

2009 - 2010



# Inginerie Software pentru Comunicatii (ISC / RST)

Titular curs: **Eduard-Cristian Popovici**

Suport curs: <http://discipline.elcom.pub.ro/isc/>

Moodle: <http://electronica07.curs.ncit.pub.ro/course/category.php?id=4>

## Continut curs

### 1. Introducere in ingineria software

- 1.1. Necesitatea unei abordari sistematice a dezvoltarii software
- 1.2. Abordari si metodologii larg utilizate in ingineria software

### 2. Introducere in limbajul UML

- 2.1. Definirea, rolul si istoricul limbajului de modelare unificat (UML)
- 2.2. Tipuri de diagrame UML. Organizarea ierarhica a diagramelor

### 3. Diagrame UML statice

- 3.1. Diagrame UML de clase
- 3.2. Diagrame UML de obiecte
- 3.3. Diagrame UML de pachete
- 3.4. Diagrame UML de componente
- 3.5. Diagrame UML de structuri compozite
- 3.6. Diagrame UML de *deployment* (amplasare)

## Continut curs

### 4. Diagrame UML dinamice

- 4.1. Diagramele UML de caz de utilizare
- 4.2. Diagrame UML de comunicare si de robustete
- 4.3. Diagrame UML de secventa si de sumar al interactiunilor
- 4.4. Diagrame UML de masini de stari
- 4.5. Diagrame UML de activitati
- 4.6. Diagrame UML de timp

### 5. Introducere in procesul de dezvoltare Rational unificat (RUP)

- 5.1. Organizarea iterativa a proiectelor
- 5.2. Fazele si activitatile procesului RUP

### 6. Introducere in managementul si organizarea proceselor de dezvoltare

### 7. Elemente de reutilizabilitate a software-ului. Pattern-uri de proiectare

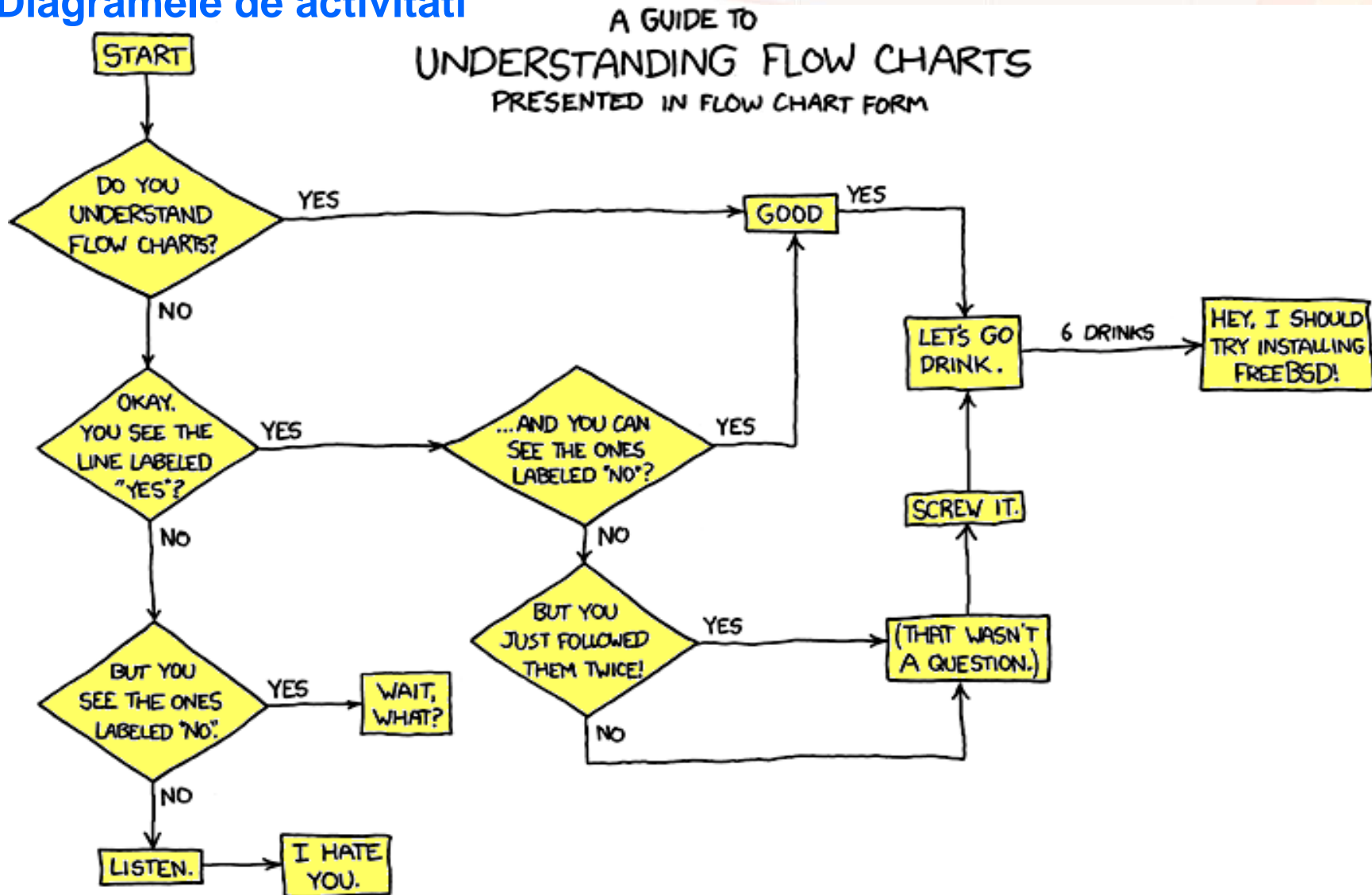
*A picture is worth more than 1024 lines of code*



## 4. Diagrame UML dinamice

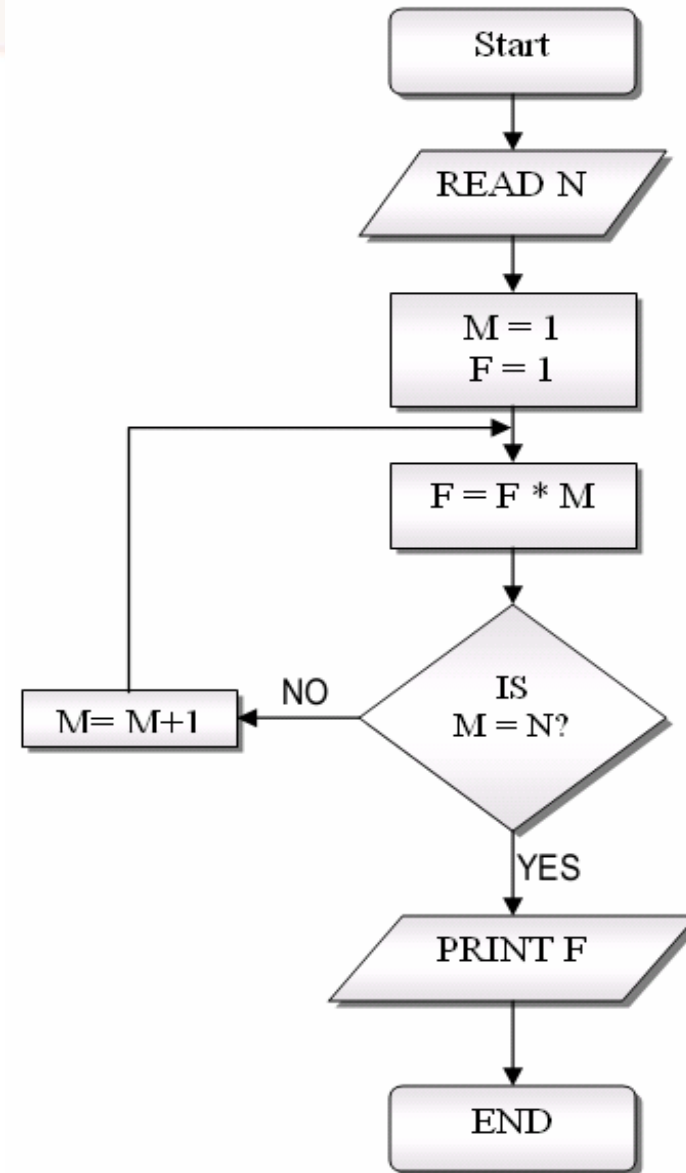
### 4.5. Diagrame UML de activitati

## Diagramele de activitati



### Diagramele de activitati

- cunoscute si ca **flowcharts** (diagrame de flux de control, sau **organigrame**)
- sunt destinate in primul rand **reprezentarii comportamentului intern**
  - al unei **operatii** sau
  - al unui caz de utilizare (**UC**)
- sunt organizate in raport cu **actiunile** si **activitatile**



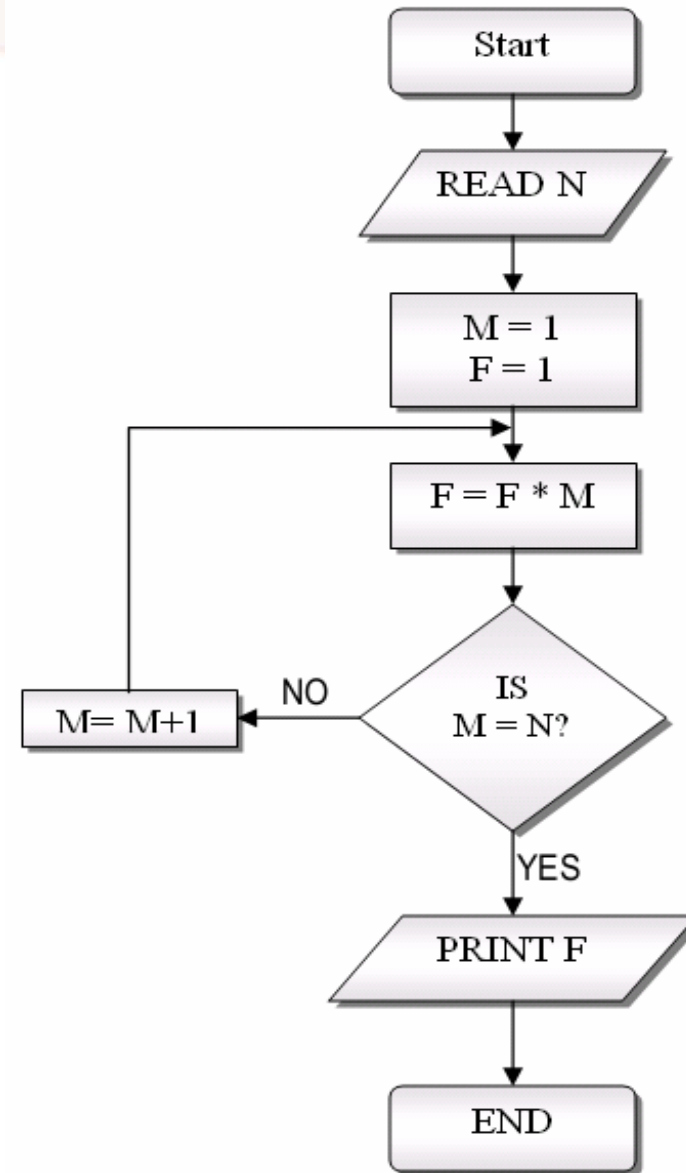
### Diagramele de activitati

➤ sunt **variante** ale diagramelor masinilor de stari (*statecharts*, **FSM**)

➤ reprezinta executia unui mecanism

➤ sub forma unei **derulari de etape** regrupate **secvential**

➤ in **ramificatii paralele** de fire de executie (fluxuri de control)



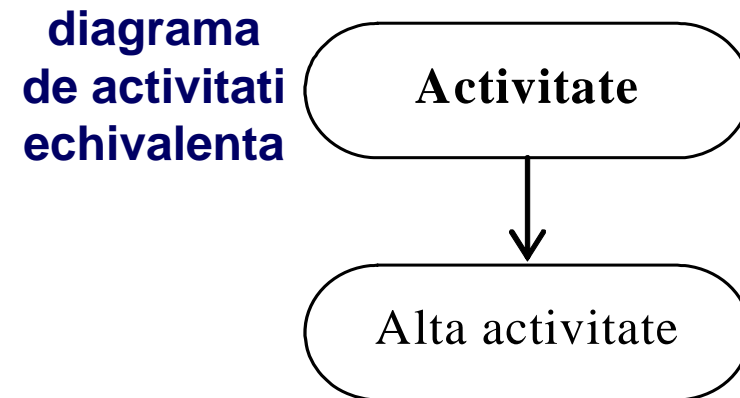
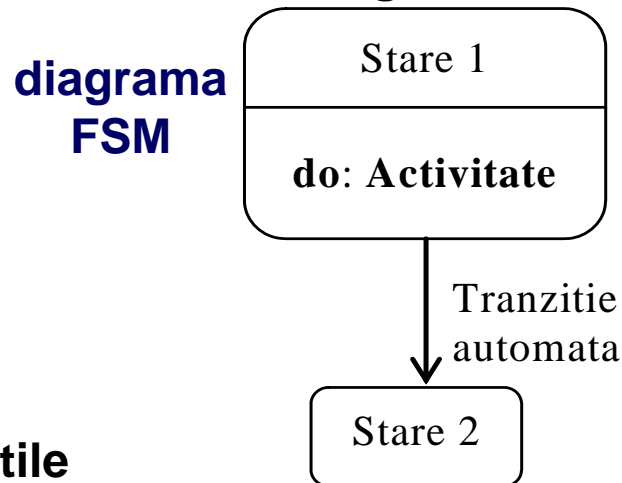


### Activitatea

- reprezinta o **etapa particulara in executia operatiei / UC** care o inglobeaza
- in UML este reprezentata printr-un **dreptunghi rotunjit in lateral** in **forma de semicerc**



- pe cand **starea din diagrama FSM** este un **dreptunghi rotunjit in colturi**



### Activitatile

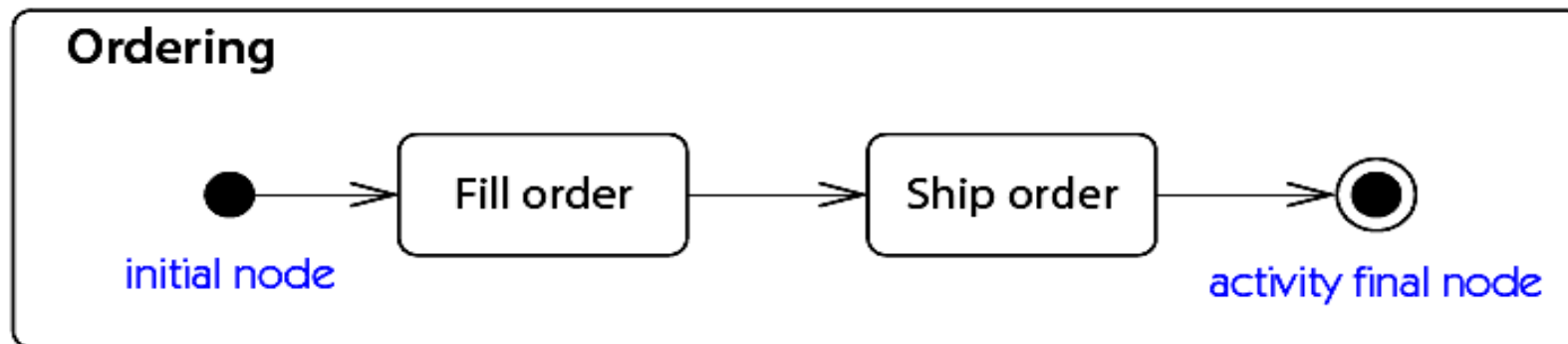
- sunt **legate prin tranzitii automate**, reprezentate prin sageti, ca si tranzitiile din diagramele FSM



### Activitatile

**Notatii speciale** sunt folosite pentru

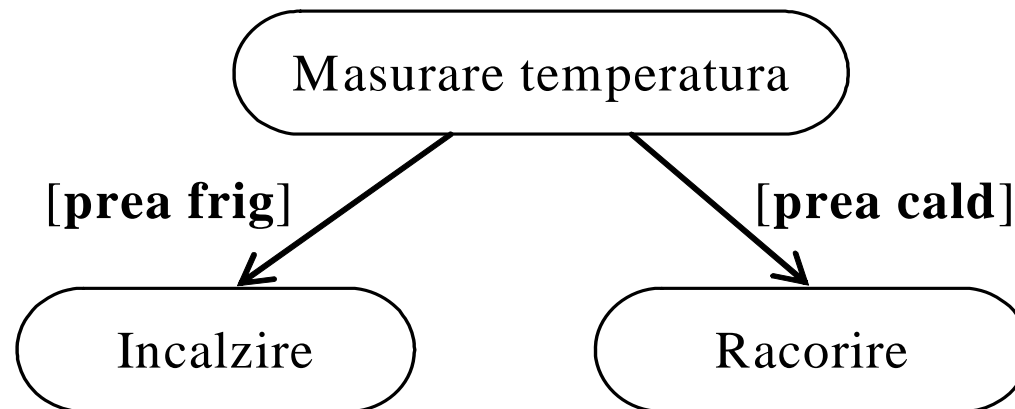
- **momentul initial (start)** al intregii diagrame de activitati sau al unei activitati composite
  - reprezentat printr-un **punct mare negru**
- **momentul final (stop)** al intregii diagrame de activitati sau al unei activitati composite
  - reprezentat printr-un **cerc care contine un mic punct negru**



### Diagramele de activitati

#### Tranzitiile intre activitati

- pot fi **decorate cu conditii logice** (*guard expressions*)
  - **mutual (reciproc) exclusive**
- plasate **in apropierea tranzitiilor** carora le **valideaza declansarea**

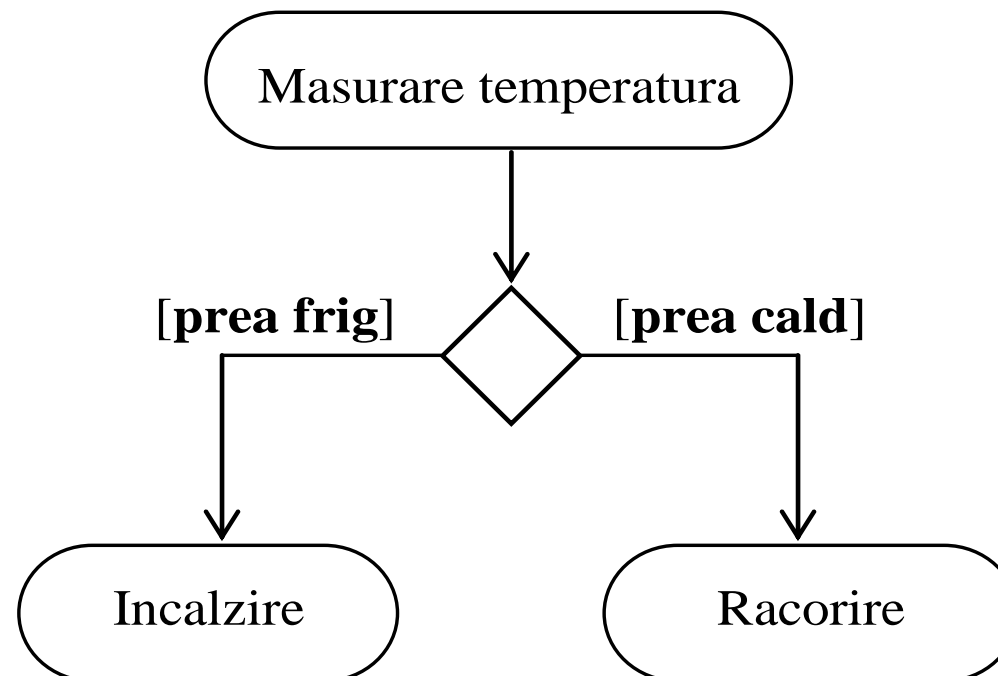


### Diagramele de activitati

#### Conditia logica (*guard expression*)

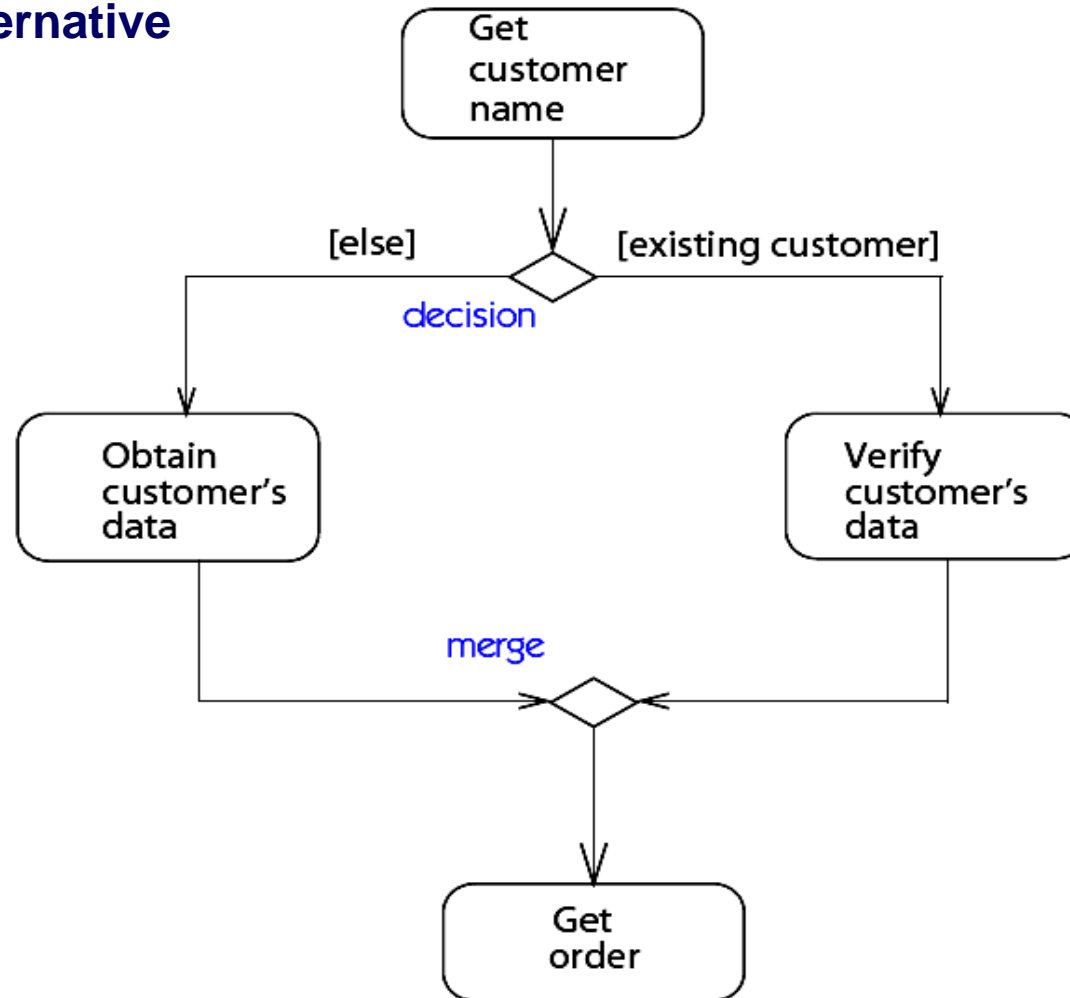
- poate fi **reprezentata explicit** printr-un **romb** (*branch / decision*) din care pleaca mai multe tranzitii

Diagrama anterioara redesenata, cu **conditia logica reprezentata explicit**



### Diagramele de activitati

Tot printr-un romb este reprezentata si **reunirea (merge)** a mai multe ramificatii alternative

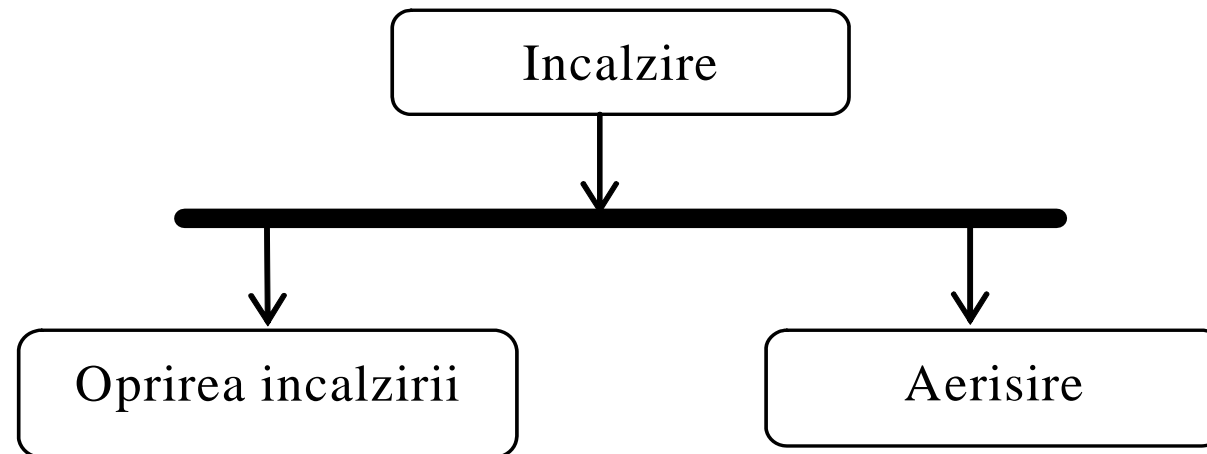


### Diagramele de activitati

#### Barele de sincronizare

- reprezinta **sincronizarile** intre fluxuri de control
- permit **deschiderea** si **inchiderea ramificatiilor paralele**
  - in interiorul unui fir de executie a unei operatii/UC

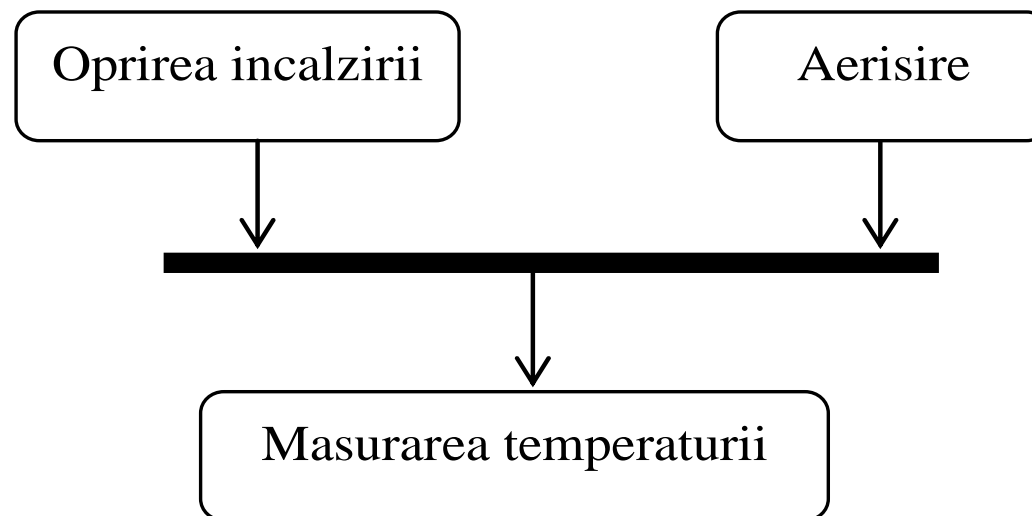
**Tranzitiile care pleaca dintr-o bara (*fork*) sunt declansate simultan**



### Diagramele de activitati

O bara de sincronizare poate fi depasita doar

- atunci **cand toate tranzitiile care intra in bara (*join*)** de sincronizare au fost declansate

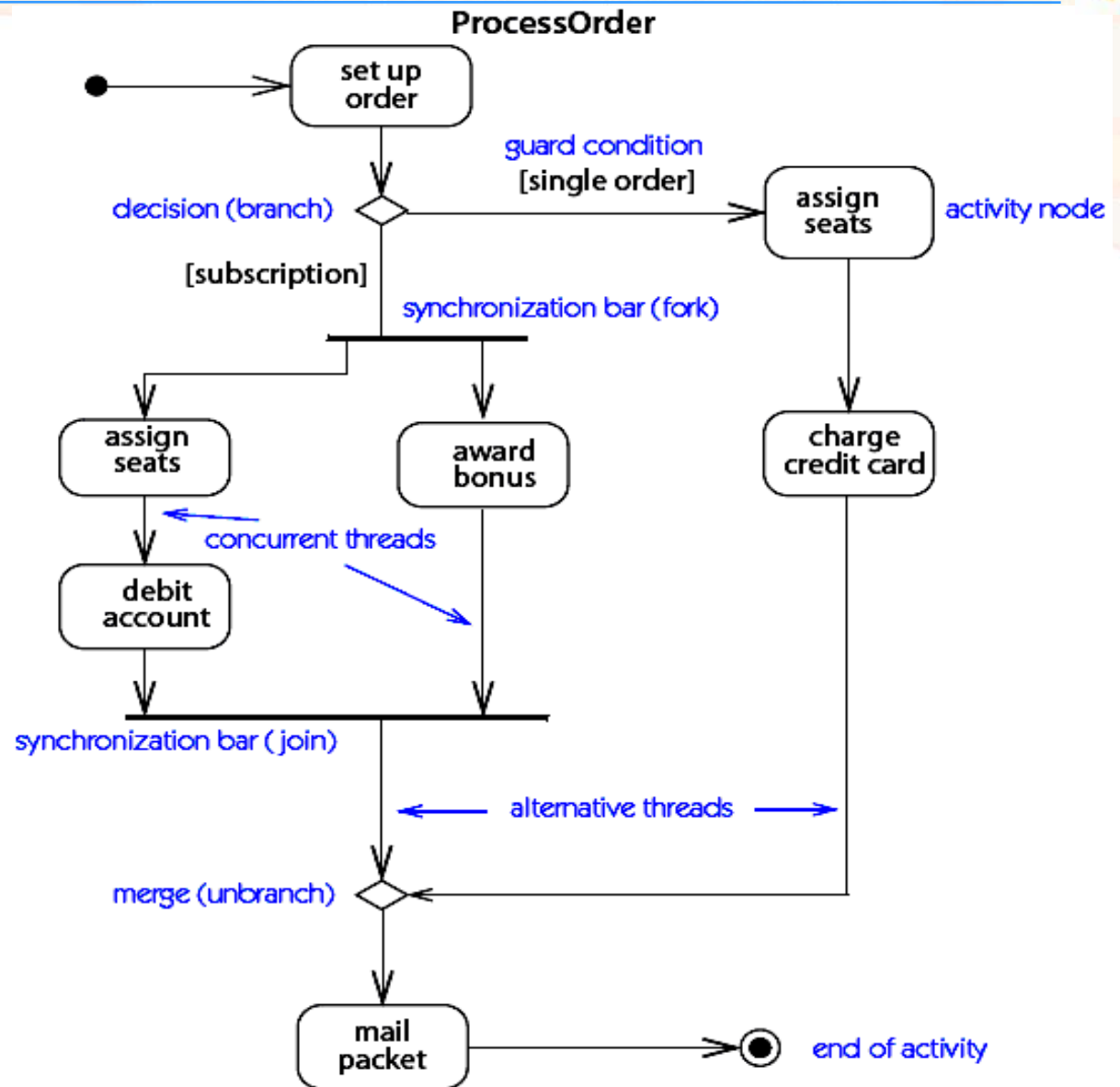


## 4.5. Diagrame UML de activitati



### Diagramele de activitati

Reprezentarea firelor de executie (*threads*) concurente



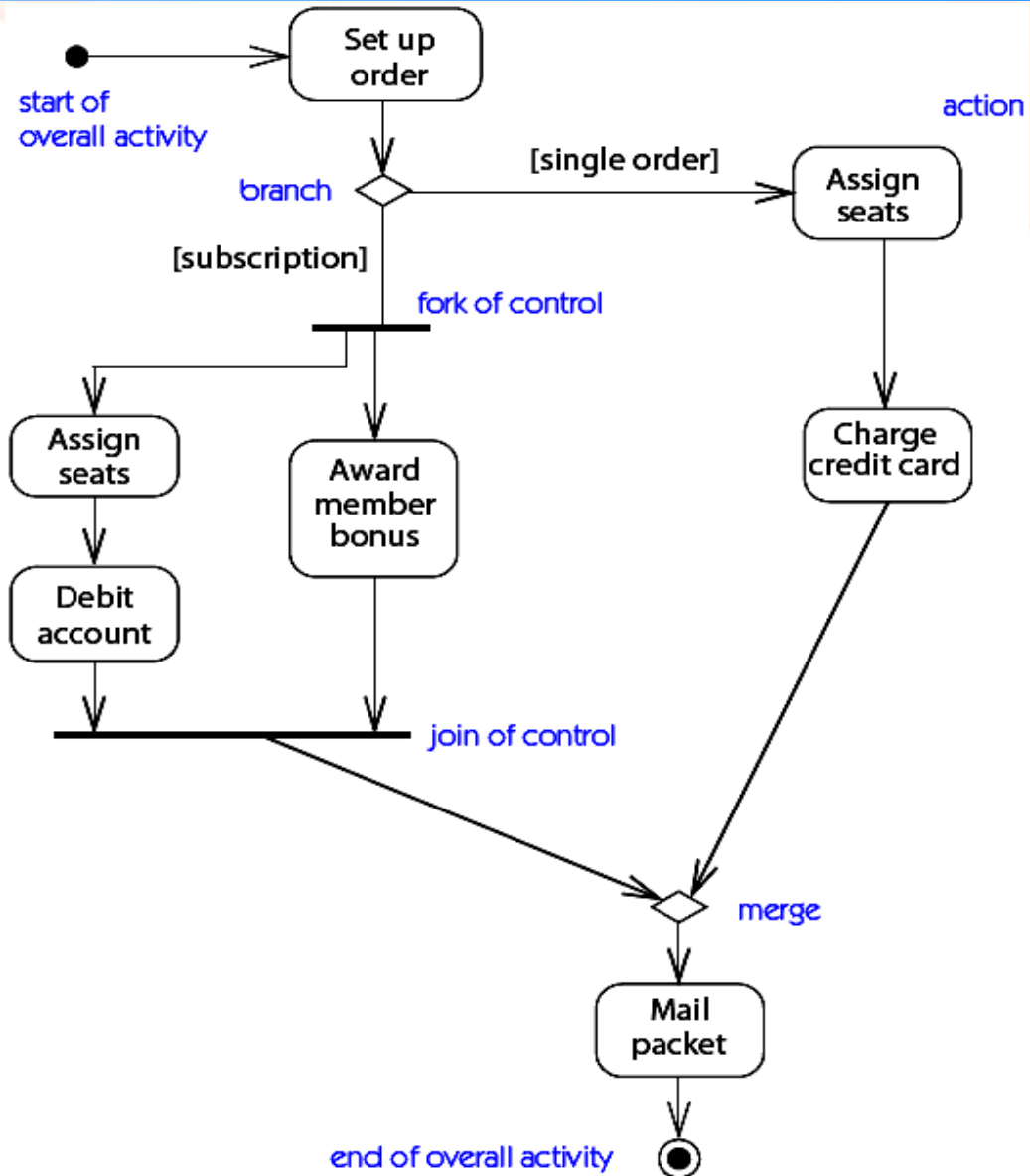


# 4.5. Diagrame UML de activitati



## Diagramele de activitati

Reprezentarea firelor de executie (*threads*) concurente

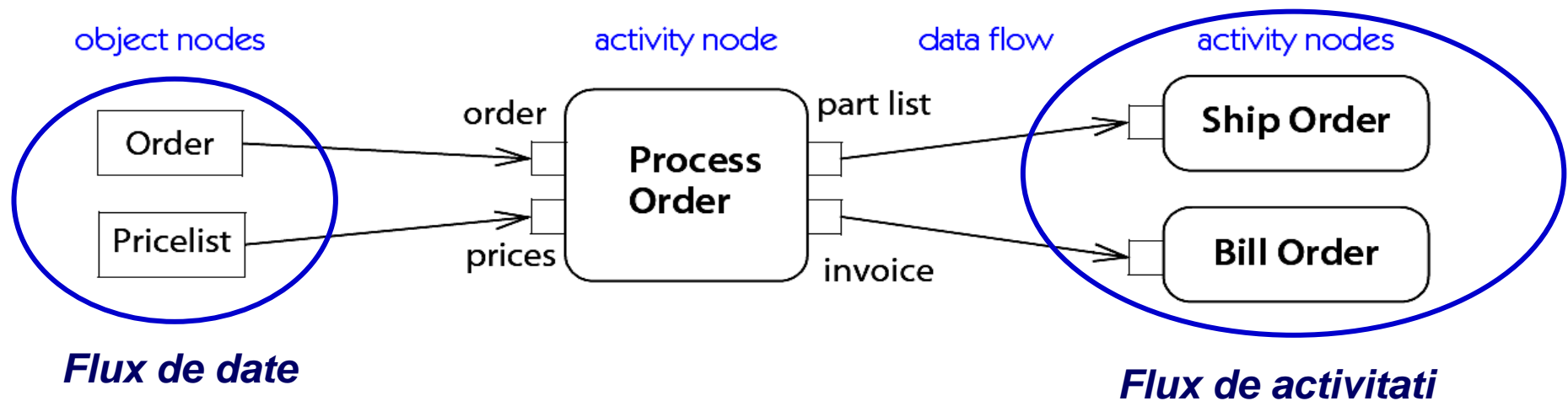


### Diagramele de activitati

Unei activitati ii pot fi precizate

- valorile de intrare (*input pins*) - parametri
- valorile de iesire normale (*output pins*) – valori returnate
- si valorile de iesire **exceptionale**

Exemplu

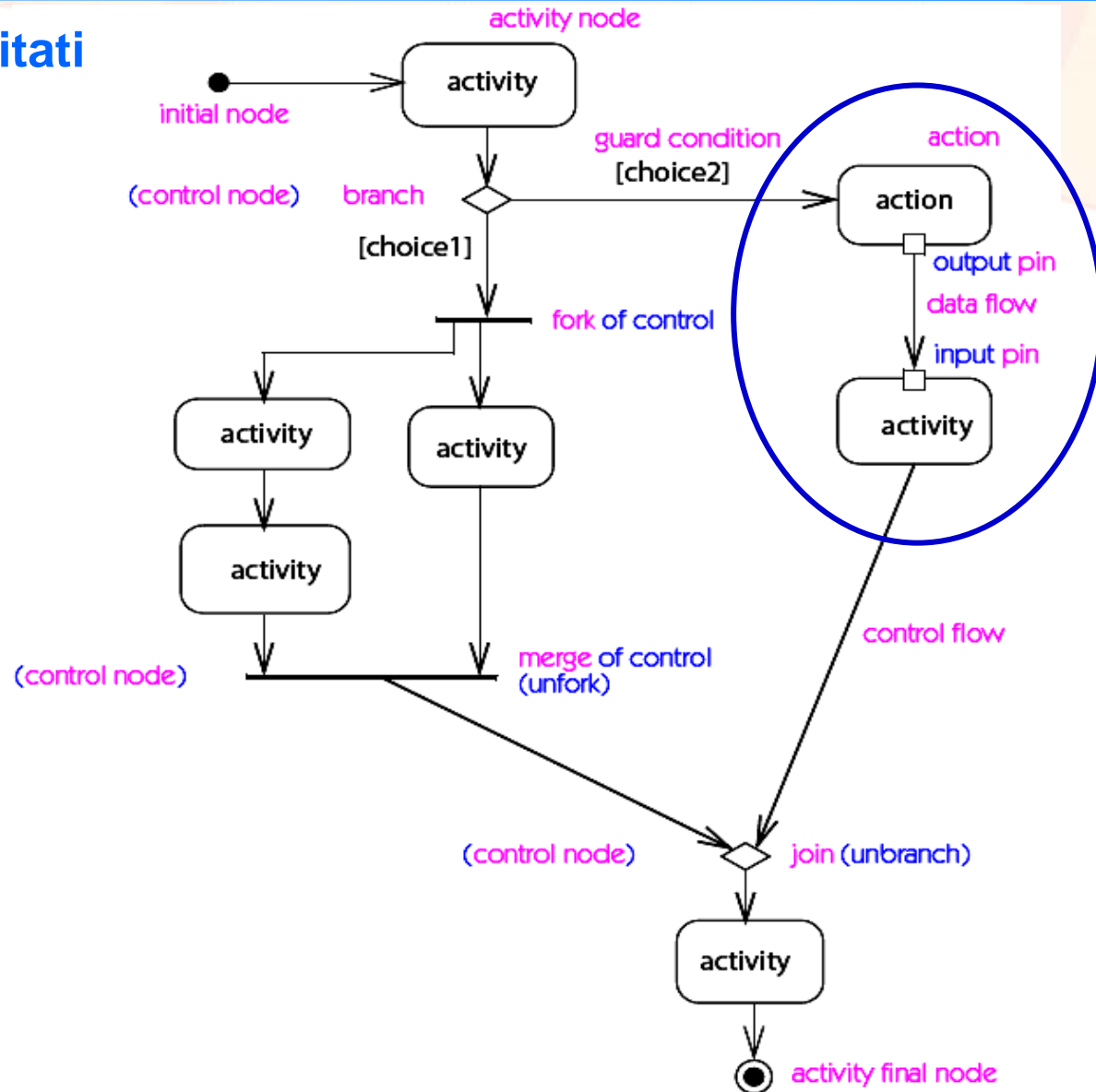


# 4.5. Diagrame UML de activitati



## Diagramele de activitati

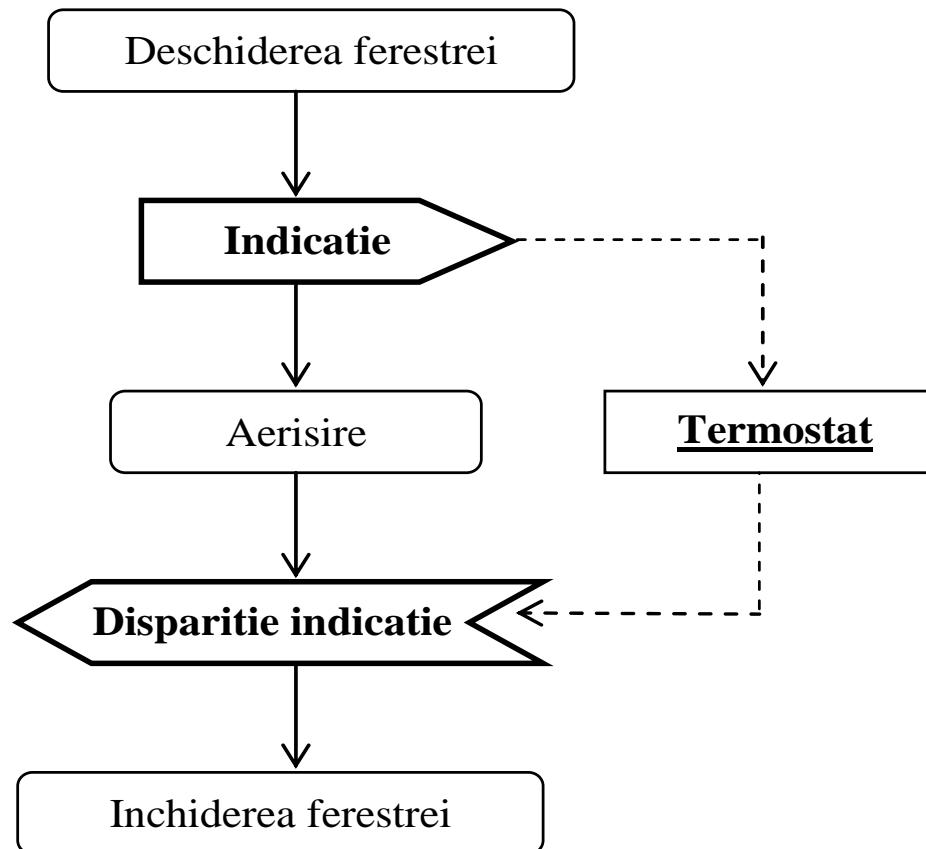
Reprezentarea parametrilor si a valorilor returnate



### Diagramele de activitati

UML defineste **stereotipuri** pentru

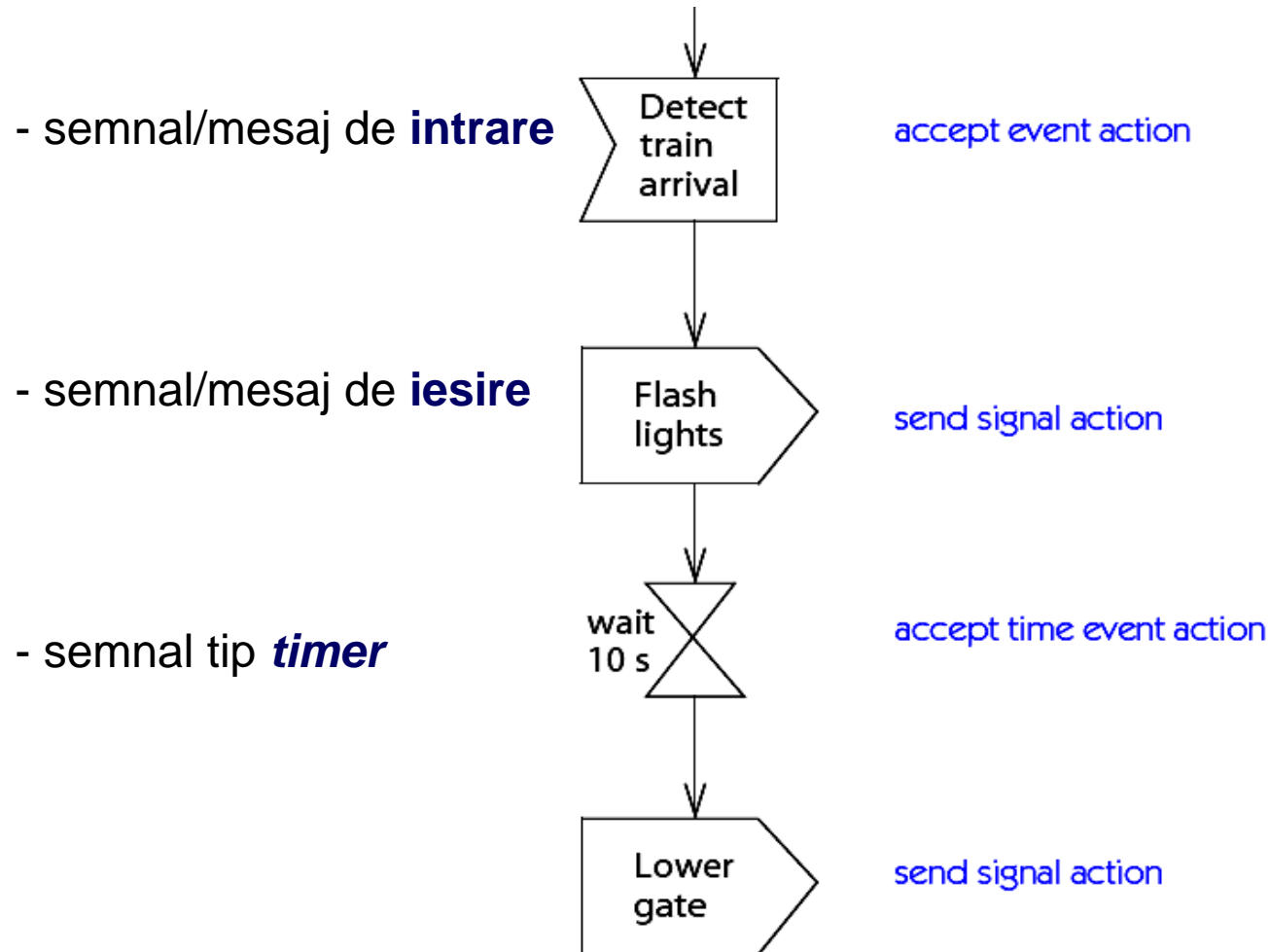
- **trimiterea unui semnal** - simbolizata printr-un **pentagon convex**
- **primirea unui semnal** - simbolizata printr-un **pentagon concav**



**Fluxurile de date (obiecte)** sunt reprezentate prin **sageti punctate**

### Diagramele de activitati

#### Notatiile speciale pentru activitatile de comunicatie

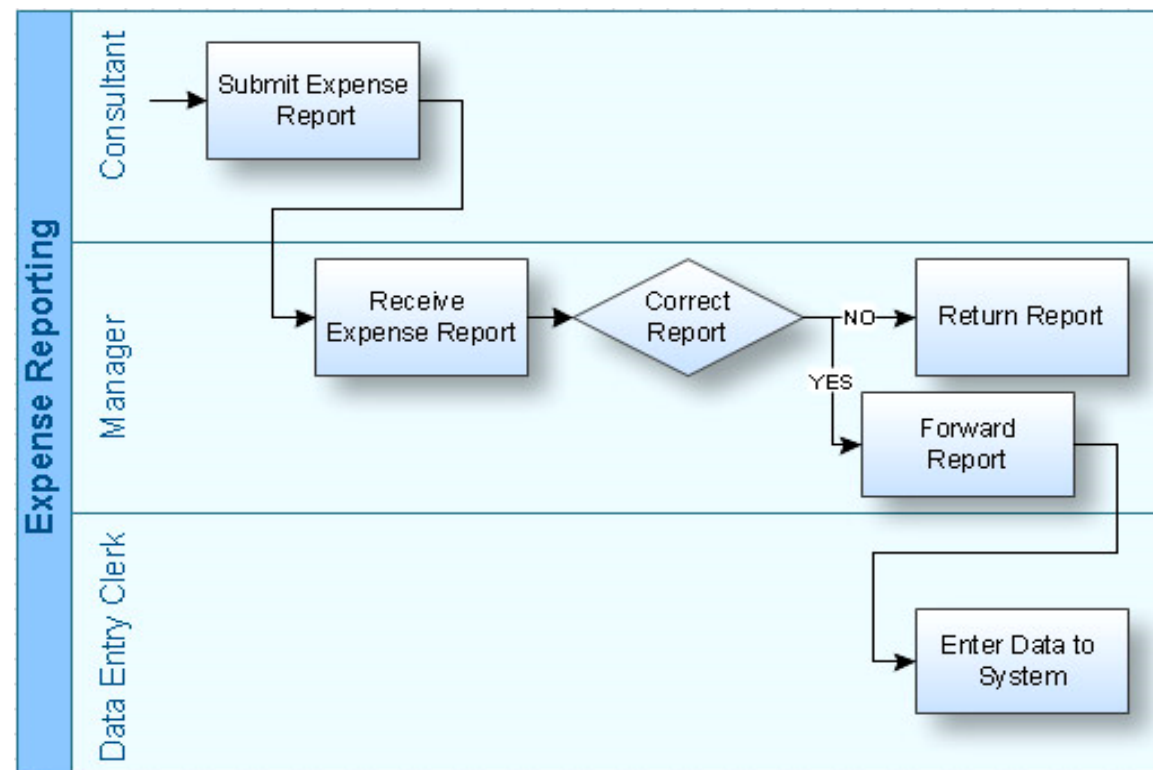


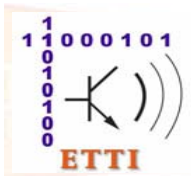
### Diagramele de activitati

Diagramele de activitati

- pot fi **decupate** in **partitii** (**swimlanes**, culoare / coridoare de activitati)
- asa cum o piscina este separate in culoare de natatie
- **pentru a arata diferitele responsabilitati** in cadrul unui mecanism

Ce este incorect din punct de vedere al notatiilor UML in aceasta diagrama de activitati?

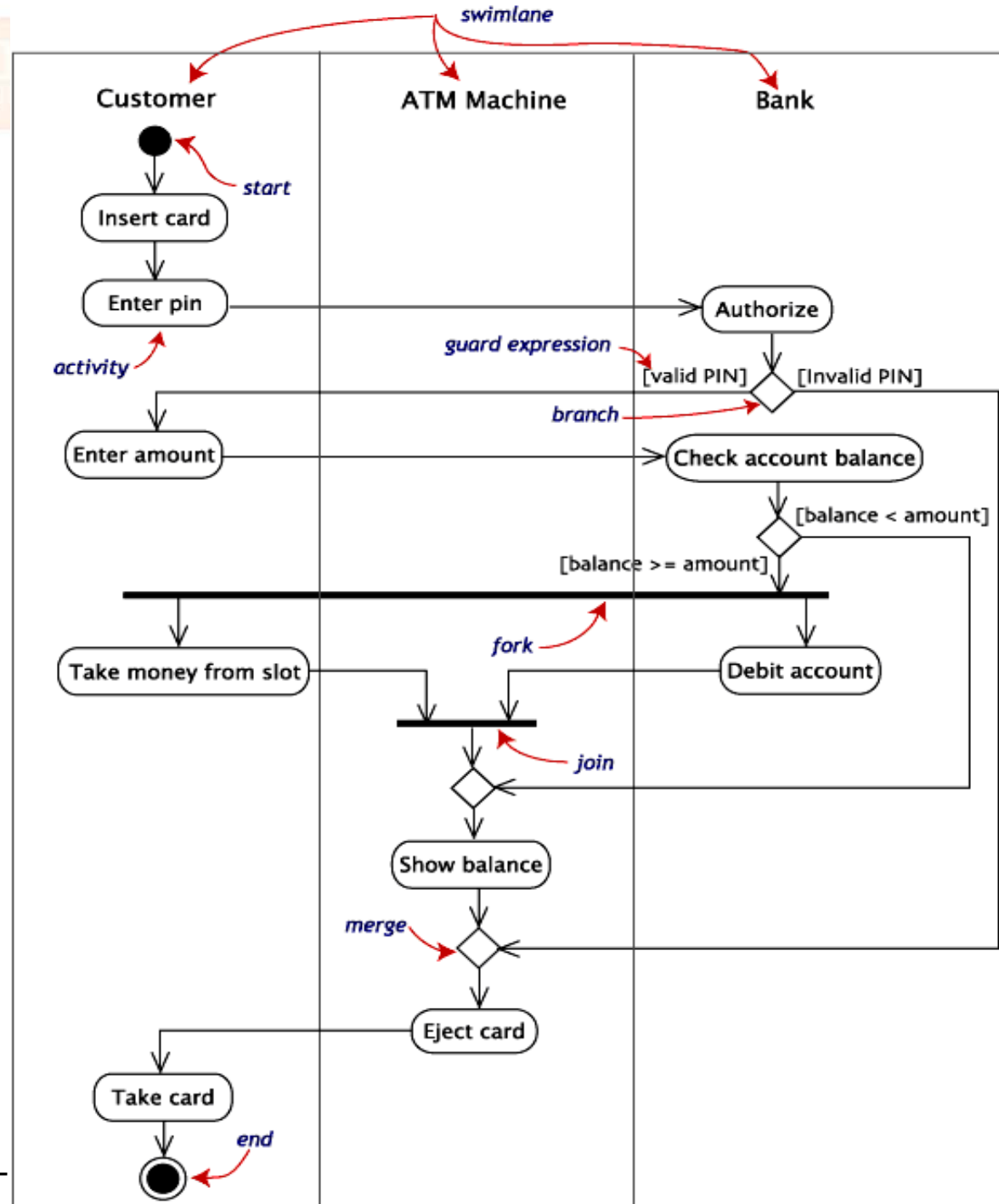




## Diagramele de activitati

Pozitia relativa a partiilor nu are nici o semnificatie

Tranzitiile sunt libere sa traverseze partiile la care nu se refera



Procesul modelat:

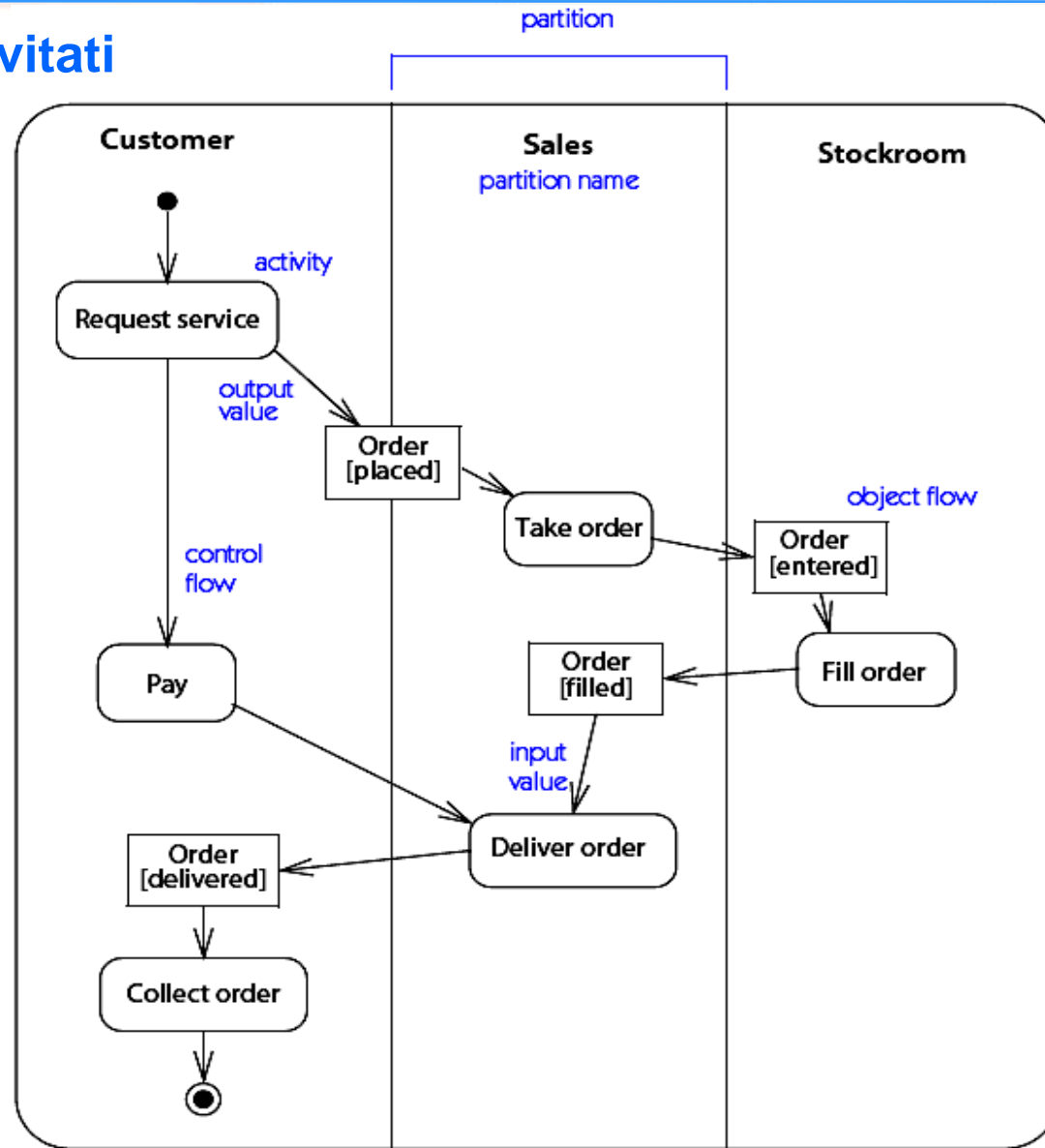
**Withdraw money from a bank account through an ATM**





## Diagramele de activitati

Partitii si fluxuri de obiecte



### Ilustrarea *Business UC* cu diagrame de activitati

Exemplu de diagrama de activitati care ilustreaza un *business UC*

#### B3: NonMember Reserves CarModel

1. The **NonMember** tells the **Assistant** which **CarModel** to reserve
2. The **Assistant** asks for a deposit and **License**
3. **While** the **NonMember** is looking for the deposit and **License**, the **Assistant** looks for the **CarModel** on the **Auk** system
4. Once everything has been found, the **Assistant** checks the deposit and the **License**
5. **If** the deposit and **License** are valid, the **Assistant** makes the **Reservation** and the activity is finished
6. **Otherwise**, the activity is finished

