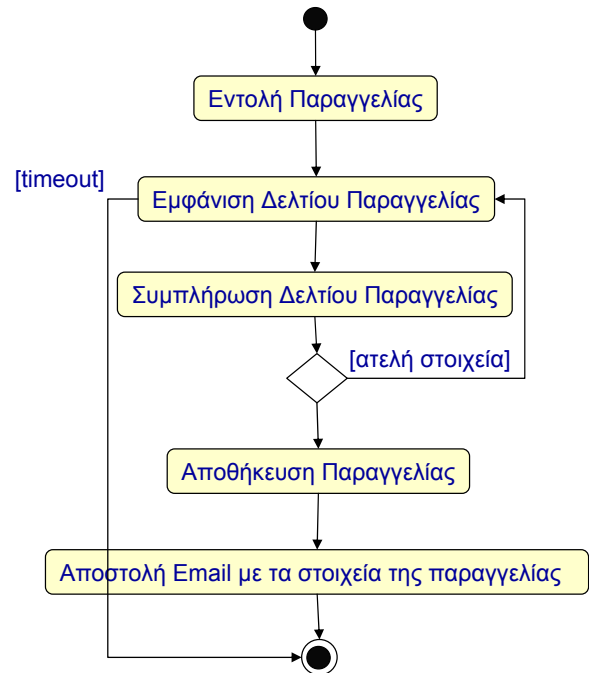
Καθημερινά παραδείγματα (πηγή: <http://odl-skopje.etf.ukim.edu.mk/uml-help/>)





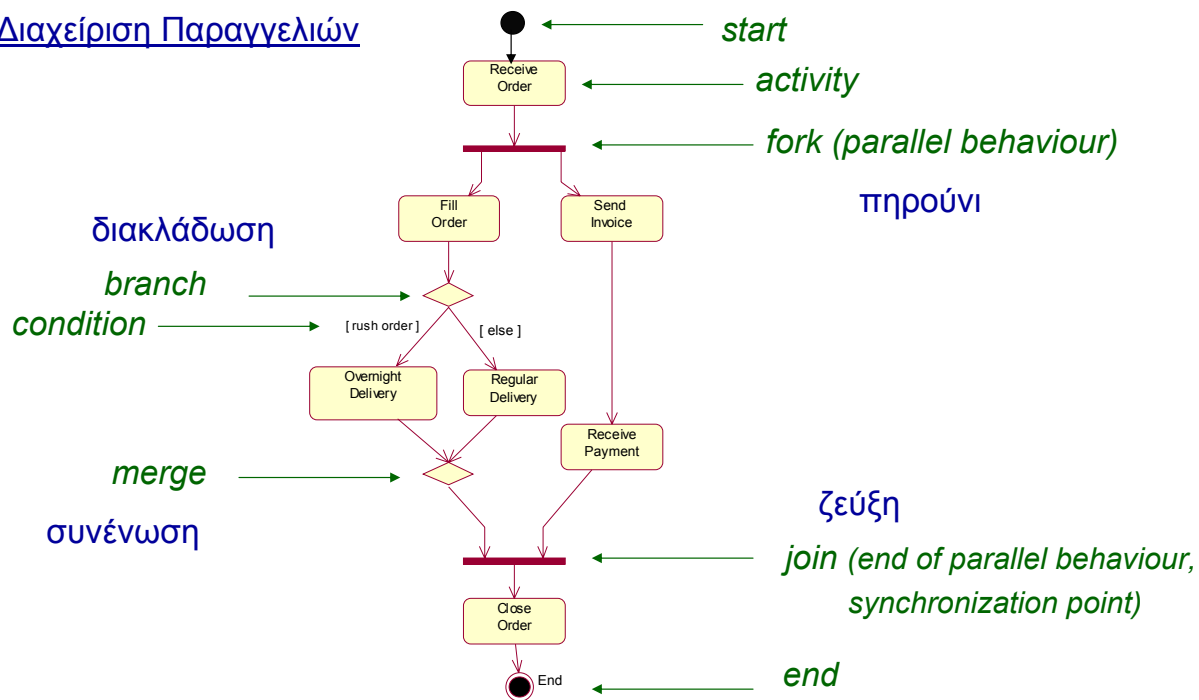
Ένα απλό παράδειγμα Διαγράμματος Δραστηριοτήτων

Περιγράφει τον «κλασικό» τρόπο
Λήψεως Παραγγελιών σε ένα
ιστότοπο.



Ένα διάγραμμα δραστηριοτήτων

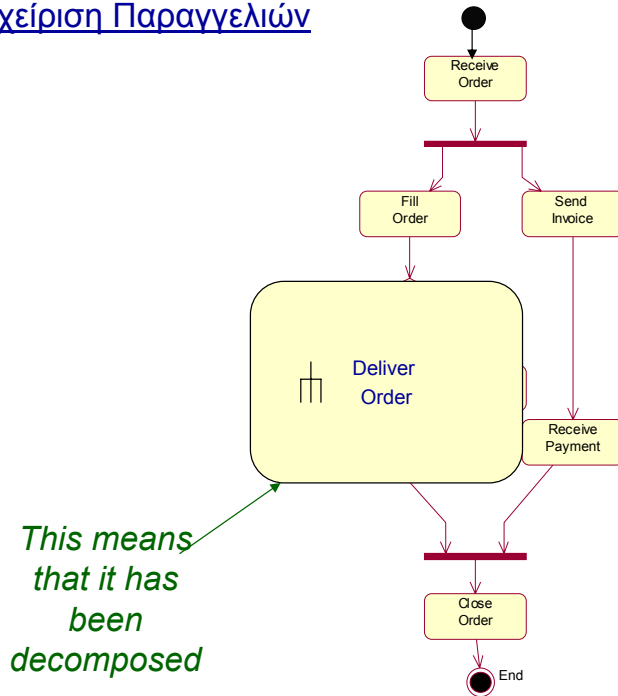
Διαχείριση Παραγγελιών





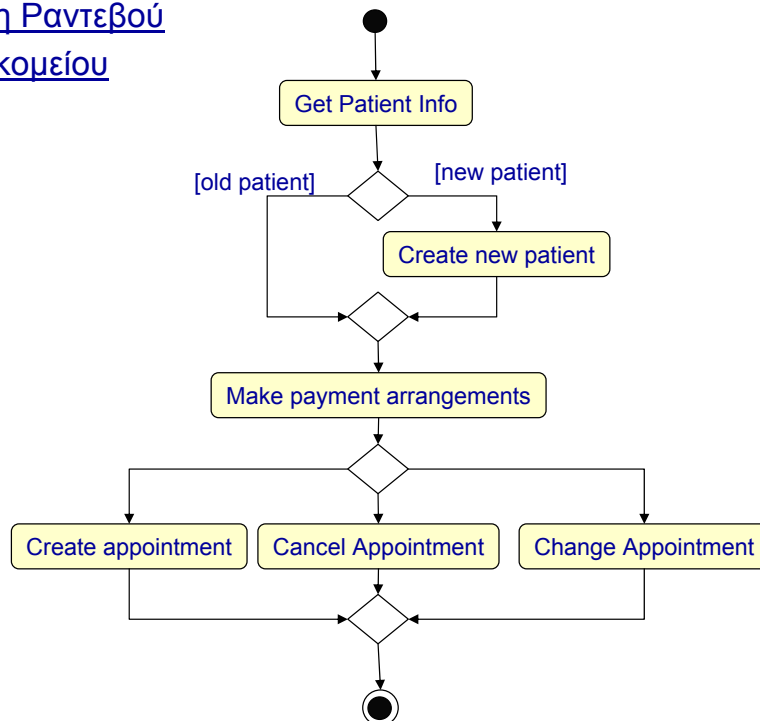
Ένα διάγραμμα δραστηριοτήτων

Διαχείριση Παραγγελιών



Άλλο ένα παράδειγμα

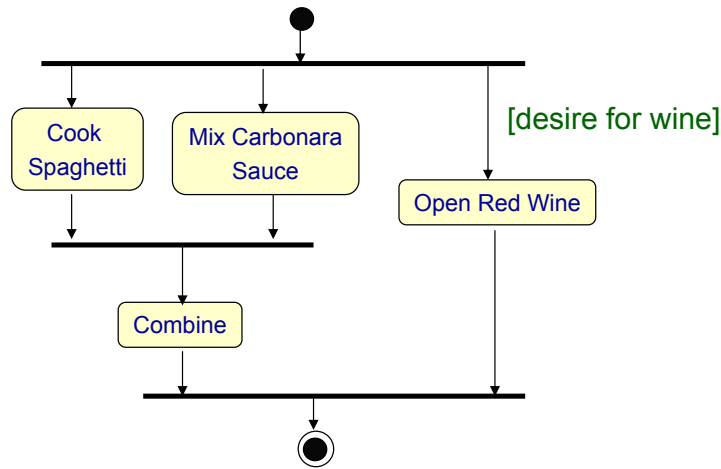
Διαχείριση Ραντεβού Νοσοκομείου





Another Example with conditional thread Παράδειγμα με «υποθετικό» νήμα

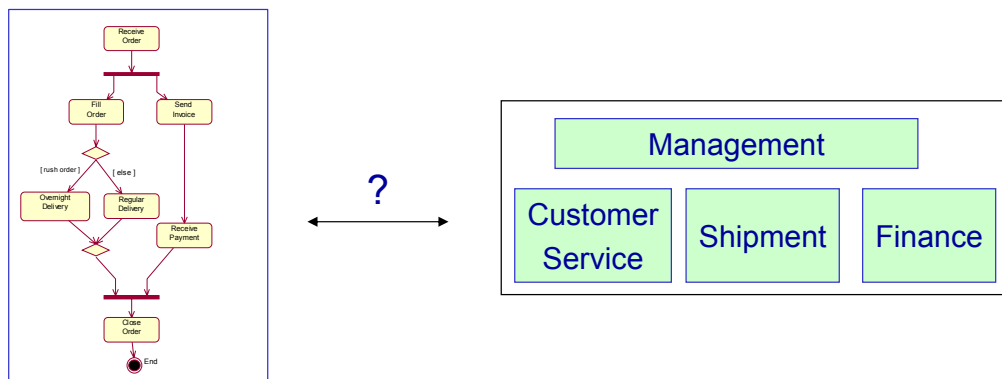
Ετοιμασία Δείπνου



Γραμμές Κολύμβησης Swimlanes

Τα ΔΔρ περιγράφουν τι συμβαίνει αλλά **δεν μας λένε ποιος κάνει τι**

- Π.χ. δεν μας λέει ποια κλάση είναι υπεύθυνη για την κάθε εργασία, ποιο τμήμα ενός οργανισμού είναι υπεύθυνο για την περιγραφόμενη δραστηριότητα κλπ.



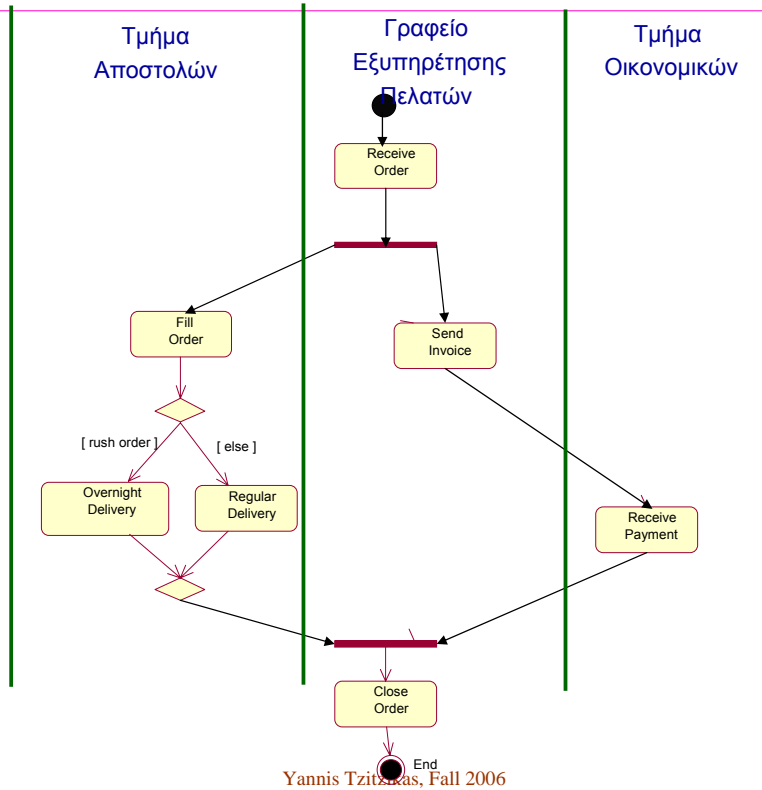
=> Γραμμές Κολύμβησης (**Swimlanes**)

- Χωρίζουν το διάγραμμα σε κατακόρυφες ζώνες. Κάθε ζώνη είναι αρμοδιότητα ενός συγκεκριμένου δράστη ή κλάσης
 - Στην UML V.2.0: οι γραμμές κολύμβησης ονομάζονται **Partitions**



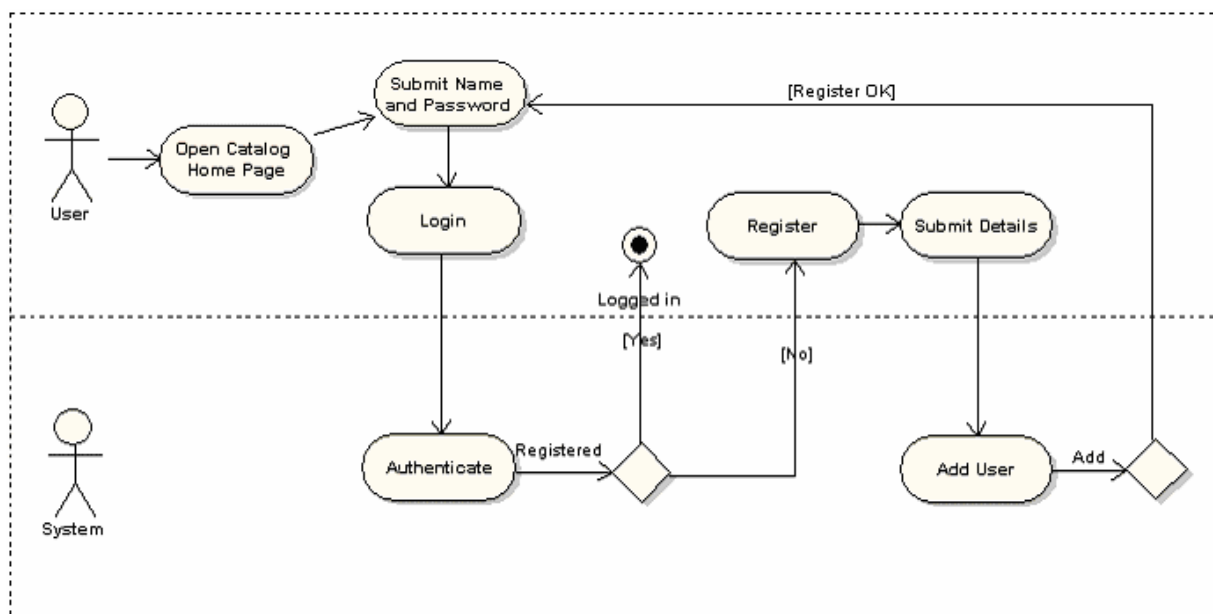
Γραμμές Κολύμβησης: Παράδειγμα

Διαχείριση Παραγγελιών



Παράδειγμα: Login (and register if new user)

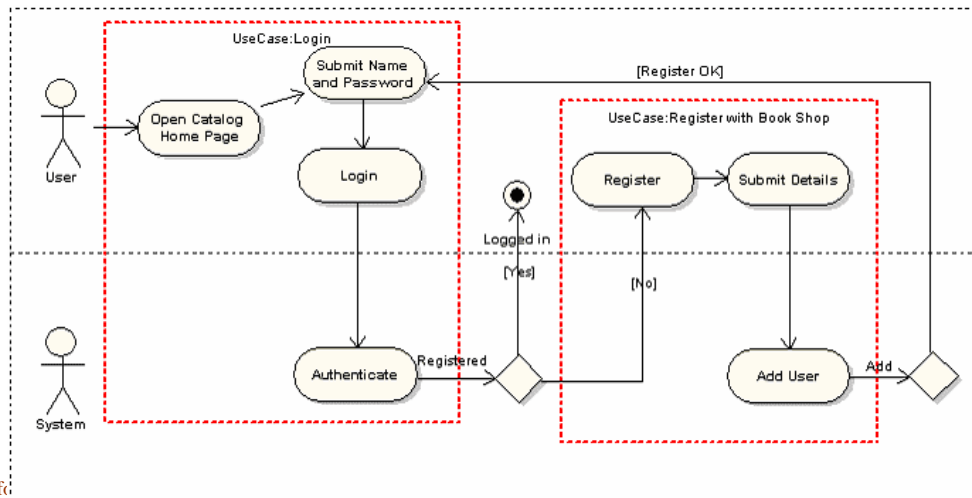
- Πηγή: http://www.sparxsystems.com.au/resources/map_uc.html (Entreprise Architect)





Παράδειγμα: Login (and register if new user)

Enterprise Architect provides a method of overlaying Use Cases onto activity diagrams. In the example below, the Use Case 'Login to System' has been overlaid onto the previous diagram. The Red bordered areas above are linked to existing Use Cases by setting the classifier. These 'Use Case Instances' then can be overlaid onto the associated Activities to indicate how particular business process is going to be implemented in the proposed system. Mapping Use Cases onto Activity Diagrams provides a good means of visualizing the overlay of system behavior onto business process.



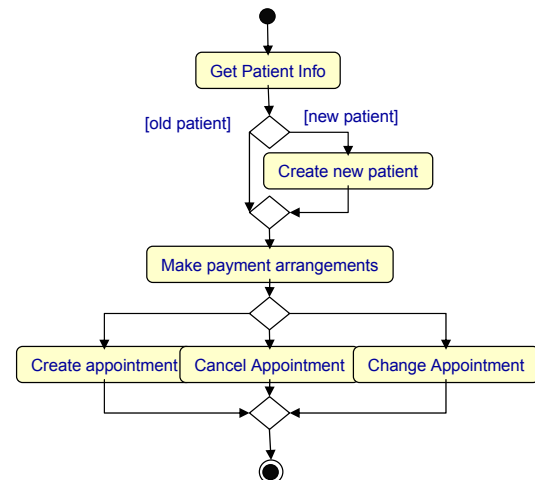
U. of Crete, Info

51



Ροή Αντικειμένων (Object flow)

Έστω ότι έχουμε μοντελοποιήσει κάπως τη δομική πλευρά του συστήματος και έχουμε ήδη ένα σύνολο κλάσεων (που είτε αντιστοιχούν στις βασικές έννοιες του πεδίου εφαρμογής ή στις κλάσεις του υπό κατασκευή συστήματος).

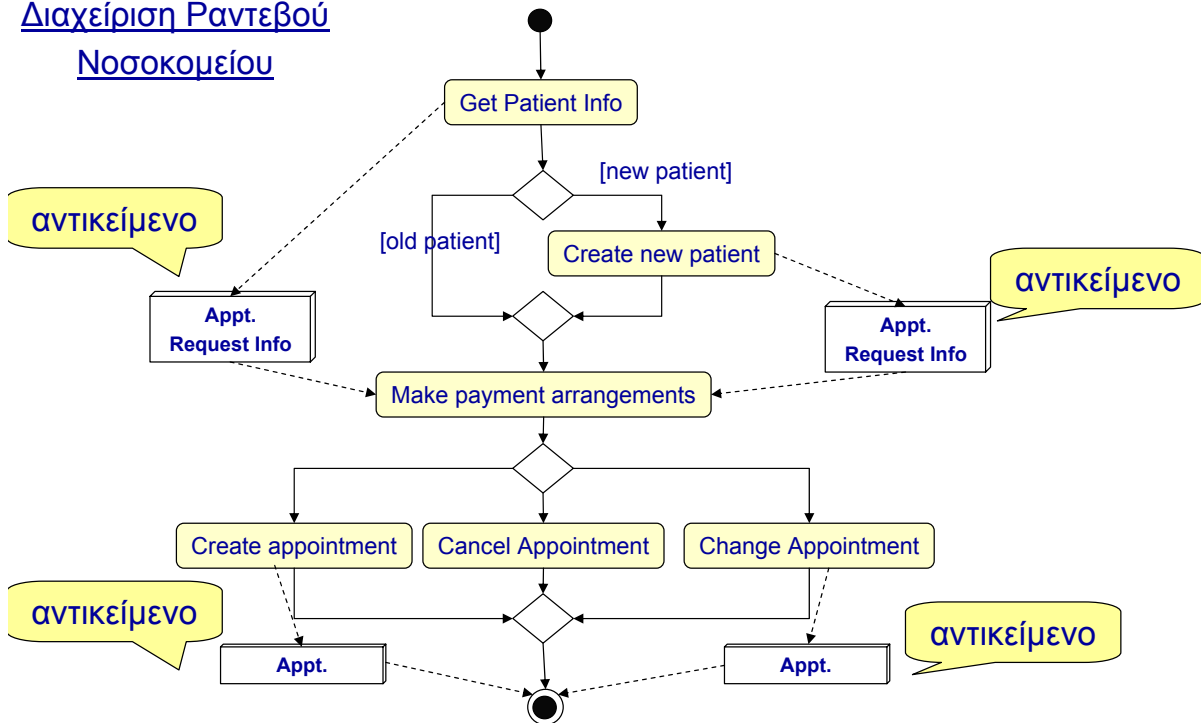


Πως μπορούμε να δείξουμε τα αντικείμενα που εμπλέκονται στη ροή ελέγχου?



Ανάδειξη της Ροής Αντικειμένων (Adding Object Flow)

Διαχείριση Ραντεβού Νοσοκομείου



Διαφορετικοί αλλά ισοδύναμοι τρόποι απεικόνισης μιας ακμής



connectors
(useful to avoid crossings)



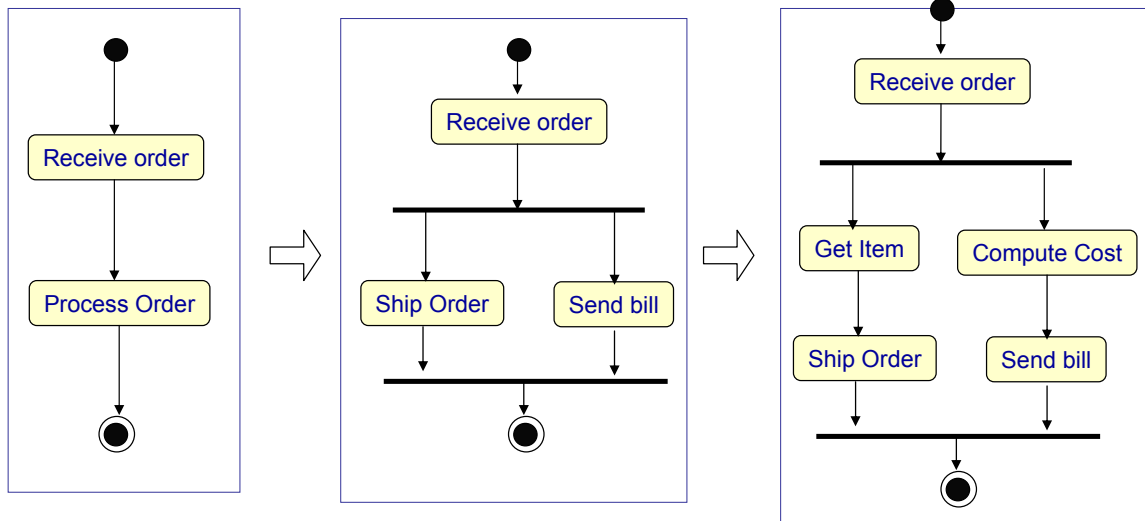
Object flow



Pins
(can be used to denote transformations)



Παραδείγματα σταδιακής ανάλυσης μιας δραστηριότητας

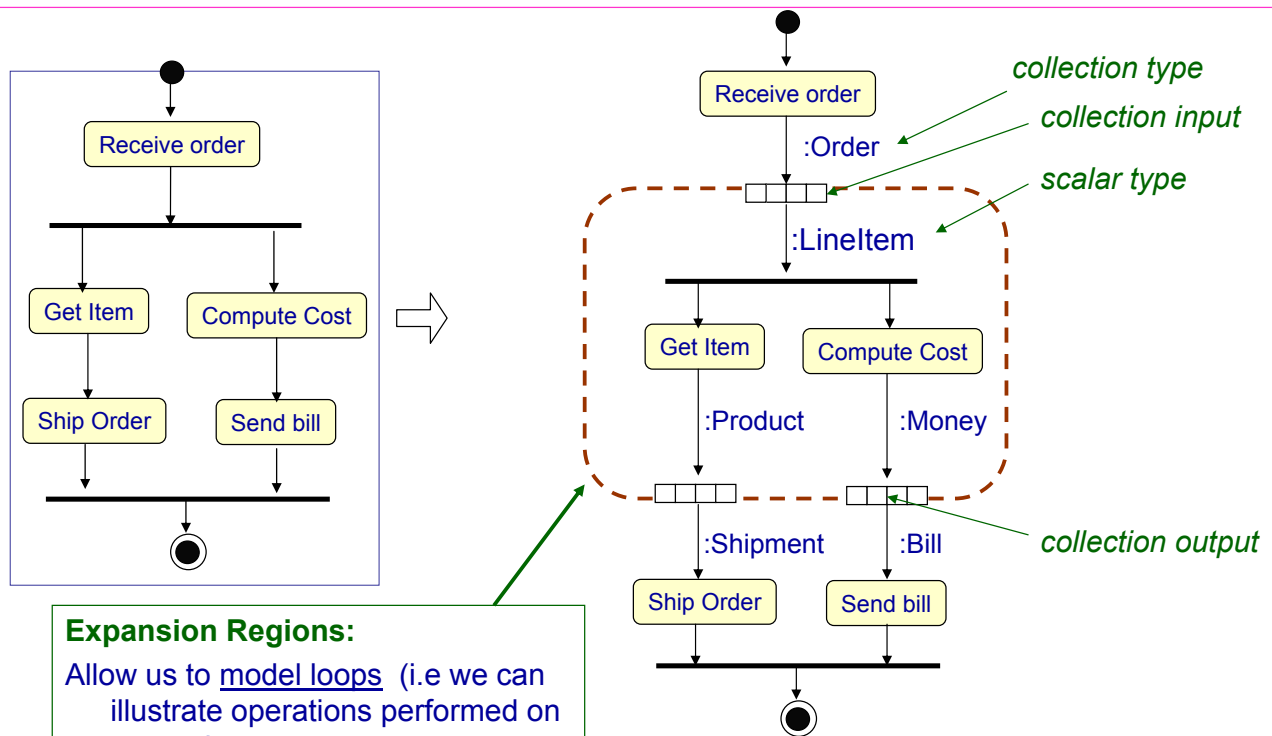


Τι λείπει ?

Μια παραγγελία μπορεί να αφορά πολλά στοιχεία (Items)
Πως μπορούμε να παραστήσουμε κάτι τέτοιο;



Παραδείγματα ανάλυσης μιας δραστηριότητας Περιοχές Επέκτασης (Expansion Regions)

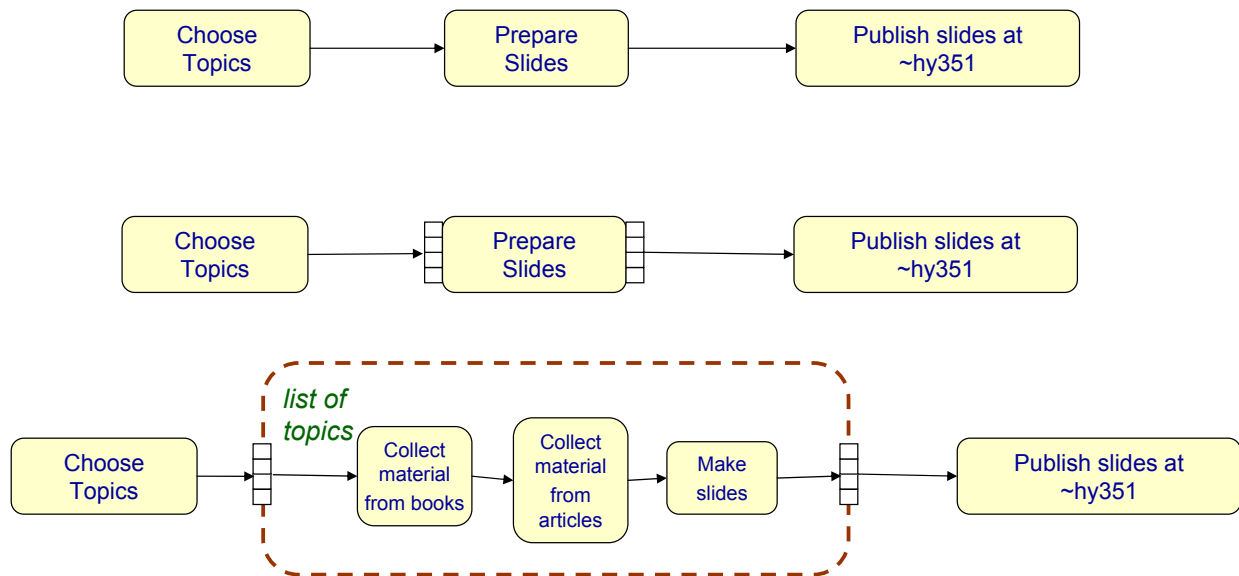


Expansion Regions:
 Allow us to model loops (i.e we can illustrate operations performed on a set of elements)



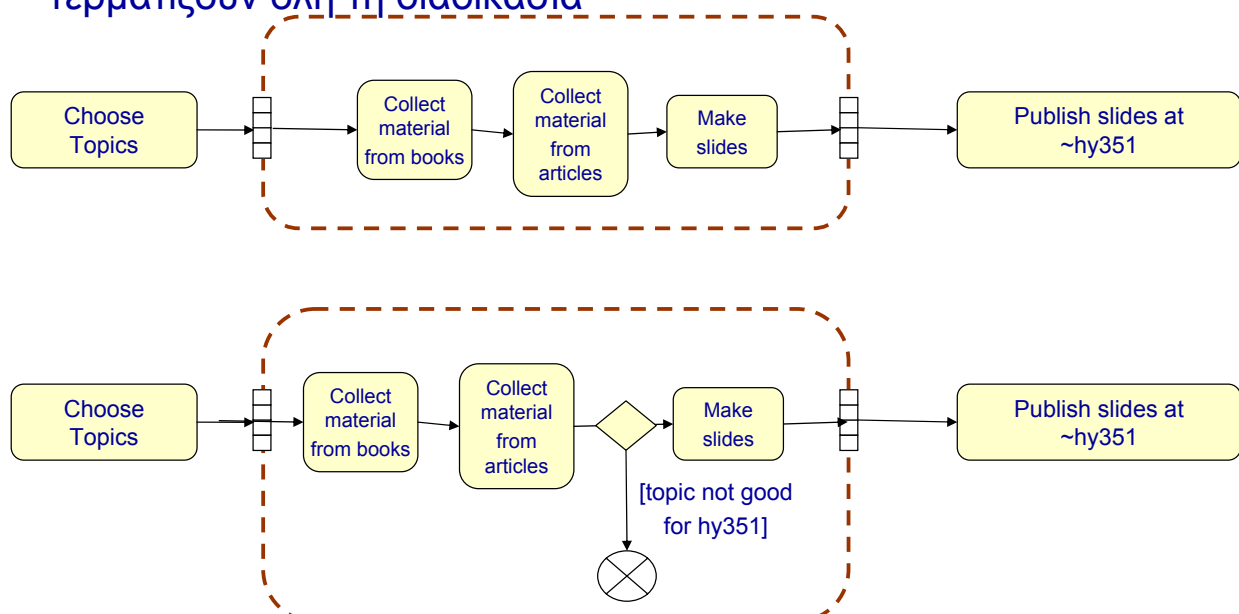
Άλλο ένα παράδειγμα με περιοχές επέκτασης

Προετοιμασία HY351



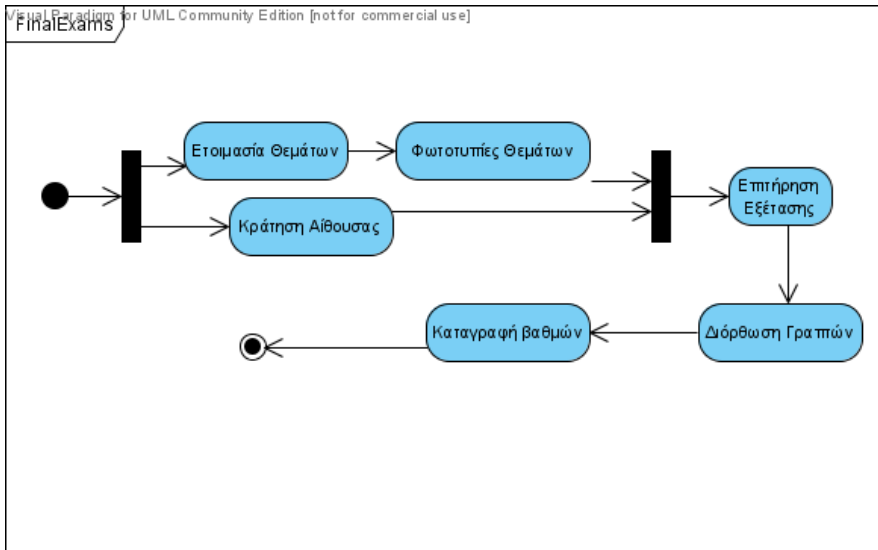
Περιοχές επέκτασης και Τέρματα Ροής (Flow finals)

Υποδηλώνουν το τέλος μια συγκεκριμένης ροής, χωρίς να τερματίζουν όλη τη διαδικασία

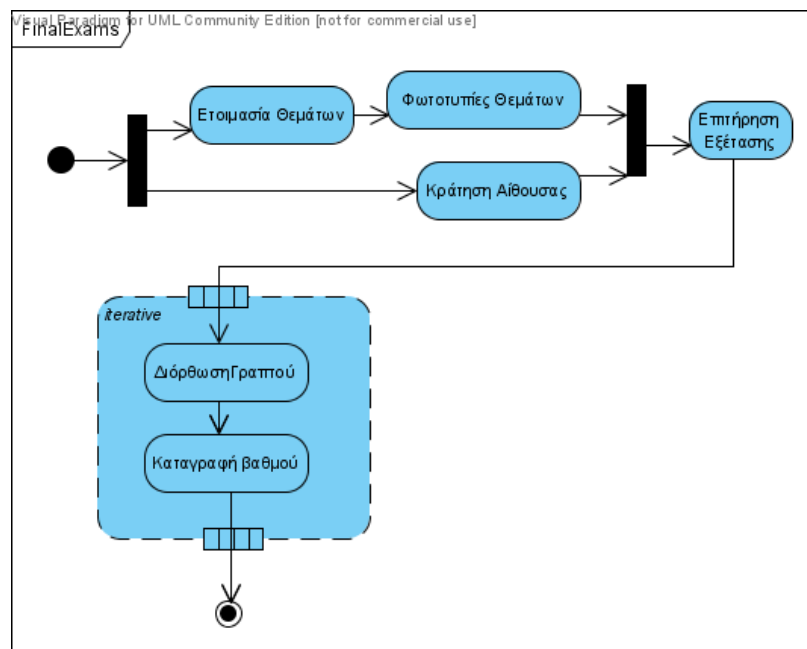




Παράδειγμα: Δραστηριότητα Τελική Εξέταση

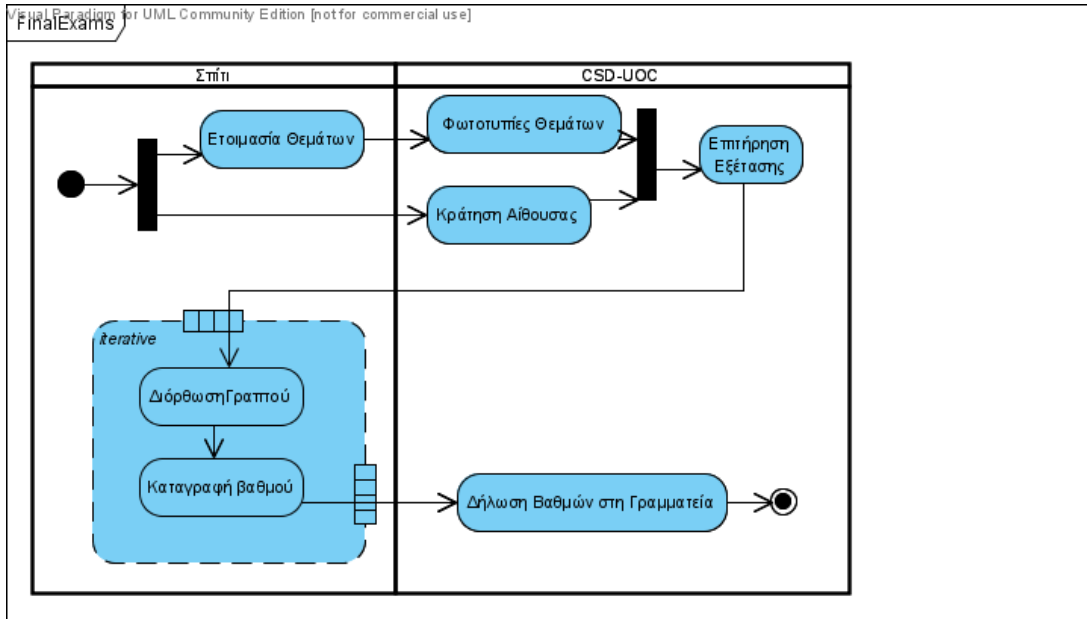


Δραστηριότητα Τελική Εξέταση

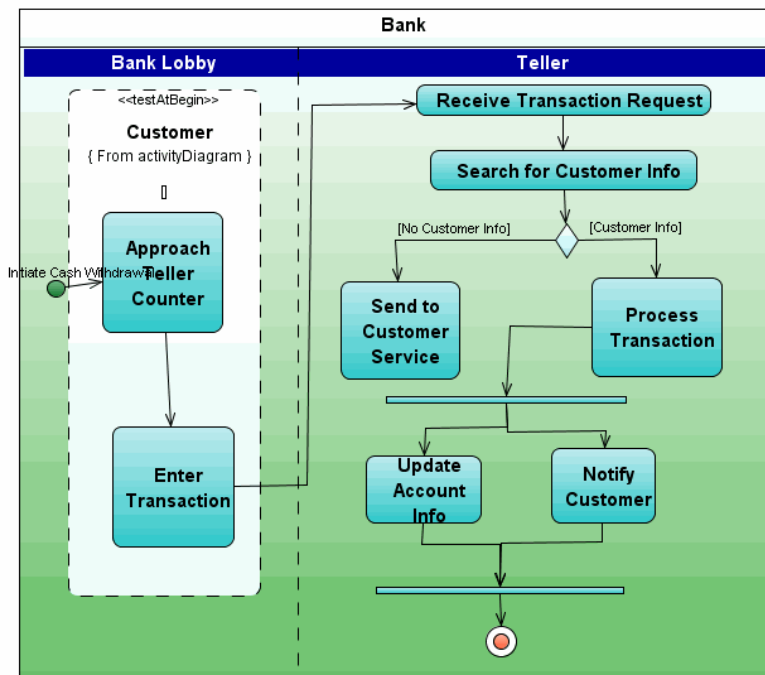




Δραστηριότητα Τελική Εξέταση προσθήκη partitions



Παράδειγμα: Ανάλυση Χρημάτων





Σήματα Signals

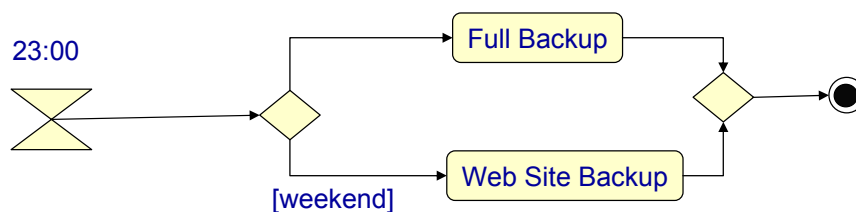
- Τα ΔΔρ έχουν ένα ξεκάθαρο σημείο εκκίνησης (start point)
 - Που μπορεί να είναι η κλήση ενός προγράμματος ή μιας ρουτίνας
- Μπορούμε όμως να έχουμε και actions που ανταποκρίνονται σε σήματα

Τύποι Σημάτων

- **Χρόνου (Time signal)**
 - Μπορεί να υποδηλώνουν το τέλος ενός μήνα, το τέλος του ακαδημαϊκού έτους, ή το τέλος ενός χιλιοστού του δευτερολέπτου σε έναν ελεγκτή πραγματικού χρόνου (real time controller)
- **Λήψης (Accept signal)**
- **Αποστολής (Send signals)**



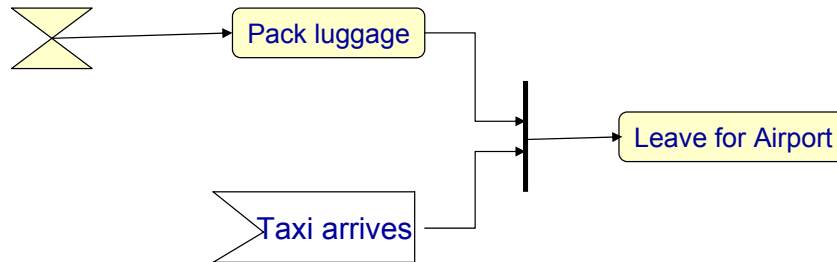
Παράδειγμα με Χρονικό Σήμα (Time Signal)





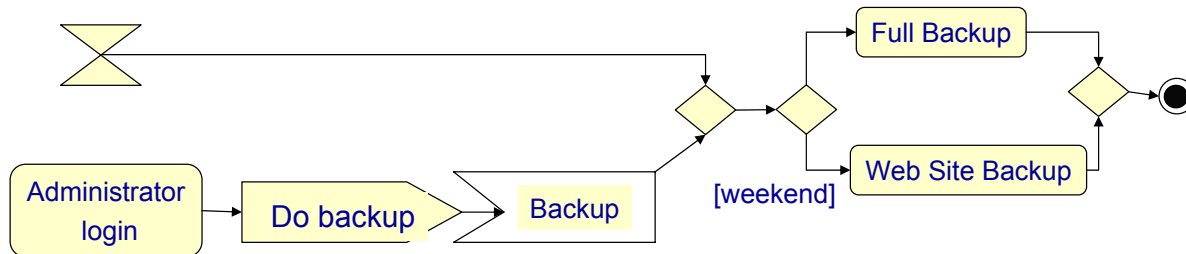
Παράδειγμα με Σήμα Χρόνου και Σήμα Αποδοχής (Time and Accept signals)

4 hours before flight



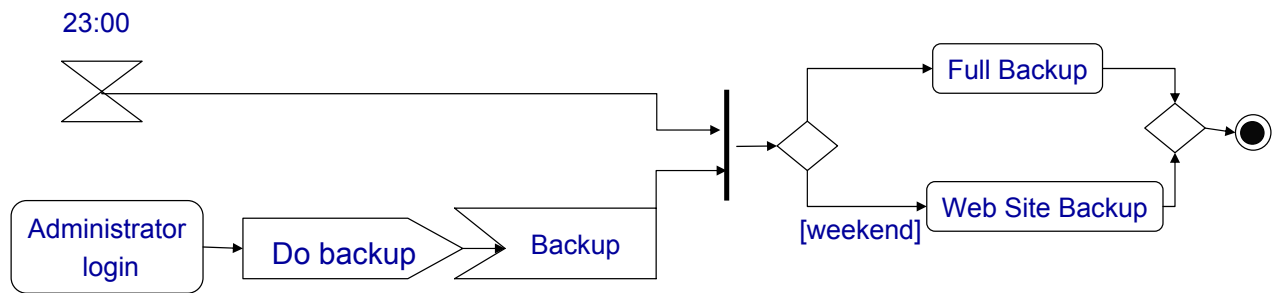
Παράδειγμα με Σήμα Χρόνου και Σήμα Αποστολής (Example with Time and Send Signal)

23:00





Παράδειγμα με Σήμα Χρόνου και Σήμα Αποστολής(2) (Example with Time and Send Signal)

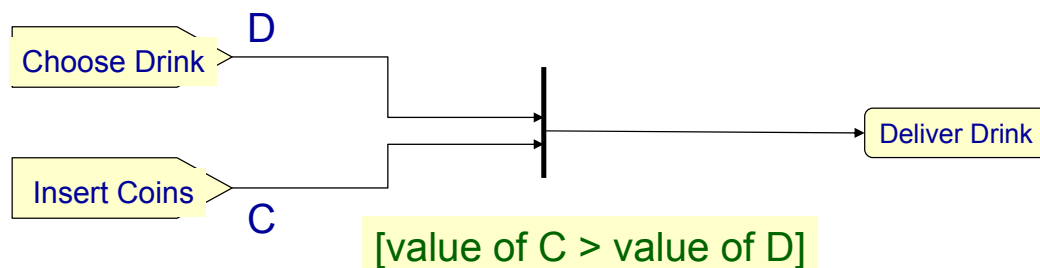


What's the difference?



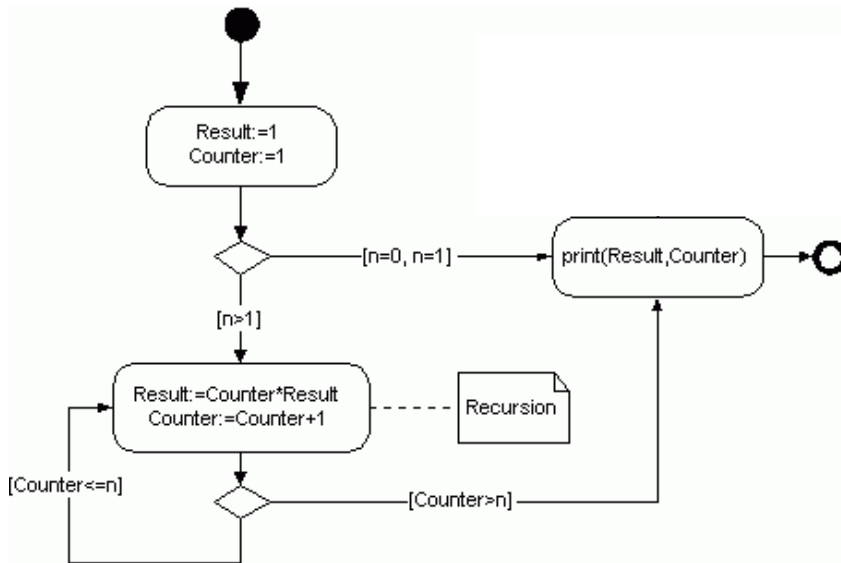
Περιγραφή των Ζεύξεων Join specifications

Λογικές εκφράσεις (boolean) μπορούν να προσδεθούν στις ζεύξεις
Boolean expressions attached to joins





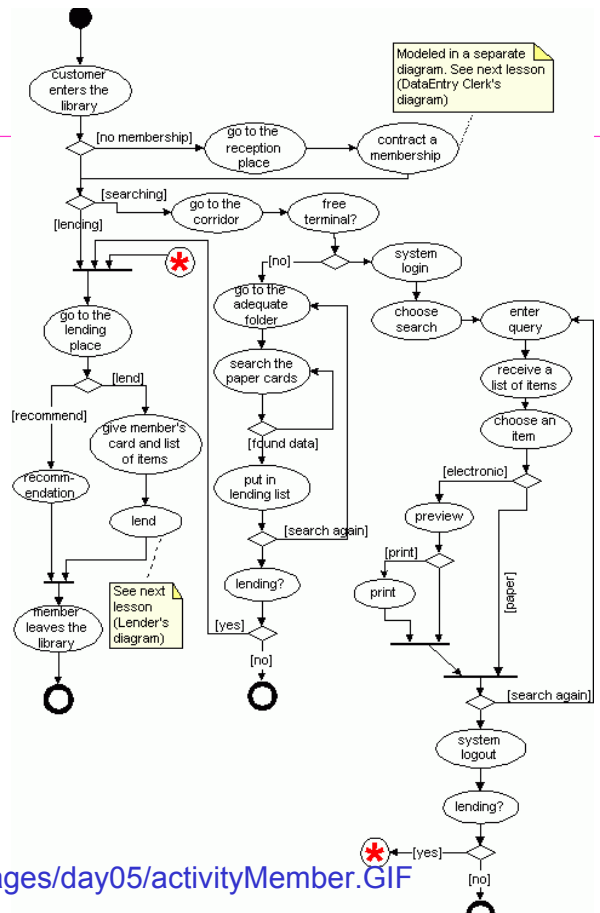
Παράδειγμα



Τι περιγράφει;

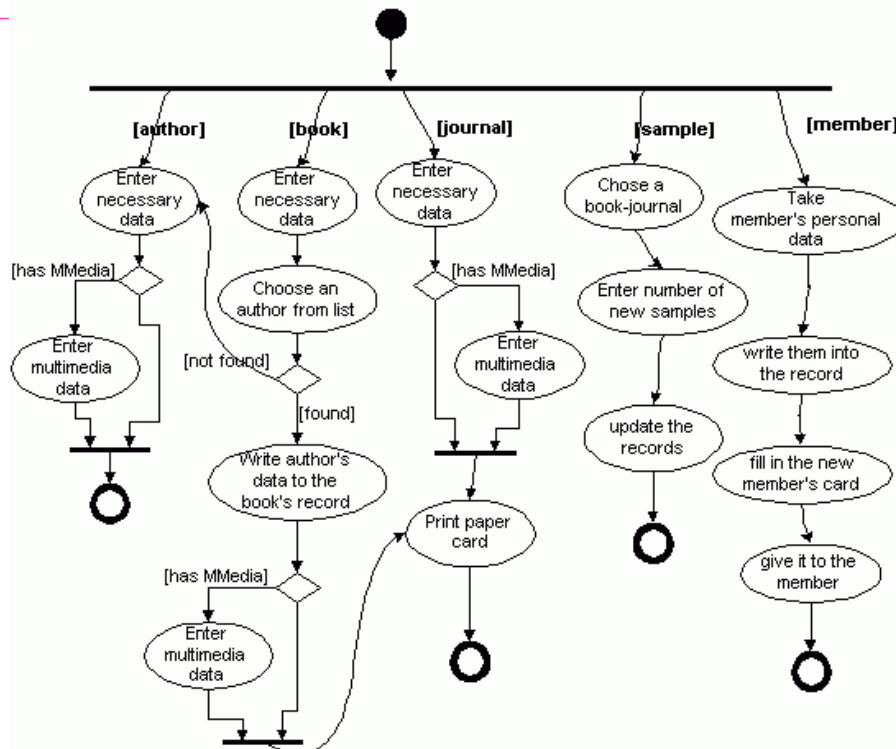
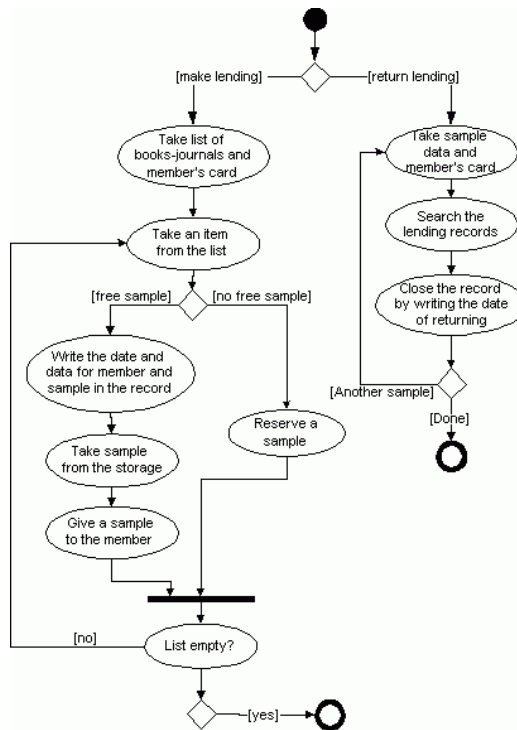


Library Example



Source:

<http://odl-skopje.etf.ukim.edu.mk/uml-help/html/images/day05/activityMember.GIF>





Μοντελοποίηση Επιχειρηματικών Διαδικασιών με Διαγράμματα Δραστηριοτήτων



Διαγράμματα Δραστηριότητας και Μεθοδολογίες Ανάπτυξης Πληροφοριακών Συστημάτων

- Αντικειμενοστρεφείς Μεθοδολογίες (Object-Oriented Methodologies)
 - Τα Διαγράμματα Δραστηριοτήτων χρησιμοποιούνται για να μοντελοποιήσουν τη συμπεριφορά μιας επιχειρηματικής δραστηριότητας ανεξάρτητα από τα αντικείμενα
- Δομημένες Μεθοδολογίες (Structured Methodologies)
 - Θα μπορούσαν να θεωρηθούν ως *σύνθετα διαγράμματα ροής δεδομένων* τα οποία χρησιμοποιούνται σε συνδυασμό με την δομημένη ανάλυση (structured analysis)

- Αρχικά η Αντικειμενοστρεφής Αν&Σχ δεν περιελάμβανε Διαγράμματα Δραστηριοτήτων (αλλά μόνο Περιπτώσεις Χρήσης)
- Σήμερα η τάση είναι να φτιάχνουμε μερικά από τις απαιτήσεις αφού συχνά βοηθούν την επικοινωνία με τον πελάτη
 - Δεν πρέπει όμως να υπερβάλλουμε και να καταλήξουμε να κάνουμε αναλυτική αποσύνθεση λειτουργιών (functional decomposition)

Στις περισσότερες περιπτώσεις κάθε δραστηριότητα θα συσχετισθεί με μια Περίπτωση Χρήσης



Πότε **πρέπει** και πότε **δεν πρέπει** να χρησιμοποιούμε Διαγράμματα Δραστηριότητας

- **Πότε πρέπει**
 - Για κατανόηση και εικονογράφηση των επιχειρηματικών ροών (business workflow)
 - Για εικονογράφηση της ροής δραστηριοτήτων σε μια Περίπτωση χρήσης
 - Για μοντελοποίηση πολυνηματικού (multithreaded) προγραμματισμού
 - Για μοντελοποίηση ενός σύνθετου σειριακού αλγορίθμου (flow chart)
- **Πότε δεν πρέπει**
 - Για περιγραφή του τρόπου συνεργασίας των αντικειμένων (τότε χρησιμοποιούμε Interaction Diagrams)
 - Για περιγραφή του τρόπου συμπεριφοράς ενός αντικειμένου κατά τη διάρκεια της ύπαρξης του (τότε χρησιμοποιούμε State Diagram)
 - Για περιγραφή σύνθετων αποφάσεων με πολλές συνθήκες (τότε καλύτερα να χρησιμοποιήσουμε έναν Πίνακα Αλήθειας)



Οδηγίες για τη σύνταξη Διαγραμμάτων Δραστηριότητας

Βήματα:

- 1/ Καθορισμός της εμβέλειας του διαγράμματος
- 2/ Καθορισμός τίτλου διαγράμματος
- 3/ Αναγνώριση των δραστηριοτήτων και της ροής ελέγχου μεταξύ αυτών
- 4/ Αναγνώριση πιθανών αποφάσεων που αφορούν τη διαδικασία που μοντελοποιείται
- 5/ Εντοπισμός ενδεχομένης παραλληλίας στη διαδικασία
- 6/ Ζωγράφισμα του διαγράμματος

Άλλες συμβουλές:

- προσπαθήστε να ελαχιστοποιήσετε τις **διασταυρώσεις** και βελτιώστε την **αναγνωσιμότητα** του διαγράμματος
- Βεβαιωθείτε για την ορθότητα των «**θαυματουργών**» (miracle) δραστηριοτήτων (αυτών χωρίς είσοδο και των «**μαύρων οπών**» (δραστηριότητες χωρίς έξοδο)



- **Systems Analysis and Design with UML Version 2.0** (2nd edition) by A. Dennis, B. Haley Wixom, D. Tegarden, Wiley, 2005. CHAPTER 6
- **UML Distilled: A Brief Guide to the Standard Object Modeling Language** (3rd Edition) by Martin Fowler, Addison Wesley, 2004. Chap. 11 (2nd edition: [Chapter 9.](#))
- **The Unified Modeling Language User Guide** (2nd edition) by G. Booch, J. Rumbaugh, I. Jacobson, Addison Wesley, 2004, Chap 20