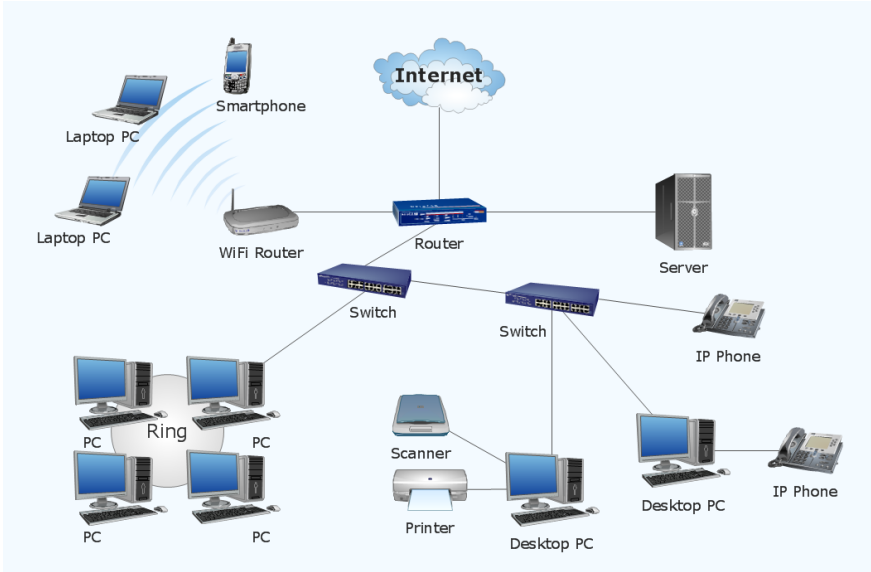
Sisteme și algoritmi distribuiți
Curs 1
FMI – UNIBUC
2024 - 2025


Internet
Smartphone
Laptop PC
Laptop PC
WiFi Router
Router
Server
Switch
Switch
IP Phone
Ring
PC
PC
PC
PC
Scanner
Printer
Desktop PC
Desktop PC
IP Phone

Organizare

Curs 2h/saptamana: A. P.
Laborator 2h/saptamana: Marius Mihailescu

Evaluare:

1. Examen scris (50%)
2. Teme laborator (50%)

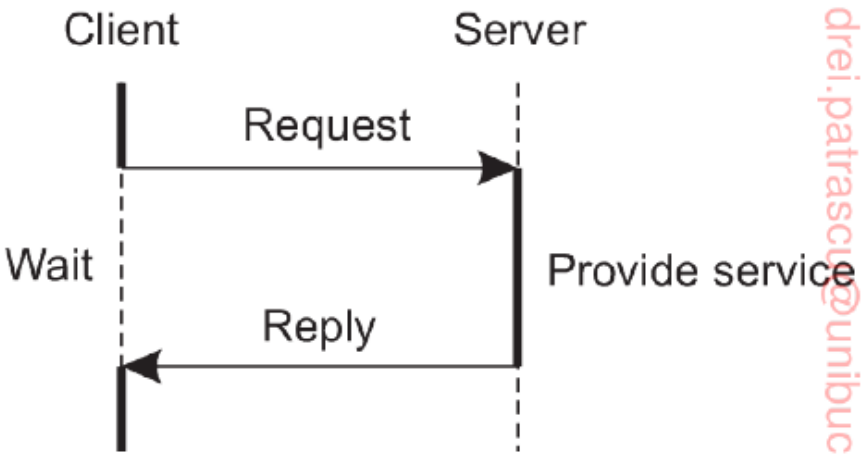
Promovare: nota 5 examen + nota 5 laborator.
Materiale și comunicare: platforma Teams.
Adresa e-mail: andrei.patrascu@fmi.unibuc.ro

Plan materie

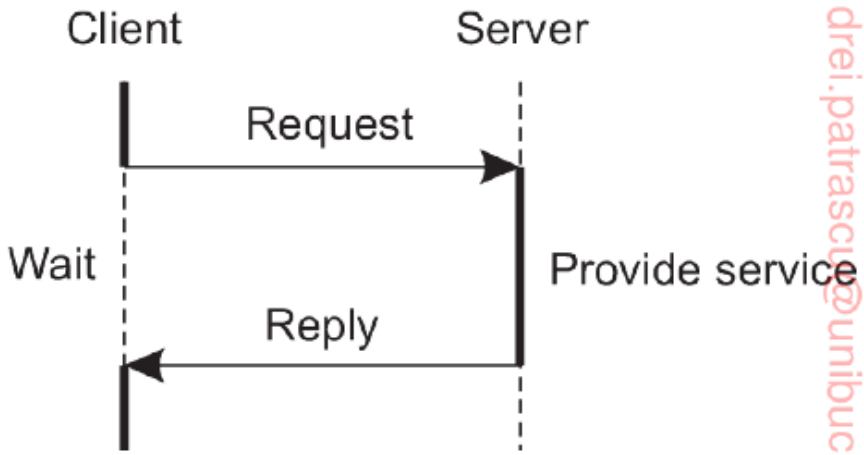
1. Introducere. Modele și arhitecturi SD.
2. Comunicație în SD. Ceasuri logice.
3. **Sisteme sincrone**
    - 3.1 Algoritmi OSD: alegere lider, consens, sincronizare
    - 3.2 Defecte
    - 3.3 Consens și alte probleme numerice: medie, evaluare funcții, sisteme liniare
    - 3.4 Toleranță la defecte
4. **Sisteme asincrone**
    - 4.1 Organizarea SDa: teorema imposibilitate, consens relaxat
    - 4.2 Medie și sisteme liniare în context asincron
    - 4.3 Toleranță la defecte
5. **Algoritmi distribuiți aleatori**:
    - 5.1 Paradigma gossip
    - 5.2 Consens și alte probleme
    - 5.3 Toleranță la defecte

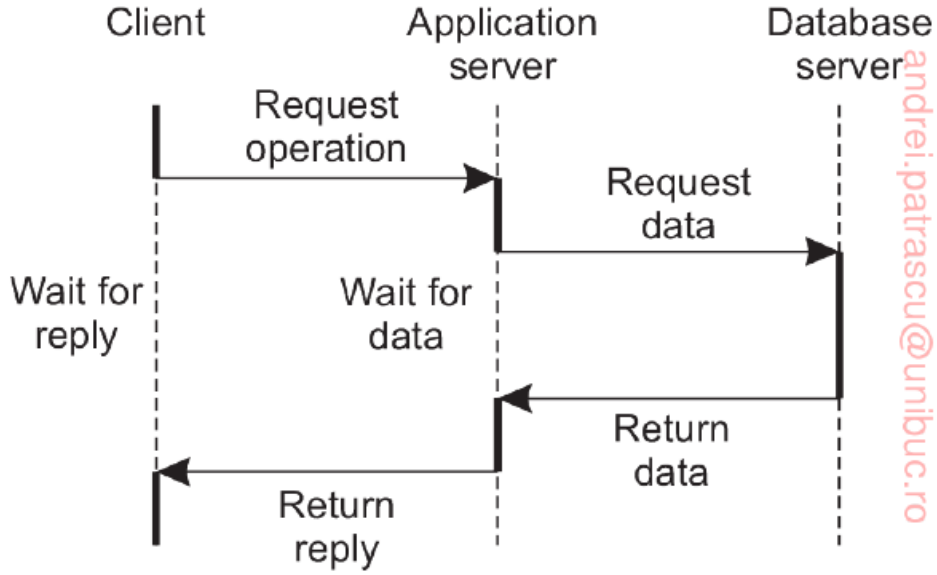
Client-Server

- Un nod/proces are calitatea de client sau server
- **Server** = entitate care ofera un serviciu
- **Client** = entitate care apeleaza un serviciu



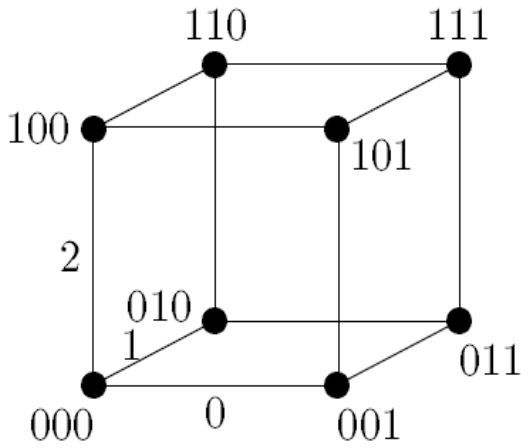
Client-Server

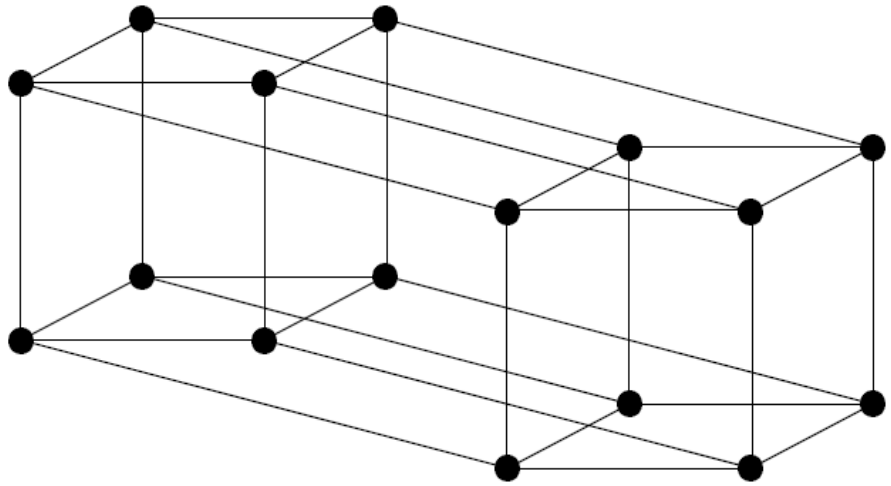




Sistem multiprocesor structurat

- Fiecare nod are aceleași capacități
- Comunicație sigură între procese
- Identificator unic per proces
- Folosită în “cluster (parallel) computing”

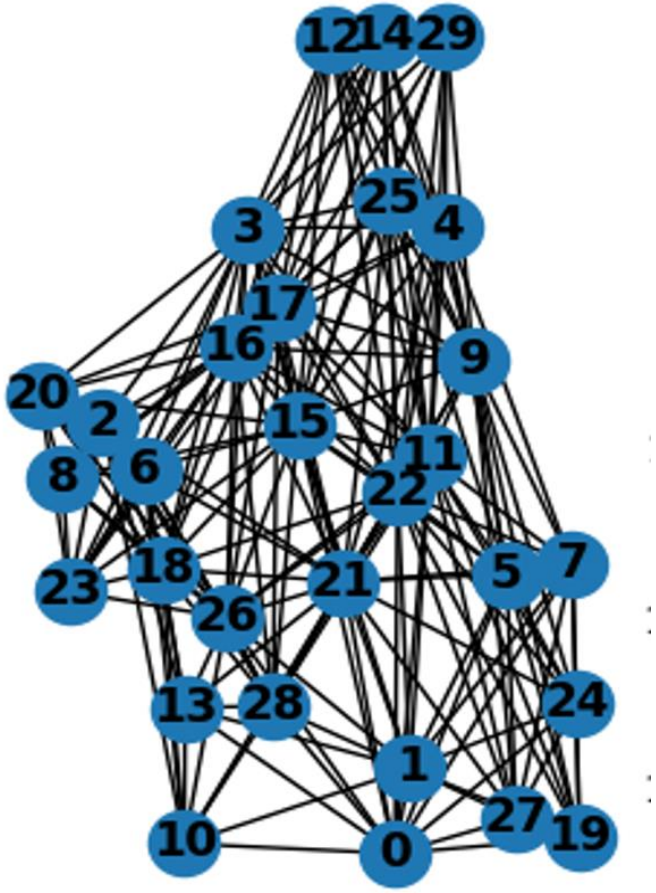




Sisteme nestructurate (peer-to-peer)

Exemplu: Cloud Computing, Sisteme de stocare

- Specifice modelelor generale de “**sisteme distribuite**”
- Nodurile nu au un identificator unic global
- Comunicatia intre doua noduri se realizeaza prin muchiile disponibile
- Topologia potential variabila in timp
- Posibile defecte pe legaturi sau noduri



Sistem distribuit

Sistem distribuit = o colecție de procese autonome care comunică peste o rețea (topologie) cu următoarele proprietăți**:**

- **Fiecare nod are o „vedere” locală asupra sistemului.** Un nod al sistemului cunoaște și comunică cu propria vecinătate, neavând acces la informații globale. În general, există o separare geografică a nodurilor**.**

- **Nu există un ceas fizic comun**. Acestui aspect se datorează caracterul “distribuit” al sistemului și este cel care cauzează lipsa sincronizării între noduri.

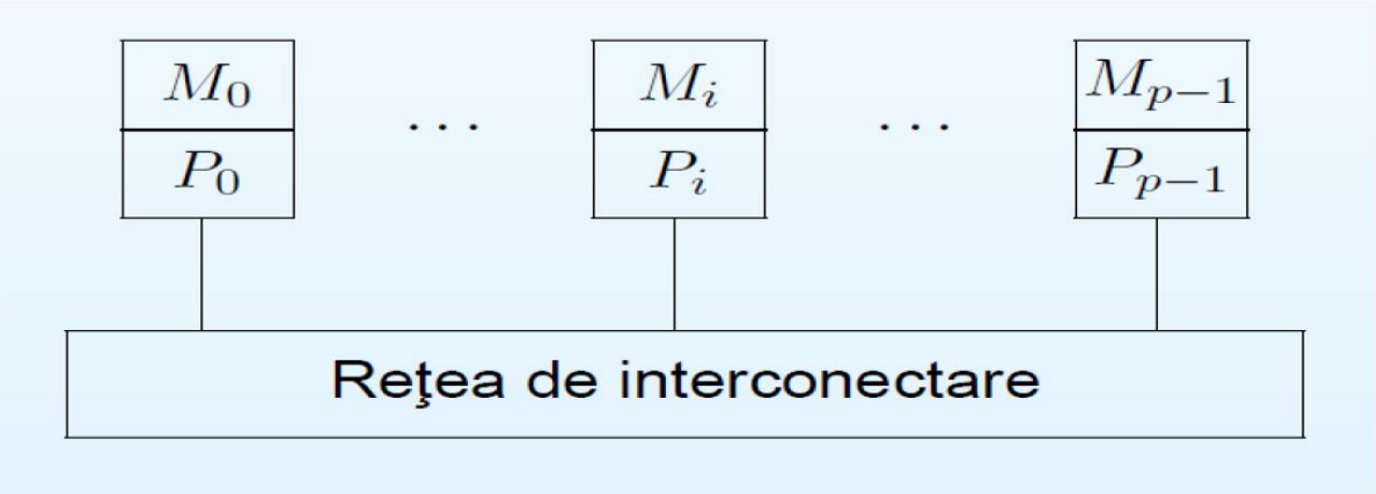
- **Nu există memorie partajată.** Proprietate care aduce necesitatea comunicației prin mesaje (în absența unui ceas global).

- **Autonomia și eterogenitatea nodurilor.** Noduri sunt “slab cuplate”, au viteze diferite de execuție, au sisteme de operare diferite.

Sistem distribuit vs. Sistem paralel

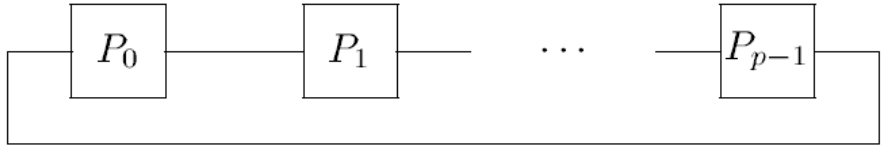
Sistem MIMD cu memorie distribuită

- Fiecare procesor are memorie proprie (arhitectura locală cu RISC şi memorie ierarhică, de obicei)
- Comunicaţia se face printr-o reţea de comunicaţie, prin mesaje explicite
- Operaţii favorizate: paralele, la nivel de bloc
- Comunicaţia prin mesaje necesită algoritmi dedicaţi

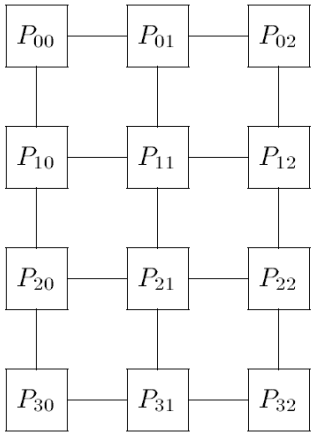


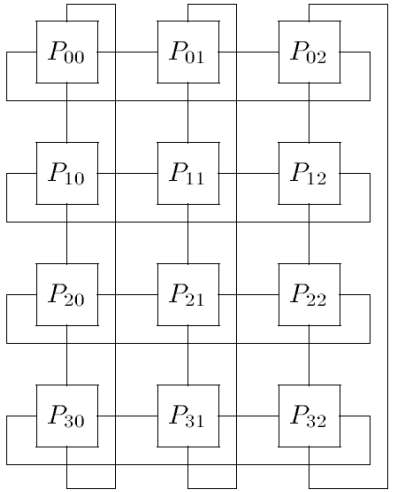
MIMD cu memorie distribuită

Inel

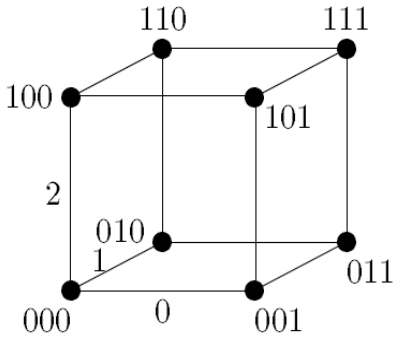


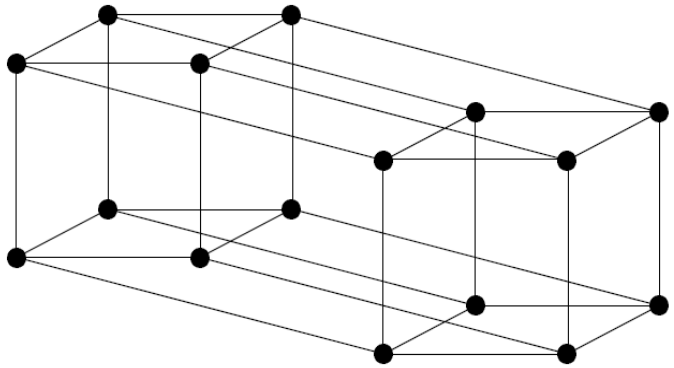
Grila/Tor





Hipercub



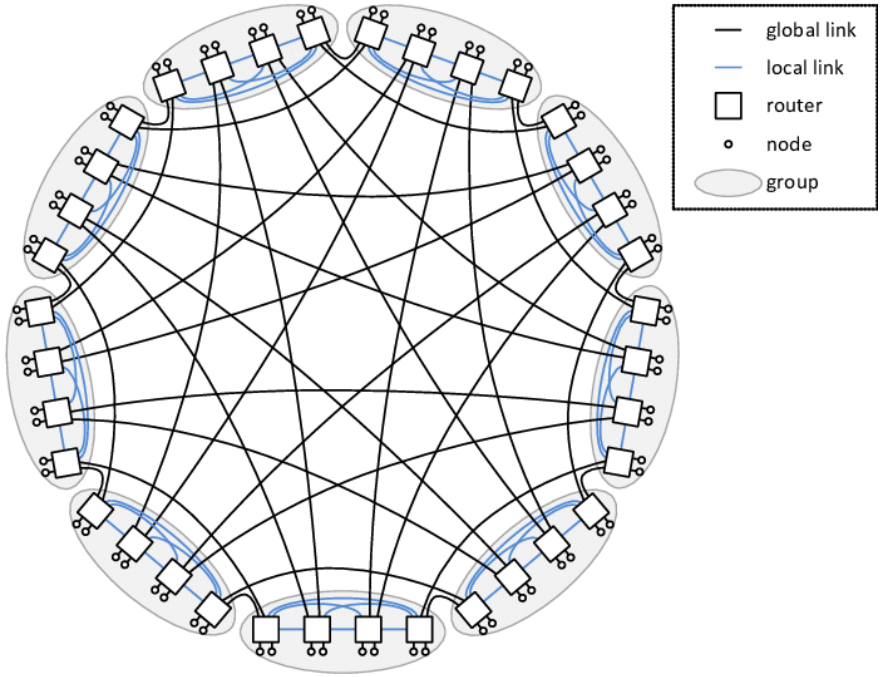


Frontier

Rank	System	Cores	Rmax (PFlop/s)	Rpeak (PFlop/s)	Power (kW)
1	**Frontier** - HPE Cray EX235a, AMD Optimized 3rd Generation EPYC 64C 2GHz, AMD Instinct MI250X, Slingshot-11, HPE DOE/SC/Oak Ridge National Laboratory United States	8,699,904	1,206.00	1,714.81	22,786
2	**Aurora** - HPE Cray EX - Intel Exascale Compute Blade, Xeon CPU Max 9470 52C 2.4GHz, Intel Data Center GPU Max, Slingshot-11, Intel DOE/SC/Argonne National Laboratory United States	9,264,128	1,012.00	1,980.01	38,698
3	**Eagle** - Microsoft NDv5, Xeon Platinum 8480C 48C 2GHz, NVIDIA H100, NVIDIA Infiniband NDR, Microsoft Azure Microsoft Azure United States	2,073,600	561.20	846.84	
4	**Supercomputer Fugaku** - Supercomputer Fugaku, A64FX 48C 2.2GHz, Tofu interconnect D, Fujitsu RIKEN Center for Computational Science Japan	7,630,848	442.01	537.21	29,899
5	**LUMI** - HPE Cray EX235a, AMD Optimized 3rd Generation EPYC 64C 2GHz, AMD Instinct MI250X, Slingshot-11, HPE EuroHPC/CSC Finland	2,752,704	379.70	531.51	7,107

Frontier cluster

- 9.472 CPU AMD Epyc 7713 (64 cores)
  - 606.208 core-uri
- 74 rack-cabinets
- 1 rack-cabinet are 64 server-blades
- 1 server-blade are 2 noduri
- Nodurile sunt structurate pe grupuri; grupurile sunt conectate intr-o topologie “dragonfly”

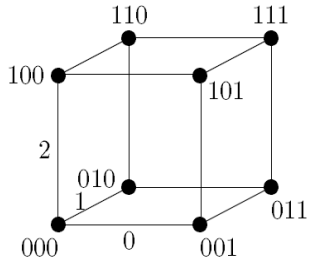


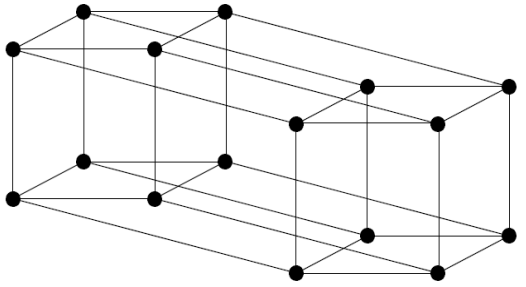
Sistem paralel (Cluster computing)

- **Informații globale disponibile**: număr de noduri ale rețelei, topologia rețelei, distribuția datelor în rețea, indexarea globală a nodurilor
  - Control asupra distribuției datelor
  - Control asupra execuției locale per nod
  - Control asupra implicării nodurilor în rețea

- **Timp de comunicație** inter-noduri neglijabil/mărginit (apropiere geografică)
  - Sincronizare: ceas fizic comun

- Topologie statică
- Probabilitatea scăzută a defectelor
- Complexitatea timp vs. complexitatea mesaj

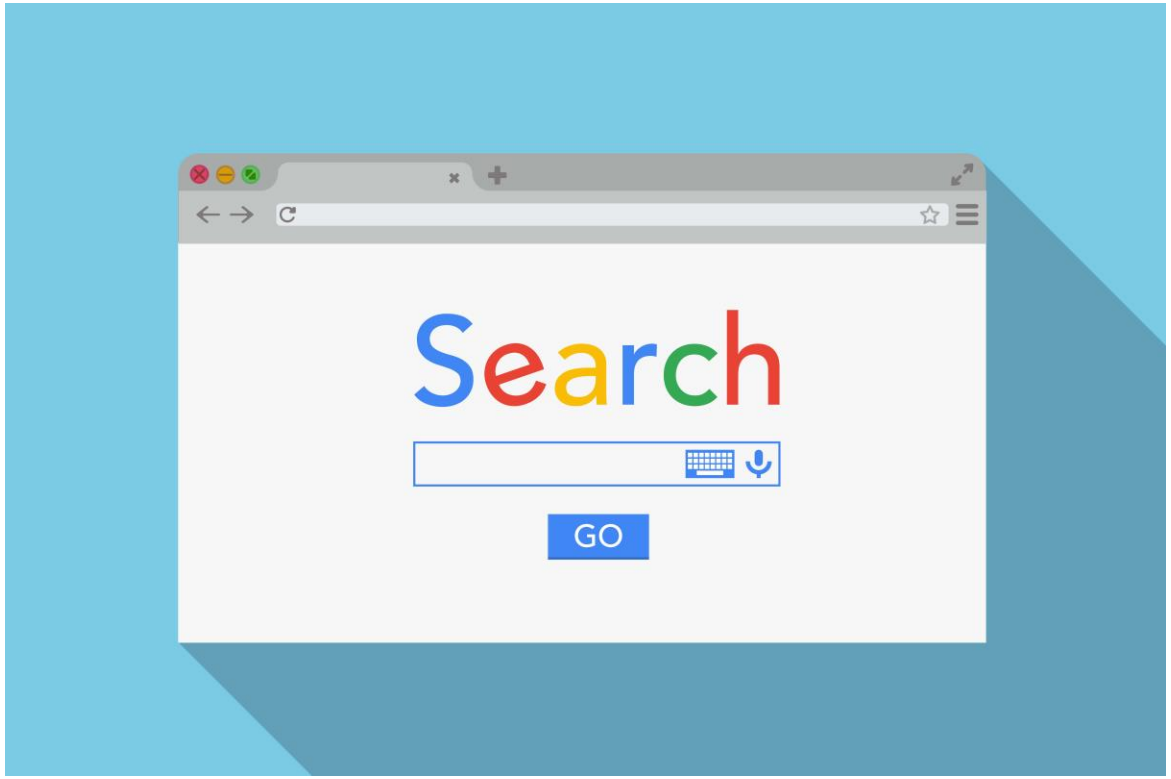




Exemple (clasificare software)

Motoare de căutare

- Alg. PageRank
- Combină paradigmele anterioare



Exemple (clasificare software)

Sistem bancare

- Sistem informațional
- Nod = replică baze de date
- Probleme de consistență etc.

Exemple (clasificare software)

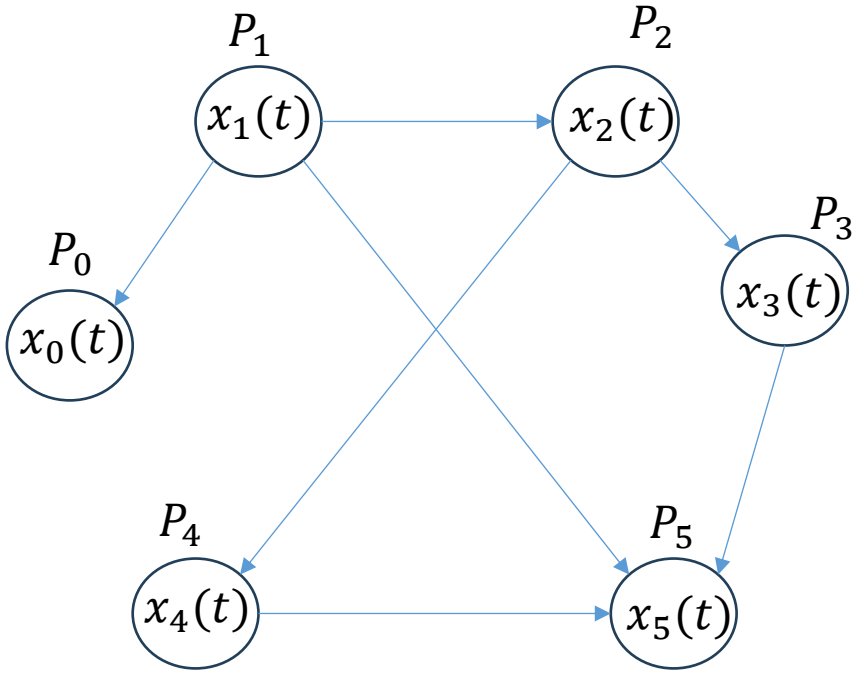
- Distributed **Computing** Systems
  - Folosite pentru calcul de înaltă performanță
  - Sisteme Cluster - Cloud

- Distributed **Information** Systems
  - Integrare funcții de business
  - Procesare tranzacții

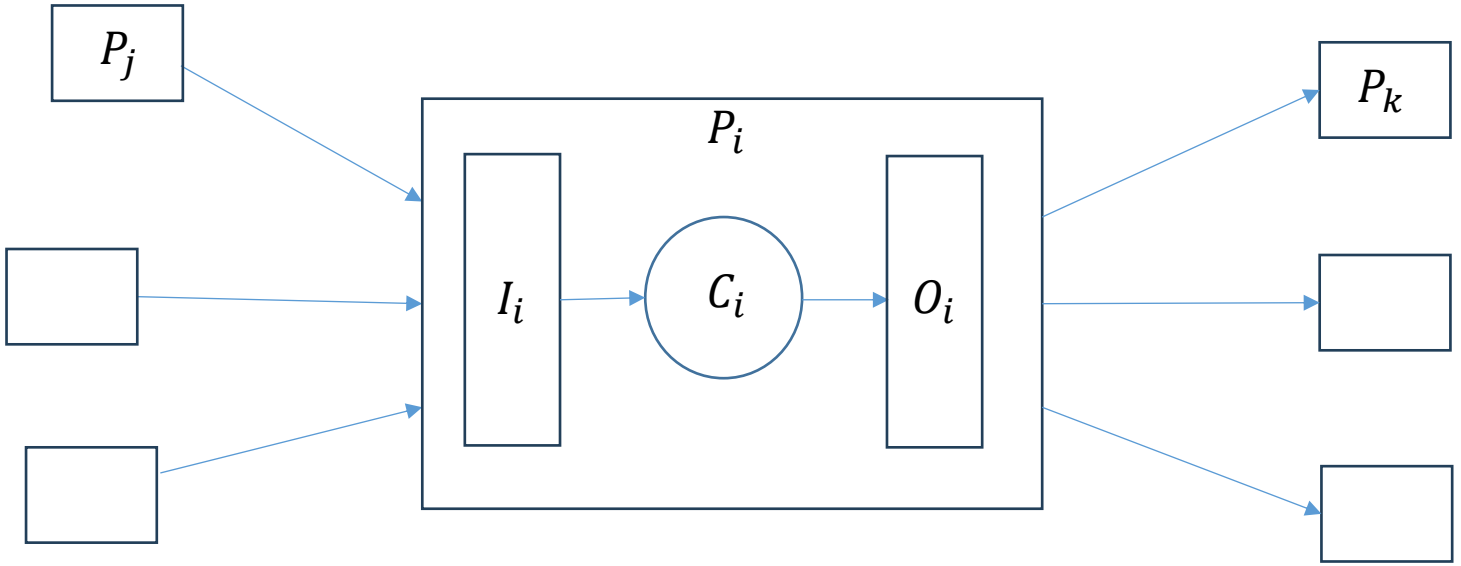
Model SD

- Noduri: $P_0, P_1, \dots, P_{n-1}$
- Rețea de comunicație $G = (V, E)$
  - $V = \{P_0, P_1, \dots, P_{n-1}\}$
  - $(i, j) \in E$ dacă există muchie între nodurile P_i și P_j
- Starea nodului P_i se exprima $x_i: \mathbb{N} \to \mathcal{D}$

- Stare nod P_i la momentul de timp t: $x_i(t)$
  - Temperatură
  - Locație-Viteză
  - Vot-Opinie

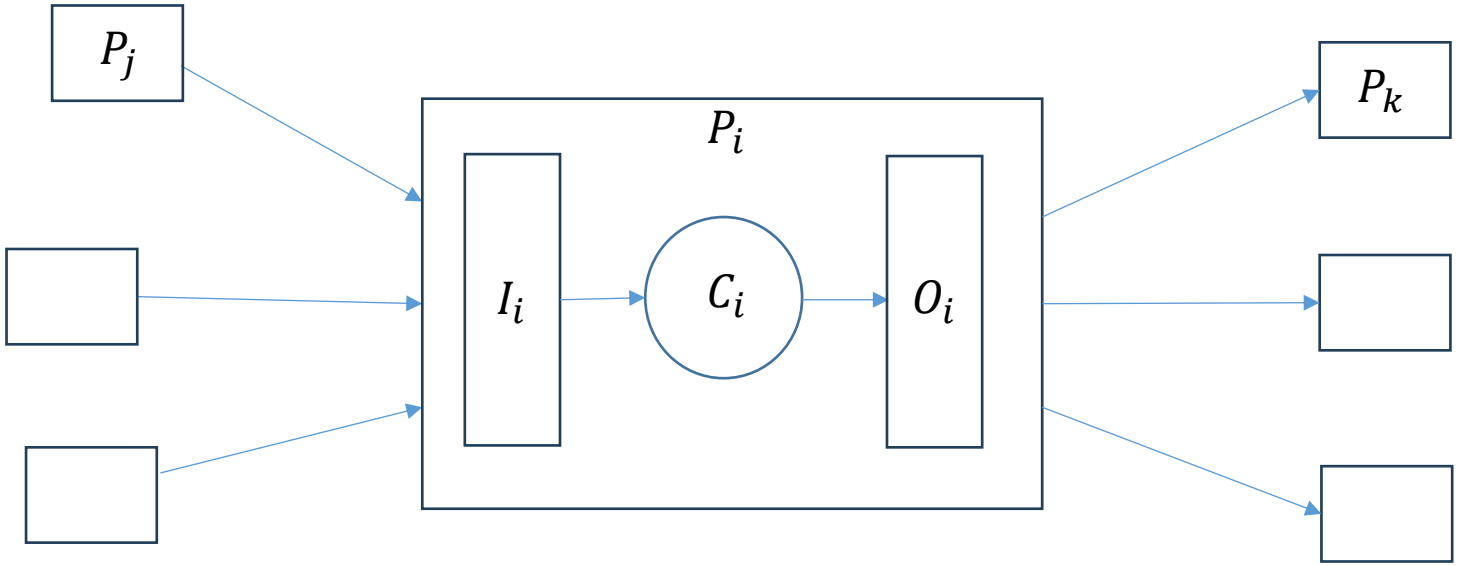


Model Proces



- Un proces primește informație la intrare (mesaje de la vecini), stocată în I_i
- Calculează noua stare pe baza info de input; în plus, depune în O_i mesaje pentru transmitere
- Transmite mesajele din O_i către vecinii de ieșire
- Reprezentăm transferul de mesaje ca evenimente separate reușite/eșuate

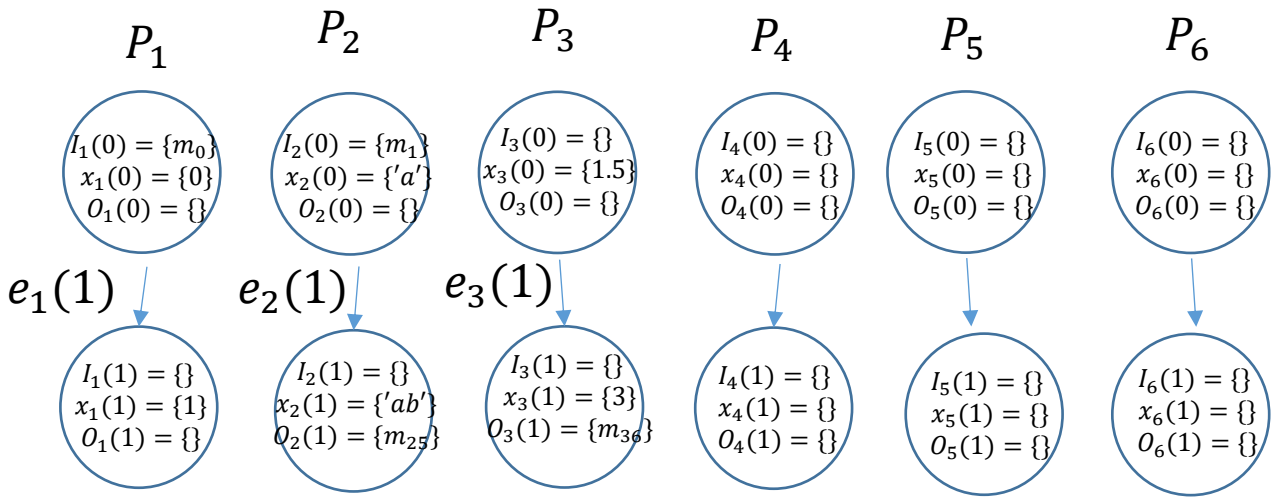
Model Proces

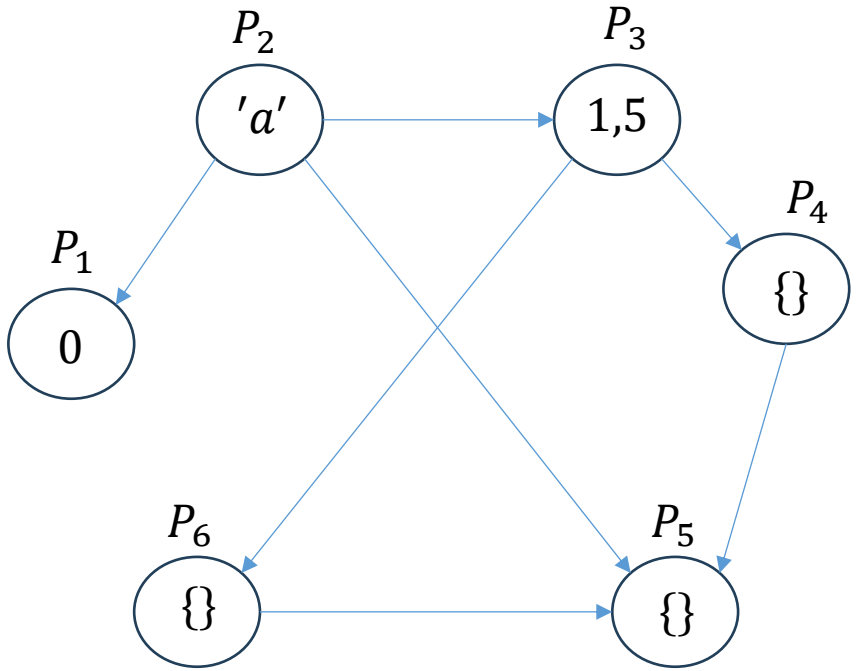


Mai formal:

- $I_i(t), O_i(t), x_i(t)$ buffer-ele de intrare și ieșire, și starea la momentul t
- Nodul i primește mesajul m la intrare:$I_i(t+1) = I_i(t)/\{m\}$
- Calculează noua stare pe baza intrării: $\mathrm{f}_i(x_i(t), m) \rightarrow (x_i(t+1), \{m_1, \ldots, m_k\})$
- Transmite mesajele de ieșire: $O(t+1) = O_i(t) \cup \{m_1, \ldots, m_k\}$
- Considerăm I/O evenimente separate în rețele supuse la defecte

Modelul principal în SD sincron

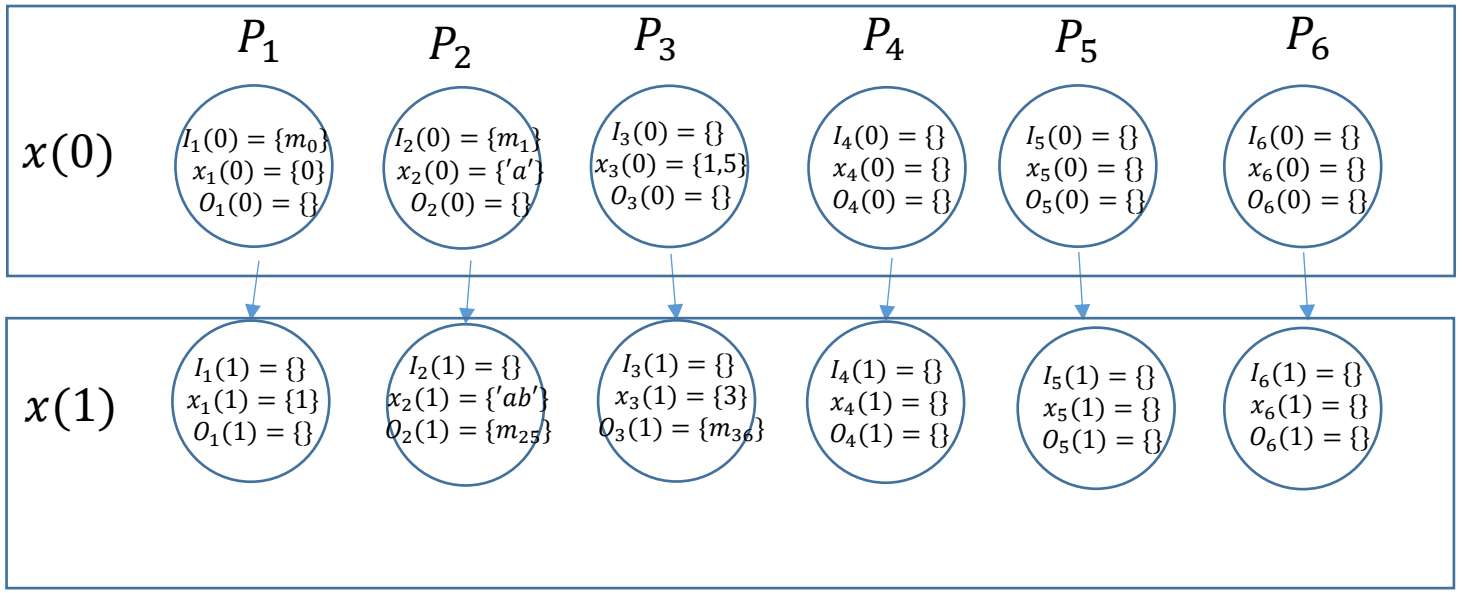


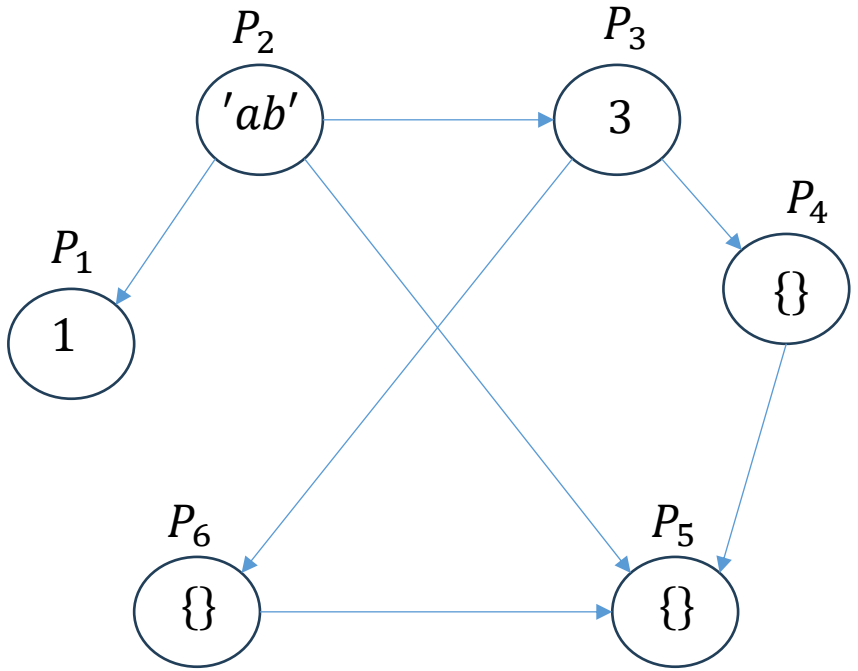


Evenimente posibile:

- Operații locale pe baza $I_i(t)$ și $x_i(t)$: e.g. calcule numerice
- Evenimente de livrare de mesaje

Modelul principal în SD sincron





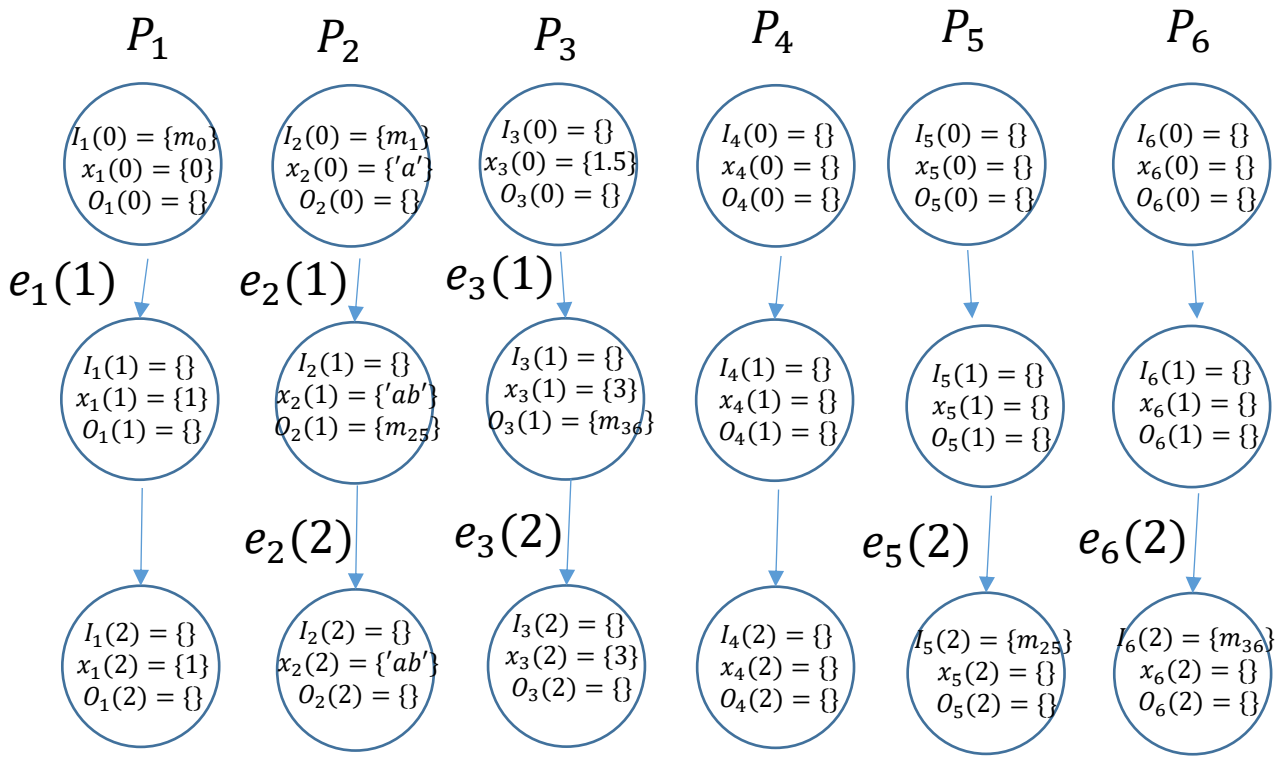
Starea globală a sistemului la momentul t :

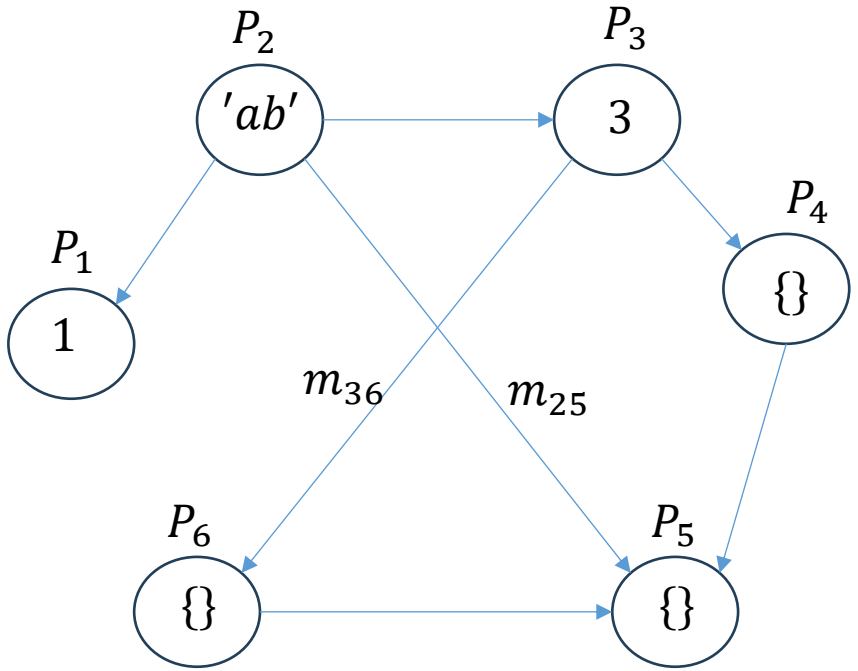
$$x(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \\ x_5(t) \\ x_6(t) \end{bmatrix}$$

Starea inițială a sistemului:

$$x(0) = \begin{bmatrix} 0 \\ a \\ 1,5 \\ NULL \\ NULL \\ NULL \end{bmatrix} \Rightarrow \qquad x(1) = \begin{bmatrix} 1 \\ ab \\ 3 \\ NULL \\ NULL \\ NULL \end{bmatrix}$$

Modelul principal in SD sincron





Modelul principal in SD sincron

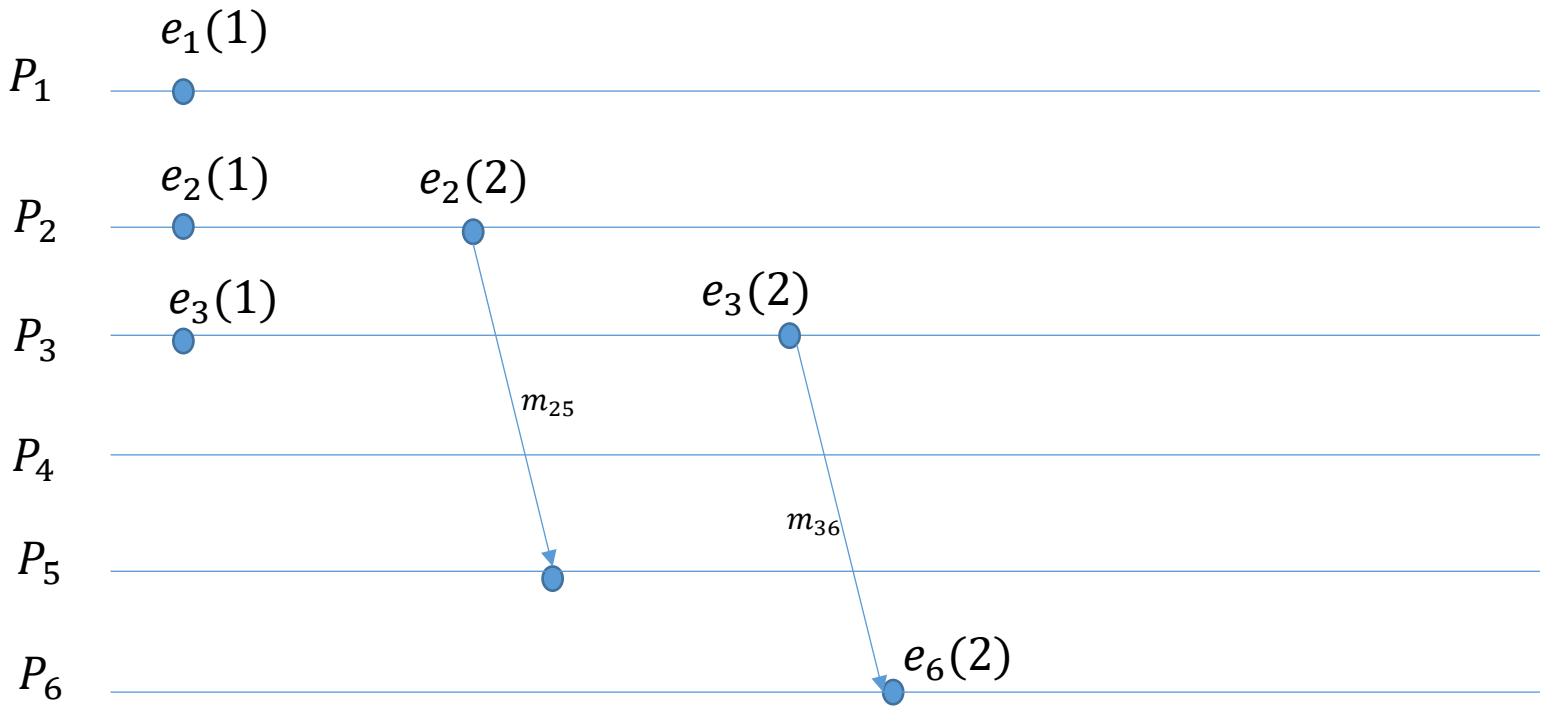


Diagrama spatiu-timp:

- Fiecare proces are propria vedere locală asupra evenimentelor
- Exprimă precedența cauzală între evenimente

Executie - traiectorie

Rețele supuse la defecte: posibile pierderi pe comunicația de mesaje (*packets loss*) sau defecte pe noduri (*crash-faults*). Procesele pornesc din starea inițială $x(0)$.

- O traiectorie/execuție: un șir (in)finit

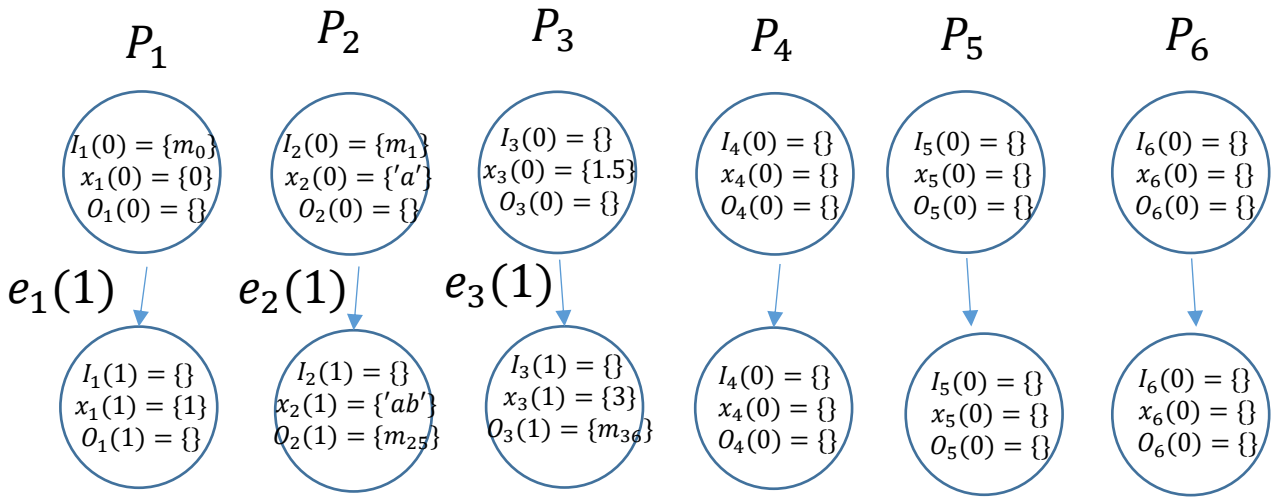
$$x(0), e(1), x(1), e(2), x(2), \ldots \ldots$$

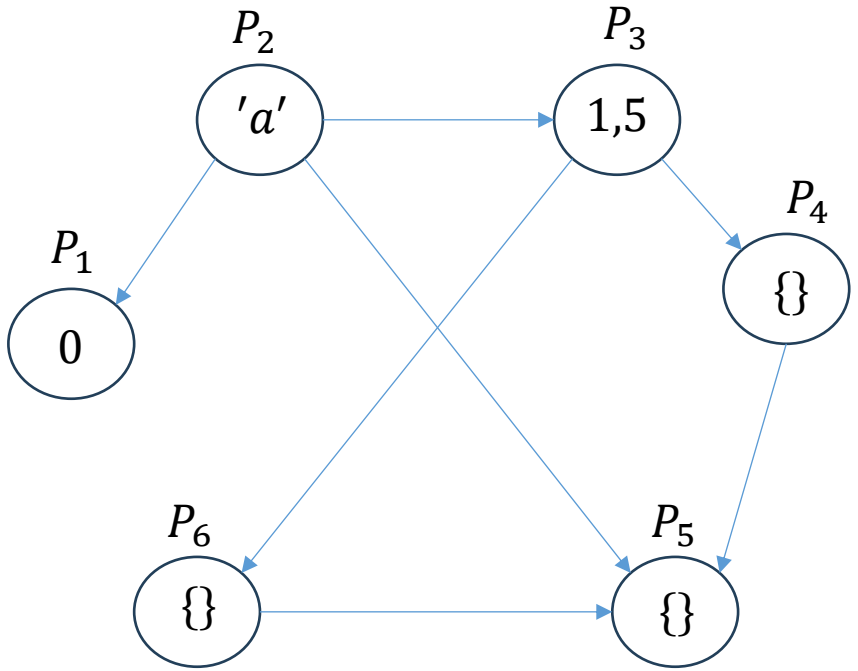
Rețele sigure: nu se iau în calcul pierderi de pachete sau defecte

- O traiectorie/execuție: un șir (in)finit

$$x(0), x(1), x(2), \ldots \ldots$$

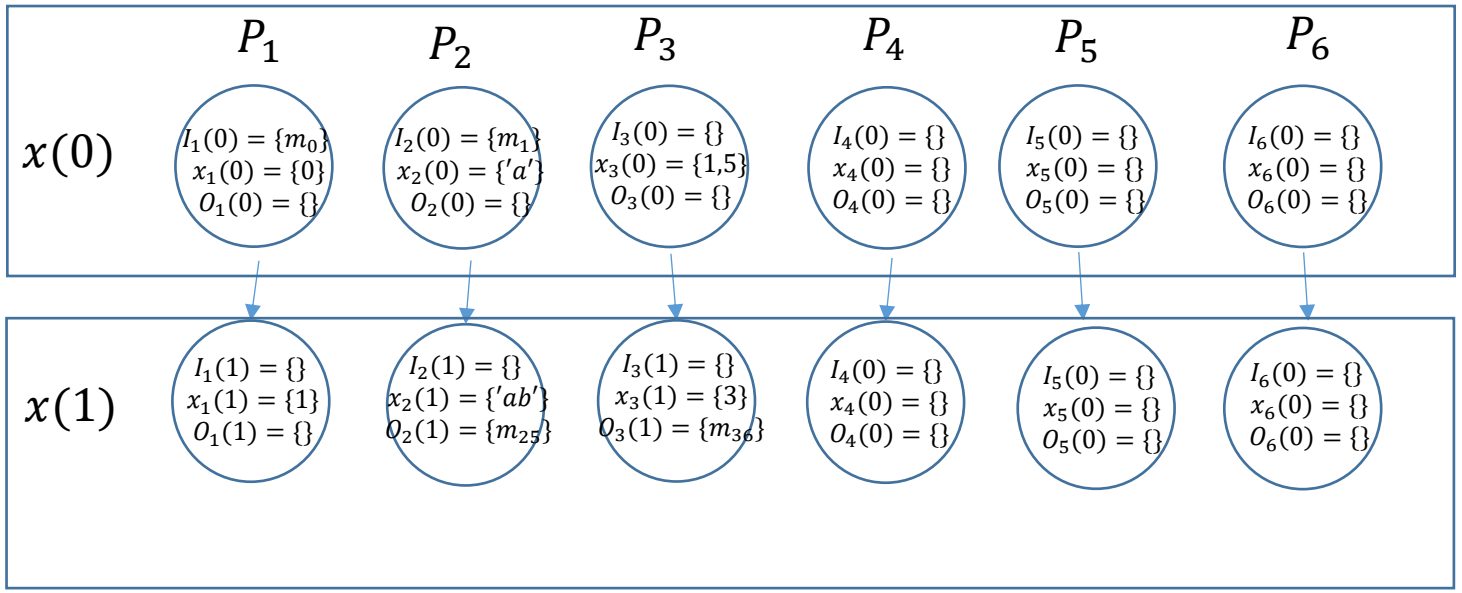
Model in SD asincron

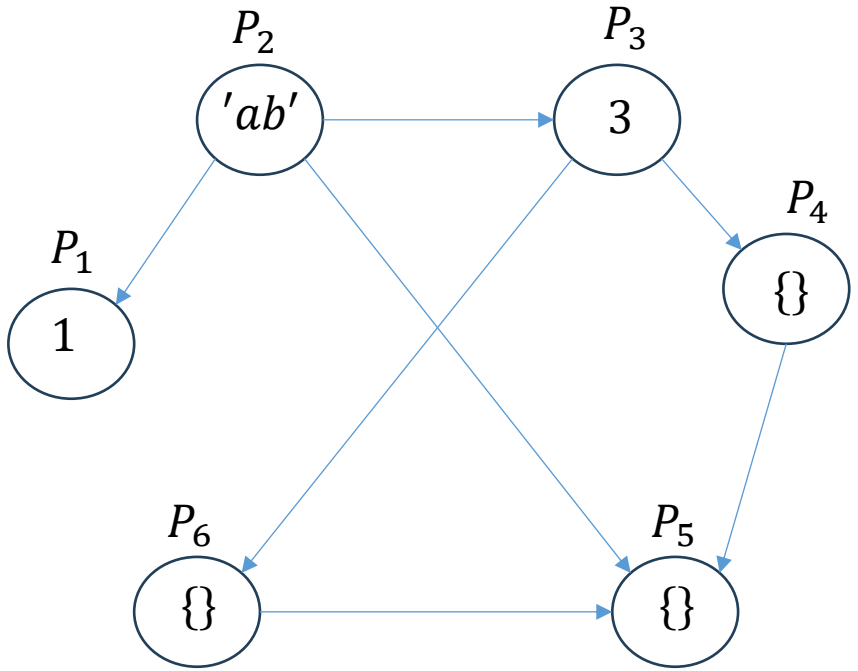




In contextual asincron, P_i are propriul ceas local, cu increment independent față de celelalte noduri.

Modelul principal în SD sincron





Starea globală a sistemului la momentul t :

$$x(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \\ x_5(t) \\ x_6(t) \end{bmatrix}$$

Starea inițială a sistemului:

$$x(0) = \begin{bmatrix} 0 \\ a \\ 1{,}5 \\ NULL \\ NULL \\ NULL \end{bmatrix} \Rightarrow \qquad x(1) = \begin{bmatrix} 1 \\ ab \\ 3 \\ NULL \\ NULL \\ NULL \end{bmatrix}$$

Modelul principal în SD sincron