

Petri-Nets

Cocheci Cristiana

Visan Sebastian

Cuprins

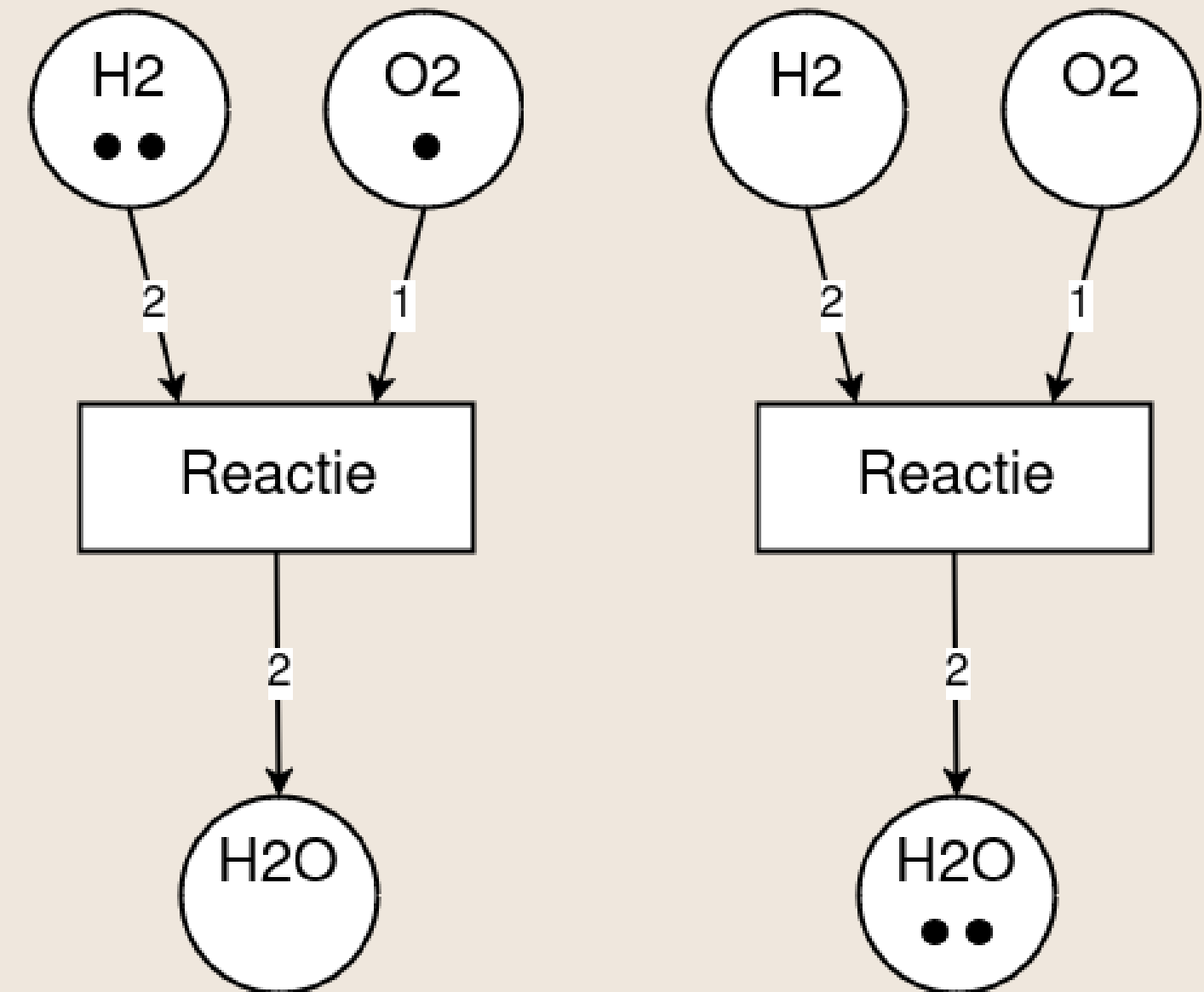
- 1 Introducere
- 2 Structura unui Petri-Net
- 3 State Graph
- 4 Work Clusters
- 5 Detalii Implementare
- 6 Exemple (Mutex, Filozofi & Santa)
- 7 Incheiere

Introducere

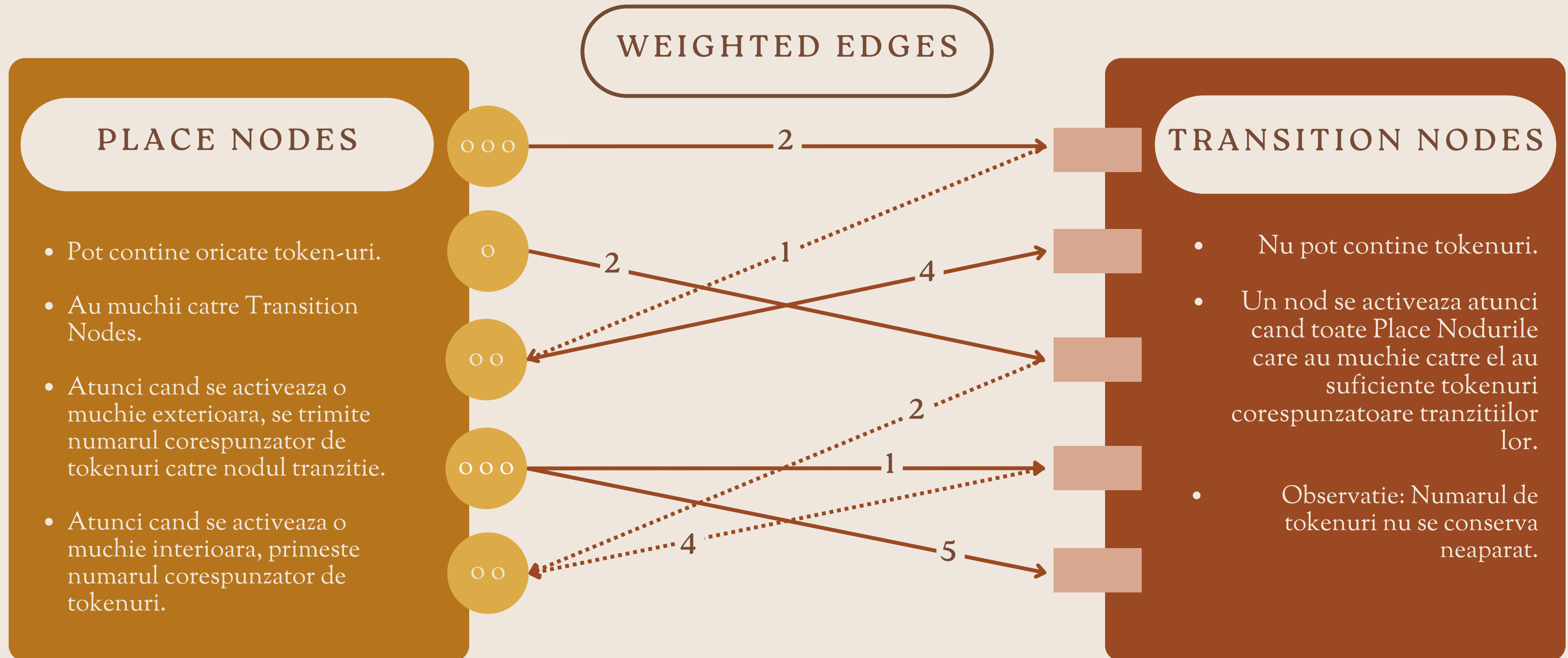
Petri Nets au fost create de catre Carl Petri in 1939 cu scopul de a modela reactii chimice.

Petri nets modeleaza programe concurente si contribuie la procesul de design al unui program concurent.

Un Petri Net este un graf directionat bipartit in care unele noduri pot contine tokenuri.



Structura unui Petri-Net



State Graph

Stare initiala

10	12	34	0	0	0	0
----	----	----	---	---	---	---



Fired Transition 1

7	10	34	1	0	0	0
---	----	----	---	---	---	---



Fired Transition 1

4	8	$\frac{3}{4}$	2	0	0	0
---	---	---------------	---	---	---	---



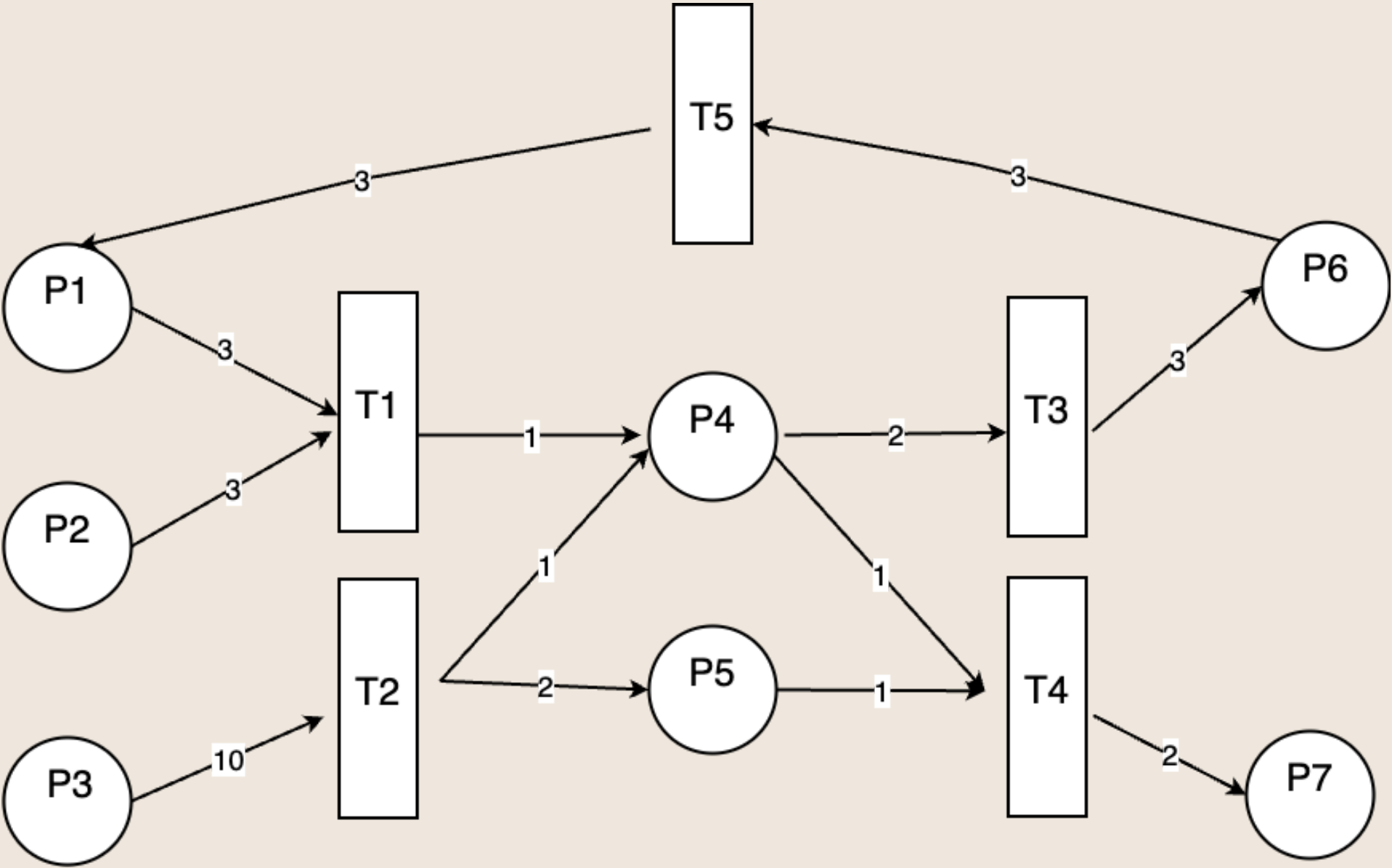
Fired Transition 1

1	6	34	3	0	0	0
---	---	----	---	---	---	---



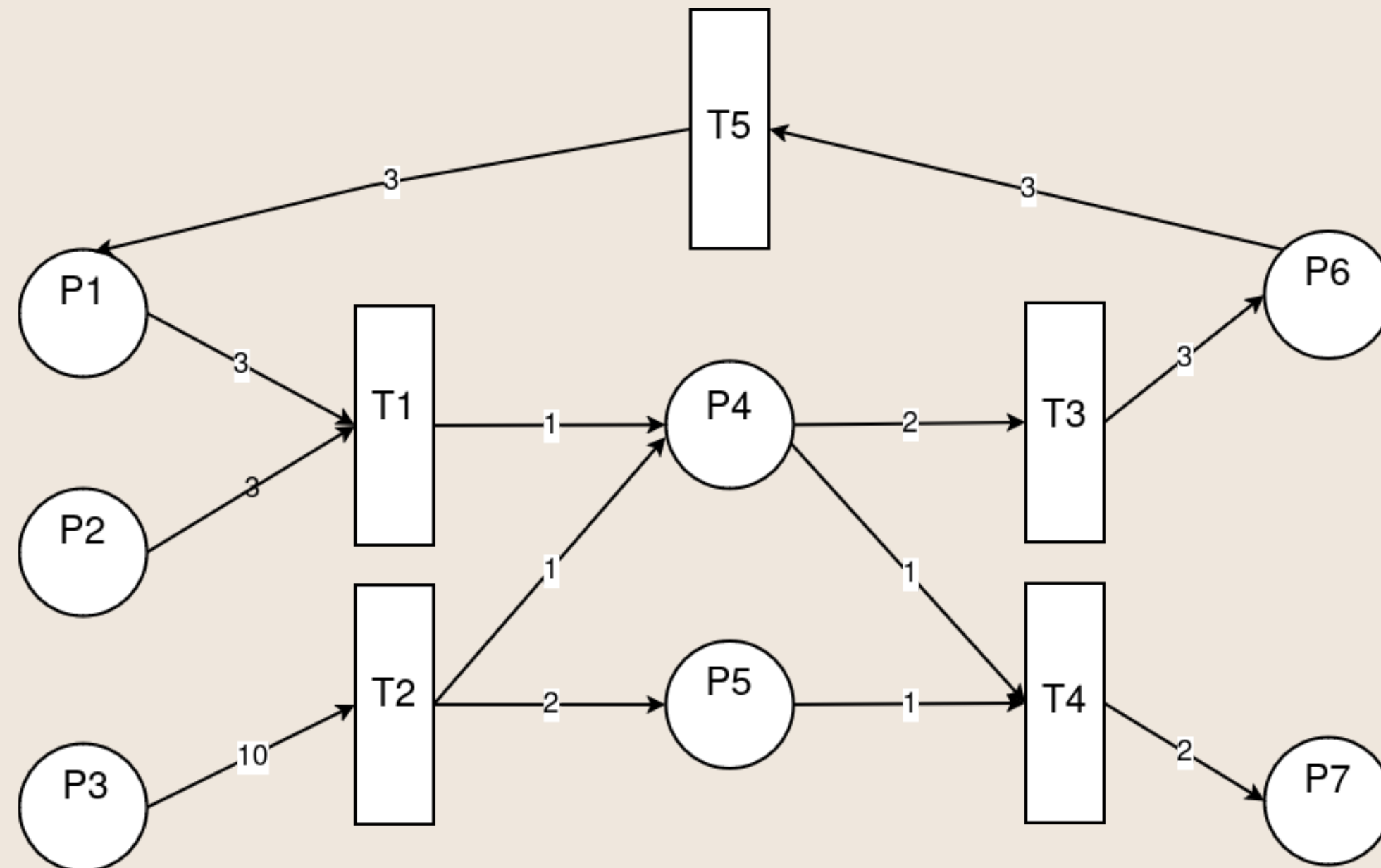
Fired Transition 2

1	6	24	4	2	0	0
---	---	----	---	---	---	---



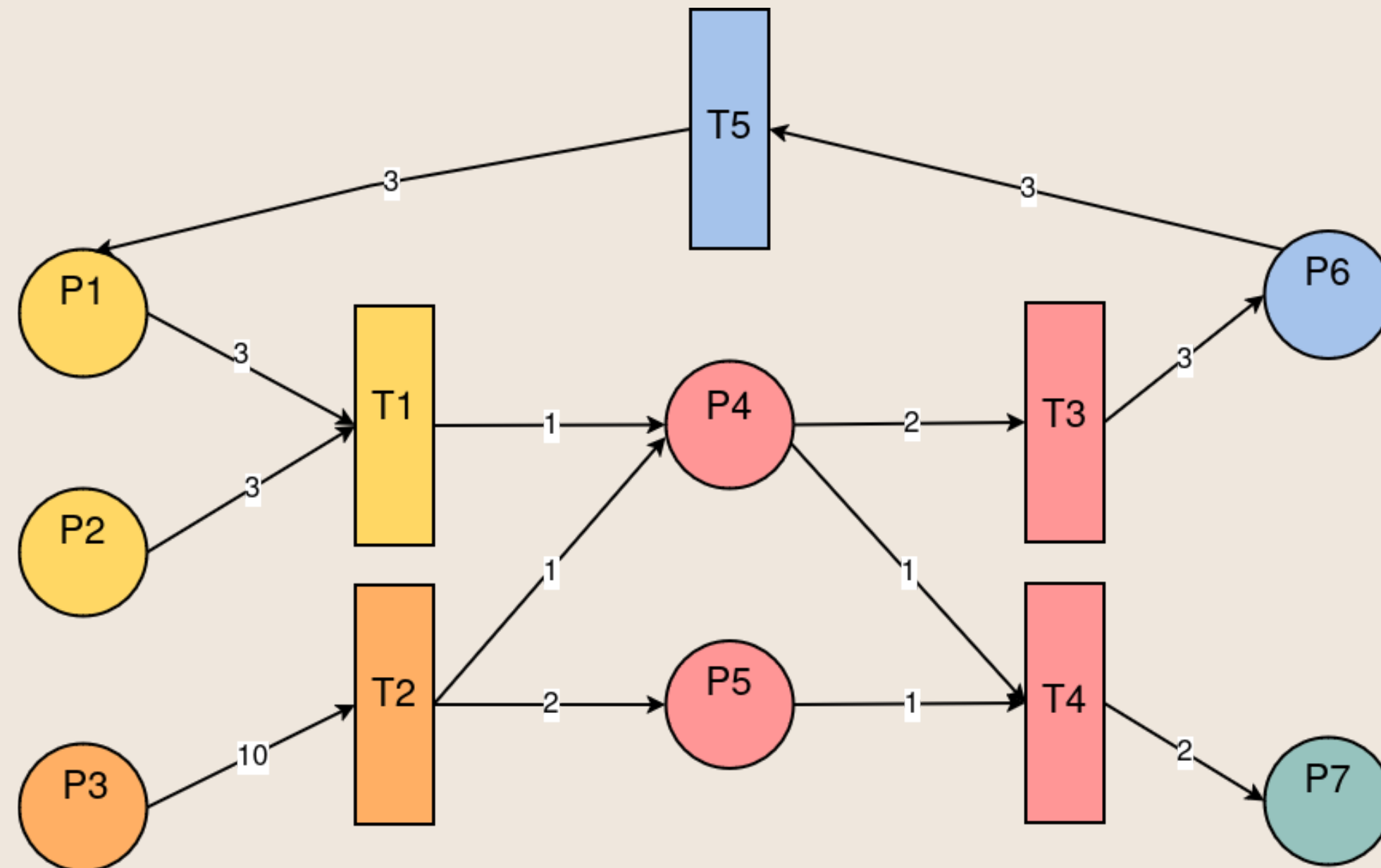
Work Clusters

Nodurile de tranzitie sunt partitionate a.i. un PlaceNode sa nu aiba muchii directionate catre doua partitii diferite.



Work Clusters

Nodurile de tranzitie sunt partitionate a.i. un PlaceNode sa nu aiba muchii directionate catre doua partiti diferite.



NONDETERMINISM

In cazul in care mai multe tranzitii pot fi activate in acelasi moment de timp, alegerea se face aleator (nedeterminist).

WORK CLUSTERS

- Nodurile de tranzitie sunt partitionate a.i. un PlaceNode sa nu aiba muchii directionate catre doua partiti diferite.
- Fiecare partitie este echivalenta unui Work Cluster, respectiv unui thread unic.
- Fara aceasta partitionare apar race conditions.

Detalii de Implementare

GO CHANNELS

- Atunci cand se declanseaza o tranzitie, notificam prin canale work-clusterele afectate ca trebuie sa verifice daca pot da fire la randul lor.
- Pentru a opri executia unui Petri-Net, folosim un canal pe care anuntam momentul finalizarii:
 - cand nu se mai produce o tranzitie prea mult timp (patience)
 - cand expira perioada acceptata de rulare (timeout)

STATE GRAPH

O stare reprezinta numarul de tokenuri care se afla la un moment de timp in fiecare Place nod.

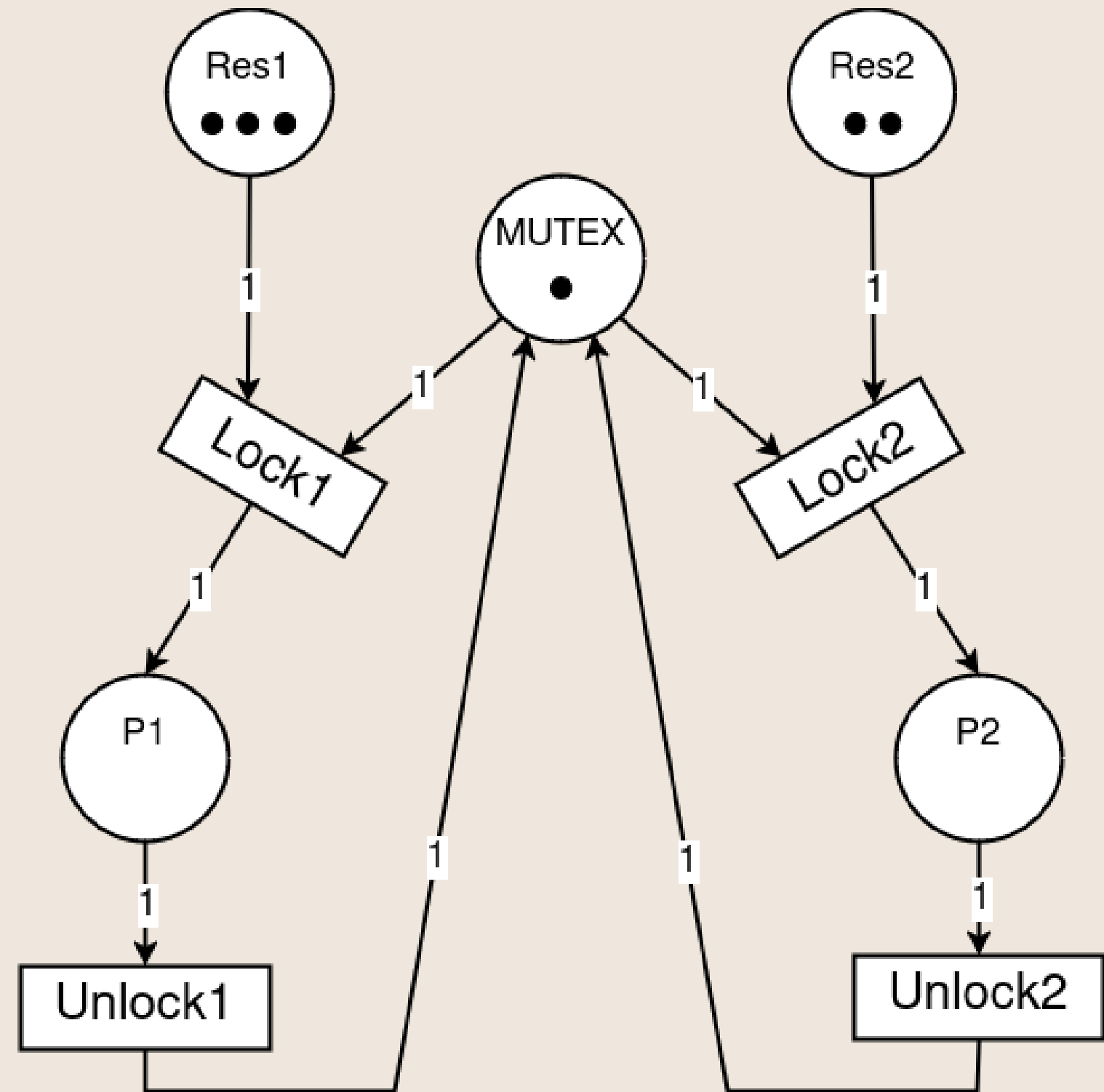
In implementarea noastra, simulam graful de stari afisand intr-un fisier separat toate starile prin care trece Petri-Netul.

Mutex

Putem modela excluziunea mutuala folosind un nod de tipul place care are exact un jeton (cheia lacatului).

Odata ce o tranzitie foloseste acest token, toate celelalte tranzitii care depind de cheie sunt blocate.

Cand se termina procesarea resurselor, jetonul cheie este inapoiat, lasand o alta tranzitie sa il foloseasca.

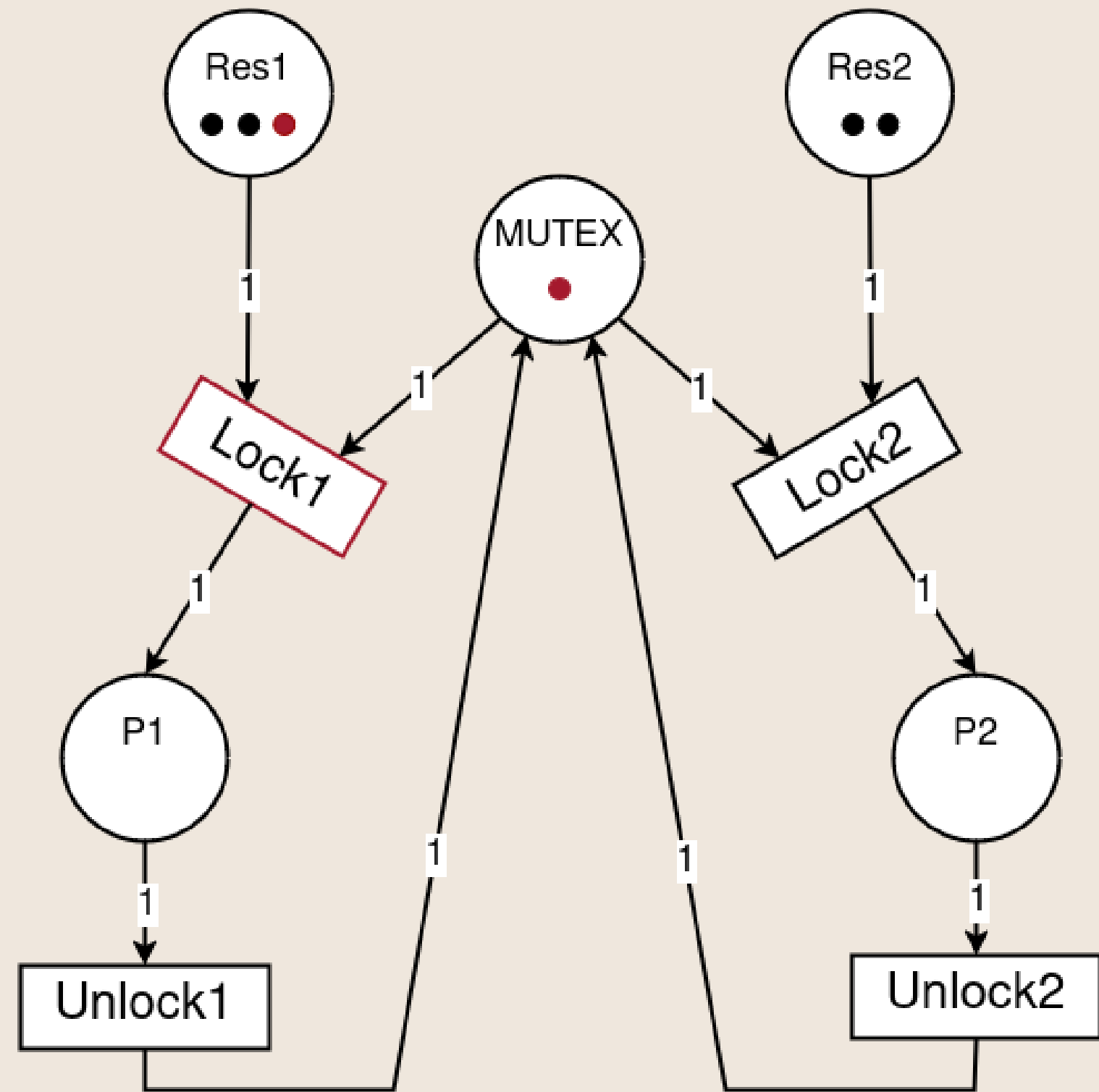


Mutex

Putem modela excluziunea mutuala folosind un nod de tipul place care are exact un jeton (cheia lacatului).

Odata ce o tranzitie foloseste acest token, toate celelalte tranzitii care depind de cheie sunt blocate.

Cand se termina procesarea resurselor, jetonul cheie este inapoiat, lasand o alta tranzitie sa il foloseasca.

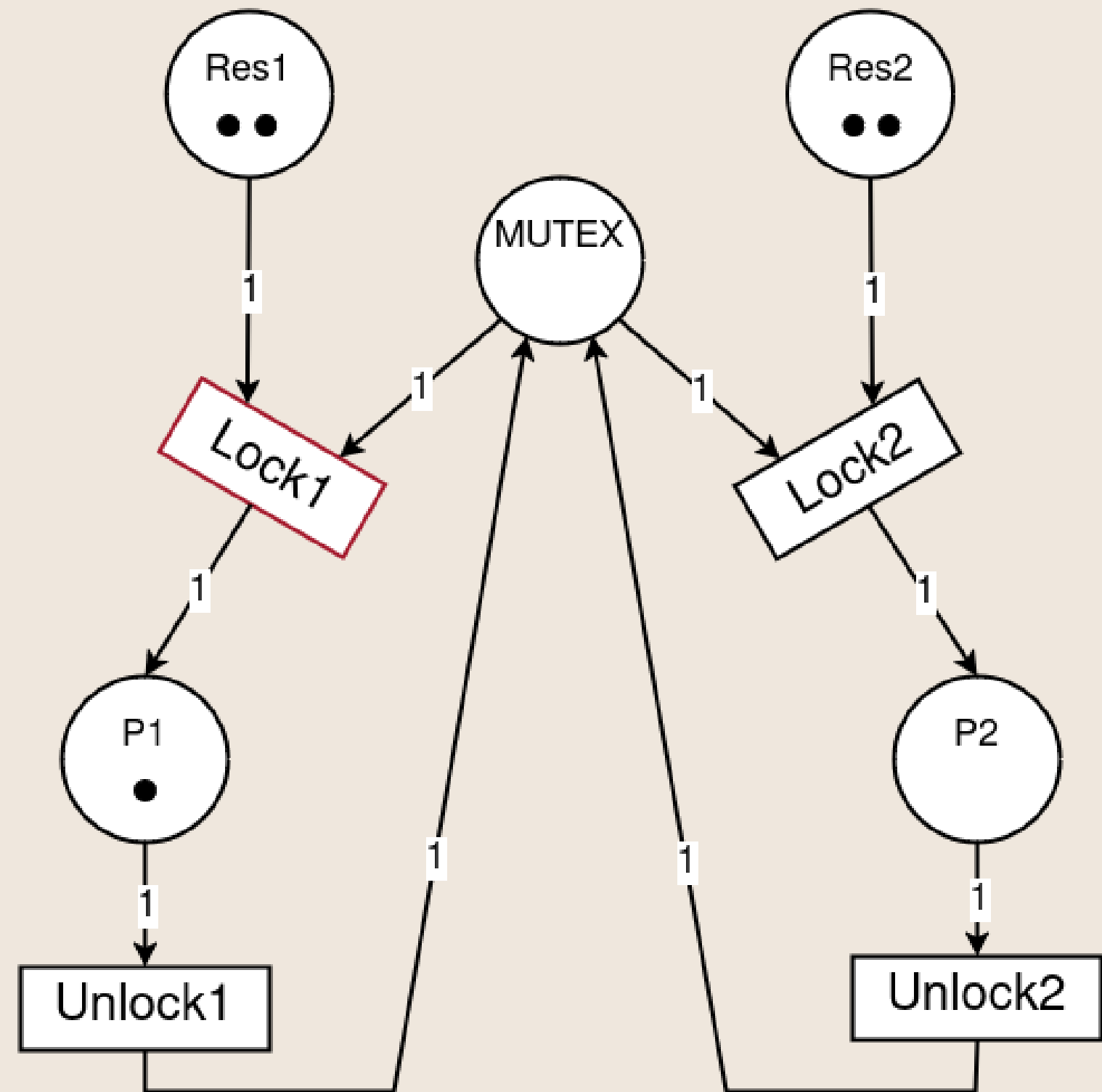


Mutex

Putem modela excluziunea mutuala folosind un nod de tipul place care are exact un jeton (cheia lacatului).

Odata ce o tranzitie foloseste acest token, toate celelalte tranzitii care depind de cheie sunt blocate.

Cand se termina procesarea resurselor, jetonul cheie este inapoiat, lasand o alta tranzitie sa il foloseasca.

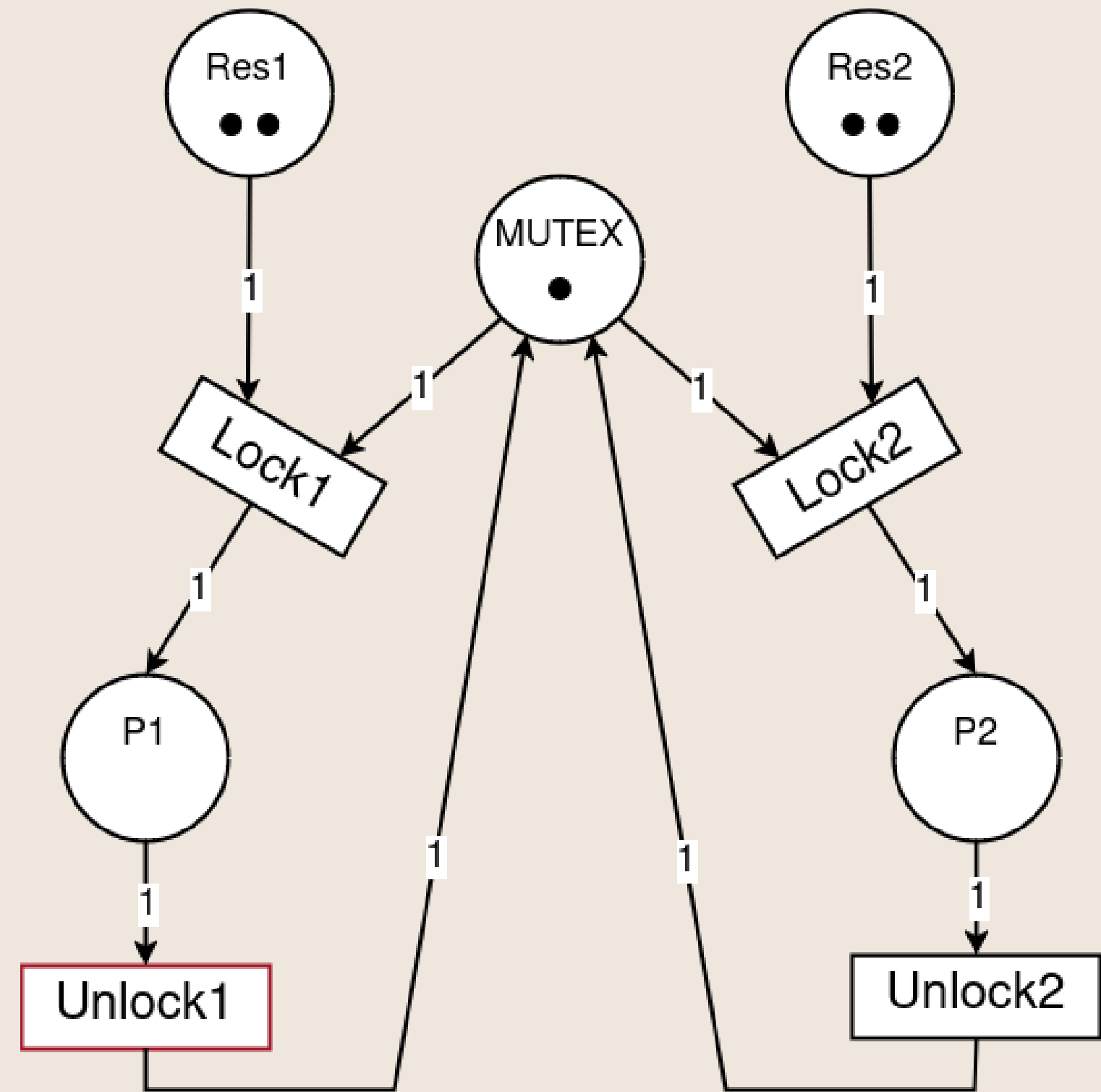


Mutex

Putem modela excluziunea mutuala folosind un nod de tipul place care are exact un jeton (cheia lacatului).

Odata ce o tranzitie foloseste acest token, toate celelalte tranzitii care depind de cheie sunt blocate.

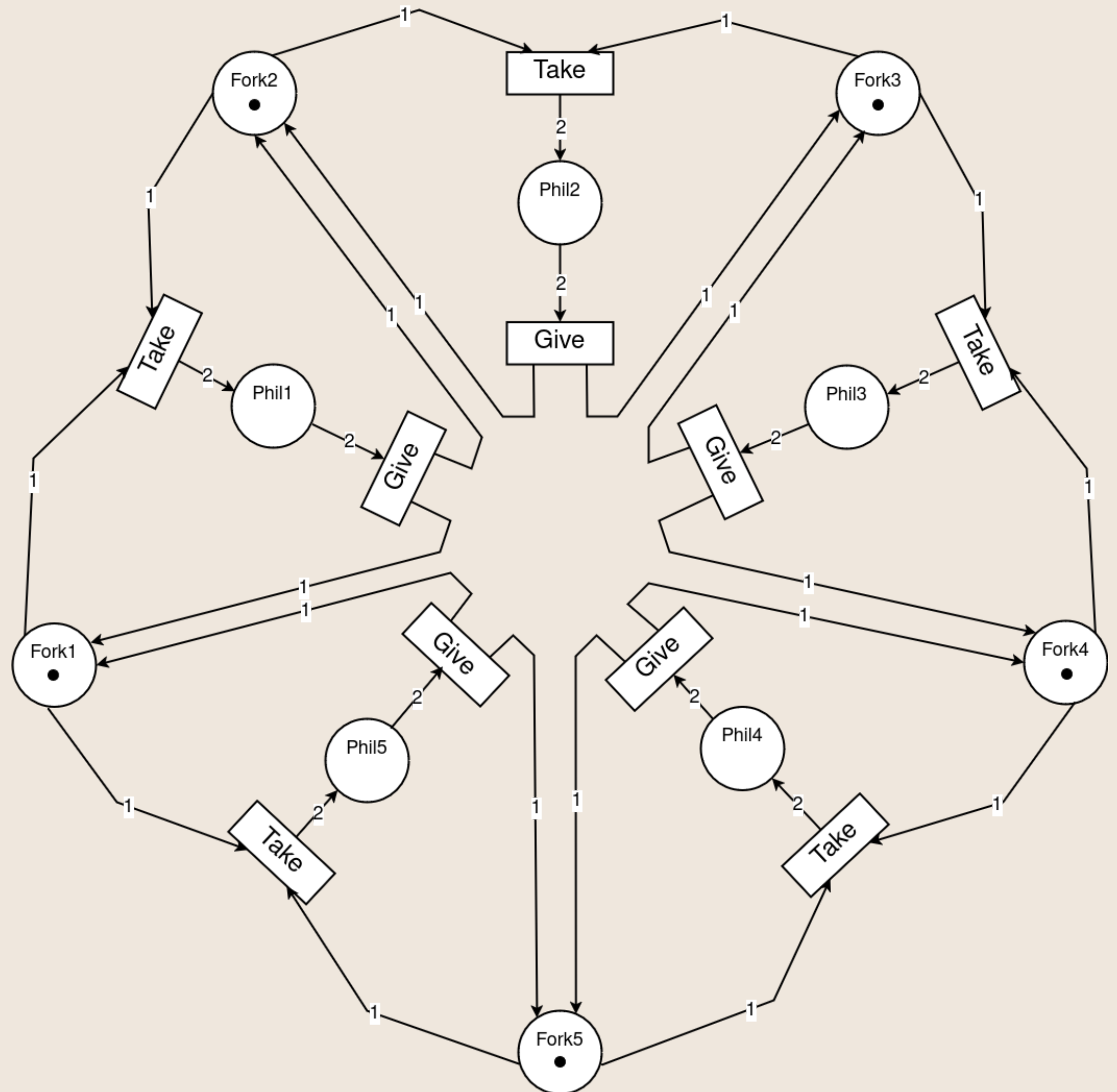
Cand se termina procesarea resurselor, jetonul cheie este inapoiat, lasand o alta tranzitie sa il foloseasca.



Problema Filozofilor

Problema filozofilor este rezolvata de faptul ca o tranzitie se declaseaza atomic.

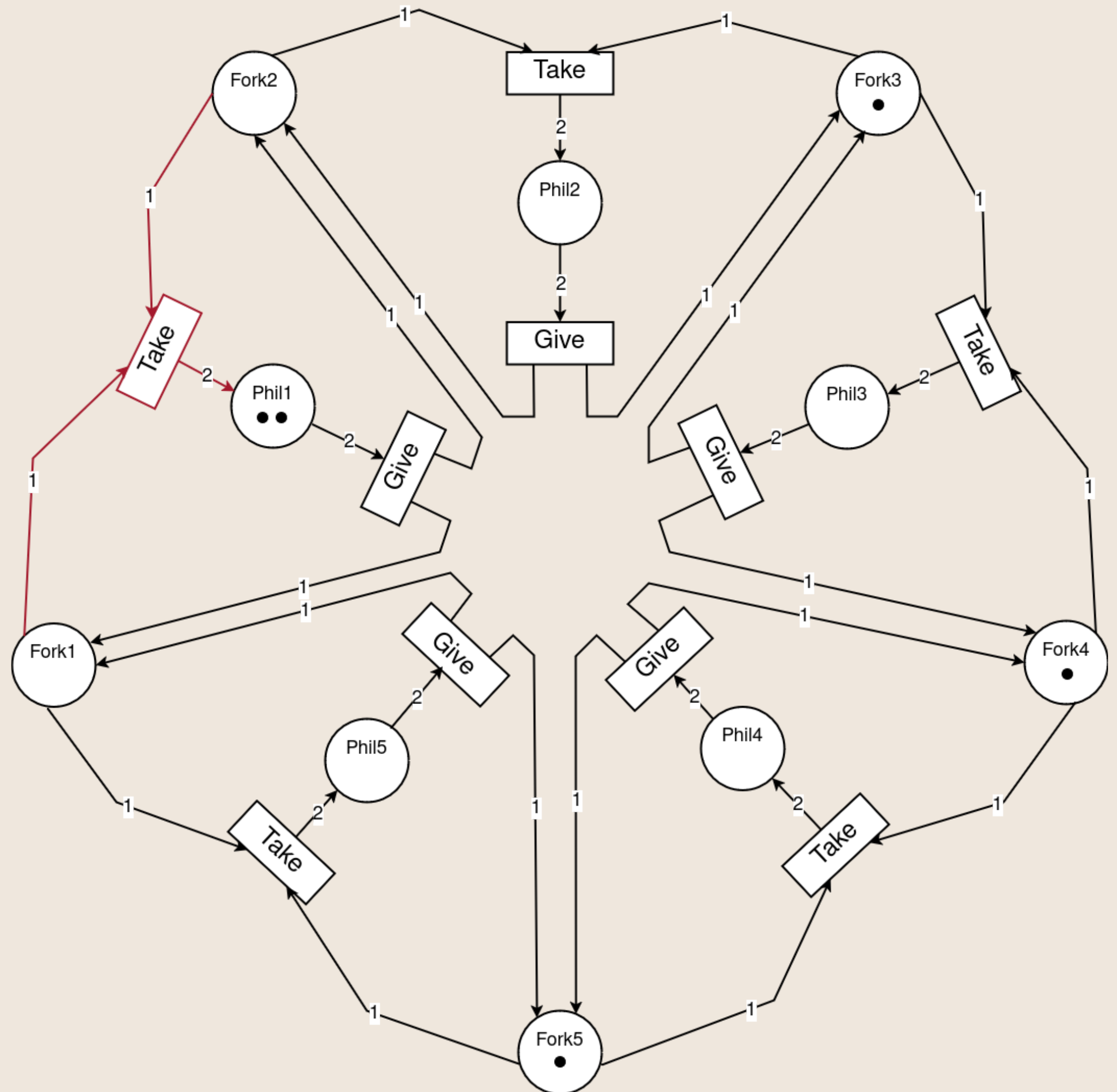
De exemplu, atunci cand primul filosof decide sa ia cele 2 resurse, ia ambele jetoane din nodurile place Fork1 si Fork2. Daca un jeton lipseste, atunci tranzitia nu se declanseaza, iar filosoful asteapta.



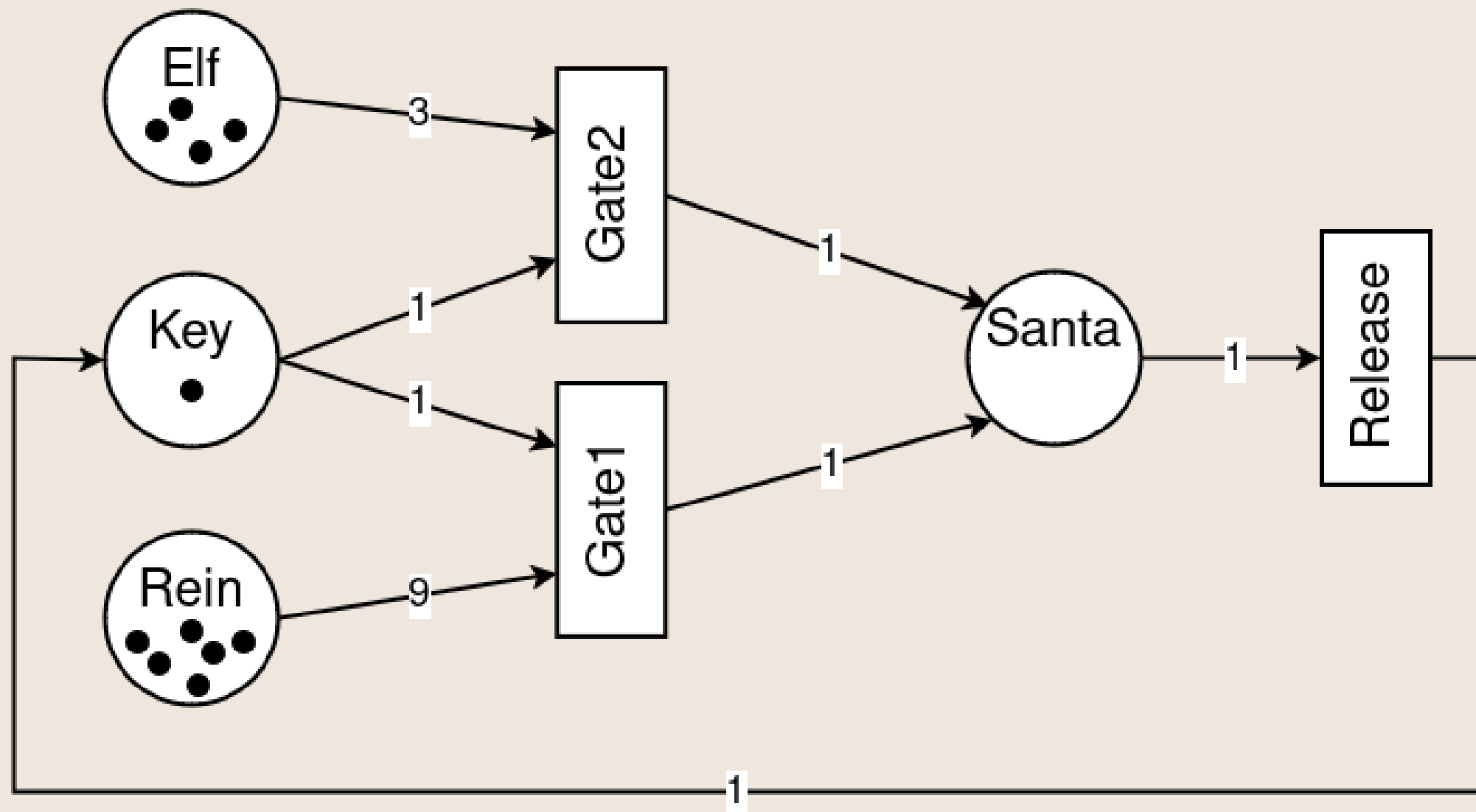
Problema Filozofilor

Problema filozofilor este rezolvata de faptul ca o tranzitie se declaseaza atomic.

De exemplu, atunci cand primul filosof decide sa ia cele 2 resurse, ia ambele jetoane din nodurile place Fork1 si Fork2. Daca un jeton lipseste, atunci tranzitia nu se declanseaza, iar filosoful asteapta.



Santa Claus



- Problema Santa Claus se modeleaza foarte natural cu un Petri Net.
- Considerand fiecare poarta ca o tranzitie, putem folosi un nod Key cu un singur jeton pentru excluziunea mutuala (la Santa intra un singur grup odata).
- Capacitatea fiecarui grup este data de muchia exterioara a nodului Elf/Rein.

Continuarea proiectului

