

# Színezett Petri hálók

dr. Bartha Tamás

BME Méréstechnika és Információs Rendszerek Tanszék

# Bevezetés

- Mik a színezett Petri hálók?
  - “A színezett Petri hálók olyan modellek, amik a *grafikus reprezentációt* ötvözik a *jól definiált (matematikai) szemantikával*, így lehetővé téve a *formális analízist*.”
  - A színezetlen hálók kiterjesztései **rugalmas adatszerkezetekkel és adatmanipulációs nyelvvel**
  - CPN háló diagram = háló struktúra + deklarációk + háló jelölések, kifejezések + inicializáló kifejezések

# Színezetlen és színezett Petri hálók összehasonlítása

## Színezetlen (P-T) Petri hálók

- színezetlen tokenek
- tokenek halmaza (számosság)
- token manipuláció
- kezdeti jelölés
- tiltó élek
- élsúlyok
- tranzíció engedélyezése
- konfliktus különböző engedélyezett tranzíciók között
- *~ assembly nyelv*

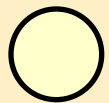
## Színezett Petri hálók

- színes tokenek
- tokenek multihalmaza
- adat manipuláció
- inicializáló kifejezések
- őrfeltételek
- élkifejezések (változókkal)
- lekötés engedélyezése
- konfliktus ugyanazon tranzíció engedélyezett lekötései között
- *~ magas szintű programnyelv*

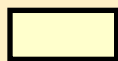
# Színezett Petri hálók felépítése

# Színezett Petri háló alkotóelemei

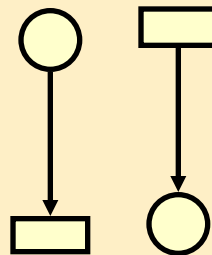
- Háló struktúra:
  - megjeleníti a rendszer vezérlési-/adatfolyam struktúráját
  - helyek, tranzíciók, (tiltó)élek
- Deklarációk:
  - definiálják: adatstruktúrákat + felhasznált függvényeket
  - deklarációk, jelölések, kódrészletek CPN ML-ben
  - színsztályok, változók, élkifejezések



Helyek



Tranzíciók



Élek

③  $3(q,0)$

Tokenek

# Színezett Petri háló alkotóelemei

- Jelölések, elnevezések:
  - megadják a háló szintaktikai és adatmanipulációs elemeit
  - nevek, színosztályok, bemenő/kimenő élkifejezések, örkifejezések, aktuális állapot
- Inicializációs kifejezések:
  - Megadják a modell kezdőállapotát
  - konstansok, kezdőállapot

```

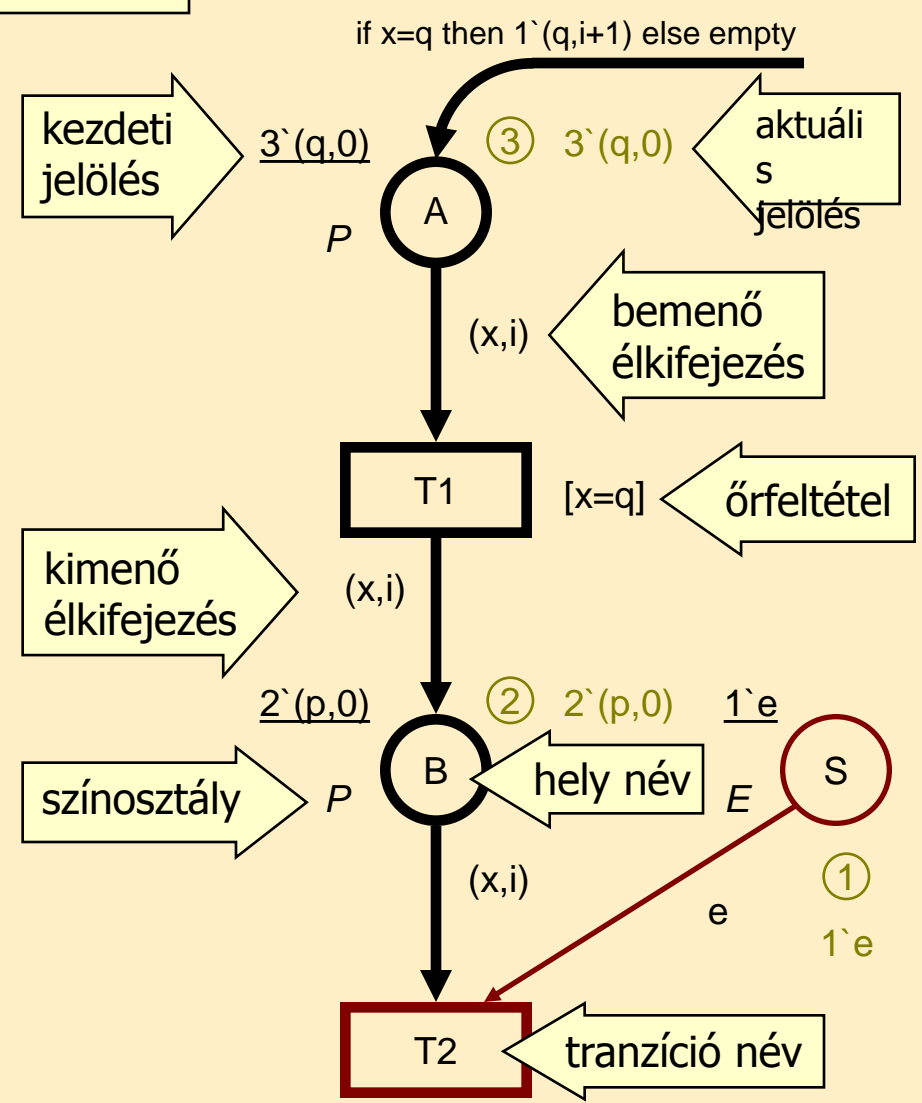
color U = with p | q;
color I = int;
color P = product U * I;
color E = with e;
var x : U;
var i : I;

```

← deklarációs mező

• CPN háló alkotóelemei:

- Helyek
  - név
  - színosztály
  - kezdő tokeneloszlás
  - aktuális tokeneloszlás
- Tranzíciók
  - név
  - őrfeltétel
- Élek
  - élkifejezések



# Multihalmazok

- Multihalmaz: azonos elemből több példány is lehet benne
  - leképezés:  $Bag(A)$ ,  $a \in [A \rightarrow \square]$
  - formálisan:  $a = \sum_{x \in A} a(x) \cdot x$  más jelölés (CPN):  $a = \sum_{x \in A} a(x)'x$
- Műveletek multihalmazokkal:
  - összehasonlítás:  $a_2 \neq a_1$  ha  $\exists x \in A, a_2(x) \neq a_1(x)$   
 $a_2 \leq a_1$  ha  $\forall x \in A, a_2(x) \leq a_1(x)$
  - számosság:  $|a| = \sum_{x \in A} a(x)$
  - összegzés:  $a_1 + a_2 = \sum_{x \in A} (a_1(x) + a_2(x)) \cdot x$
  - különbség:  $a_1 - a_2 = \sum_{x \in A} (a_1(x) - a_2(x)) \cdot x$  feltéve, hogy  $a_2 \leq a_1$
  - szorzás skalárral:  $n \cdot a = \sum_{x \in A} (n \cdot a(x)) \cdot x$



# Műveletek multihalmazokkal

Összeadás:  $a_1 + a_2$



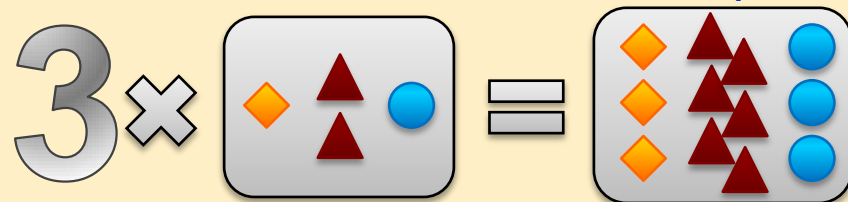
Összehasonlítás:  $a_1 \leq a_2, a_1 \neq a_2$



Számosság:  $|a_1|$



Szorítás skalárral:  $n \cdot a_1$



Kivonás:  $a_1 - a_2$  (csak ha  $a_2 \leq a_1$ )



# Multihalmazok (folyt.)

- Unió, multihalmazok egyesítése:  $a_1 \cup a_2 \cup \dots \cup a_m$ 
  - tartomány:  $A_1 \cup A_2 \cup \dots \cup A_m$
  - eleme:  $e_i \in \bigcup_1^m A_k$  ha  $\exists A_j, e_i \in A_j$
- n-esek képzése:  $\langle A_1, A_2, \dots, A_n \rangle$ 
  - tartomány:  $A_1 \times A_2 \times \dots \times A_n$
  - eleme:  $\langle e_1, e_2, \dots, e_n \rangle \in \prod_1^n A_j$  ha  $\forall e_i \in A_i$
  - általánosítás:  $\langle a_1, a_2, \dots, a_n \rangle$

# Színezett Petri hálók formális definíciója

$$\text{CPN} = (\Sigma, P, T, A, C, G, E, M_0)$$

Színhalmazok  $\Sigma = \{\sigma_1, \sigma_2, \dots, \sigma_\kappa\}$

Helyek  $P = \{p_1, p_2, \dots, p_\pi\}$

Tranzíciók  $T = \{t_1, t_2, \dots, t_\tau\}$

$$P \cap T = \emptyset$$

Élek  $A \subseteq (P \times T) \cup (T \times P)$

Színkészlet  $C : P \mapsto \Sigma$

Örfeltétel  $G : \forall t \in T, [\text{Type}(G(t)) = \mathbf{B} \wedge \text{Type}(\text{Var}(G(t))) \subseteq \Sigma]$

Élkifejezés  $E : \forall a \in A, [\text{Type}(E(a)) = C(p)_{\text{MS}} \wedge \text{Type}(\text{Var}(E(a))) \subseteq \Sigma]$

Kezdőállapot  $M_0 : \forall p \in P, [\text{Type}(M_0(p)) = C(p)_{\text{MS}}]$

# A formális definíciókban alkalmazott jelölések

- egy  $v$  változó típusa (színosztálya):  $\text{Type}(v)$
- egy  $expr$  kifejezés típusa:  $\text{Type}(expr)$
- egy  $expr$  kifejezésben szereplő változók halmaza:  $\text{Var}(expr)$
- a  $v$  változók egy halmazának lekötése:  $b(v) \in \text{Type}(v)$
- kifejezés által  $b$  lekötésre visszaadott érték:  $expr\langle b \rangle$ 
  - ahol  $v \in \text{Var}(expr)$  és  $b(v) \in \text{Type}(v)$

# Coloured Petri Nets (CPN) hálók eszközészlete

# CPN hálók: színosztályok definiálása

- Egyszerű színosztályok

- színezetlen tokenek:  
`unit`
- alapvető típusok:  
`int, bool, real, string`
- részhalmaz:  
`with 1..4;`
- felsorolás:  
`with true | false;`
- indexelés:  
`index d with 1..4;`

- Alábbi elemek

definíciójában szerepelnek:

- összetett színosztályok
- hely jelölések (kezdőállapot)
- változók, konstansok
- függvények, operátorok

# Összetett színosztályok

- Módszerek kombinált színosztályok létrehozására
  - unió képzés:  
`union s1:S + s2:S + T;`
  - n-esek képzése (Descartes szorzat):  
`product P * Q * R;`
  - rekord (címkézett n-esek):  
`record p:P * q:Q * r:R;`
  - lista:  
`list int with 2..6;`

# További CPN háló elemek: változók

- változók

Tokenek szimbolikus nevei

- változódeklaráció:

```
var proc : P;
```

- konstansok

Áttekinthetőbb, kezelhetőbb

- konstansdeklaráció:

```
val n = 10;
```

```
val d1 = d(1) : D;
```

- az alábbi kifejezésekben:

- élkifejezések
- őrfeltételek

- alábbi deklarációkban:

- színosztályok
- függvények, operátorok
- élkifejezések, őrfeltételek, inicializáló kifejezések



# További CPN háló elemek: függvények

- függvények
  - mellékhatás-mentes SML nyelvű függvények
- műveletek, operátorok
  - infix jelölésrendszer
- az alábbi kifejezésekben:
  - színosztályok
  - függvények, operátorok, konstansok
  - élkifejezések, őrfeltételek, inicializáló kifejezések

# További CPN háló elemek: kifejezések

- háló kifejezések
  - értéke: a változók egy adott lekötésével értékelhető ki
  - típusa: az összes lehetséges kiértékelési eredmény halmaza
  - példák:  
`x=q`  
`2` (x,i)`  
`if x=q then 2`i else empty`  
`Mes (s)`  
`let n=5 in n * x + 2 end`
- felhasználásuk:
  - élkifejezések, őrfeltételek, inicializáló kifejezések

# Színezett Petri háló példa: elosztott adatbázis

# CPN háló példa: elosztott adatbázis

- Specifikáció:

- $n$  különböző szerver, minden szerverhez egy adatbázis példány, amit egy lokális **adatbázis menedzser** kezel

$$\text{DBM} = \{d_1, d_2, \dots, d_n\}, n \geq 3$$

- lokális frissítés lehetséges, de a menedzsernek minden más menedzsert egy **üzenettel** értesítenie kell a frissítésről
- üzenet fejléc: **küldő** és **fogadó** (címezett)

$$\text{MES} = \{(s,r) \mid s,r \in \text{DBM} \wedge s \neq r\}, \text{Mes}(s) = \sum_{r \in \text{DBM} - \{s\}} 1 \cdot (s,r)$$

- rendszer állapota: *Active, Passive*
- adatbázis menedzserek állapota:  
*Inactive, Waiting* (nyugtázásra vár), *Performing*
- üzenetek állapota: *Unused, Sent, Received, Acknowledged*

# Elosztott adatbáziskezelő rendszer: deklarációk

## Deklarációs mező

```
val n = 4;  
color DBM = index d with 1..n declare ms;  
color PR = product DBM * DBM declare mult;  
fun diff(x,y) = (x<>y);  
color MES = subset PR by diff declare ms;  
color E = with e;  
fun Mes(s) = mult'PR(1`s, DBM--1`s)  
var s, r : DBM;
```

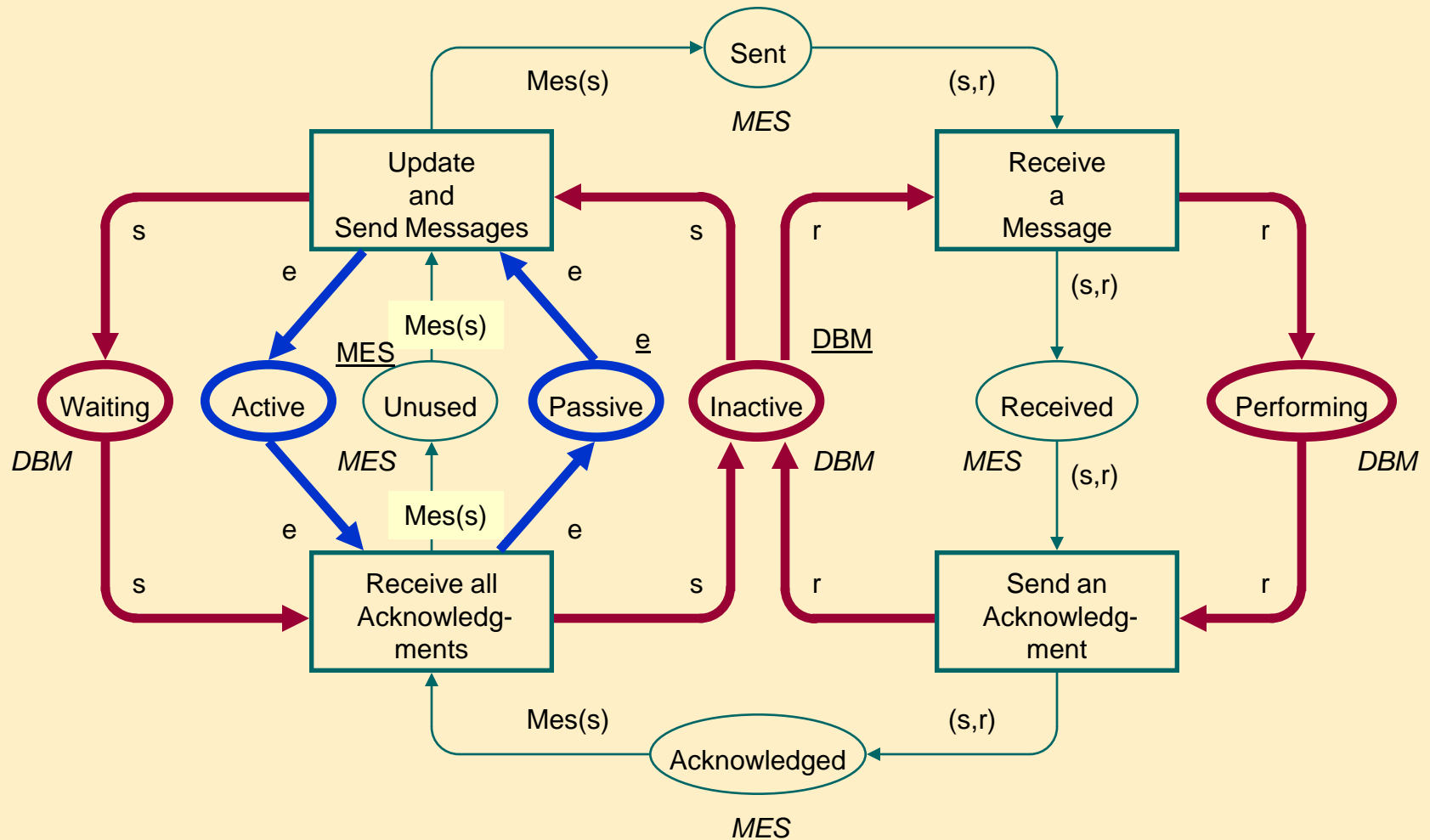
## Jelentése:

$$\text{DBM} = \{d_1, d_2, \dots, d_n\}$$

$$\text{MES} = \{(s, r) \mid s, r \in \text{DBM} \wedge s \neq r\}$$

$$\text{Mes}(s) = \sum_{r \in \text{DBM} - \{s\}} 1'(s, r)$$

# Elosztott adatbáziskezelő rendszer CP háló modellje



# Színezett Petri hálók működése

# Működés: engedélyezés és tüzelés

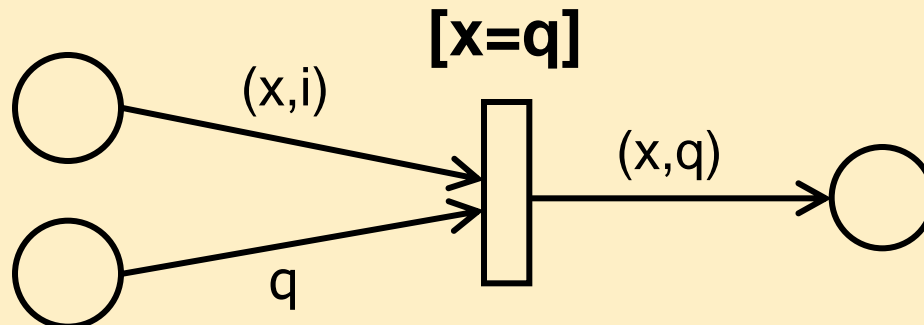
Engedélyezés fogalma megváltozik:

- **Lezárt** élkifejezés: nem tartalmaz változókat
  - **Változók**
    - rendelkeznek típussal:  $\text{Type}(v)$
    - értékük a multihalmaz egy eleme
- **Nyílt** élkifejezés: változókat le kell kötni egy értékkel
  - **lekötés**: egy konkrét értékhozzárendelés minden változóhoz
    - adott lekötéssel az élkifejezés kiértékelhető
  - **rendelkezik típussal**:  $\text{Type}(expr) = C(p)_{MS}$ 
    - az értékül kapott színosztály típusa
  - **kifejezésben szereplő változók halmaza**:  $\text{Var}(expr)$



# Őrfeltétel

- Őrfeltételek:
  - Tranzíciókhoz rendelt kifejezések
    - multihalmazok felett értelmezett kifejezések (lásd élkifejezés)
    - DE boolean visszatérési értékkel! (élkifejezés: multihalmaz v.é.)
  - „Igaz” kiértékelési érték esetén engedélyezik a tranzíciót
    - „szűrik” az engedélyezett lekötéseket
  - Jelölés: szögletes zárójelekben megadott kifejezés a tranzíció mellett



# Engedélyezettség színezett Petri hálóokban

- Tranzíció lekötése

- Érvényes lekötés:  $\forall v \in \text{Var}(t): b(v) \in \text{Type}(v) \wedge G(t)\langle b \rangle$

$$\text{Var}(t) = \{v \mid v \in \text{Var}(G(t)) \vee \exists a \in A(t) : v \in \text{Var}(E(a))\}$$

- $G(t)\langle b \rangle$  az őrfeltétel

- Az összes érvényes lekötés halmaza:  $B(t)$

- Egy érvényes lekötés engedélyezett, ha

- A bemenő helyeken „van elég színezett token”:

$$\forall p \in \bullet t : E^-(p, t)\langle b \rangle \leq M(p) \wedge E^h(p, t)\langle b \rangle > M(p)$$

# Tüzelés színezett Petri hálóokban

- Egy engedélyezett tranzíció **tüzelhet**, ha

- Magasabb prioritású tranzíció nem engedélyezett:

$$\forall t' : \pi(t') > \pi(t), \exists p \in \bullet t' : E^-(p, t') \langle b' \rangle > M(p) \vee E^h(p, t') \langle b' \rangle \leq M(p)$$

- Magasabb prioritású tranzíció őrfeltétele nem teljesül

$$\neg G(t') \langle b' \rangle$$

# Tüzelés színezett Petri hálóokban

- Tüzelés menete:

- Engedélyezett lekötések keresése

- meghatározzák a bemenő élkifejezések, őrfeltételek

- Tranzíció engedélyezett adott lekötéssel → tüzelés

- Színezett tokenek elvétele a bemenő helyekről

- Színezett tokenek kitétele a kimenő helyekre

$$\forall p \in P : M'(p) = M(p) - \sum_{p \in \bullet t} E^-(p, t) \langle b \rangle + \sum_{p \in t \bullet} E^+(t, p) \langle b \rangle$$

- Ekkor  $M'$  közvetlenül elérhető  $M$ -ből:  $M \xrightarrow{(t, b)} M'$

# Lekötött és lekötetlen változók

- **Lekötött változók**

- Az értékhozzárendelést a bemenő élek határozzák meg
- Konzisztencia: változó értéke lekötésen belül azonos!
  - minden a tranzícióhoz tartozó élen: azonos név → azonos érték

- **Lekötetlen változók**

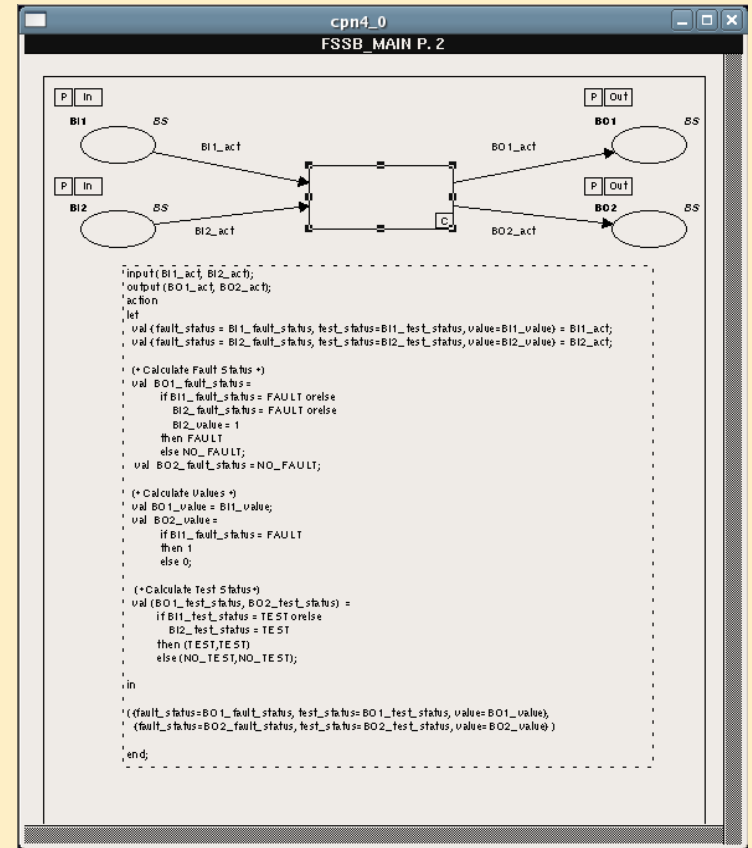
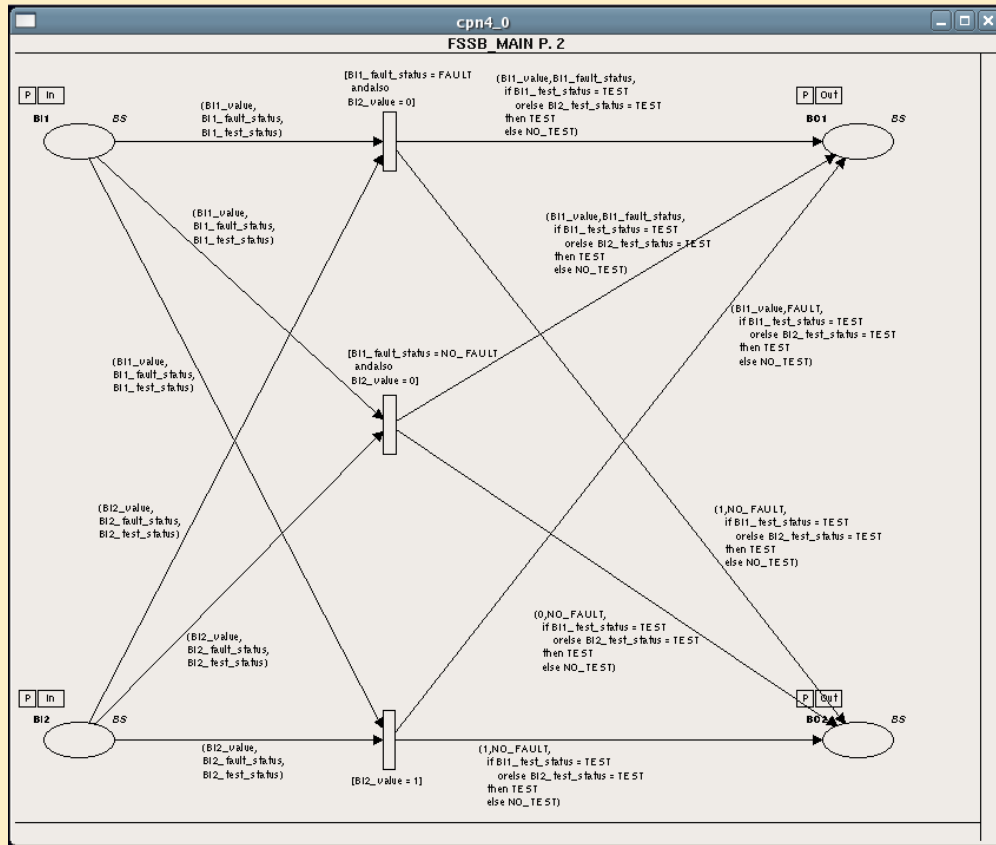
- Csak kimenő élkifejezésekben szereplő változók
- Az engedélyezés nem rendelt hozzá értéket: lekötetlen
- Tüzeléshez le kell kötni!
  - a színosztályából bármilyen értéket felvehet
  - annyi lehetséges lekötés, amennyi a színosztály számossága
  - nondeterminisztikus választás

# Színezett Petri hálók széthajtogatása

# Színezett Petri hálók felépítésének lehetőségei

- CPN hálók: strukturális és vezérlési elemek
  - tisztán strukturális információ, nincs adattartalom:
    - széthajtogatott háló
  - tisztán adat és vezérlési információ, nincs struktúra:
    - egy hely + egy tranzíció, komplex színosztályok és élkifejezések
  - kompromisszumra van szükség
    - érthető, áttekinthető CPN hálóra van szükség!

# Modellezési lehetőségek



vezérlési folyamat struktúrában kifejezve

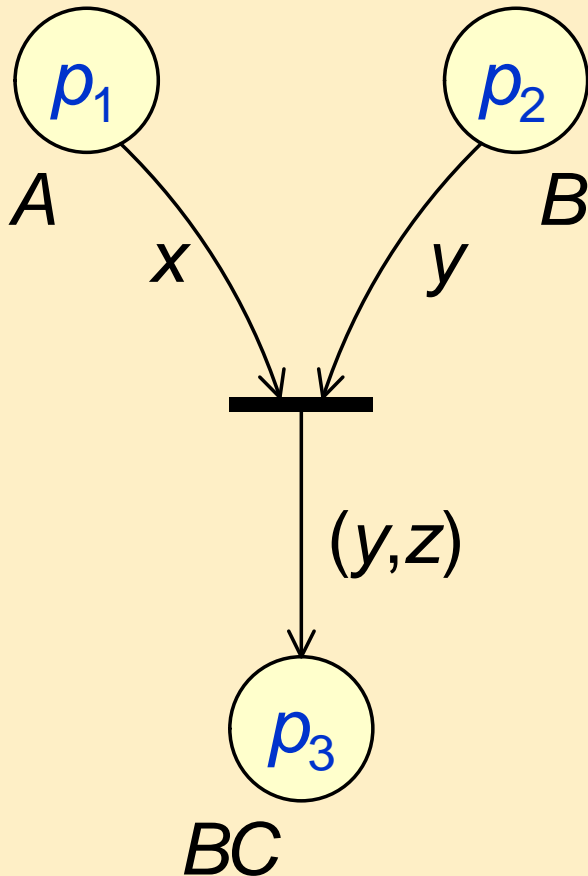
ugyanaz csak kódban  
 („összehajtogatva”)



# Széthajtogatás

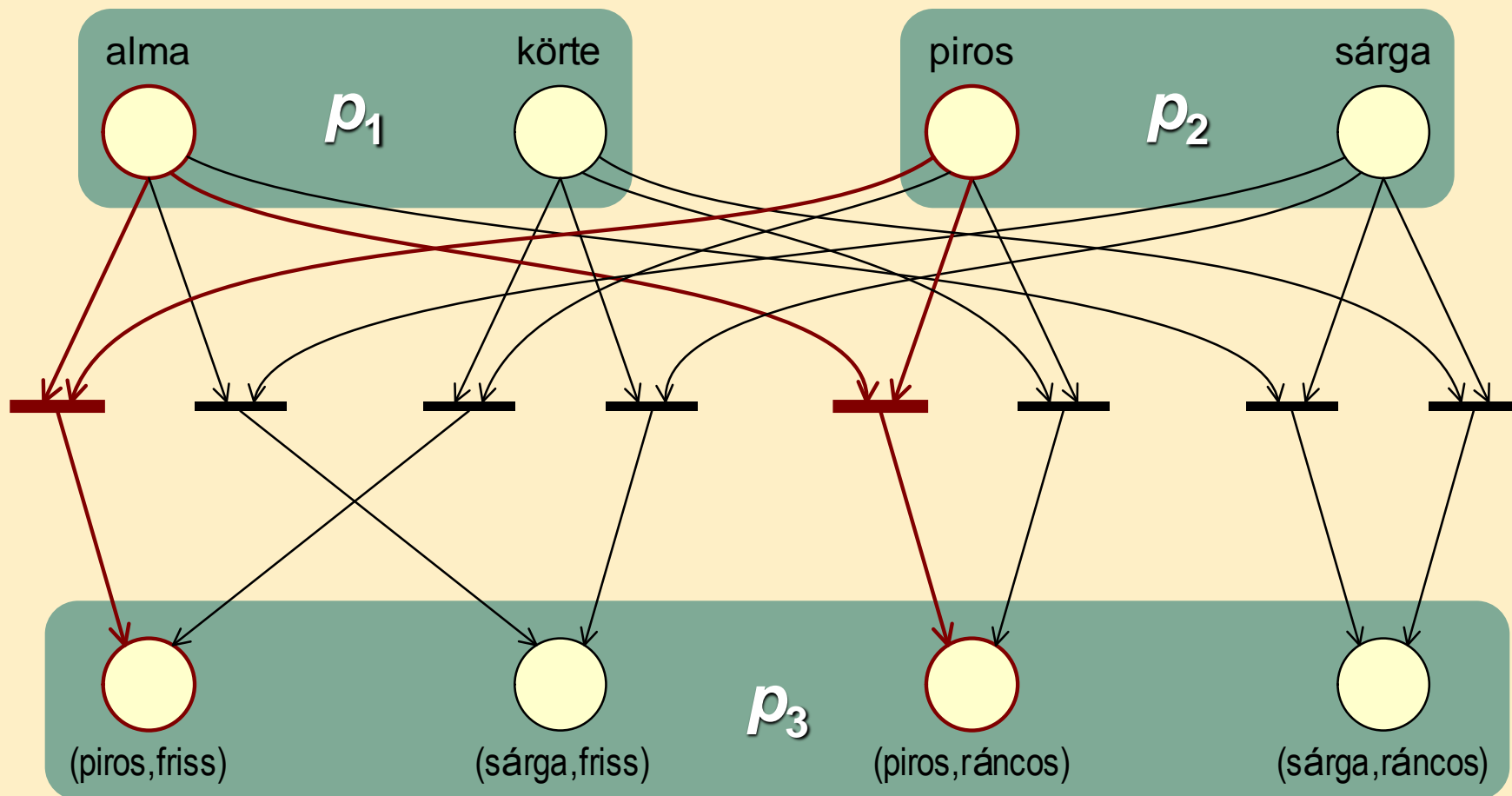
- (Prioritásos) színezett hálók modellező ereje megfelel a tiltó éllel kiegészített (prioritásos) színezetlen hálókénak
  - minden színezett hálónak megfeleltethető egy (automataelméleti értelemben → biszimuláció) ekvivalens működésű színezetlen háló
  - ekvivalens színezetlen háló: széthajtogatott háló
  - **széthajtogatás:**
    - tokenek adattartalmát struktúrában fejezzük ki
    - minden eseménynek az eredeti hálóban megfelel egy és csak egy esemény a széthajtogatott hálóban

# Egyszerű színezett háló széthajtogatása



```
color A = with alma | körte;  
color B = with piros | sárga;  
color C = with friss | ráncos;  
color BC = product B*C declare mult;  
var x: A;  
var y: B;  
var z: C;
```

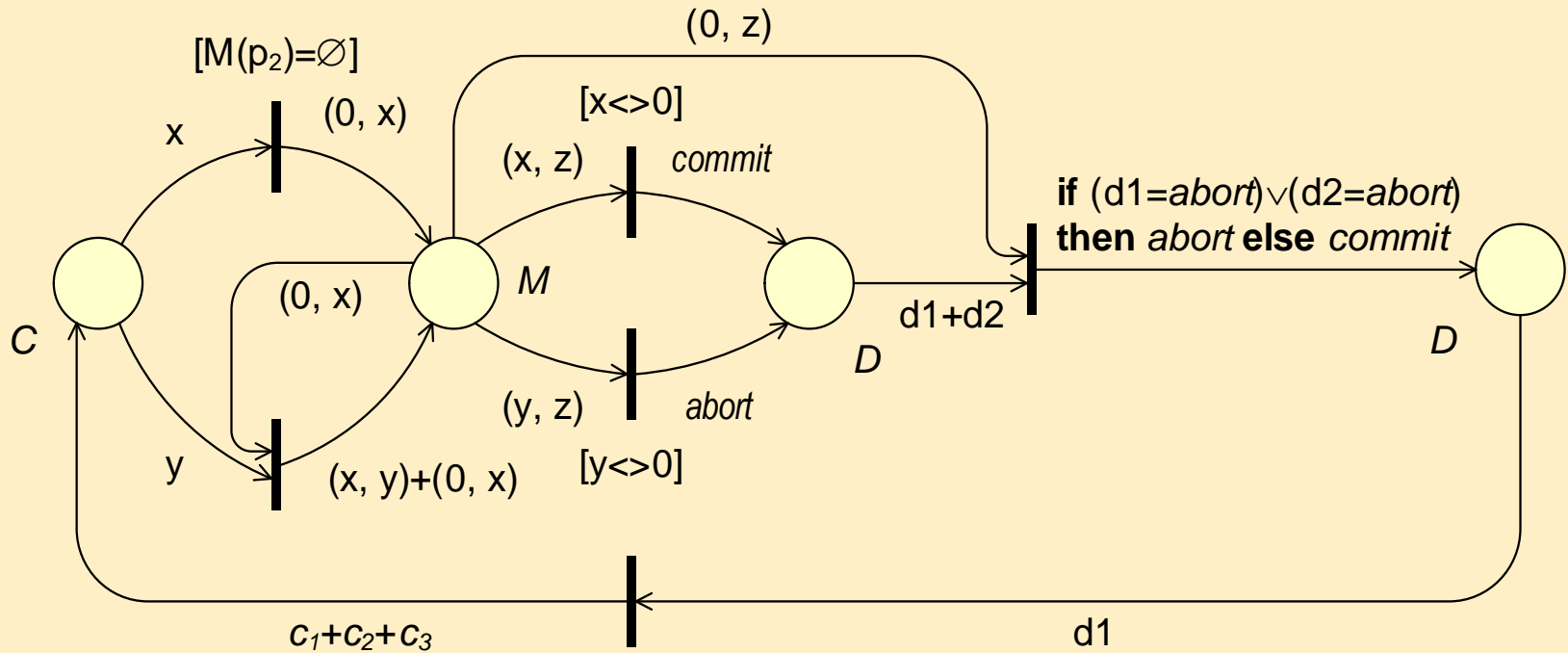
# A széthajtogatott, színezetlen modell



# Összetett példa: kétfázisú commit protokoll

- Specifikáció
  - $n$  ( $=3$ ) darab szerver egy tranzakciót le akar zárni
  - kétfázisú commit protokollt alkalmaznak
  - először egy koordinátor választanak
    - többiek: résztvevők
  - a koordinátor minden résztvevőnek küld egy kérést
  - a résztvevők commit vagy abort választ adnak
    - ha mindenki commit választ adott: koordinátor commit-ot dönt
    - ha van legalább egy abort válasz: koordinátor abort-ot dönt
    - a protokoll további részét nem modellezzük

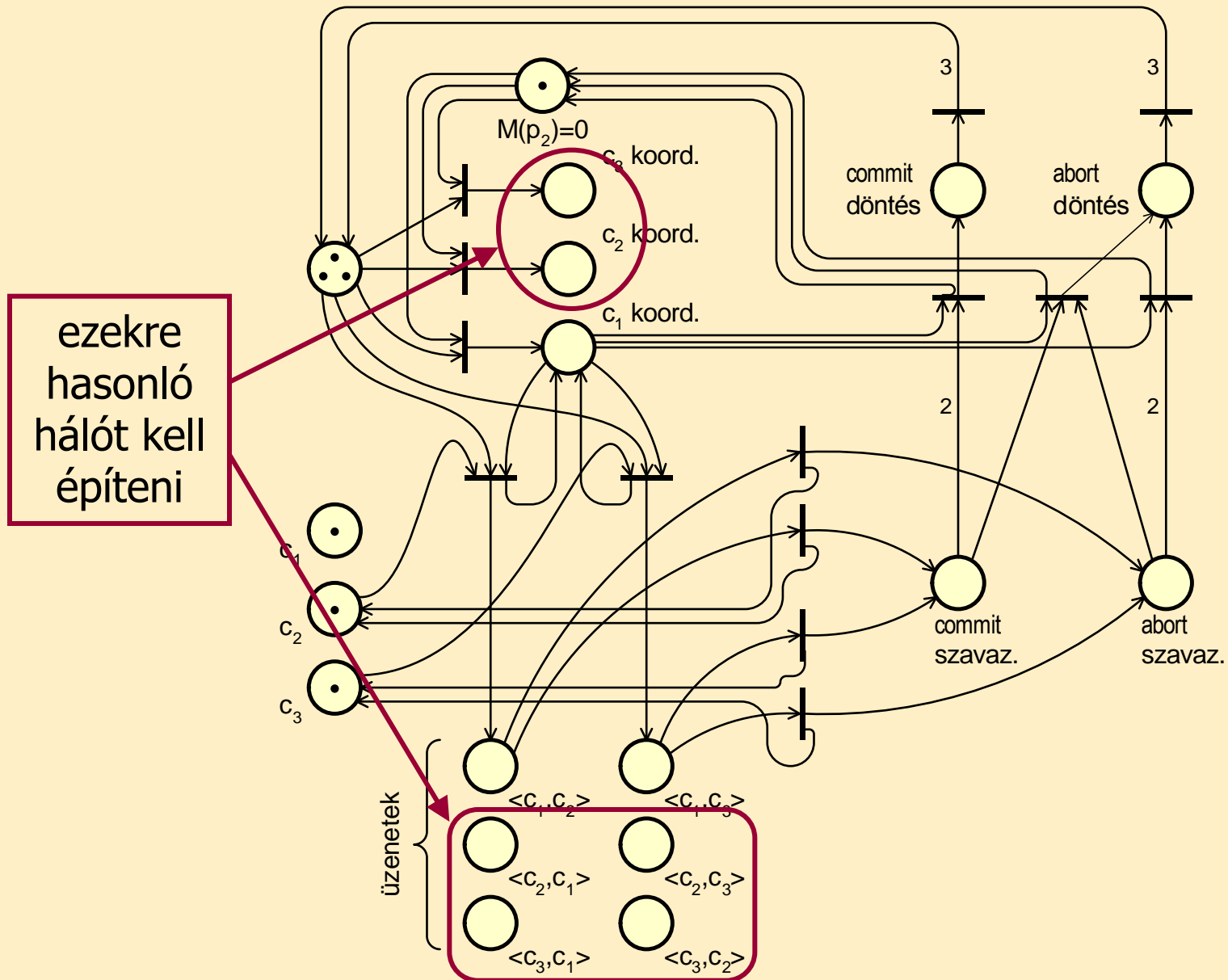
# Kétfázisú commit protokoll CP-net modellje



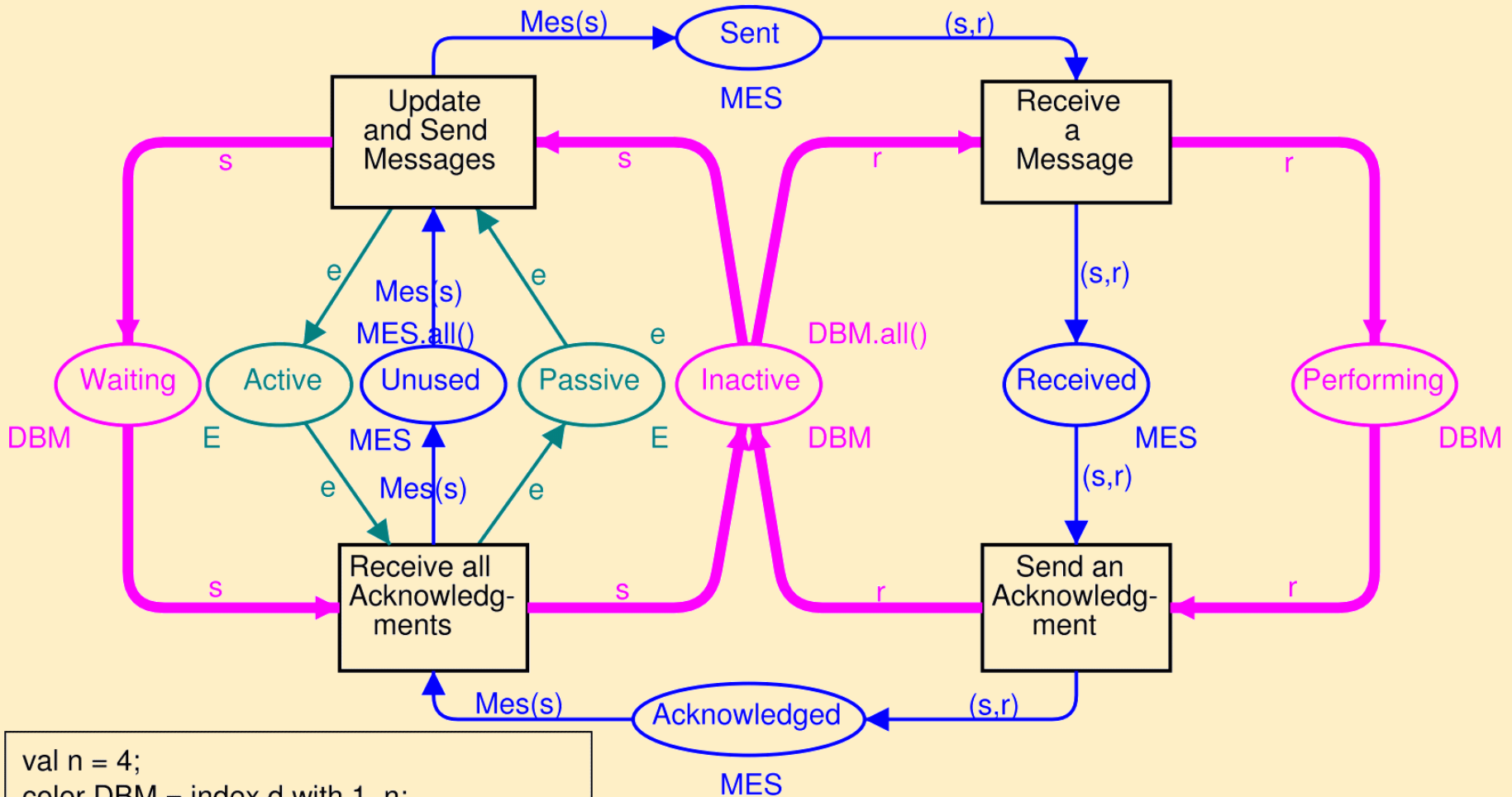
```

color C1 = index c with 1..3 declare ms;
color C2 = with 0;
color C = union C1 + C2 declare ms;
color D = with commit | abort;
color M = product C*C declare mult;
var x, y, z: C;
var d1, d2: D;
    
```

# Modell részleges széthajtogatása ( $c_1$ a koordinátor)



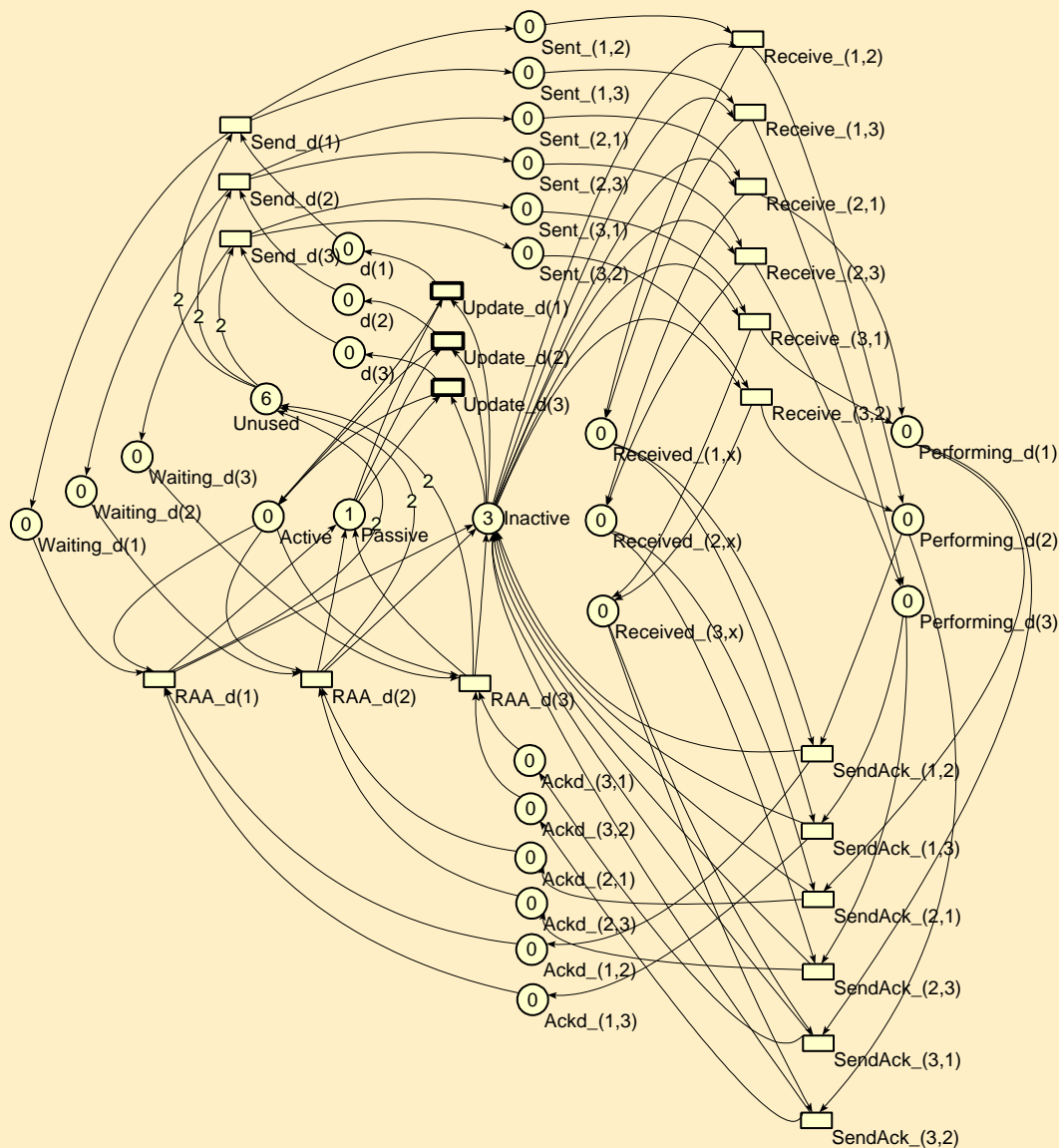
# Elosztott adatbáziskezelő rendszer CP háló modellje



```

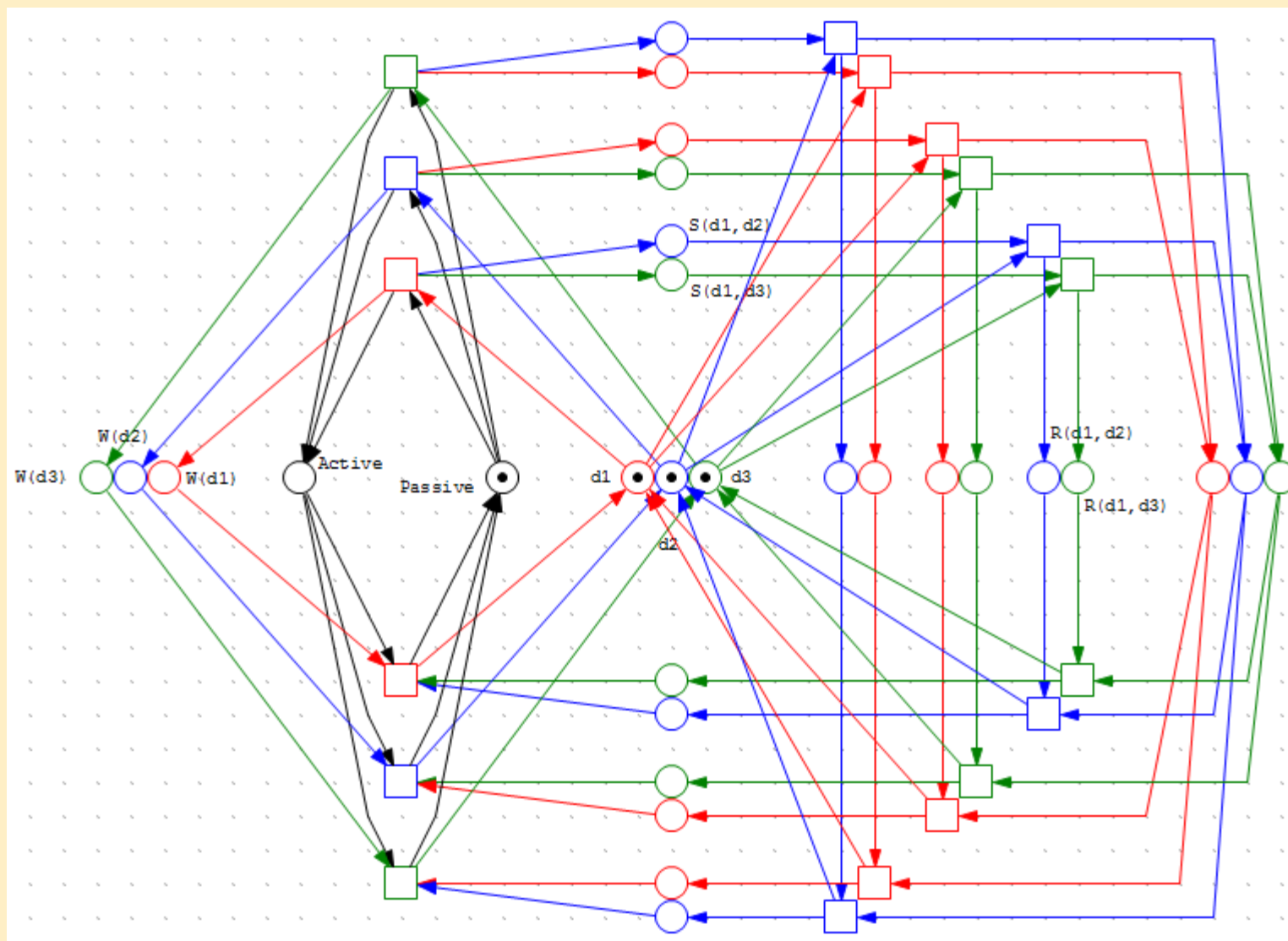
val n = 4;
color DBM = index d with 1..n;
color PR = product DBM * DBM;
fun diff(x,y) = (x<>y);
color MES = subset PR by diff;
color E = with e;
fun Mes(s) = PR.mult(1`s, DBM.all()--1`s);
var s,r: DBM;
    
```

# Az adatbázis-kezelő modell széthajtogatása





# Széthajtogatott adatbázis-kezelő háló Snoopy-ban



# Színezett Petri hálók dinamikus tulajdonságai

# Színezett Petri hálók dinamikus tulajdonságai

- A színezetlen hálóknál megismert tulajdonságok kiterjesztései multihalmazokra

- Korlátosság

Egy hely korlátos, ha a tokenek száma bármely állapotban korlátos

- $n$  egy felső integer korlát  $p$ -re, ha  $\forall M \in [M_0\rangle : |M(p)| < n$
- $m$  egy felső multihalmaz korlát  $p$ -re, ha  $\forall M \in [M_0\rangle : M(p) < m$

- Visszatérő tulajdonság

Egy visszatérő állapotba mindig lehetséges visszajutni

- $M$  egy visszatérő állapot, ha  $\forall M' \in [M_0\rangle : M \in [M'\rangle$
- $X$  egy visszatérő csoport, ha  $\forall M' \in [M_0\rangle : X \cap [M'\rangle \neq \emptyset$

# Színezett Petri hálók dinamikus tulajdonságai

- Élőség

Az élőség garantálja, hogy a lekötési elemek egy része aktív marad

– halott állapot (deadlock): egy lekötési elem sem engedélyezett

$$\forall b \in BE : \neg M[b \rangle$$

– halott tranzíció: egyik lekötése sem válhat engedélyezetté

$$\forall M' \in [M \rangle, b \in B(t) : \neg M'[b \rangle$$

– élő tranzíció: nincs olyan állapot, amelyben minden lekötése halott

$$\forall M' \in [M_0 \rangle, \exists M'' \in [M' \rangle, \exists b \in B(t) : M''[b \rangle$$

# Színezett Petri hálók dinamikus tulajdonságai

- Fair tulajdonság

Fairség megmutatja, hogy egy lekötési elem milyen gyakran tüzel

- elfogulatlan (impartial) tranzíció: végtelen sokszor tüzel

$$\forall b \in B(t), |\sigma| = \infty : OC_b(\sigma) = \infty$$

- fair tranzíció: végtelen sok engedélyezés  $\rightarrow$  végtelen sok tüzelés

$$\forall b \in B(t), |\sigma| = \infty : EN_b(\sigma) = \infty \Rightarrow OC_b(\sigma) = \infty$$

- igazságos (just) tranzíció: perzisztens engedélyezés  $\Rightarrow$  tüzelés

$$\forall b \in B(t), \forall i \geq 1 : \left[ EN_{b,i}(\sigma) \neq 0 \Rightarrow \exists k \geq i : \left[ EN_{b,k}(\sigma) = 0 \vee OC_{b,k}(\sigma) \neq 0 \right] \right]$$

- Elérhetőségi gráf

Színezetlen hálók elérhetőségi gráfjának generálásával analóg módon

- állapot ekvivalencia osztályok, erősen összekötött komponensek

# Színezett Petri hálók strukturális tulajdonságai

# T-invariáns színezett Petri hálóknban

- Tranzíció invariáns

Olyan tüzelési szekvencia, ami nem hat az állapotra

$$\forall p \in P: \underbrace{M'(p) = M(p)}_{M'(p) - M(p) = 0} - \sum_{p \in \bullet t, b \in \sigma} E^-(p, t) \langle b \rangle + \sum_{p \in t \bullet, b \in \sigma} E^+(t, p) \langle b \rangle$$

$$\forall p \in P: \sum_{p \in \bullet t, b \in \sigma} E^-(p, t) \langle b \rangle = \sum_{p \in t \bullet, b \in \sigma} E^+(t, p) \langle b \rangle$$

# P-invariáns színezett Petri hálóknban

- Hely invariáns

Alapötlet: egyenlet, ami minden (elérhető) állapotban fennáll

– súlyozott összeg:

$$W_{p_1}(M(p_1)) + W_{p_2}(M(p_2)) + \dots + W_{p_n}(M(p_n)) = m_{\text{inv}}$$

– súlyfüggvény: hely színekészletét egy közös súlyozott halmazra

– súlyozott halmazok: multihalmaz jellegűek, negatív együtthatókkal

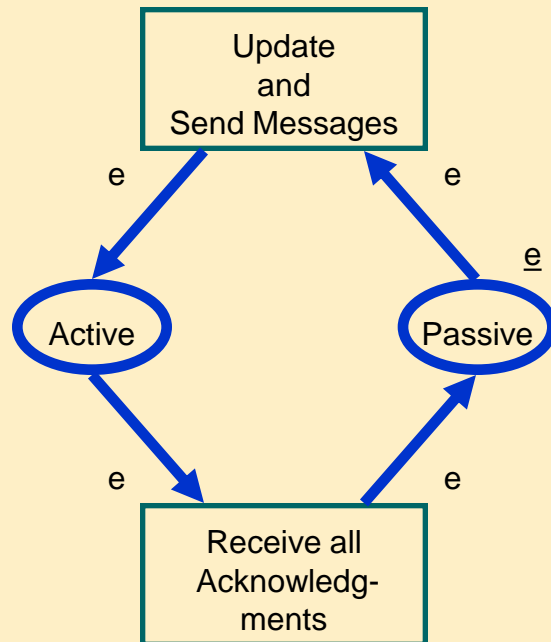
– hely-folyam:  $\forall b \in B(t) : \sum_{p \in P} W_p(E^-(p, t)\langle b \rangle) = \sum_{p \in P} W_p(E^+(t, p)\langle b \rangle)$

– P-invariáns:  $\forall M \in [M_0] : \sum_{p \in P} W_p(M(p)) = \sum_{p \in P} W_p(M_0(p))$

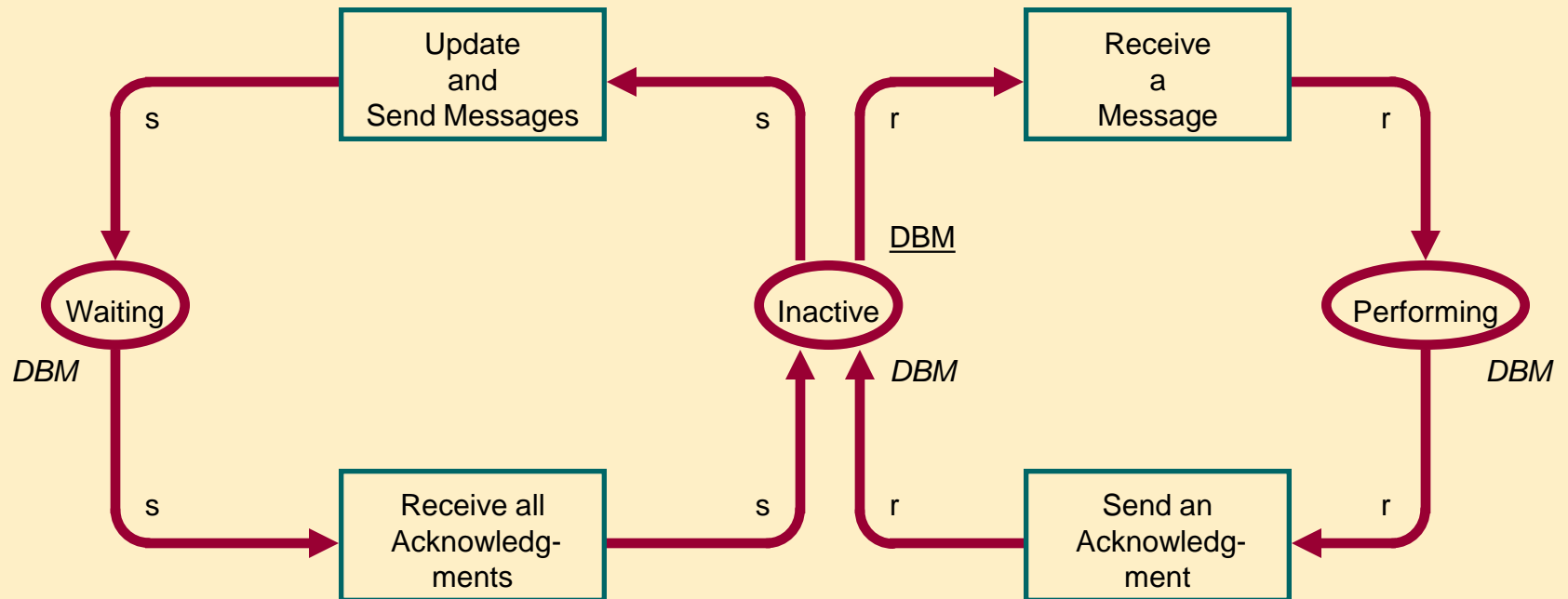
– Tétel:  $W$  egy hely-folyam  $\Leftrightarrow W$  egy P-invariánst határoz meg



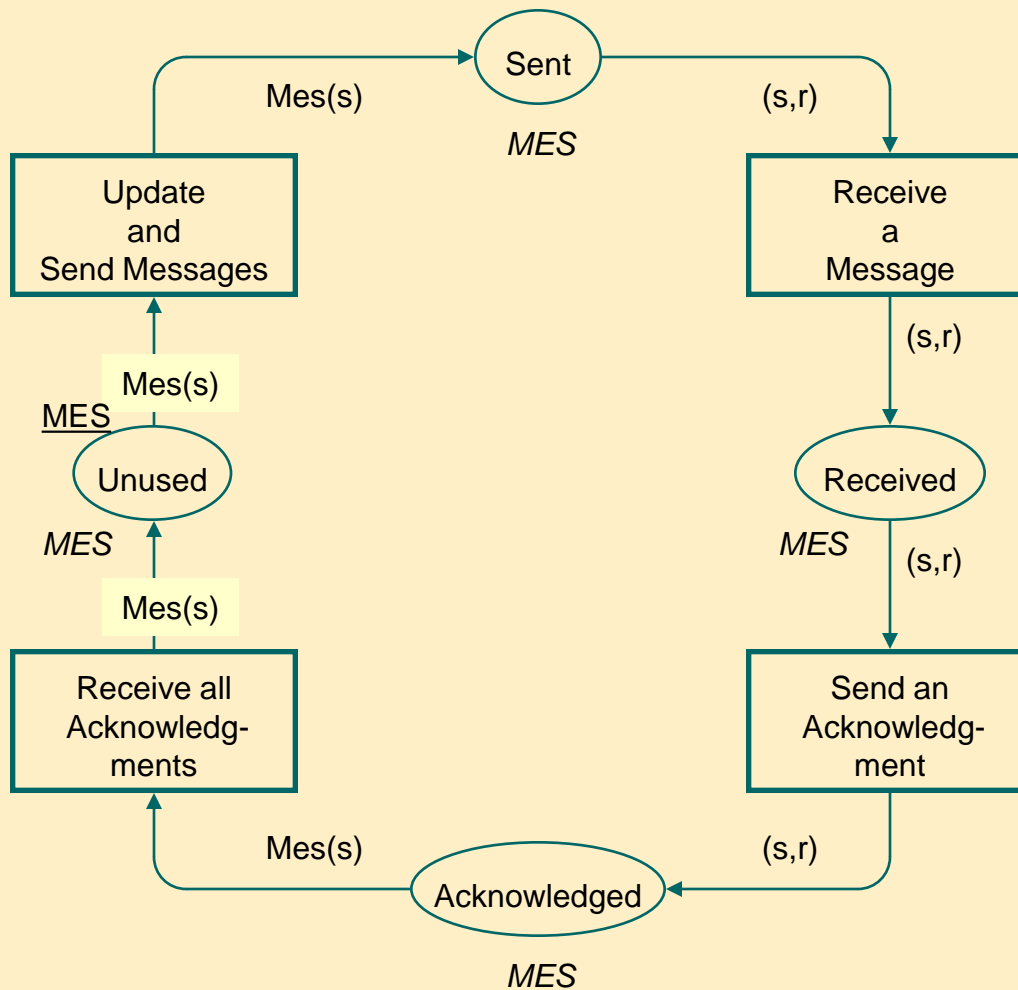
# Dekompozíció: a rendszer állapota



# Dekompozíció: adatbázis menedzserek



# Dekompozíció: üzenetvábbító alrendszer



# A modell dinamikus tulajdonságai

- Konkurencia:

Update & Send → Receive önmagával konkurens

- Kauzalitás:

Update & Send → Receive → Send Ack. ⇒ Receive Ack.

- Konfliktus:

Update & Send

- minden Mes(s) lekötési elemre engedélyezett
- de csak egyre tüzelhet

# A modell tulajdonságai

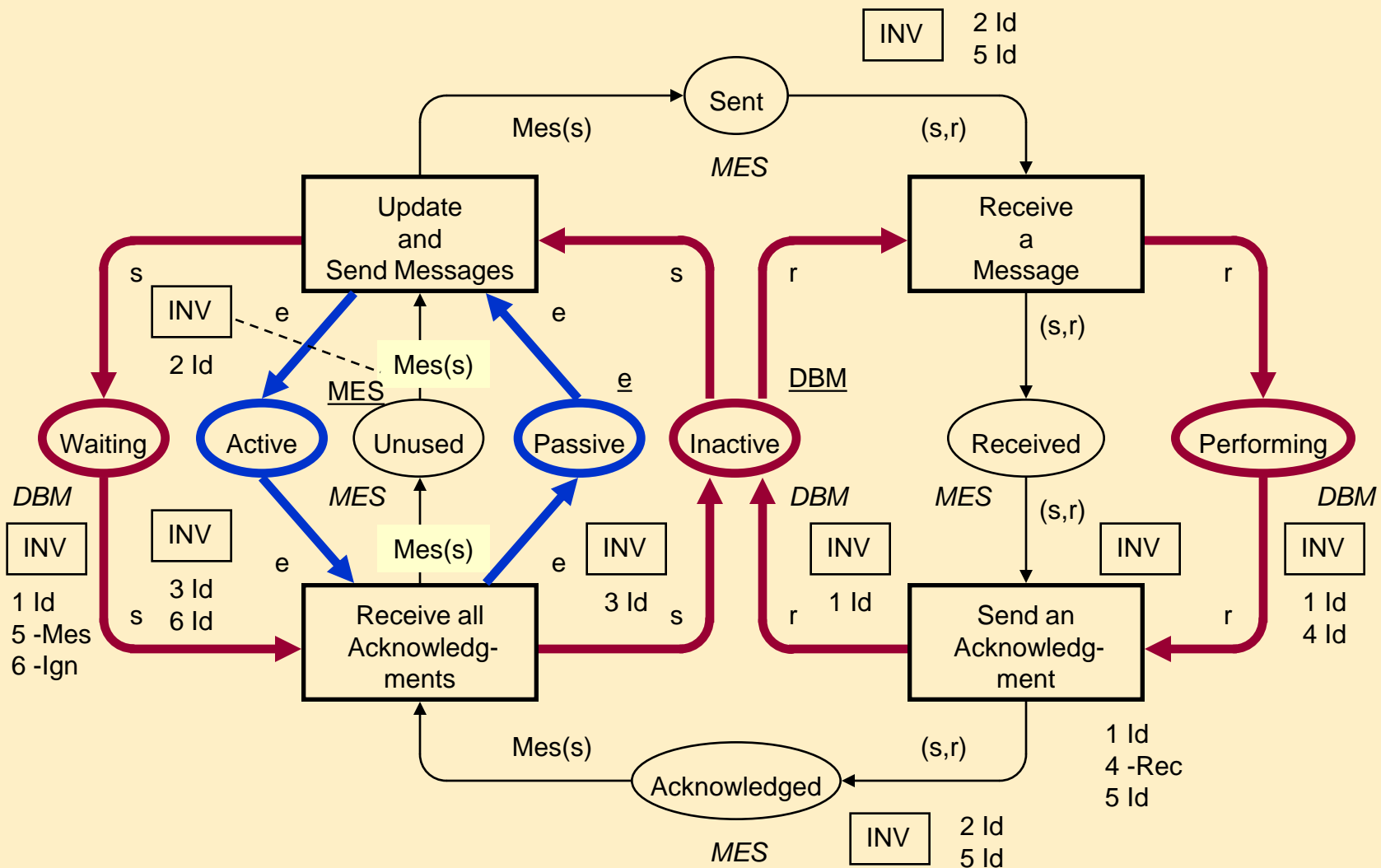
- Strukturális tulajdonságok: a háló uniform és konzervatív
- Korlátosság:

	multihalmaz	integer
– Inactive	<b>DBM</b>	<b>n</b>
– Waiting	<b>DBM</b>	<b>1</b>
– Performing	<b>DBM</b>	<b>n - 1</b>
– Unused	<b>MES</b>	<b><math>n*(n - 1)</math></b>
– Sent, Received, Acknowledged	<b>MES</b>	<b>n - 1</b>
– Passive, Active	<b>E</b>	<b>1</b>

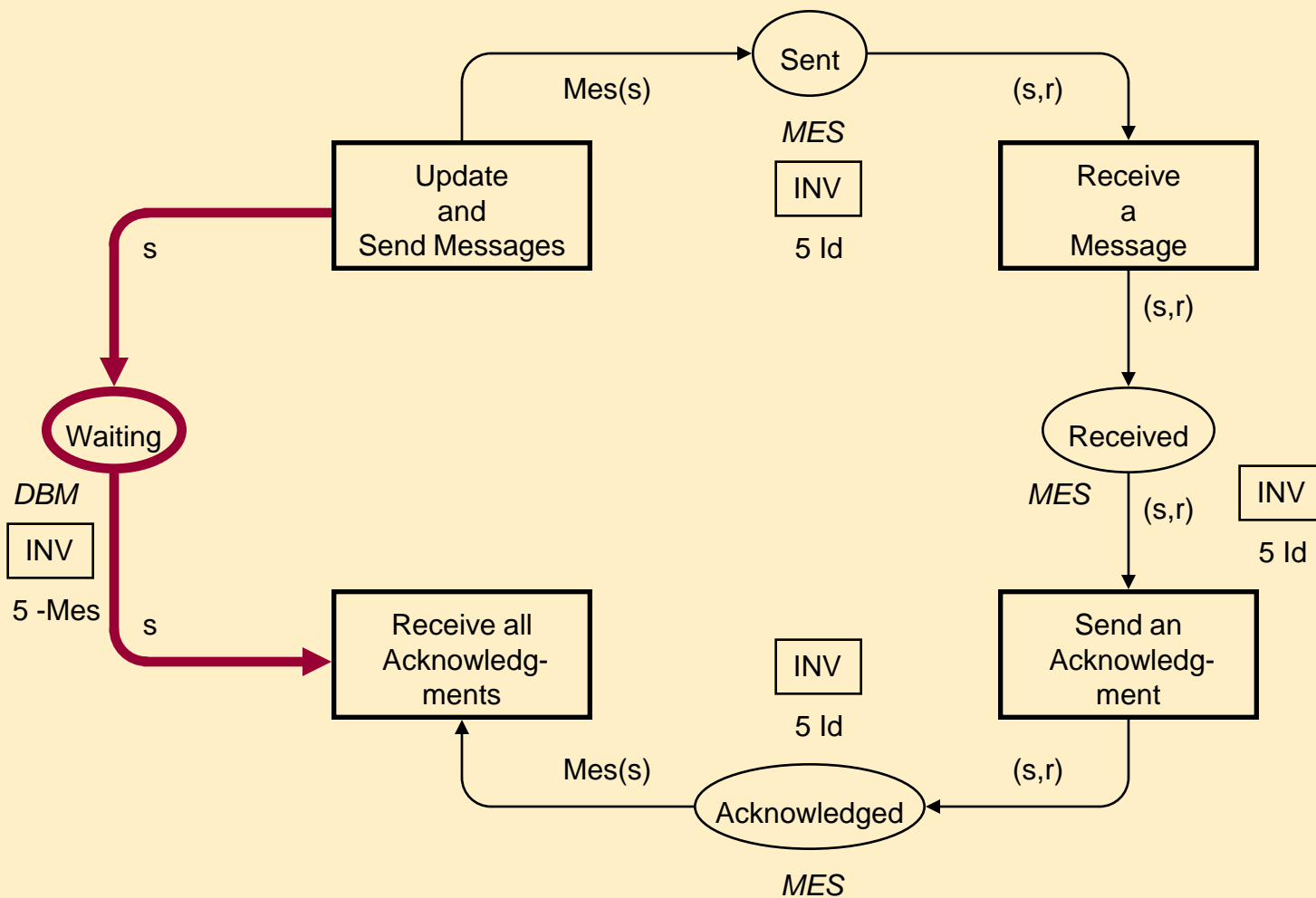
# P-invariánsok az adatbázis-kezelő modellben

- $M(\text{Inactive}) + M(\text{Waiting}) + M(\text{Performing}) = \text{DBM}$
- $M(\text{Unused}) + M(\text{Sent}) + M(\text{Received}) + M(\text{Acknowledged}) = \text{MES}$
- $M(\text{Active}) + M(\text{Passive}) = 1 \cdot e$
- $M(\text{Performing}) - \text{Rec}(M(\text{Received})) = \emptyset$
- $M(\text{Sent}) + M(\text{Received}) + M(\text{Acknowledged}) - \text{Mes}(M(\text{Waiting})) = \emptyset$
- $M(\text{Active}) - \text{Ign}(M(\text{Waiting})) = \emptyset$ 
  - $\text{Ign}()$  függvény tetszőleges színű tokent  $e \in E$  színű tokenre vált

# P-invariánsok az adatbázis-kezelő modellben



# Az adatbázis-kezelő modell egyik P-invariánsa





# A modell elérhetőségi gráfjának részlete

Sent, Received, Acknowledged

