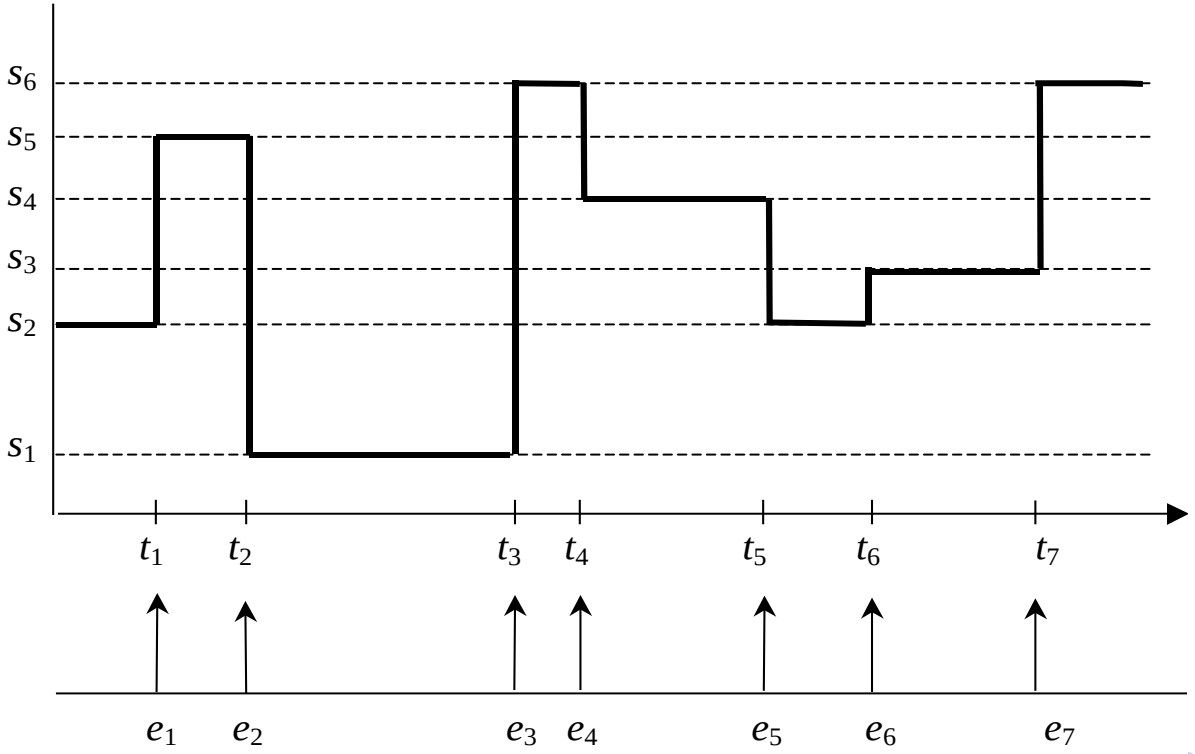
CURS 3

Modelare cu Retele Petri

Sisteme cu Evenimente Discrete

- ***Un Sistem cu Evenimente Discrete (SED) este un sistem cu stari discrete, care evolueaza prin evenimente, adica evolutia sa depinde in intregime de aparitia asincrona a u***



Obiective ale studiului sistemelor (1)

- ***Modelare si analiza***: Se dezvolta un model pentru a verifica daca poate reproduce comportamentul pertinent al sistemului fizic corespunzator. Daca modelul este corect, atunci el permite analiza functionarii sistemului fizic in diferite conditii de lucru.
- ***Proiectare si sinteza***: Se utilizeaza tehnici de modelare deja verificate pentru construirea de sisteme care sa aiba un anumit “comportament dorit”. Pentru acesta se combina diverse componente si se dau valori anumitor parametri.
- ***Conducere***: Scopul activitatii de conducere este de a determina functiile de intrare care vor asigura un comportament dorit al sistemului condus in contextul unor conditii de operare variate si uneori posibil adverse. Este necesar un model al sistemului condus, precum si existenta unor tehnici care sa permita testarea a diferite metode de conducere.

Obiective ale studiului sistemelor (2)

- ***Evaluarea performantelor***: Se poate ca mai multe tehnici de conducere sa permita obtinerea comportamentului dorit al sistemului condus. Trebuiesc stabilite criterii si tehnici de comparatie ale acestor tehnici, in vederea selectarii celei mai eficiente.
- ***Optimizare:*** Determinarea performantelor optime ale sistemului condus - necesita tehnici specializate si de regula euristice.

Exemple de SED

- Retele de calculatoare
- Sisteme de comunicatii
- Sisteme de Fabricatie
- Sisteme de Trafic
- Baze de date complexe
- Sisteme software
- …
- orice sistem complex, condus prin calculator

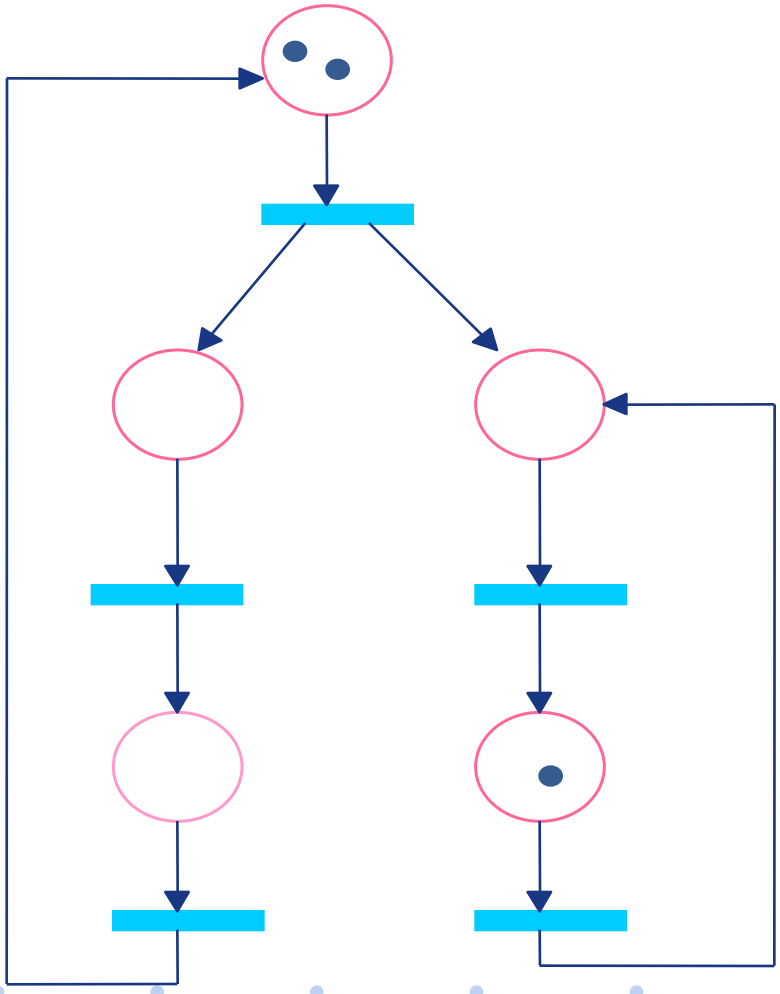
Modelare SED

- ***Concluzie***
  - Evolutia unui sistem poate fi reprezentata in termeni de stari/ conditii si evenimente

- ***Formalisme disponibile:***
  - Automate (stari/ evenimente)
  - Retele Petri (variabile de stare/ evenimente)
  - Lanturi Markov, cozi de asteptare

Retele Petri notiuni de baza

- O retea Petri (PN) e un graf **bipartit**, compus din:
  - doua tipuri de noduri
    - **Pi (1 ≤ i ≤ 5) – pozitii**
    - **Tj (1 ≤ j ≤ 5) – tranzitii**
  - arce orientate (unesc doua tipuri **diferite** de noduri).
- Fiecare pozitie poate contine un numar intreg de **jetoane sau marcaje**.
- Numarul de jetoane dintr-o pozitie s.n. **marcaj**.
- Marcajul lui Pi: m(Pi) or mi



Comentarii

- Un nod i poate fi, pentru un nod j, :
  - intrare: daca exista un arc directionat de la nodul i la nodul j ($i \rightarrow j$)
  - iesire: daca exista un arc directionat de la nodul j la nodul i ($j \rightarrow i$)
- O tranzitie fara intrari = sursa
- O tranzitie fara iesiri = capcana
- Marcajul retelei M este definit ca un vector al marcajelor pozitiilor

  $M = [m(P_1)\ \ m(P_2)\ \ m(P_3)\ \ m(P_4)\ \ m(P_5)]^T$

  si defineste starea retelei la un moment dat.

Semnificatii pozitii

- Mediu de comunicare (linie telefonica, intermediar, retea de comunicatie)
- Buffer (cutie postala, stoc, depozit)
- Locatie geografica (locatie intr-un depozit, oficiu sau spital)
- Stare a unei componente sau valoarea unei conditii (etajul la care se afla un lift, conditia ca un specialist sa fie disponibil).

Semnificatii tranzitii

- Eveniment (inceputul sau sfarsitul unei operatii, decesul unui pacient, schimbarea culorii unui semafor)
- Transformarea unui obiect (modificarea/adaptarea unui produs, reactualizarea unei baze de date, reactualizarea unui document)
- Transportul unui obiect (transportul bunurilor, expedierea unui mesaj sau a unui fisier)

Semnificatii jetoane

- Obiect fizic (produs, componenta, medicament, persoana)
- Obiect informational (mesaj, semnal, raport, fisier)
- Colectie de obiecte (camion cu bunuri, depozit cu componente, fisier cu adrese)
- Indicator de stare (indicator al starii unui proces, starea unui obiect)
- Indicator al indeplinirii unei conditii.

Marcaje, stari si evolutii

- Marcajul retelei M este definit ca un vector al marcajelor pozitiilor retelei si reprezinta starea modelului la un moment dat.
- Evolutia sistemului corespunde evolutiei marcajelor.
- Evolutia se face prin executia tranzitiilor.

Executia unei tranzitii

O tranzitie se numenste valida (executabila) daca toate pozitiile sale de intrare contin cate cel putin un jeton de marcaj.

Nota: O tranzitie sursa este totdeauna valida.

Executia unei tranzitii:

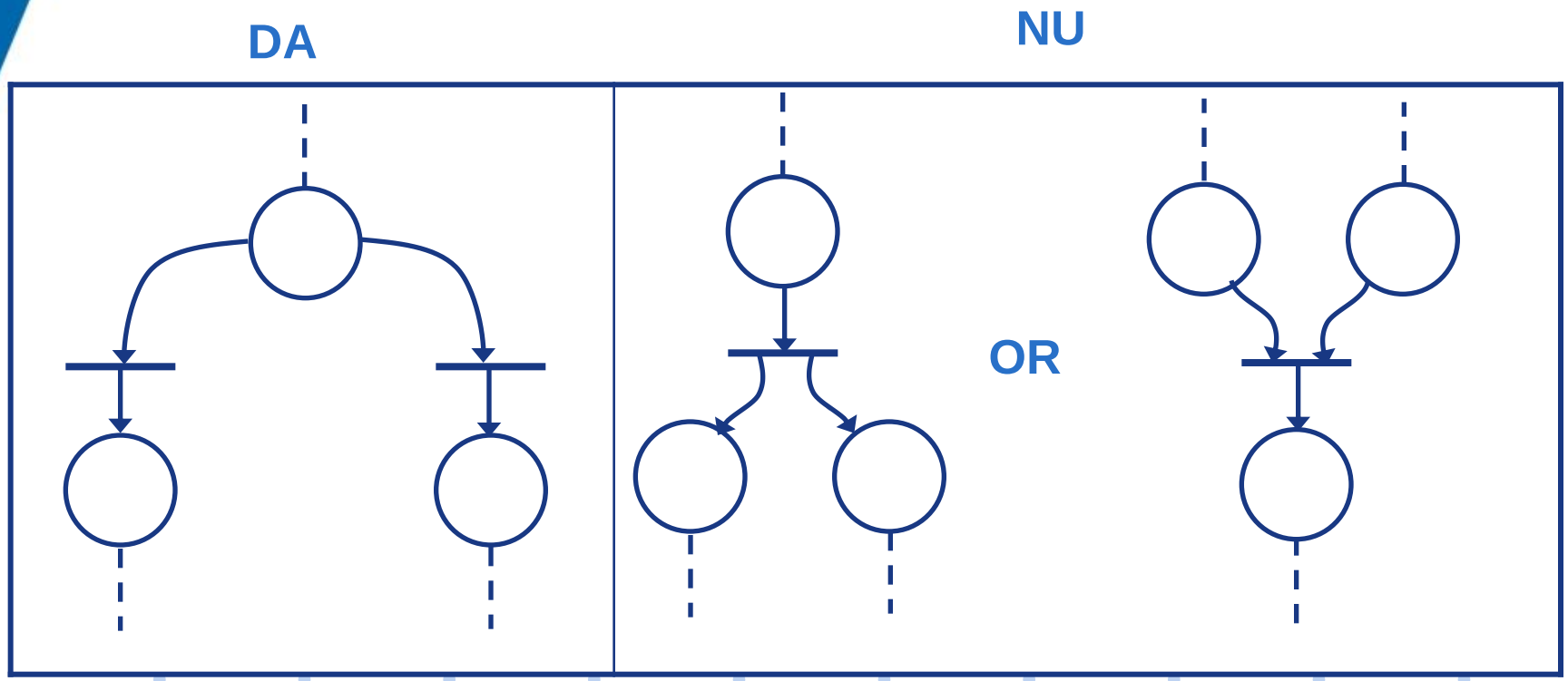
- se retrage (consuma) cate un jeton din fiecare pozitie de intrare
- se depune (produce) cate un jeton in fiecare pozitie de iesire

Ipoteze:

- executia unei tranzitii are durata nula si este indivizibila
- intr-un sistem SED poate sa aiba loc numai un eveniment la un moment dat

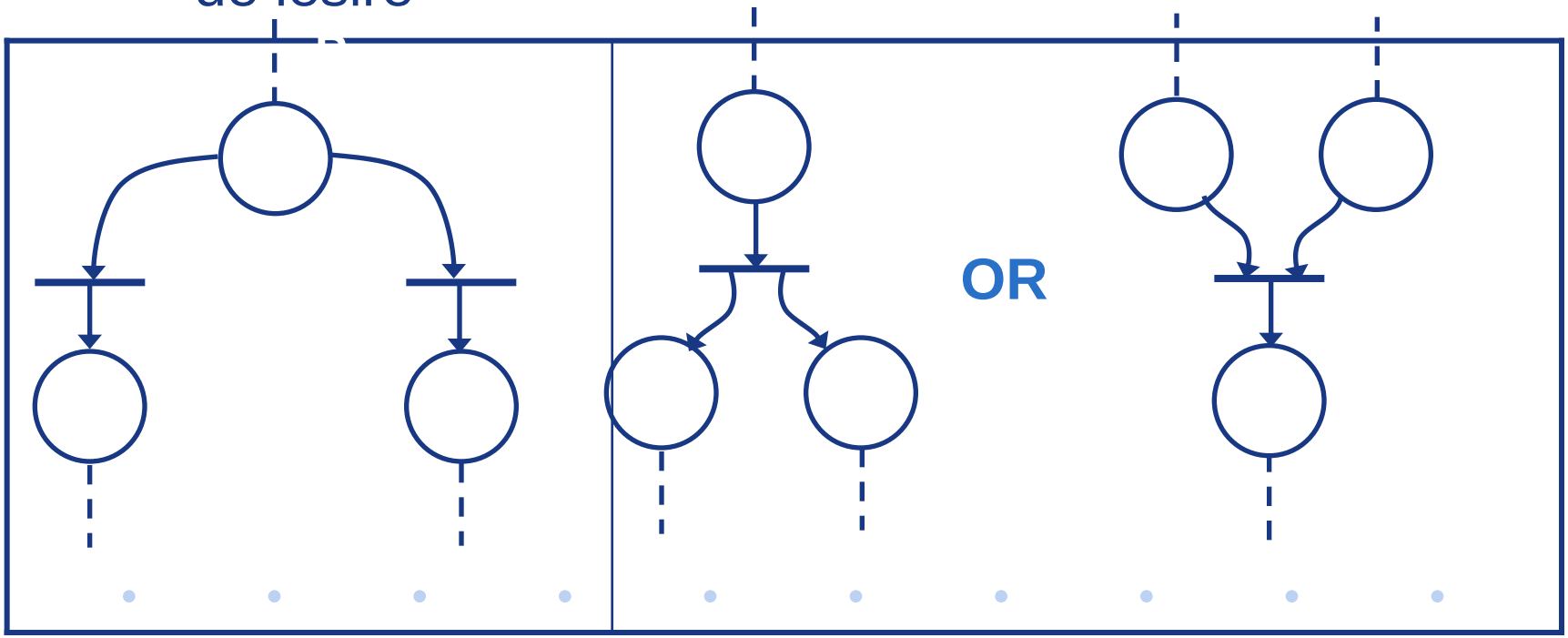
Structuri particulare (1)

- **Graf de stari:** fiecare tranzitie are exact o pozitie de intrare si o pozitie de iesire



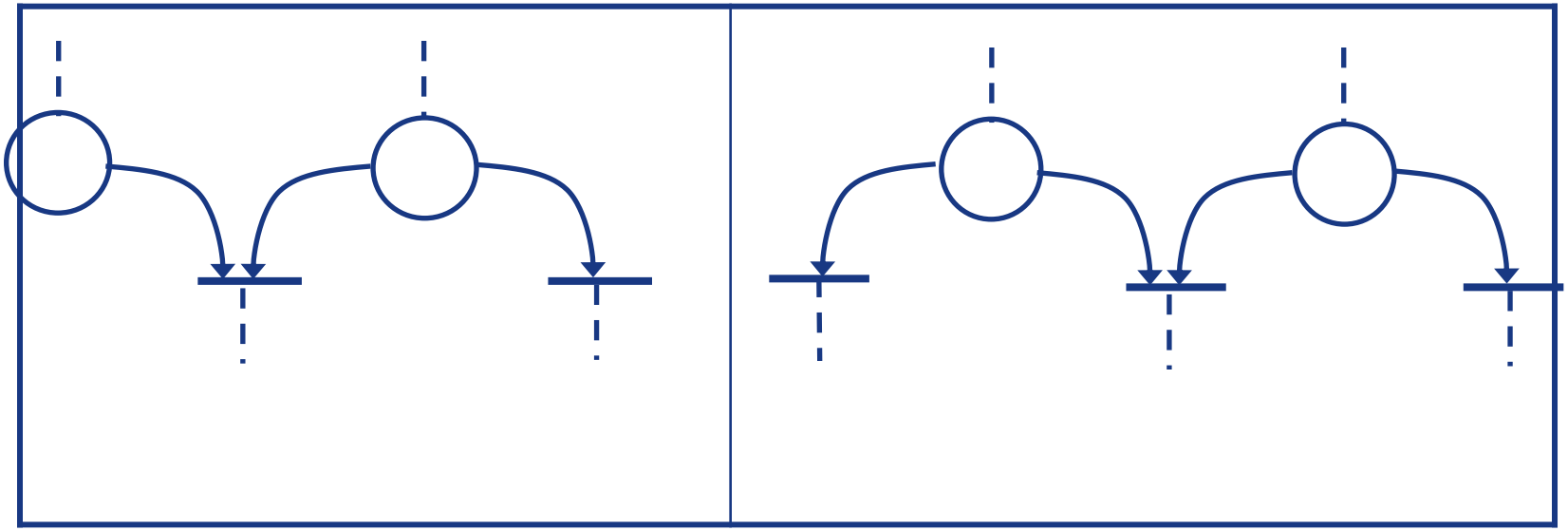
Structuri particulare (2)

- **Conflict structural**: o pozitie cu cel putin doua tranzitii de iesire: $K=< P_1, \{T_1, T_2\}>$
- **Fara conflicte**: fiecare pozitie are max. o tranzitie de iesire



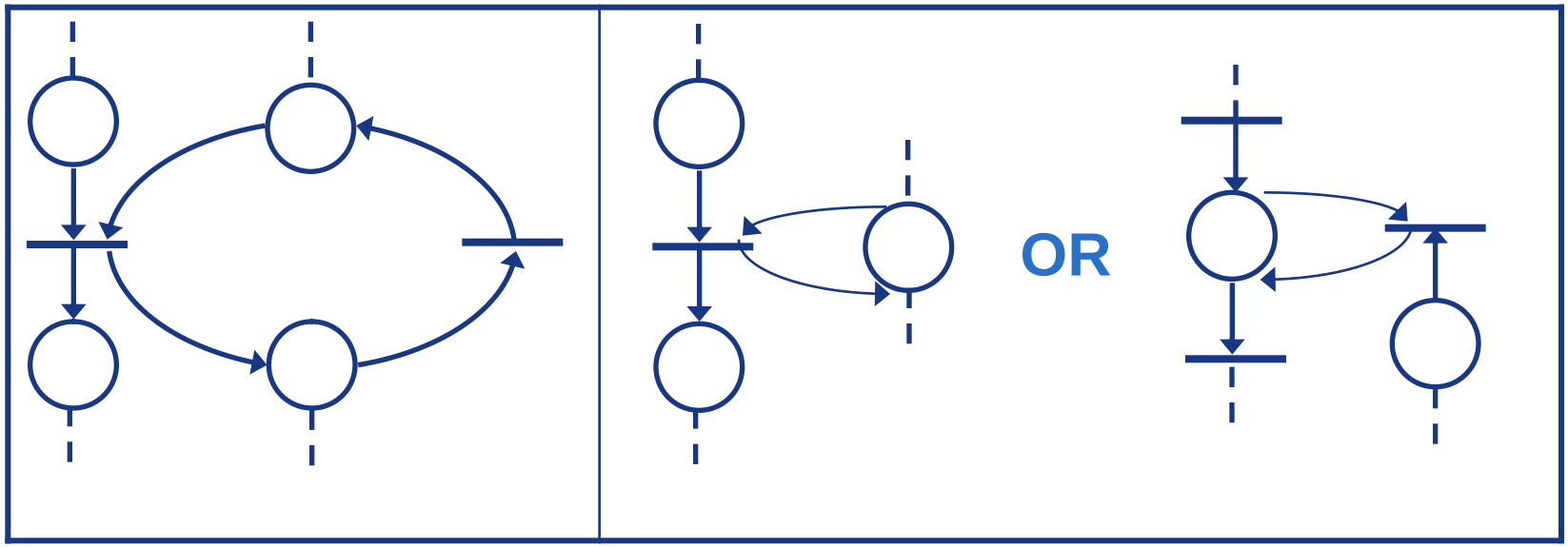
Structuri particulare (3)

- **RP simple:** fiecare tranzitie e implicata in cel mult un conflict



Structuri particulare (4)

- **RP pura:** o RP fara auto-buclare (***auto-buclare*** – o pereche pozitie P_i + tranzitie T_j - in care P_i este si intrare si iesire pentru T_j)

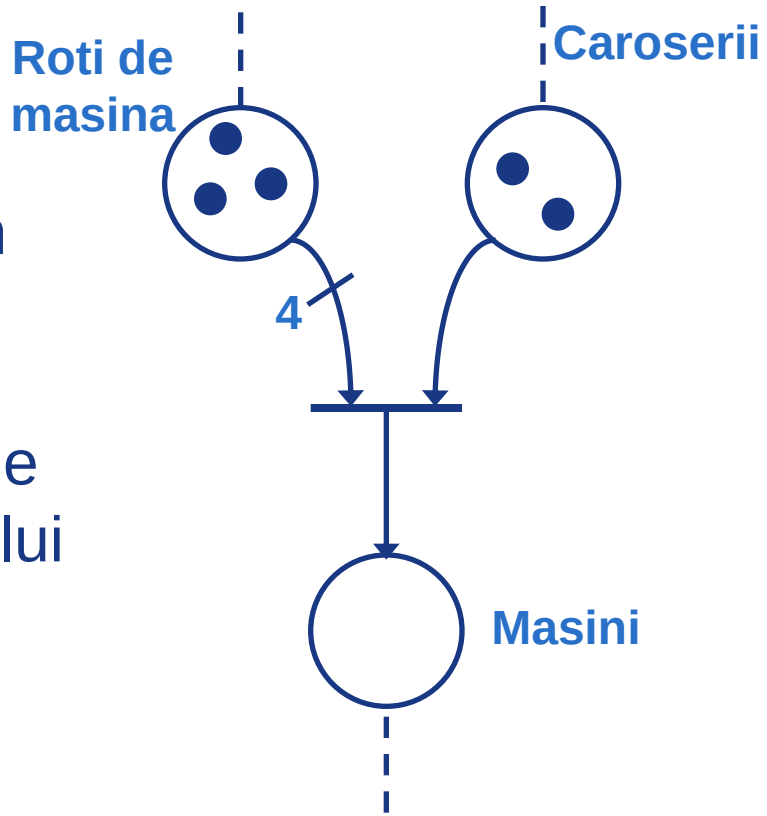


Abrevieri si Extensii

- **Abrevieri:** reprezentari simplificate pentru care exista intotdeauna o RP ordinara echivalenta
- **Extensii:** modele cu reguli functionare suplimentare si cu putere de modelare superioara
- ***Nota:* Toate proprietatile unei RP se mentin pentru abrevieri; acest lucru NU este valabil in mod obligatoriu pentru extensii.**

RP generalizate

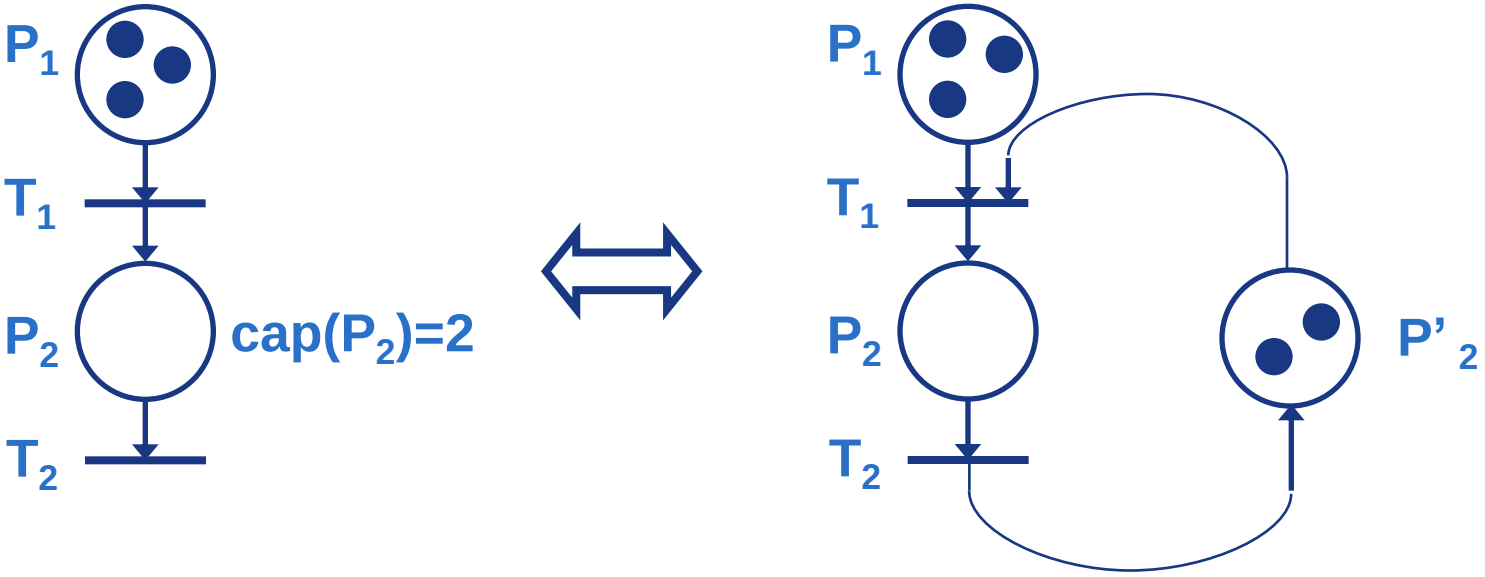
- **RP generalizate** = arcurile au asociate ponderi (intregi); pe un arc circula numai pachete de jetoane de marcaj al caror numar e egal cu ponderea arcului



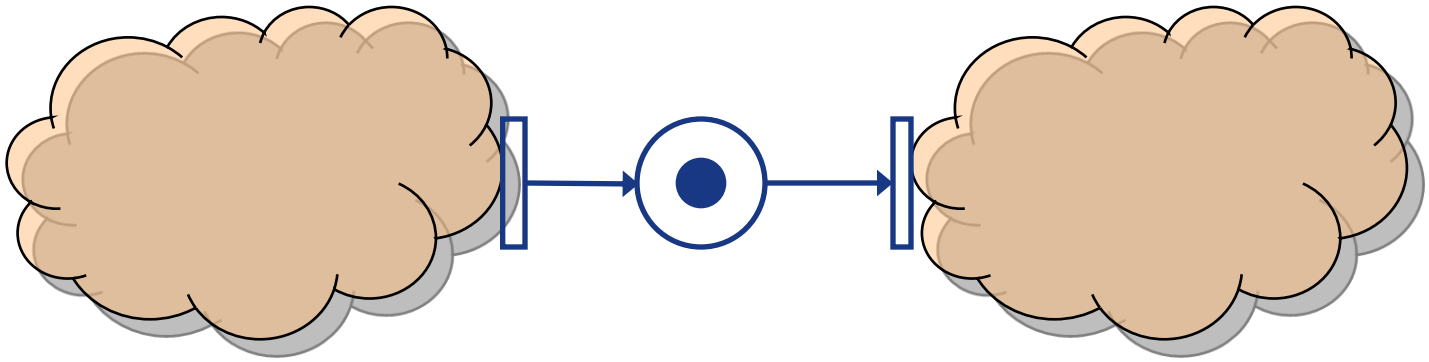
Exemplu de asamblare

RP cu capacitati

- **RP cu capacitati** = pozitiile au asociate capacitati fixe (intregi pozitivi); o tranzitie NU poate fi executata daca prin aceasta se depaseste capacitatea vreunei pozitii

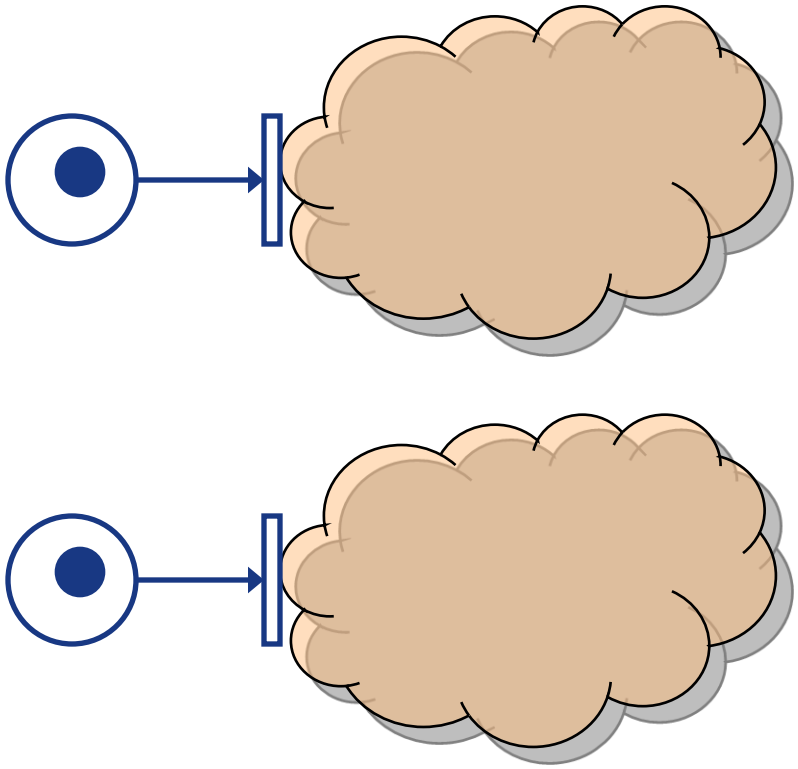


Modelarea anumitor concepte (1)



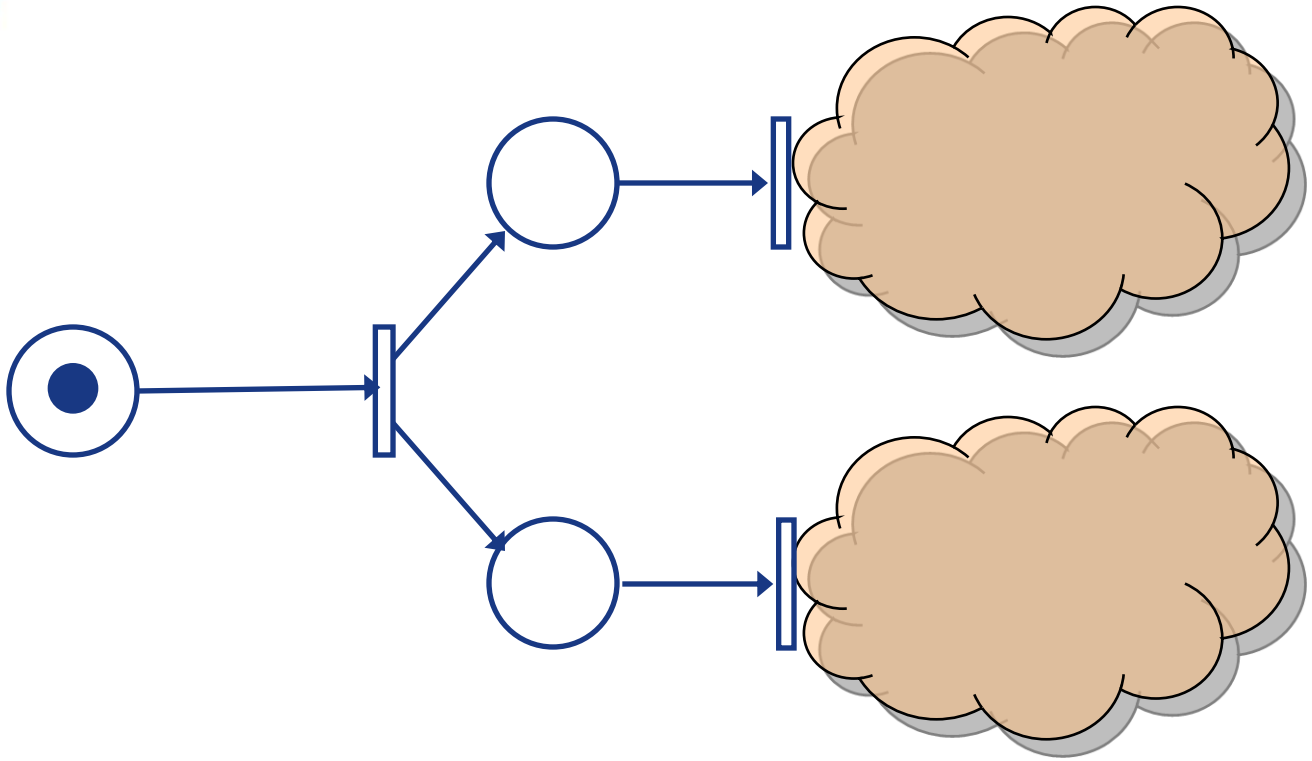
Cauzalitate

Modelarea anumitor concepte (2)



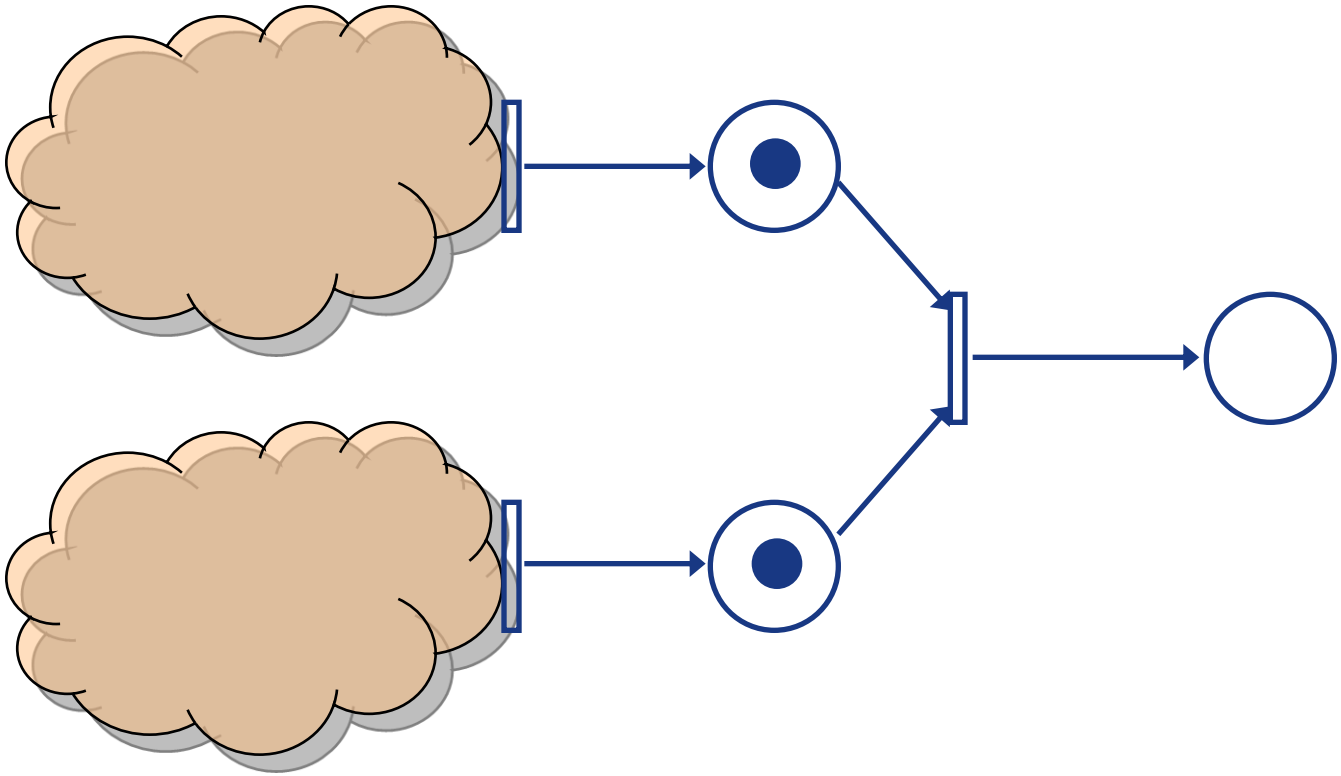
Paralelism

Modelarea anumitor concepte (3)



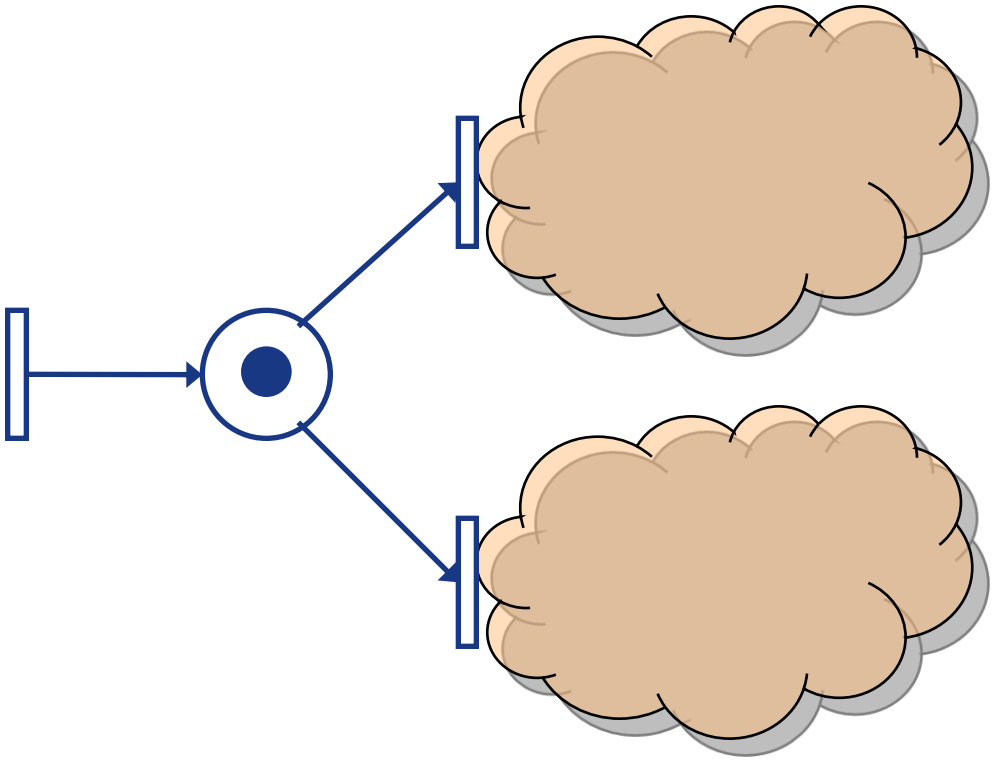
Paralelism AND-split (sincronizarea inceputului)

Modelarea anumitor concepte (4)



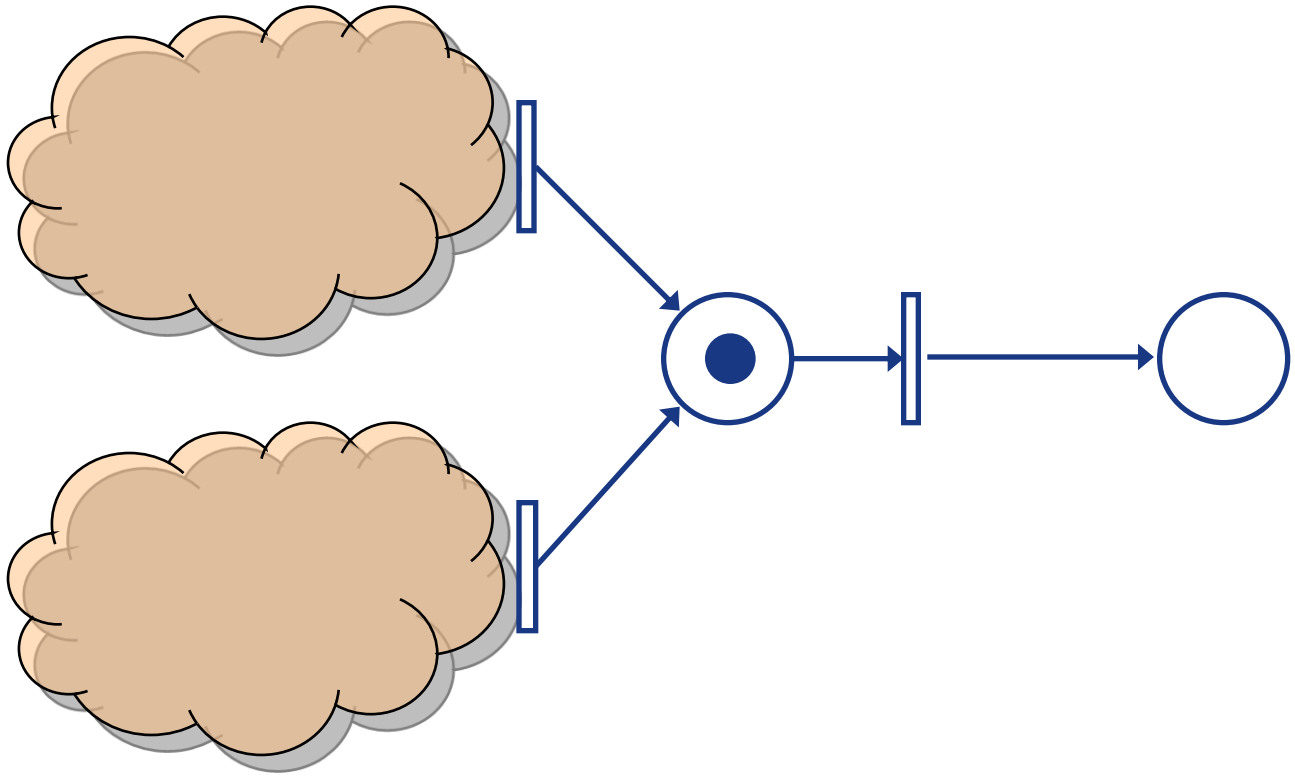
Paralelism AND-join (sincronizarea sfasitului)

Modelarea anumitor concepte (5)



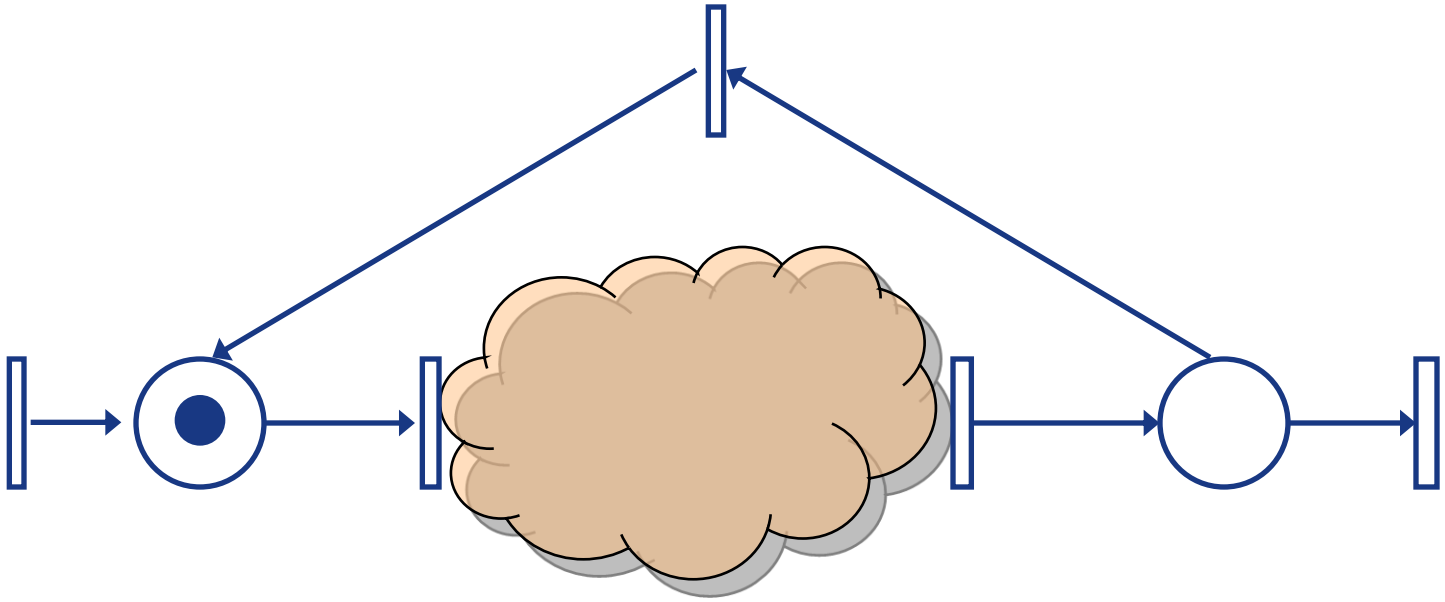
Alternativa XOR-split (la inceput)

Modelarea anumitor concepte (6)



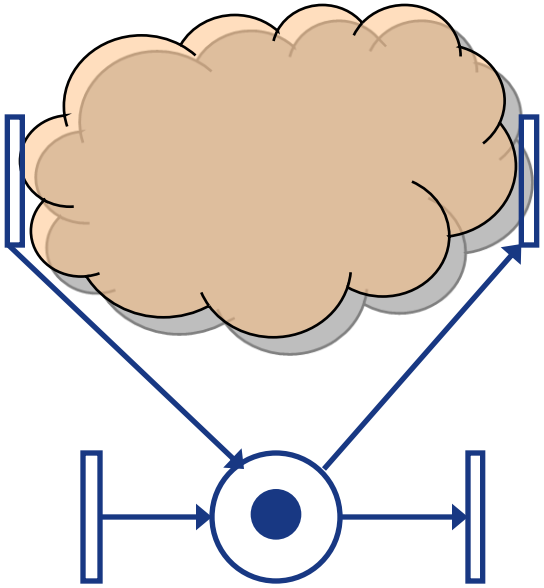
- Alternativa XOR-join (la final)

Modelarea anumitor concepte (7)



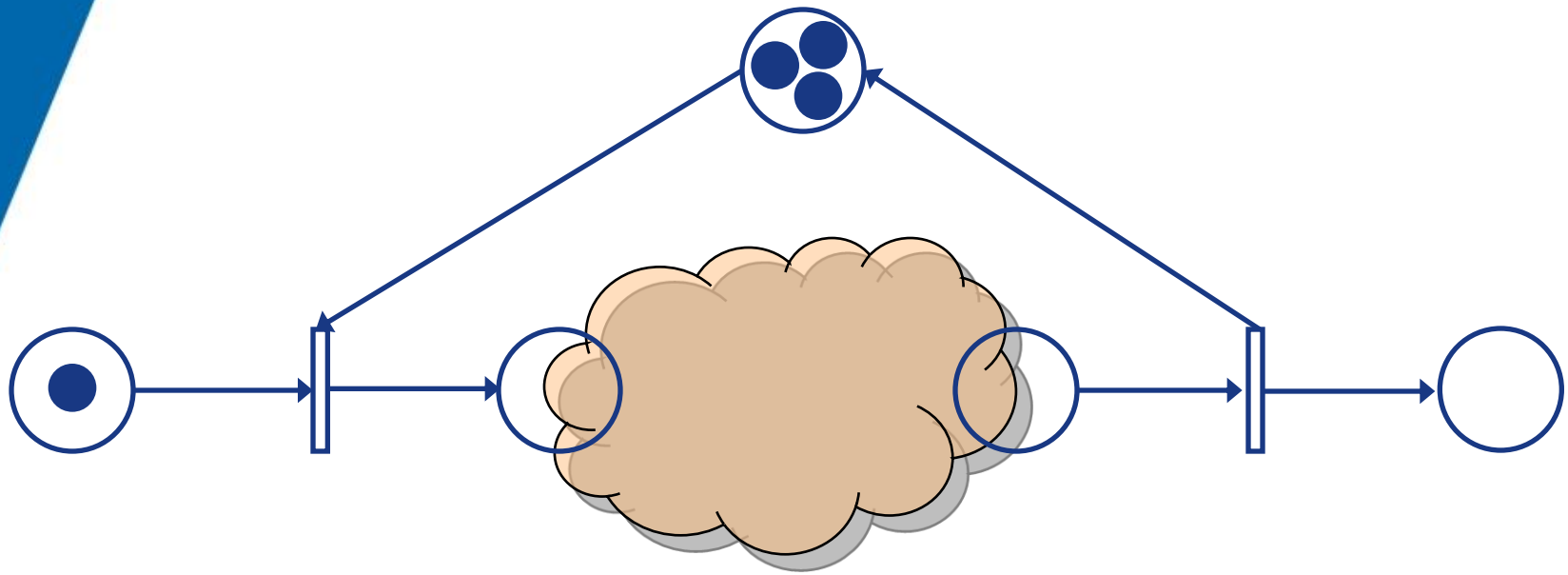
Iteratie (cel putin o data)

Modelarea anumitor concepte (8)



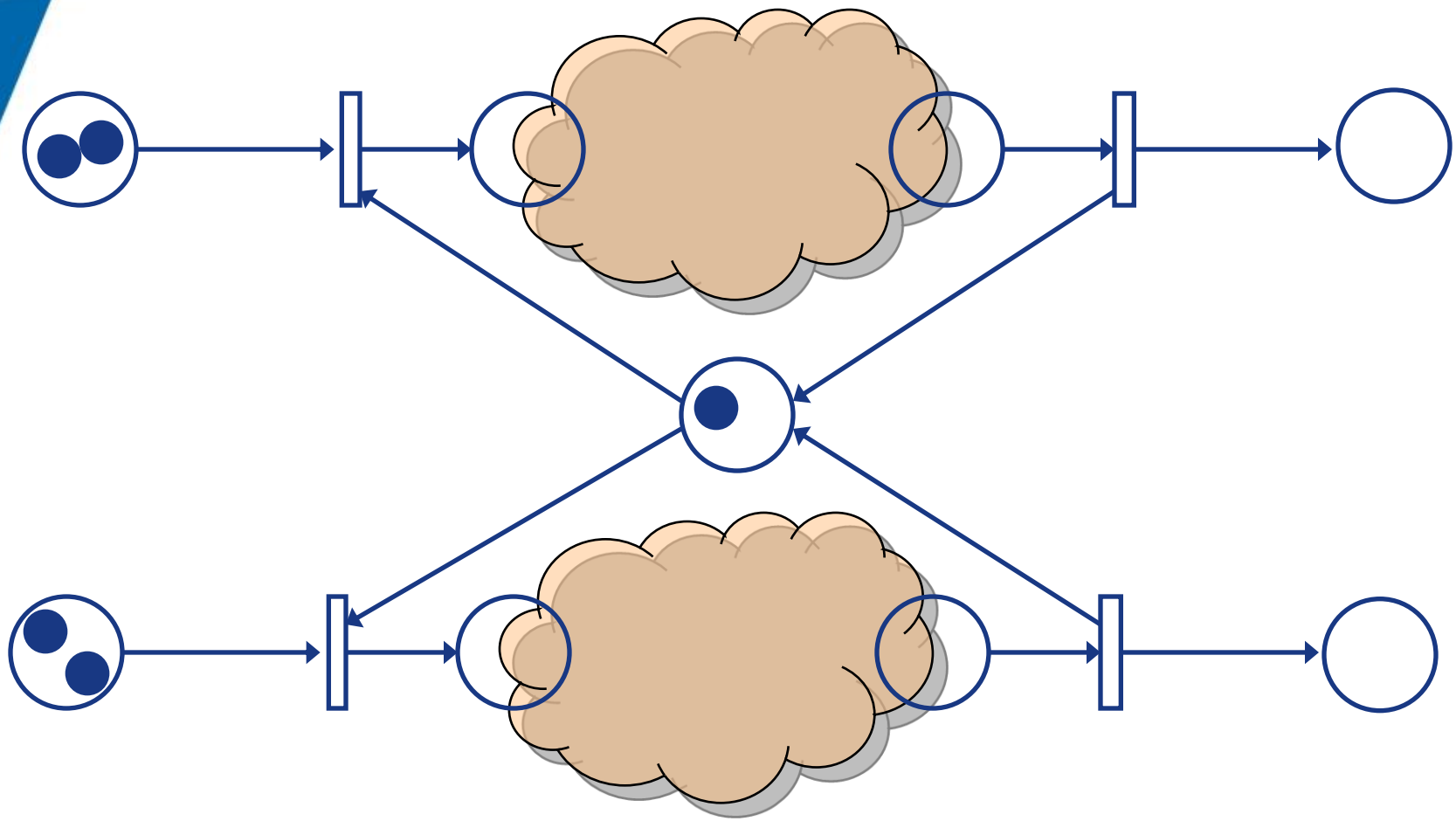
Iteratie (0 sau mai multe ori)

Modelarea anumitor concepte (9)



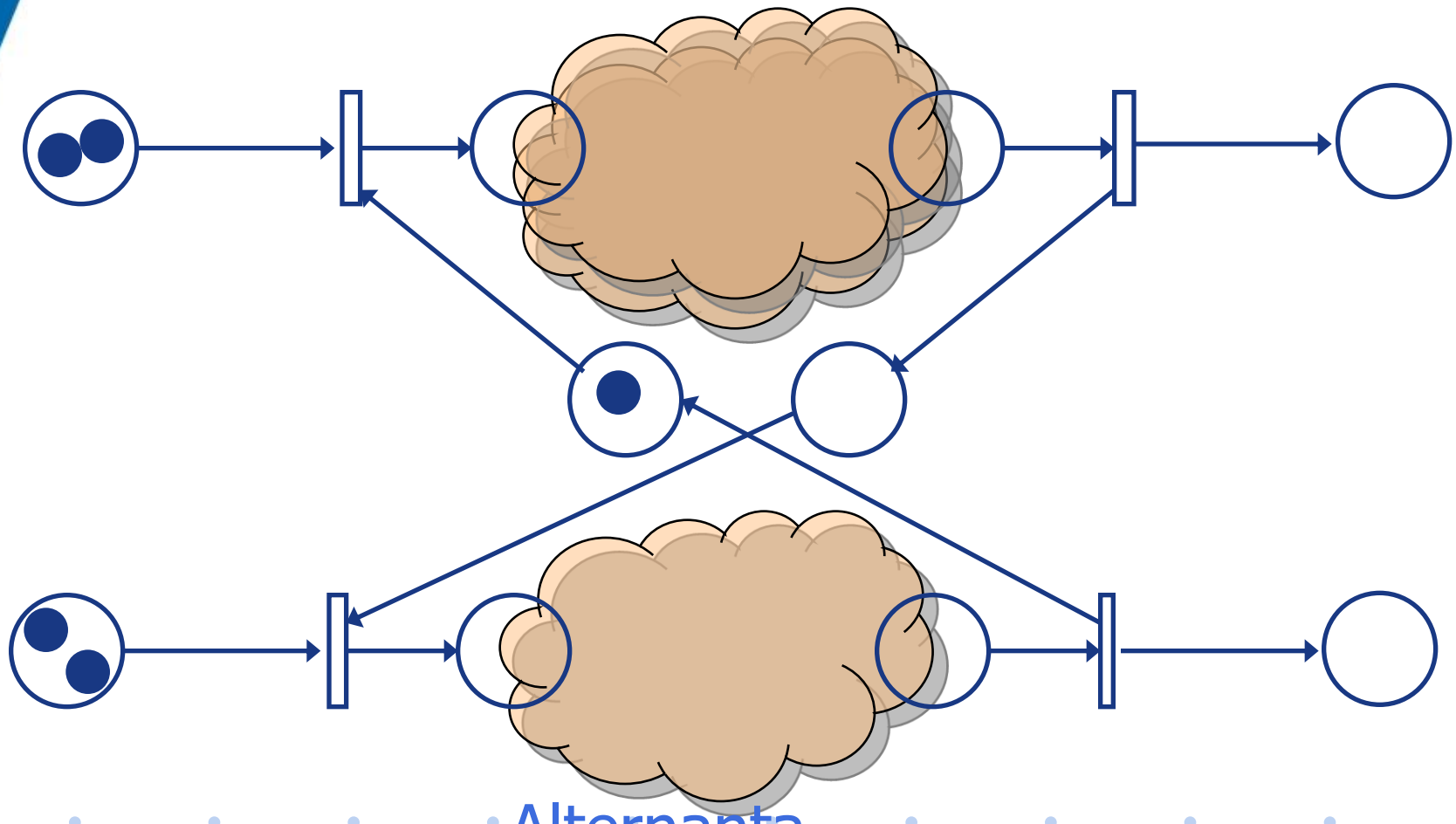
Capacitate limitata

Modelarea anumitor concepte



- Excludere mutuala (partajare resurse)

Modelarea anumitor concepte



Alternanta

Proprietati ale RP (1)

Notatii si definitii

- marcaj initial: $\mathbf{M_0}$
- din marcajul **M,** prin executia tranzitiei T_1 se ajunge in $\mathbf{M_1}$ **:** $\mathbf{M}|T_1>\mathbf{M_1}$
- $m_i(P_j)$ – numarul de jetoane din pozitia P_j corespunzator marcajului M_i**.**
- multimea marcajelor accesibile din $\mathbf{M_0}$: M_0^*
- secventa de executii: $S = T_1T_2$

Proprietati ale RP (2)

Dandu-se doi vectori de marcaj de aceeasi dimensiune, $\mathbf{M_1}$ si $\mathbf{M_2}$,

- vectorul $\mathbf{M_1}$ este superior lui $\mathbf{M_2}$ ($M_1 \geq M_2$) daca $m_1(P_i) \geq m_2(P_i)$ si exista un P_j a.i. $m_1(P_j) > m_2(P_j)$ pentru cel putin o componenta
- vectorul $\mathbf{M_1}$ este strict superior lui $\mathbf{M_2}$ ($\mathbf{M_1}>\mathbf{M_2}$) daca $\mathbf{m_1}(P_i) > \mathbf{m_2}(P_i)$ pentru orice pozitie P_i.

Proprietati ale RP (3)

1. Marginire
2. Viabilitate
3. Blocaje
4. Conflicte
5. Reinitializabilitate

Marginire

- O pozitie P_i este ***marginita*** pentru un marcaj initial $\mathbf{M_0}$ daca exista un intreg pozitiv *k* a.i. pentru orice marcaj in M_0^* numarul de jetoane din P_i nu depaseste *k*
- ***O RP*** este ***marginita*** pentru un marcaj initial $\mathbf{m_0}$ daca toate pozitiile sale sunt marginite pentru $\mathbf{M_0}$.
- O RP este ***sigura*** daca este 1-marginita.

Viabilitate

- O tranzitie T_j este ***viabila*** pentru un marcaj initial $\mathbf{M_0}$ daca pentru orice marcaj $\mathbf{m_i}$ in M_0^* exista o secventa *S* care sa contina T_j.
- O RP este viabila pentru un marcaj initial $\mathbf{M_0}$ daca toate tranzitiile sale sunt viabile pentru $\mathbf{M_0}$.
- O tranzitie T_j este ***quasi-viabila*** pentru un marcaj initial $\mathbf{M_0}$ daca exista macar o secventa *S* din $\mathbf{M_0}$ care sa contina T_j

Blocaje (deadlock)

- Un **blocaj** este un marcaj din care nu mai exista nici o tranzitie valida.
- O RP este **fara blocaje** pentru marcajul initial $\mathbf{M_0}$ daca nici un marcaj din M_0^* nu este un blocaj.

- Nota: Daca o RP are macar un blocaj, atunci nu poate fi viabila.

Observatii

- Proprietatile de marginire, viabilitate, blocaje - depind de marcajul initial al retelei.
- Toate proprietatile de pana acum se pot aplica la abrevieri si pot fi generalizate pentru extensii

Conflicte

- Intr-o RP ordinara, un ***conflict efectiv*** este o pereche formata dintr-un conflict structural

$$K = <P_i, \{T_1, T_2, \ldots\}>$$

si un marcaj **M** in care numarul de jetoane din P_i este mai mic decat numarul de tranzitii de iesire ale acesteia, validate prin **M**.

$$K^E = <K, \mathbf{M}> = < P_i, \{T_1, T_2, \ldots\}, \mathbf{M} >$$

Componenta repetitiva

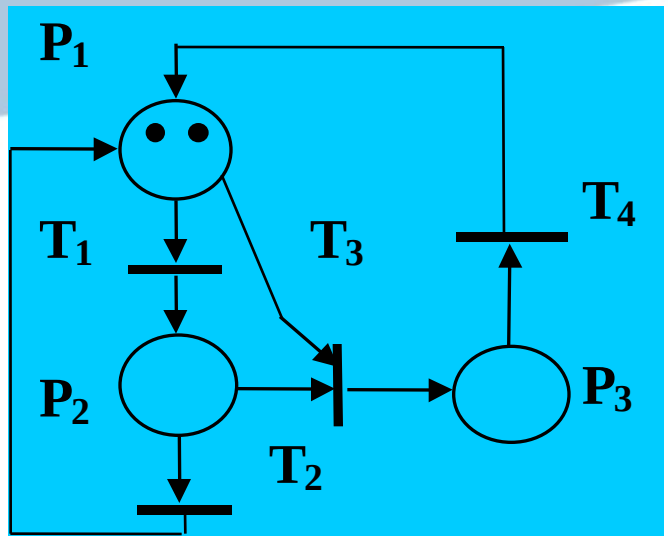
- O ***secventa repetitiva*** este o secventa *S* astfel incat $\mathbf{M_0}|S > \mathbf{M_0}$.
- O secventa repetitiva care contine toate tranzitiile retelei, este o ***secventa repetitiva completa*****.**
- Daca T' este o submultime a tranzitiilor retelei si S_k o secventa repetitiva de tranzitii din T', atunci T' este o ***componenta repetitiva***.

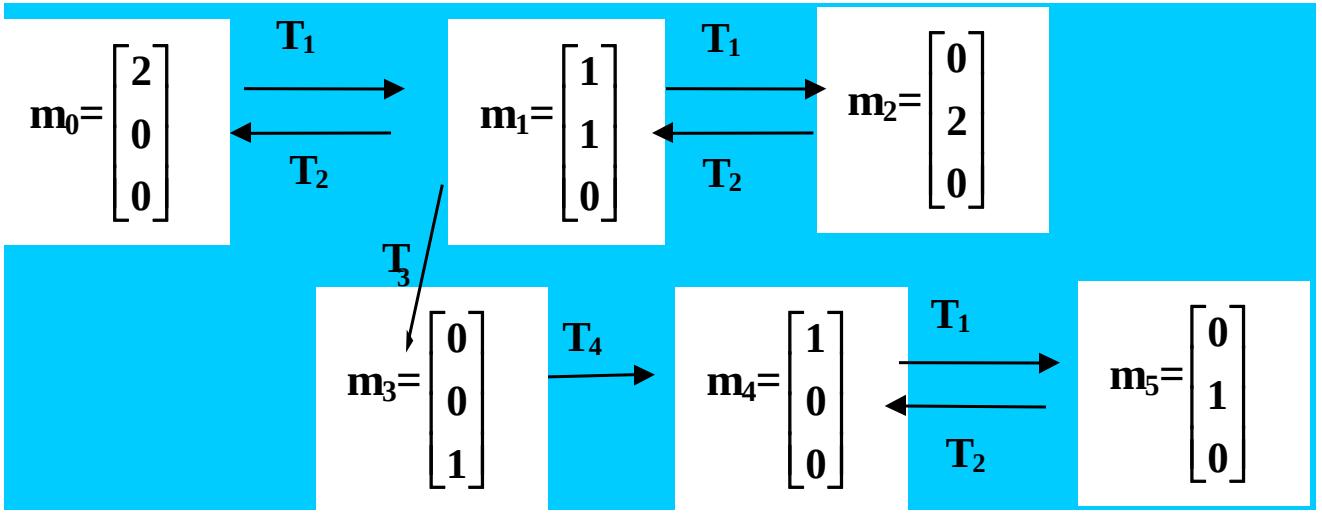
Reinitializabilitate

- O RP poarta numele de reinitializabila daca pentru orice marcaj Mi exista o secventa *S* astfel incat $\mathbf{M_i}|S > \mathbf{M_0}$.

Metode de analiza a RP

- Clase de metode:
  - ***graf de marcaje*** (arbore de acoperire)
    - ***Graful de marcaje*** - poate fi utilizat numai pentru RP marginite si include toate evolutiile posibile ale acestora.
  - ***algebra lineara***


P1
T1
T3
T4
P2
P3
T2


m0= 2 0 0
T1
T2
m1= 1 1 0
T1
T2
m2= 0 2 0
T3
m3= 0 0 1
T4
m4= 1 0 0
T1
T2
m5= 0 1 0

Algoritm de constructie

- Step 1: Din $\mathbf{m_0}$, se determina toate tranzitiile valide si marcajele succesive corespunzatoare.
- Step 2. Pentru fiecare marcaj nou m_i executa Step 2.1., Step 2.2 sau Step 2.3
  - Step 2.1. Daca exista un marcaj $\mathbf{m_j}=\mathbf{m_i}$ pe drumul de la m_0 la $\mathbf{m_i}$, atunci $\mathbf{m_i}$ nu are succesor.
  - Step 2.2. Daca nu exista un marcaj $\mathbf{m_j}$ pe drumul de la $\mathbf{m_0}$ la $\mathbf{m_i}$, graful este completat prin adaugarea tuturor succesorilor lui $\mathbf{m_i}$. Salt la Step 2.
  - Step 2.3 Daca pe calea de la m_0 la m_i exista un marcaj m_j astfel incat $m_i>m_j$ (m_i superior lui m_j), atunci pe pozitia/pozitiile de superioritate se inlocuieste valoarea numerica cu ω (simbol pentru nemarginire). Toate marcajele succesoare ale lui m_i vor pastra valorile ω de pe pozitiile lui m_i.

Exercitii (1)

- Se consideră un sistem circular de cale ferată format din 4 tronsoane (numerotate 1, 2, 3 şi 4) în care trenurile se pot deplasa într-un singur sens şi două trenuri (A şi B). Să se modeleze prin intermediul unei rețele Petri acest sistem ştiind că pe un tronson se poate afla maxim un tren şi că nu se face deosebire între trenuri.

- Se consideră un sistem circular de cale ferată format din 4 tronsoane (numerotate 1, 2, 3 şi 4) în care trenurile se pot deplasa într-un singur sens şi două trenuri (A şi B). Să se modeleze prin intermediul unei rețele Petri acest sistem ştiind că pe un tronson se poate afla maxim un tren şi că se face distincția între trenuri.

Exercitii (2)

- Se consideră un sistem circular de cale ferată format din 4 tronsoane (numerotate 1, 2, 3 şi 4) în care trenurile se pot deplasa într-un singur sens şi două trenuri (A şi B). Să se modeleze prin intermediul unei rețele Petri acest sistem ştiind că pe un tronson se poate afla maxim un tren, pentru ca un tren să poată intra pe un tronson trebuie ca tronsonul ce urmează celui pe care vrea să intre să fie liber şi că nu se face deosebire între trenuri.

- Se consideră un sistem circular de cale ferată format din 4 tronsoane (numerotate 1, 2, 3 şi 4) în care trenurile se pot deplasa într-un singur sens şi două trenuri (A şi B). Să se modeleze prin intermediul unei rețele Petri acest sistem ştiind că un tronson de cale ferată se poate afla în una din următoarele stări: **liber**, **ocupat**, **rezervat**; pentru ca un tren să poată intra pe un tronson trebuie mai întâi să rezerve tronsonul şi că nu se face deosebire între trenuri.