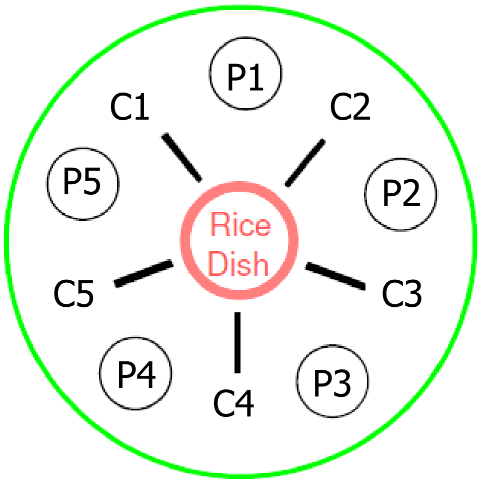
Színezett Petri-hálók

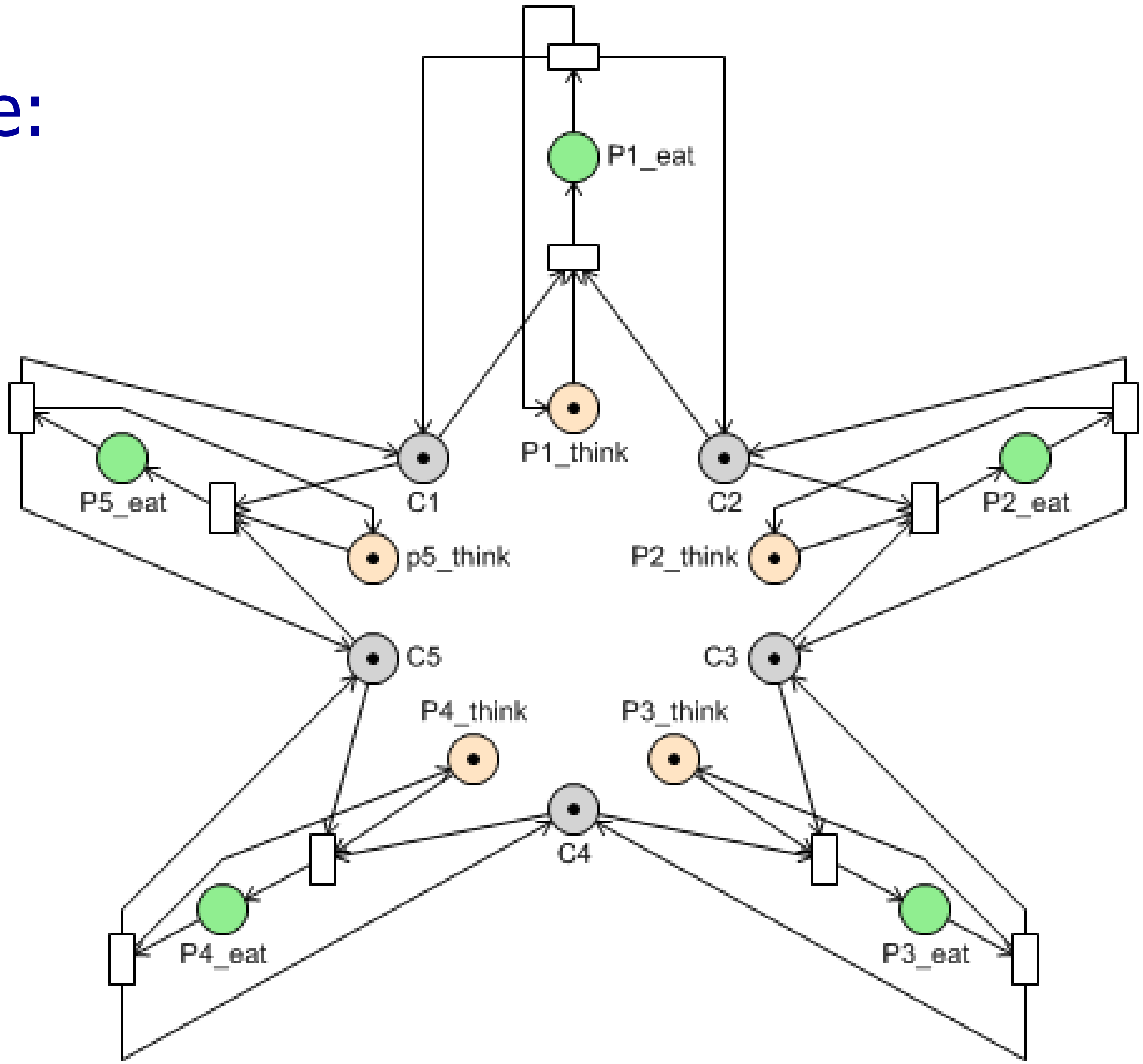
dr. Bartha Tamás
dr. Majzik István

BME Méréstechnika és Információs Rendszerek Tanszék

Motiváció 1/5

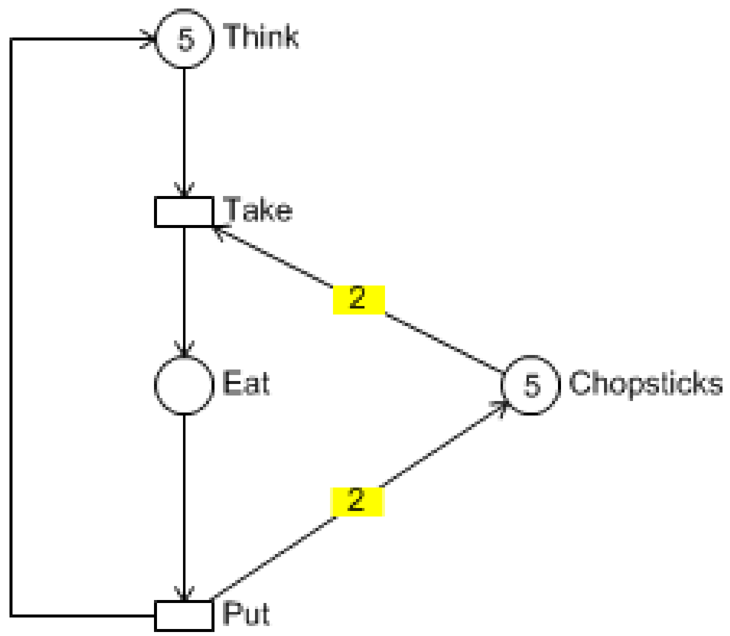
Étkező filozófusok Petri-háló modellje:





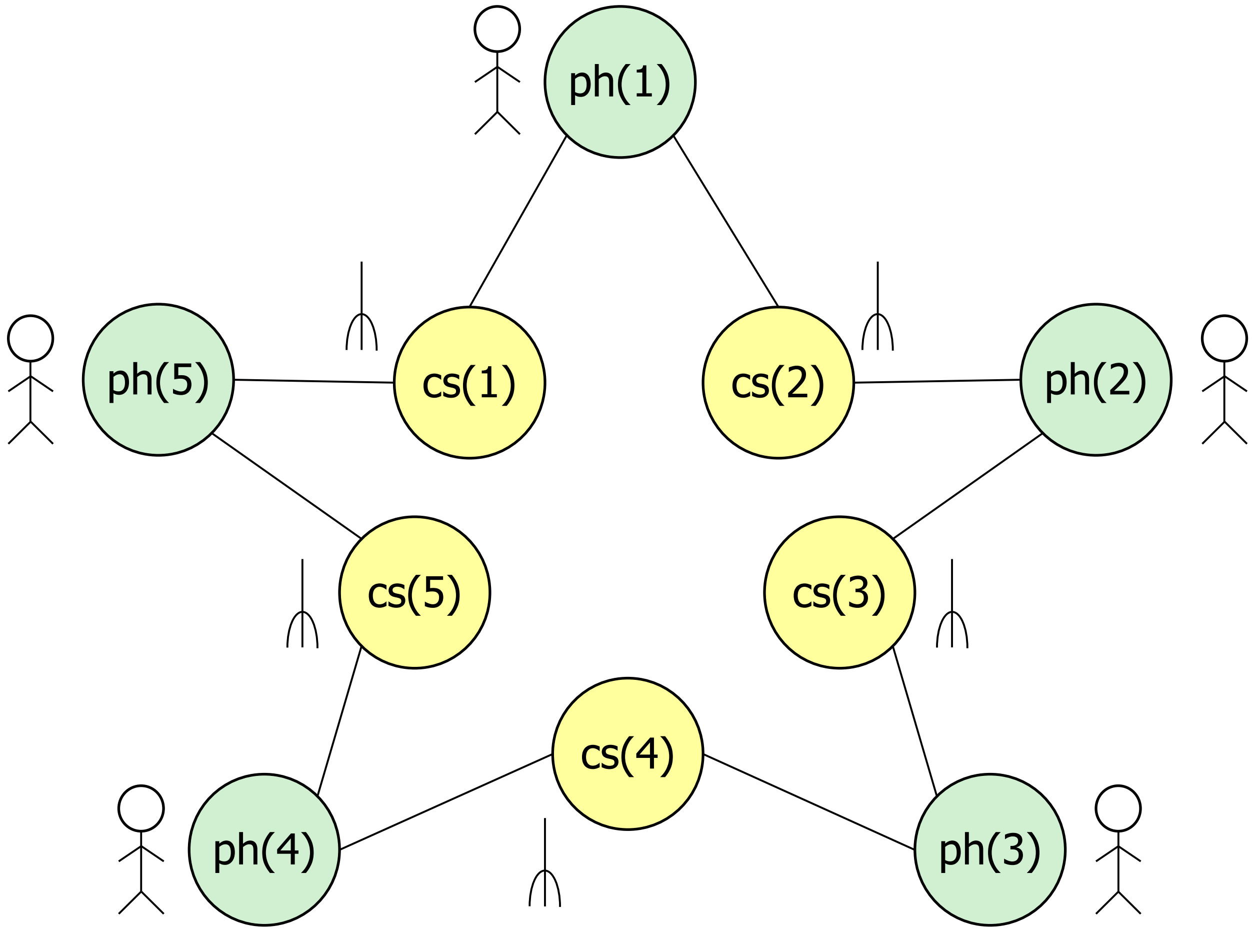
Motiváció 2/5

- Miért nem így modellezzük?



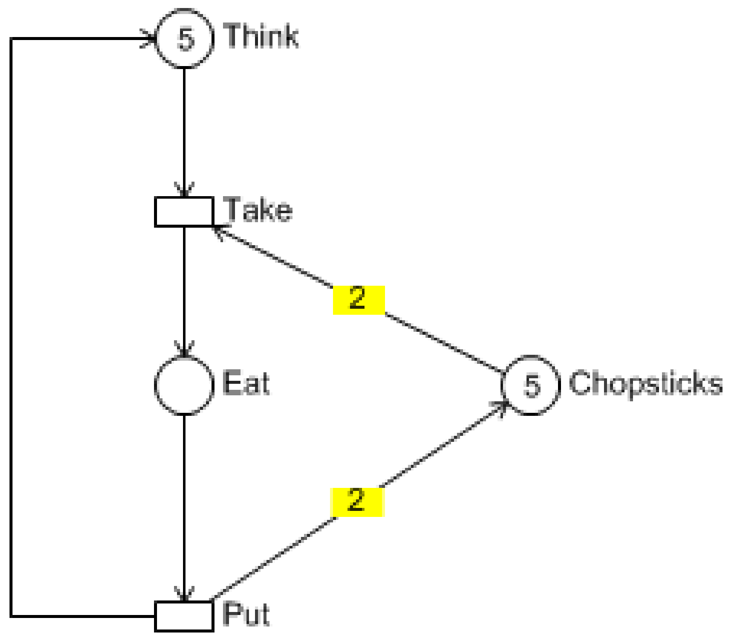
Motiváció 3/5

- Szereplők megkülönböztetése szükséges (lehetőleg paraméterezhetően)



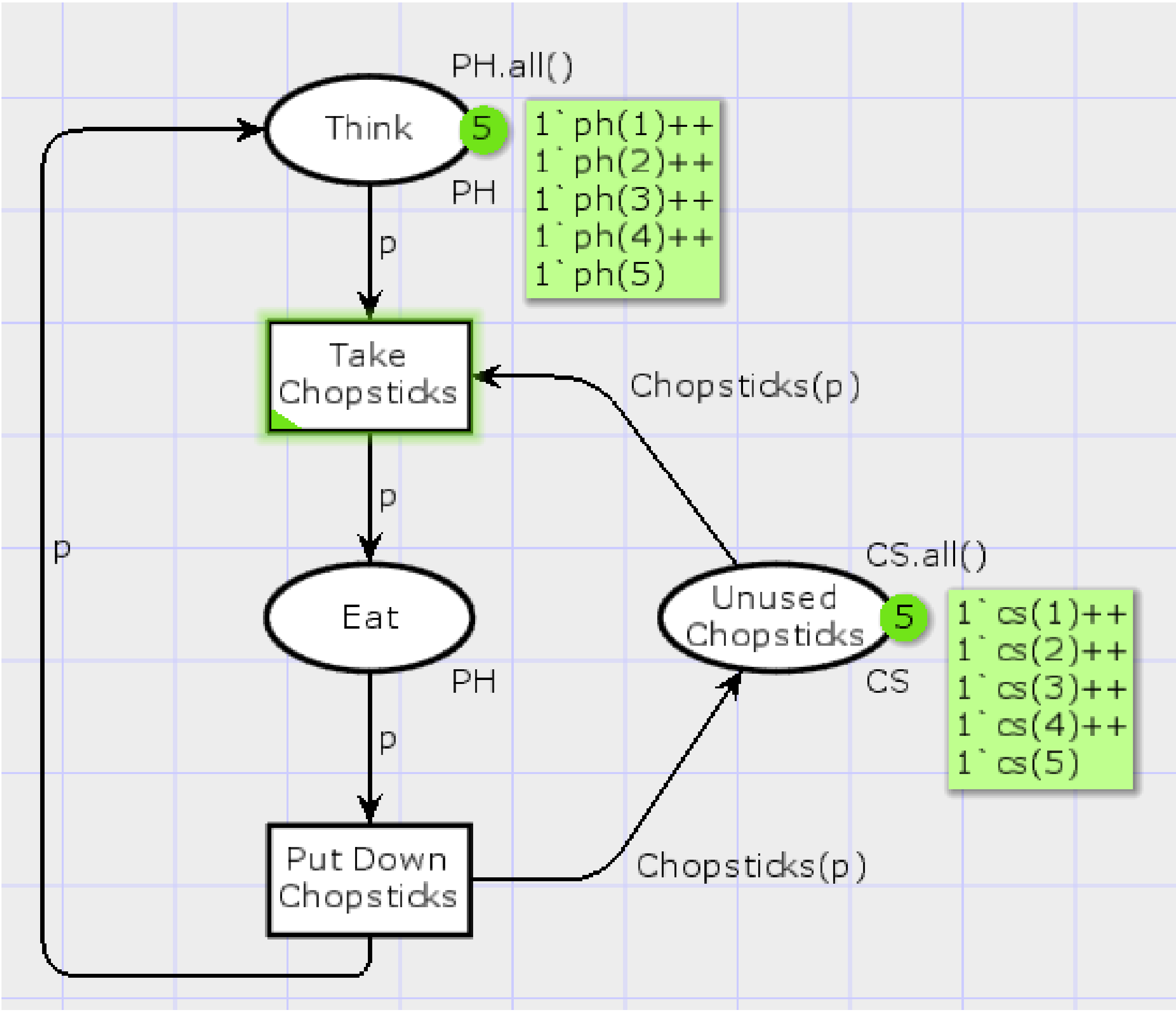
Motiváció 4/5

- Tokenek: Filozófusok és villák
  - Legyenek megkülönböztethetők
- Tranzíciók: Villa felvétele és lerakása
  - Legyen meghatározható, melyik filozófus melyik villát



Motiváció 5/5

- Megoldás: Színezett Petri-háló
  - Tokenek megkülönböztetve
  - Tranzíciók hatása részletezve




```
val n = 5;
colset PH = index ph with 1..n;
colset CS = index cs with 1..n;
var p: PH;

fun Chopsticks(ph(i)) =
  1`cs(i) ++
  1`cs(if i=n then 1 else i+1);
```

Színezett Petri-hálók

- A színezett Petri-hálók (Coloured Petri Net, CPN)
  - A színezetlen hálók kiterjesztései:
    - Rugalmas adatszerkezetekkel
    - Adatmanipulációs nyelvvel
  - A színezett Petri-háló modellek ötvözik:
    - Grafikus reprezentáció → struktúra áttekinthetősége
    - Adatmanipuláció → kifejezőerő
    - Jól definiált szemantika → formális analízis
  - CPN modell = háló struktúra + deklarációk + kifejezések + inicializáló kifejezések

Színezett Petri-hálók fő elemei (áttekintés)

- Tokenek kiterjesztései
  - Adatérték: színezett token
  - Adattípus: színhalmaz (színkészlet, színosztály)
- Helyek kiterjesztései
  - Hely típusa: fogadható tokenek adattípusa (színhalmaza)
  - Hely inicializáló kifejezése: kezdeti színezett tokenek megadása
  - Hely aktuális jelölése: típusának megfelelő színezett tokenek (multihalmaz: egy színezett tokenből több is lehet)
- Élek kiterjesztései
  - Élkifejezés: elvett ill. kirakott tokenek meghatározása
  - Leköthető változók: elvett és kirakott tokenek közötti függés
- Tranzíciók kiterjesztései
  - Őrfeltétel a tüzeléshez
  - Tüzeléskor: Élkifejezések kötése színezett tokenekhez

Színezetlen és színezett Petri-hálók összehasonlítása

Színezetlen (P-T) Petri-hálók:

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

Színezett Petri-hálók:

- színezett tokenek
- tokenek multihalmaza
- tokenszám és -szín manipuláció
- inicializáló kifejezések
- őrfeltételek
- élkifejezések (változókkal)
- lekötés engedélyezése
- konfliktus ugyanazon tranzíció engedélyezett lekötései között is
- *~ magas szintű programnyelv*

Színezett Petri-hálók felépítése

Tokenek kiterjesztései

- Színezett token
  - Adatérték reprezentálása
- Színhalmaz (színosztály)
  - Adattípus megadása (color, colset)

    Pl. felsorolás (with),
    alaptípus (int, bool, string, …)
  - Komplex is lehet

    Pl. color P = product U * I
- Változók (token hordozása)
  - Típus megadással (színhalmaz)

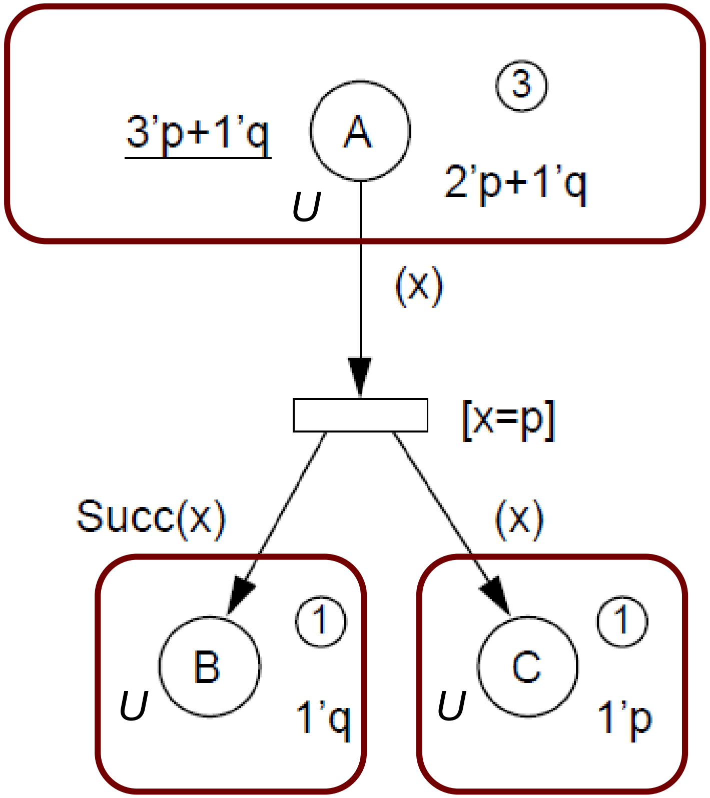
    Pl. var x: U
- Deklaráció: formális nyelven
  - Standard ML

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

Petri-háló helyek kiterjesztései

- Hely színhalmaza (típusa)
  - Milyen típusú tokeneket képes fogadni a hely (a deklarált típusok egyike)
  - Megjelenítés: hely mellett *dőlten*, pl. *U*
- Inicializáló kifejezés
  - A kezdeti jelölés megadása
  - A színhalmaz egy multi-halmaza (egy adott színű tokenből több is lehet)
  - Megjelenítés: hely mellé írva, aláhúzva

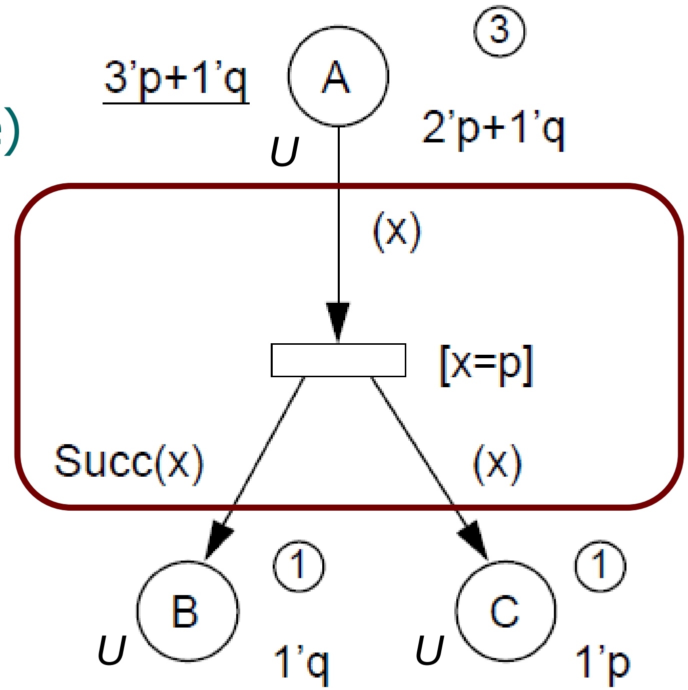
    Pl. 3′p + 1′q darabszám jelölés
- Aktuális jelölés
  - Az aktuális tokenek megadása
  - Megjelenítés: hely mellé írva; részletesen: pl. 2′p + 1′q bekarikázott szám: darabszám




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

Petri-háló élek és tranzíciók kiterjesztései

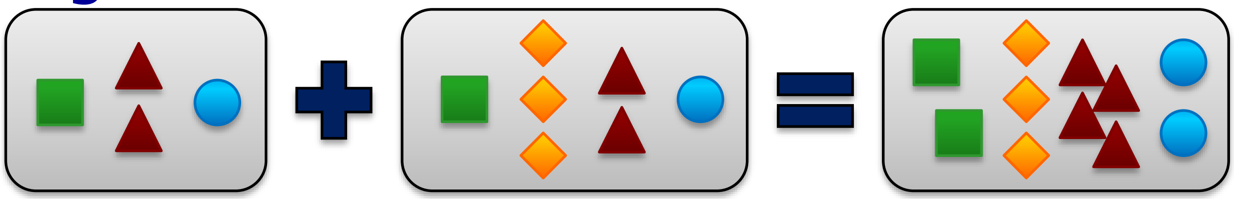
- Élkifejezések
  - Elveendő tokenek (engedélyezés feltétele), illetve kirakott tokenek (tüzelés eredménye)
  - Típusa: az élhez tartozó hely típusa (egy tranzícióhoz több „típusú” él húzható)
  - Megjelenítés: él mellett, pl. (x), Succ(x)
- Változó használható az élkifejezésben
  - Adatértékeket (színezett tokeneket) lehet hozzá kötni a bemeneti helyről
  - Típusa kell legyen (milyen színhalmaz elemei köthetők hozzá)
  - Egy tranzíció esetén: azonos kötéssel, ha a tranzíció több élkifejezésében is szerepel
- Őrfeltétel
  - Boole-kifejezés, a tranzíció engedélyezettségéhez igaz kell legyen
  - Megjelenítés: tranzíció mellett, pl. [x=p]



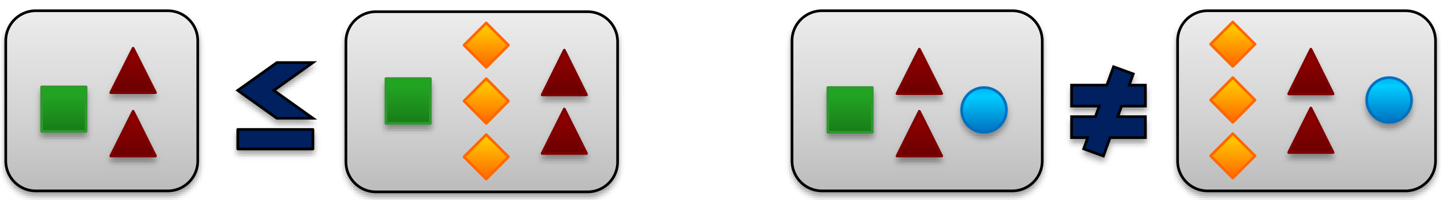

```
color U = with p | q;
color I = int;
color P = product U * I;
color E = with e;
var x : U;
var i : I;
```

Élkifejezés, őrfeltétel: Műveletek multihalmazokkal

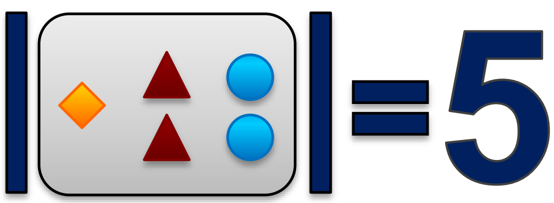
Összegzés: a1 + a2



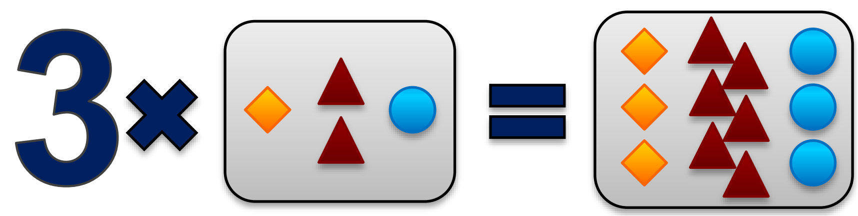
Összehasonlítás: a1 ≤ a2, a1 ≠ a2



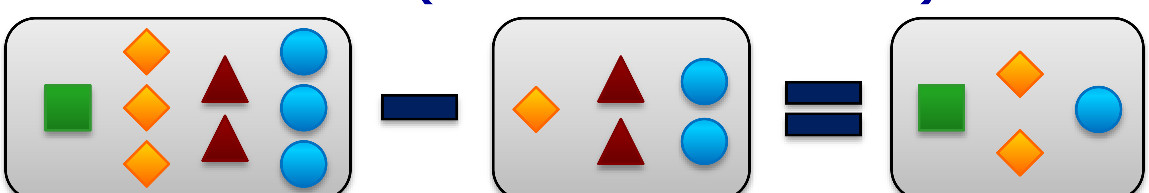
Számosság: |a1|



Szorzás skalárral: n·a1



Kivonás: a1 - a2 (csak ha a1 ≥ a2)

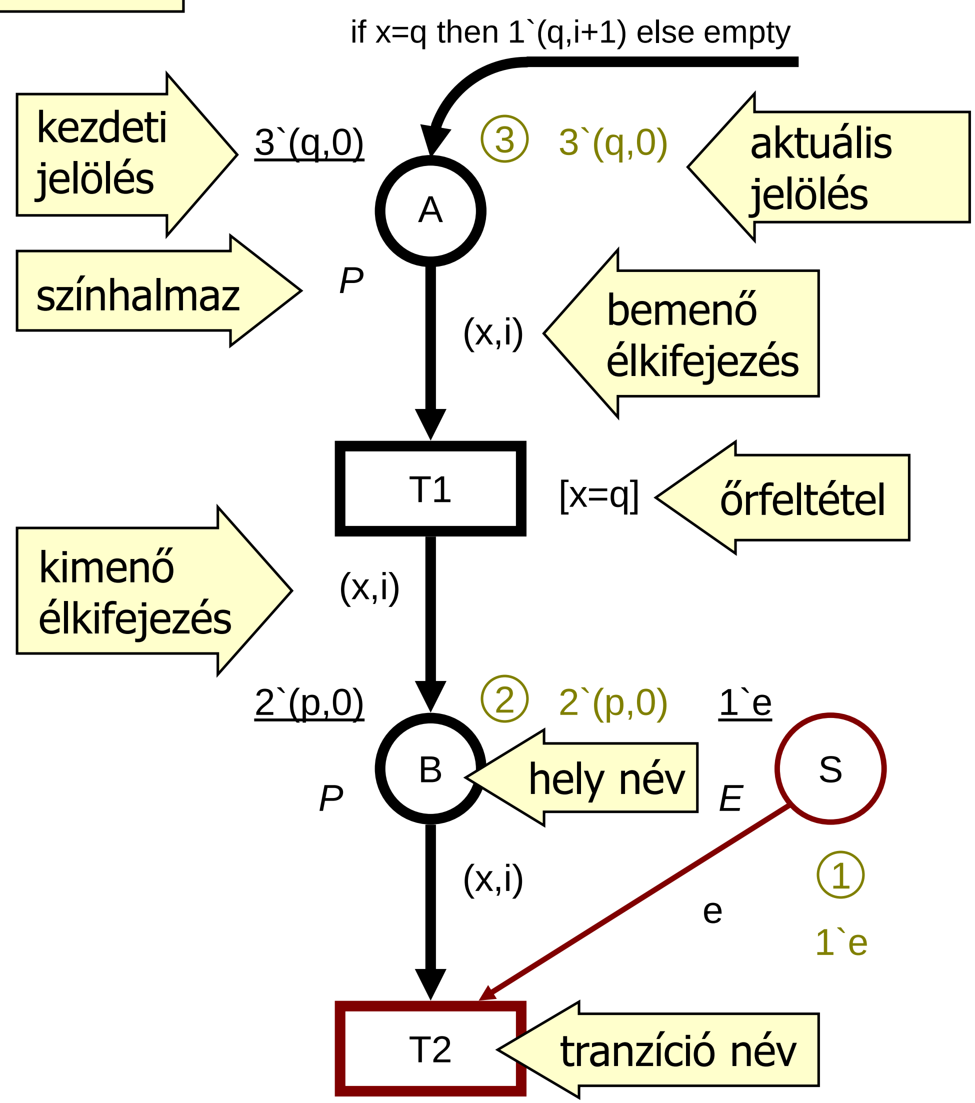


Színezett Petri-háló felépítés: Összefoglalás

- Háló struktúra (elnevezésekkel):
  - Megjeleníti a rendszer vezérlési illetve adatfolyam struktúráját
  - Helyek, tranzíciók, élek
- Deklarációk:
  - Definiálják az adatstruktúrákat és a felhasznált függvényeket
  - Színosztályok, változók, függvények megadása
- Kifejezések:
  - Megadják a háló szintaktikai és adatmanipulációs elemeit
  - Aktuális jelölések, élkifejezések, őrfeltételek
- Inicializációs kifejezések:
  - Megadják a modell kezdeti jelölését (kezdőállapotát)

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

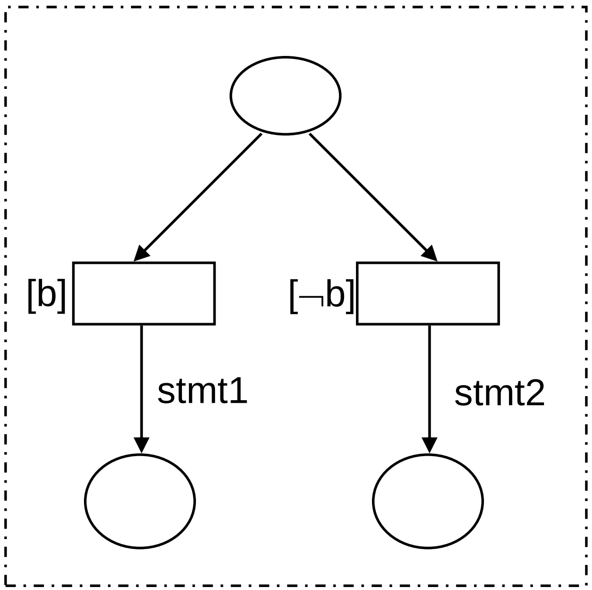
deklarációs mező



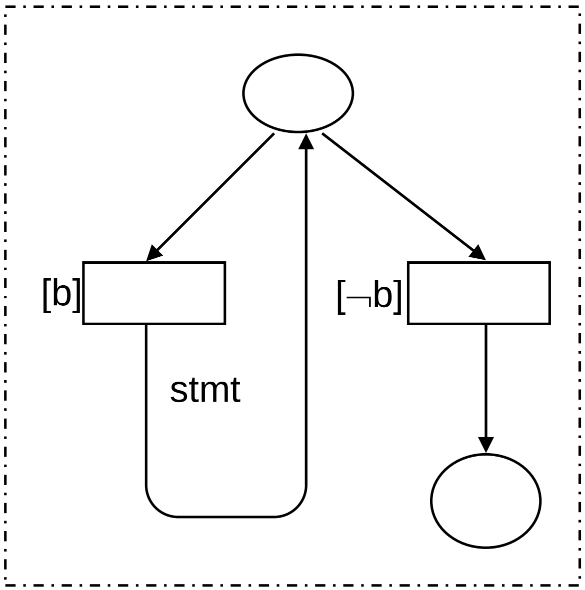
- CPN háló alkotóelemei:
  - Helyek
    - Név
    - Színhalmaz
    - Kezdeti jelölés
    - Aktuális jelölés
  - Tranzíciók
    - Név
    - Őrfeltétel
  - Élek
    - Élkifejezések (bemenő, kimenő)

Példa: Vezérlési struktúrák 1.

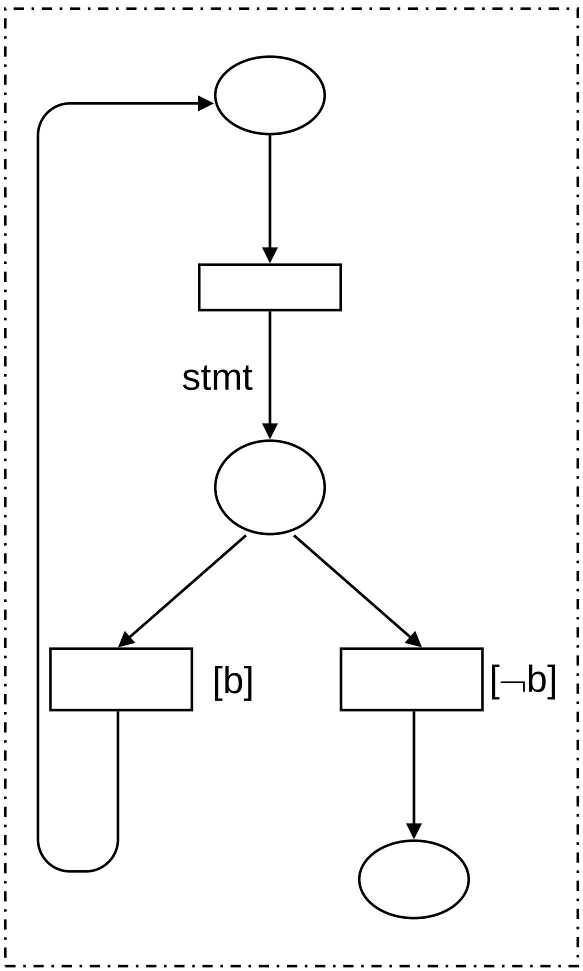
IF b THEN stmt1 ELSE stmt2



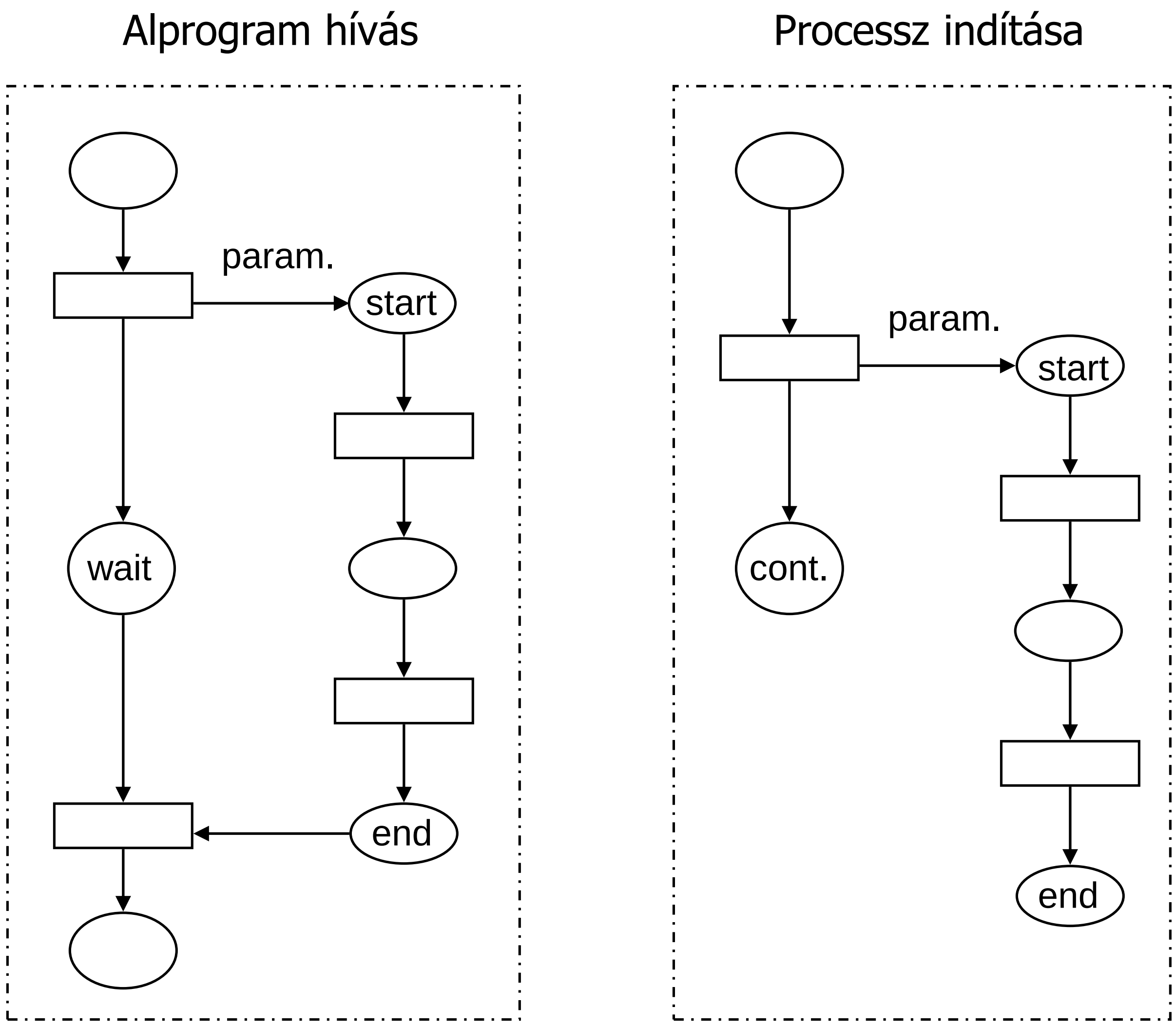
WHILE b DO stmt



REPEAT stmt UNTIL b



Példa: Vezérlési struktúrák 2.



Coloured Petri Nets (CPN) hálók eszközkészlete a CPN Tools eszközben

CPN hálók: Színosztályok definiálása

- Egyszerű színosztályok
  - Színezetlen tokenek:
    `unit`
  - Alapvető típusok:
    `int, bool, real, string`
  - Részhalmaz:
    `with 1..4;`
  - Felsorolás:
    `with true | false;`
  - Indexelés (vektor):
    `index d with 1..4;`
- Az alábbi elemek definíciójában szerepelnek:
  - Összetett színosztályok
  - Változók, konstansok
  - Függvények

Összetett színosztályok

- Módszerek kombinált színosztályok létrehozására
  - Unió képzés:
    ```
    union 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

  Rögzített értékek

  - Konstansdeklaráció:

    ```
    val n = 10;
    val d1 = d(1):D;
    ```

- Az alábbi kifejezésekben:
  - Élkifejezések
  - Őrfeltételek

- Az alábbi deklarációkban:
  - Színosztályok
  - Függvények
  - Élkifejezések, őrfeltételek, inicializáló kifejezések

További CPN háló elemek: Függvények

- Műveletek, operátorok
  - Infix jelölésrendszer
  - Példa: tokenek összegzése egy multihalmazba

```
1`cs(i) ++ 1`cs(i+1);
```

- Függvények
  - Mellékhatás-mentes SML nyelvű függvények
  - Példa:

```
fun Chopsticks(ph(i)) =
    1`cs(i) ++
    1`cs(if i=n then 1 else i+1);
```

- Az alábbi kifejezésekben:
  - Színosztályok
  - Függvények
  - Konstansok
  - Élkifejezések, őrfeltételek, inicializáló kifejezések

További CPN háló elemek: Kifejezések

- Kifejezések a hálóban
  - Értéke: a változók egy adott lekötésével értékelhető ki
  - Típusa: az összes lehetséges kiértékelési eredmény halmaza
  - Példák:

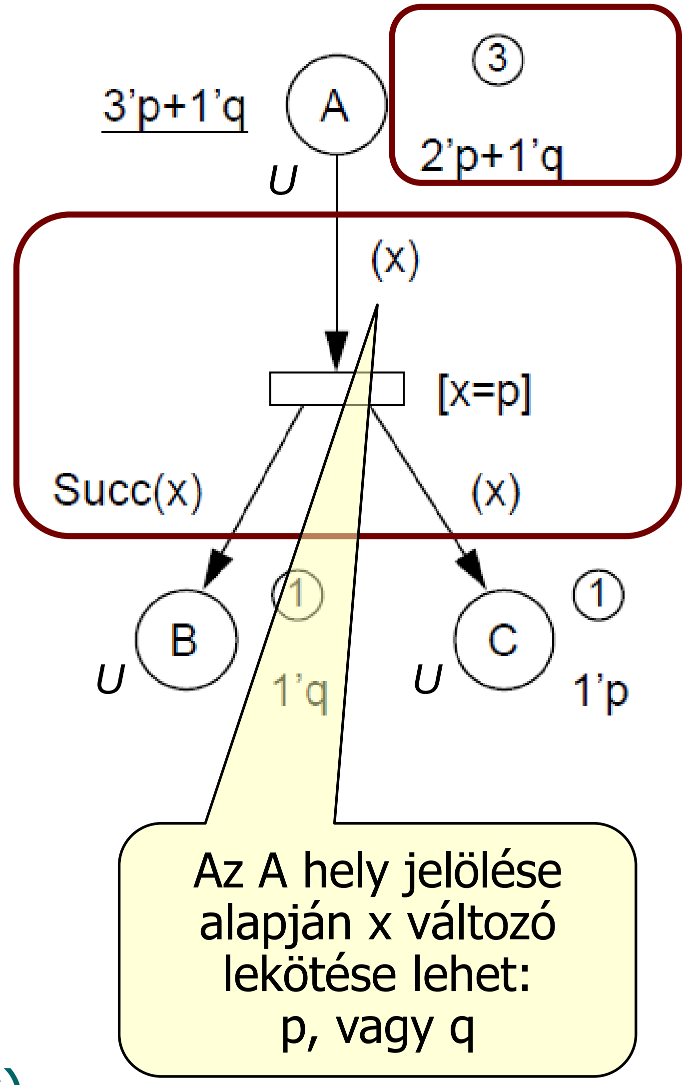
```
x=q
2`(x,i)
if x=q then 2`i else empty
Mes(s)
```

- Felhasználásuk:
  - Élkifejezések, őrfeltételek, inicializáló kifejezések

Színezett Petri-hálók működése (informális szemantika)

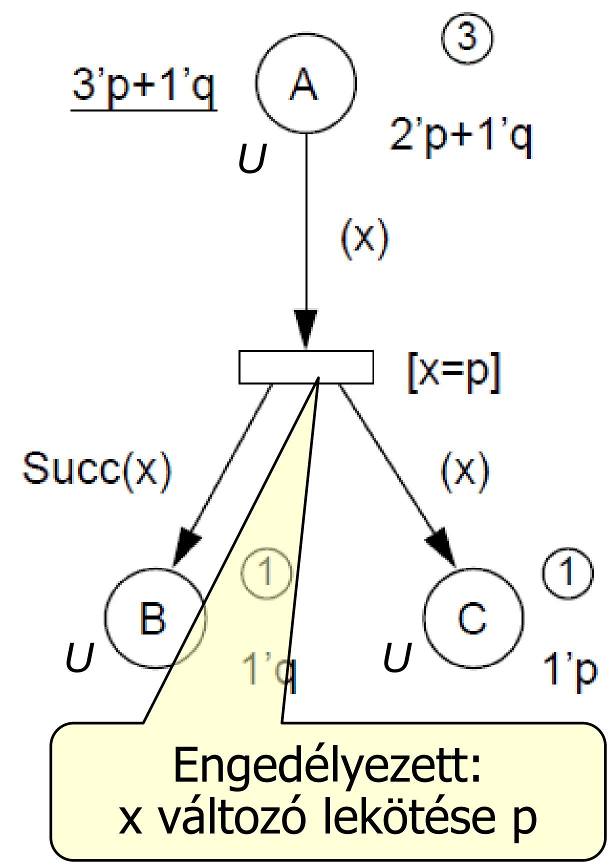
Jelölés és lekötés

- Jelölés:
  - Színezett tokenek eloszlása a helyeken
- Lekötés egy tranzíció élkifejezéseiben:
  - A változót adatértékhez (színezett tokenhez) kötjük a bemeneti helyről
  - Egy tranzíció esetén: egy adott változó minden előfordulása ugyanúgy lesz lekötött (azonos változó ugyanazt az értéket veszi fel)
  - Lekötetlen változó kimenő élen: Típusának bármely értékét felveheti
  - Különböző tranzíciók a lekötés szempontjából függetlenek (azonos nevű változó lekötései itt függetlenek)



Engedélyezettség

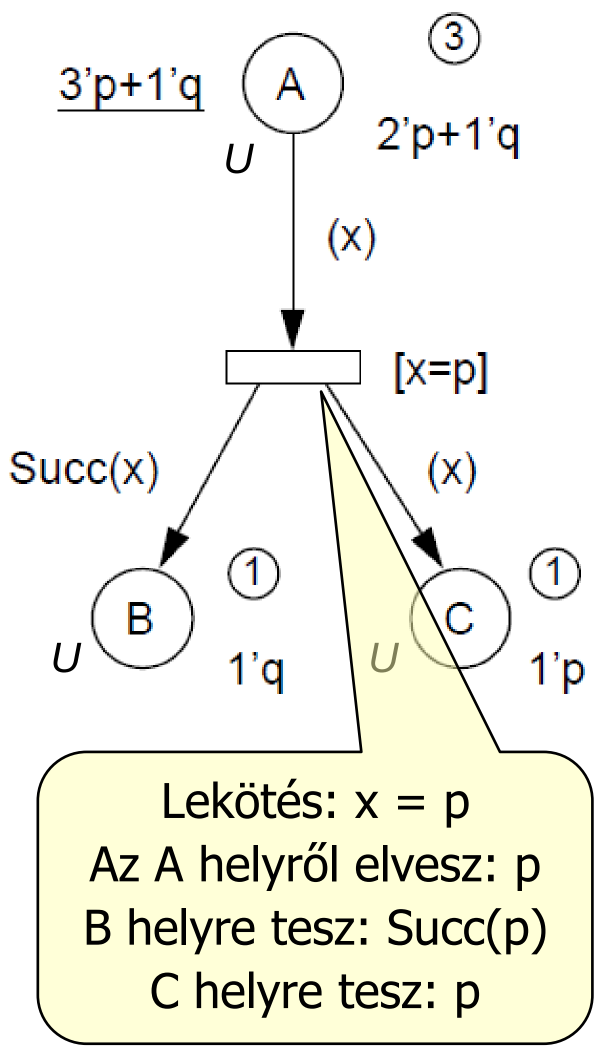
- Tranzíció engedélyezett egy adott jelölésben egy adott lekötésre:
  - A bemenő helyek tartalmazzák azokat a tokeneket, amik az élkifejezések értékei lehetnek az adott lekötésben
  - Az őrfeltétel teljesül
  - Ha egy tranzíció engedélyezett egy adott lekötésre, akkor tüzelhet
- Kötési elem tüzeléshez:
  - Egy (tranzíció, lekötés) pár, pl. (T1, <x=p>)



  - Engedélyezett lehet egy adott jelölésben → tüzelhet
  - Egy tranzíció esetén: több lekötés, ezekből több engedélyezett kötési elem képezhető; ezek tüzelhetnek

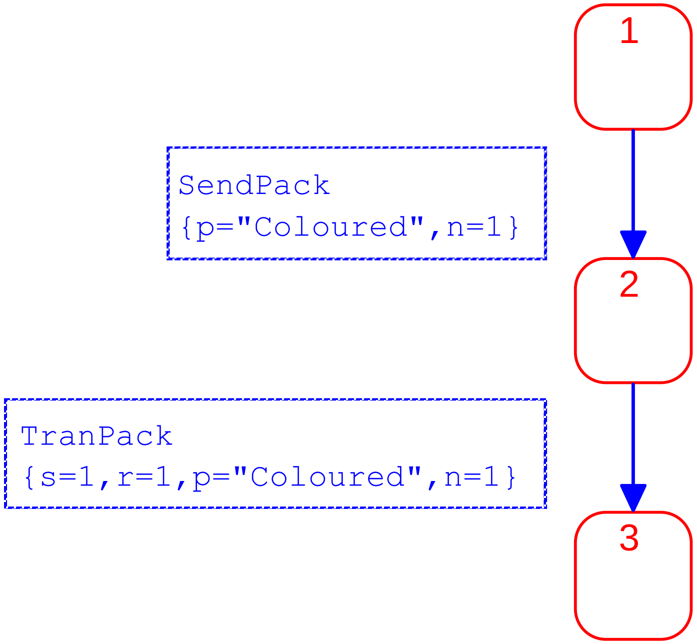
Tüzelés

- Tranzíció tüzel egy lekötésben (azaz egy kötési elem tüzel):
  - Bemenő helyekről az élkifejezés adott lekötésben lévő értéke által meghatározott (számú, színű) token elvétele
  - Kimenő helyekre az élkifejezés adott lekötésben lévő értéke által meghatározott (számú, színű) token odarakása
- Lépés (a tüzelés hatása az állapottérben):
  - A színezett Petri-háló egy jelöléséből egy másik lesz



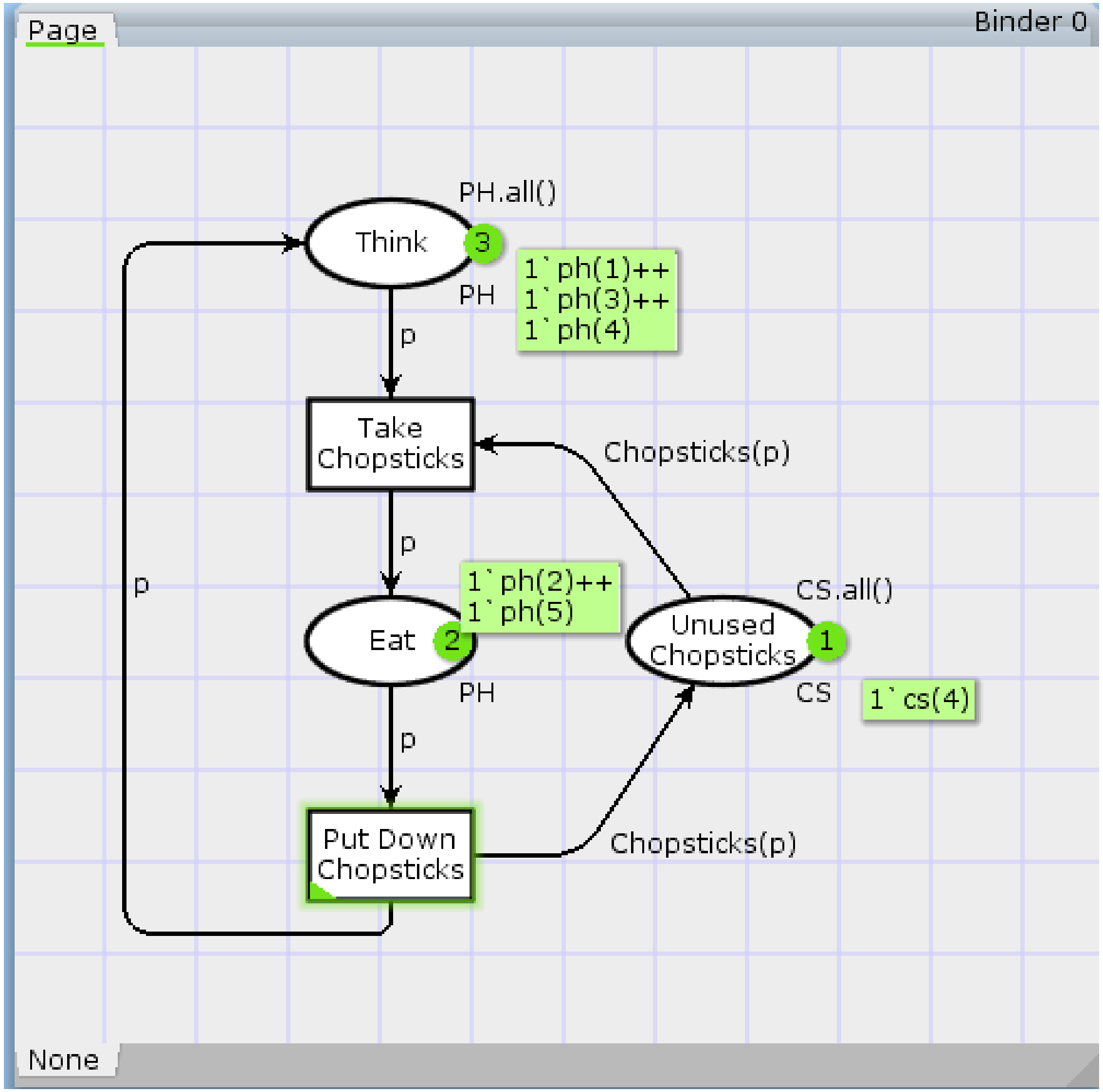
Elérhetőségi gráf

- Csomópont az elérhetőségi gráfban:
  - Egy jelölés: tokenek multihalmaza minden helyen
- Él az elérhetőségi gráfban:
  - Egy kötési elem, amiben tüzelés történt: tranzíció és a lekötés megadása
  - Definíció szerint egyszerre egy tüzelő kötési elem van feltüntetve az elérhetőségi gráfban



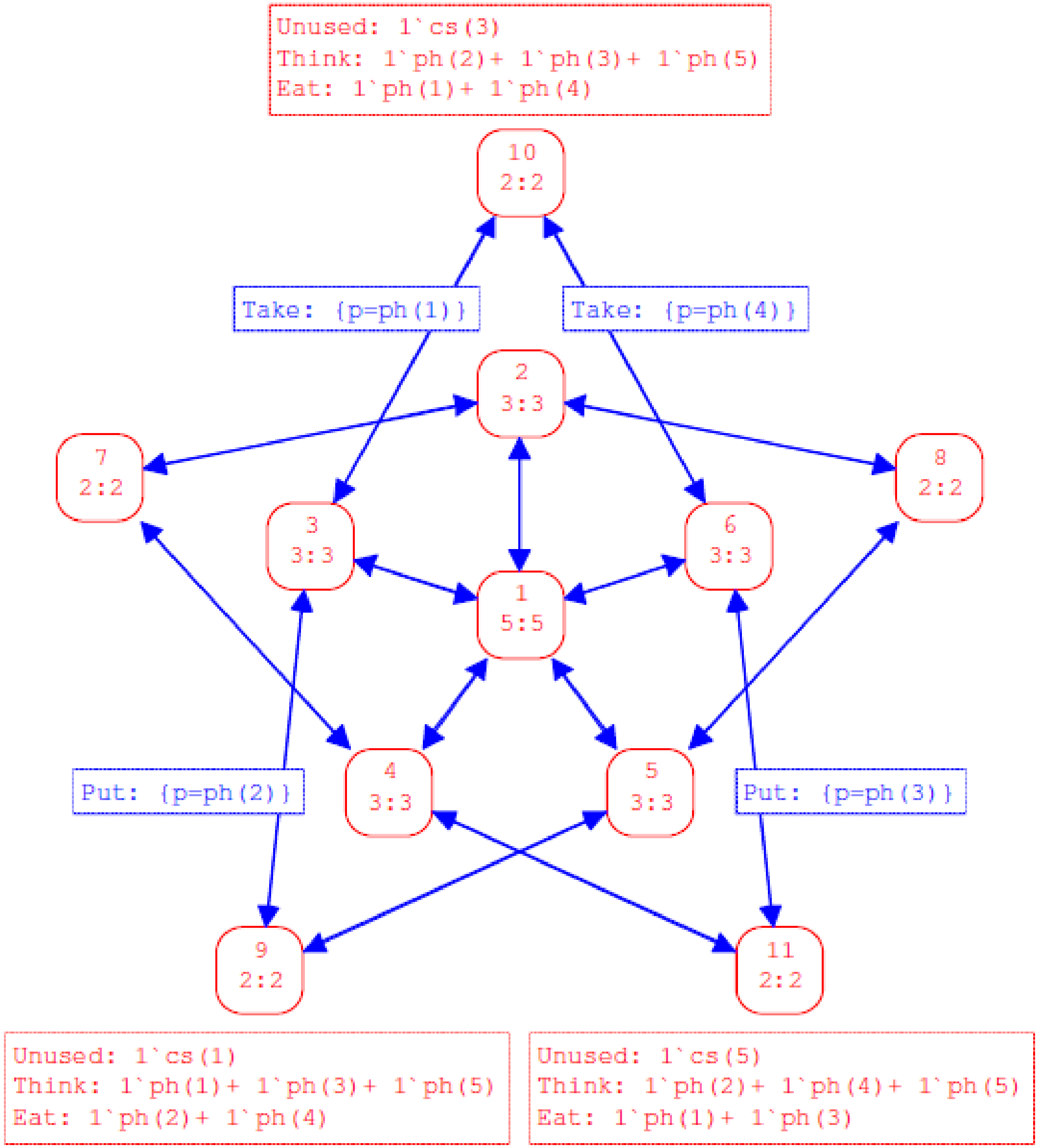
CPN Tools demo

- Étkező filozófusok modellje
- Szimuláció: Lekötés kiválasztása, tüzelés
- Elérhetőségi gráf felvétele



CPN Tools elérhetőségi gráf

- Csomópont:
  - Egy jelölés
  - Sorszám, elődök : utódok száma megadva
  - Jelölés kifejthető
- Él:
  - Egy kötési elem, amiben tüzelés történt
  - Tranzíció és a lekötés kifejthető



Példa: Egy egyszerű commit protokoll

A probléma leírása:

- Egy rendszerben három egység van: c_1, c_2 és c_3
- Közülük véletlenszerűen az egyik lesz a koordinátor, aki kérést küld a másik kettőnek
- A kérés hatására a felkért egység vagy abort, vagy commit szavazatot ad
- A két egység szavazata alapján a koordinátor dönt: ha mindkét egység commit döntést hozott, akkor a döntés commit, egyébként abort
- A koordinátornak nincs szavazata

Példa: Az egyszerű commit protokoll modellje

- Három színosztály, ebből kettő egyszerű halmaz:

    $C = \{0, c_1, c_2, c_3\}$ az egységek (0 a „senki”)

    $D = \{commit, abort\}$ a szavazatok / döntések

  Egy pedig kompozit színosztály:

    $M = C \times C$, a felkérések (ki kit kért fel szavazásra);
    a $(0, x)$ token: a koordinátort senki sem kéri fel

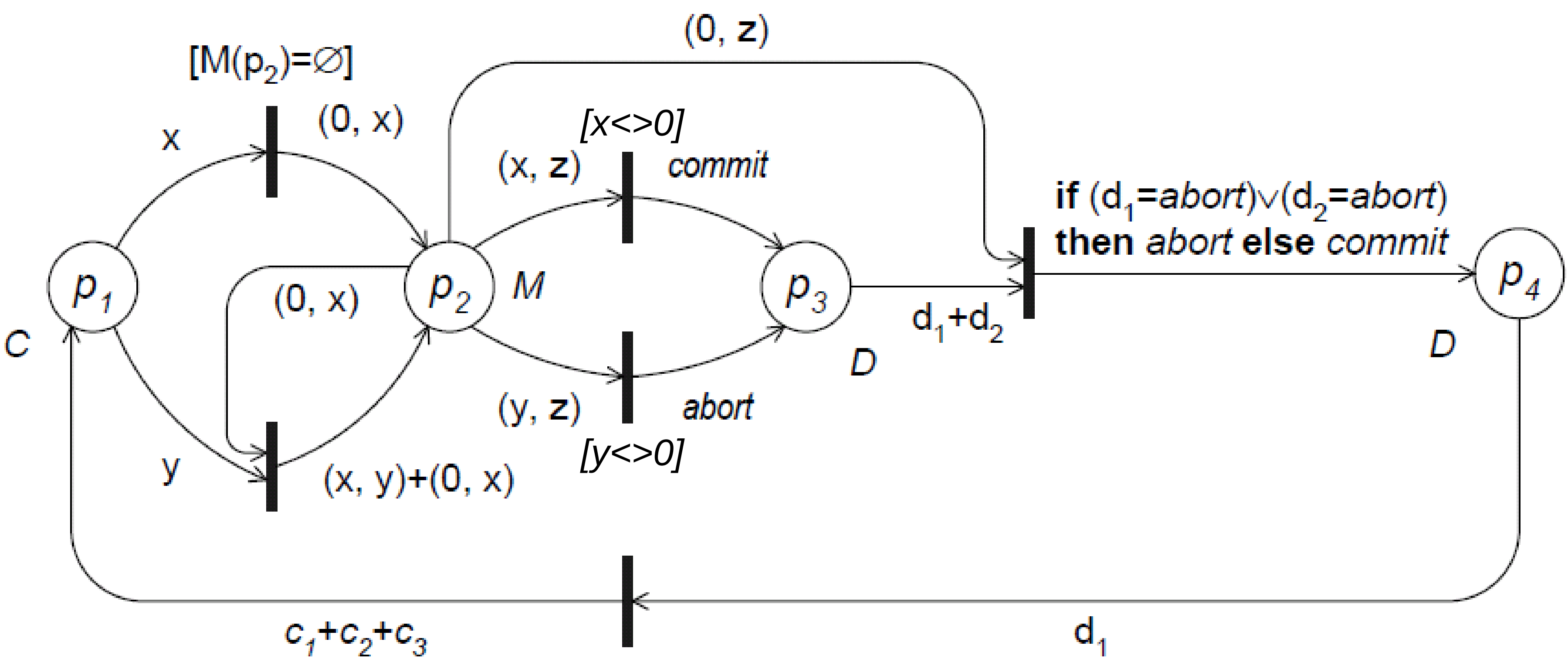
- Öt változó: $x, y, z \in C$; és $d1, d2 \in D$
- Az if élkifejezés: a programozási nyelveknél megszokott jelentéssel
- A háló kezdőállapotában a p_1 helyen 3 token van:

    $M(p_1)=1`c_1++1`c_2++1`c_3$, a többi hely üres

- A $\varnothing$ jel az üres halmazt jelöli

Példa: Az egyszerű commit protokoll modellje

- Színezett Petri-háló modell:
  - p_1: Résztvevők (kezdeti állapotban c_1, c_2, c_3 tokenek)
  - p_2: Kérések
  - p_3: Szavazatok
  - p_4: Döntés



Színezett Petri-hálók formális definíciója és szemantikája

(első olvasáskor átugorható)

Színezett Petri-hálók formális definíciója

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

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

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

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

$P \cap T = \emptyset$

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

Színkészlet: $C : P \mapsto \Sigma$

Őrfeltétel: $G : \forall t \in T, \left[\mathrm{Type}\big(G(t)\big) = \mathbb{B} \wedge \mathrm{Type}\Big(\mathrm{Var}\big(G(t)\big)\Big) \subseteq \Sigma\right]$

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

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

A formális definíciókban alkalmazott jelölések

- Egy v változó típusa (színosztálya): $\mathrm{Type}(v)$
- Egy expr kifejezés típusa: $\mathrm{Type}(\mathit{expr})$
- Egy expr kifejezésben szereplő változók halmaza: $\mathrm{Var}(\mathit{expr})$
- A v változó egy lekötése: $b(v) \in \mathrm{Type}(v)$
- Kifejezés által b lekötésre visszaadott érték: $\mathit{expr}<b>$

  ahol $v \in \mathrm{Var}(\mathit{expr})$ és $b(v) \in \mathrm{Type}(v)$

Multihalmazok

- Multihalmaz: azonos elemből több példány is lehet benne
  - Leképezés: $Bag(A)$, az A elemkészletre, $a \in [A \to \mathbf{N}]$
  - 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: $a_1 + a_2 = \sum_{x \in A} \big(a_1(x) + a_2(x)\big) \cdot x$
  - Különbség: $a_1 - a_2 = \sum_{x \in A} \big(a_1(x) - a_2(x)\big) \cdot x$ feltéve, hogy $a_2 \leq a_1$
  - Szorzás skalárral: $n \cdot a = \sum_{x \in A} \big(n \cdot a(x)\big) \cdot x$

Multihalmazok (folytatás)

- Unió, multihalmazok egyesítése: $a_1 \cup a_2 \cup \ldots \cup a_m$
  - Tartomány: $A_1 \cup A_2 \cup \ldots \cup A_m$
  - Eleme: $e_i \in \bigcup_1^m A_k \quad \text{ha } \exists A_j : e_i \in A_j$
- n-esek képzése: $\langle A_1, A_2, \ldots, A_n \rangle$
  - Tartomány: $A_1 \times A_2 \times \ldots \times A_2$
  - Eleme: $\left\langle e_1, e_2, \ldots, e_n \right\rangle \in \Diamond_1^n A_j \quad \text{ha } \forall e_i \in A_i$
  - Általánosítás: $\langle a_1, a_2, \ldots, a_n \rangle$

Élkifejezések

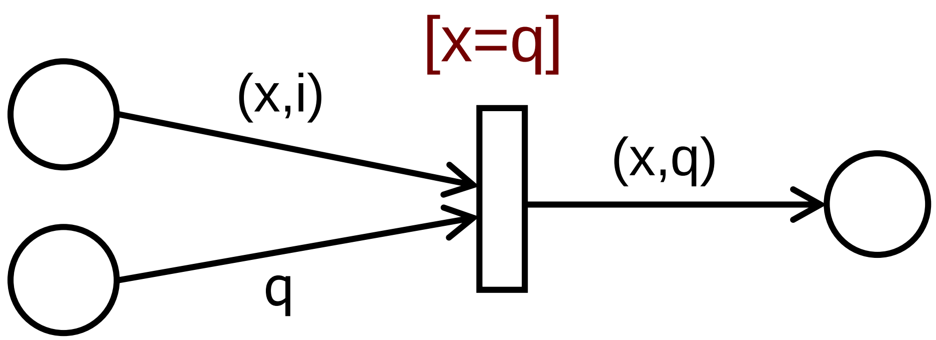
- Használhatók: Változók
  - Rendelkeznek típussal: $\mathrm{Type}(v)$
  - Értékük a típushoz tartozó multihalmaz egy eleme lehet
- Lezárt élkifejezés: nem tartalmaz változókat
- Nyílt élkifejezés: változókat tartalmaz, amelyeket le lehet kötni egy értékkel
  - Lekötés: egy konkrét értékhozzárendelés minden változóhoz
    - Adott lekötéssel az élkifejezés kiértékelhető
  - Rendelkezik típussal: $\mathrm{Type}(expr) = C(p)_{\mathrm{MS}}$
    - Az értékül kapott színosztály típusa
  - Kifejezésben szereplő változók halmaza: $\mathrm{Var}(expr)$

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 $\rightarrow$ azonos érték
- Lekötetlen változók
  - Csak kimenő élkifejezésekben szereplő változók
  - Az engedélyezés nem rendelt hozzá értéket: lekötetlen
  - Tüzeléskor le kell kötni:
    - A színosztályából bármilyen értéket felvehet
    - Annyi lehetséges lekötés, amennyi a színosztály számossága
    - Nemdeterminisztikus választás

Őrfeltételek

- Tranzícióhoz rendelt őrfeltétel
  - Multihalmazok felett értelmezett kifejezés
  - Boolean visszatérési értékkel
- „Igaz” kiértékelési érték esetén engedélyezi a tranzíciót
  - „Szűri” az engedélyezett lekötéseket



Engedélyezettség színezett Petri-hálókban

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

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

  - Az összes érvényes lekötés halmaza: $\mathrm{B}(t)$
- Egy érvényes lekötés engedélyezett, ha
  - Őrfeltétel igaz
  - A bemenő helyeken van elég színezett token (lásd $E^{-}(p,t)$<b> élkifejezések) és a tiltó élek nem tiltják le a tüzelést (lásd $E^{h}(p,t)$<b> élkifejezések):

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

Tüzelés prioritásos színezett Petri-hálókban

- Egy engedélyezett tranzíció tüzelhet, ha magasabb prioritású t′ tranzíció nem engedélyezett b′ lekötésben, azaz
  - Ennek bemenő helyein nincs elég színezett token
    (lásd $E^{-}(p,t')\langle b'\rangle$ élkifejezések),
    vagy a tiltó élei tiltják le a tüzelését
    (lásd $E^{h}(p,t')\langle b'\rangle$ élkifejezések),

$$\forall t', \pi(t') > \pi(t) : \exists p \in \bullet t' :$$

$$E^{-}(p,t')\langle b'\rangle > M(p) \vee E^{h}(p,t')\langle b'\rangle \leq M(p)$$

  - vagy az őrfeltétele nem igaz

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

Tüzelés színezett Petri-hálókban

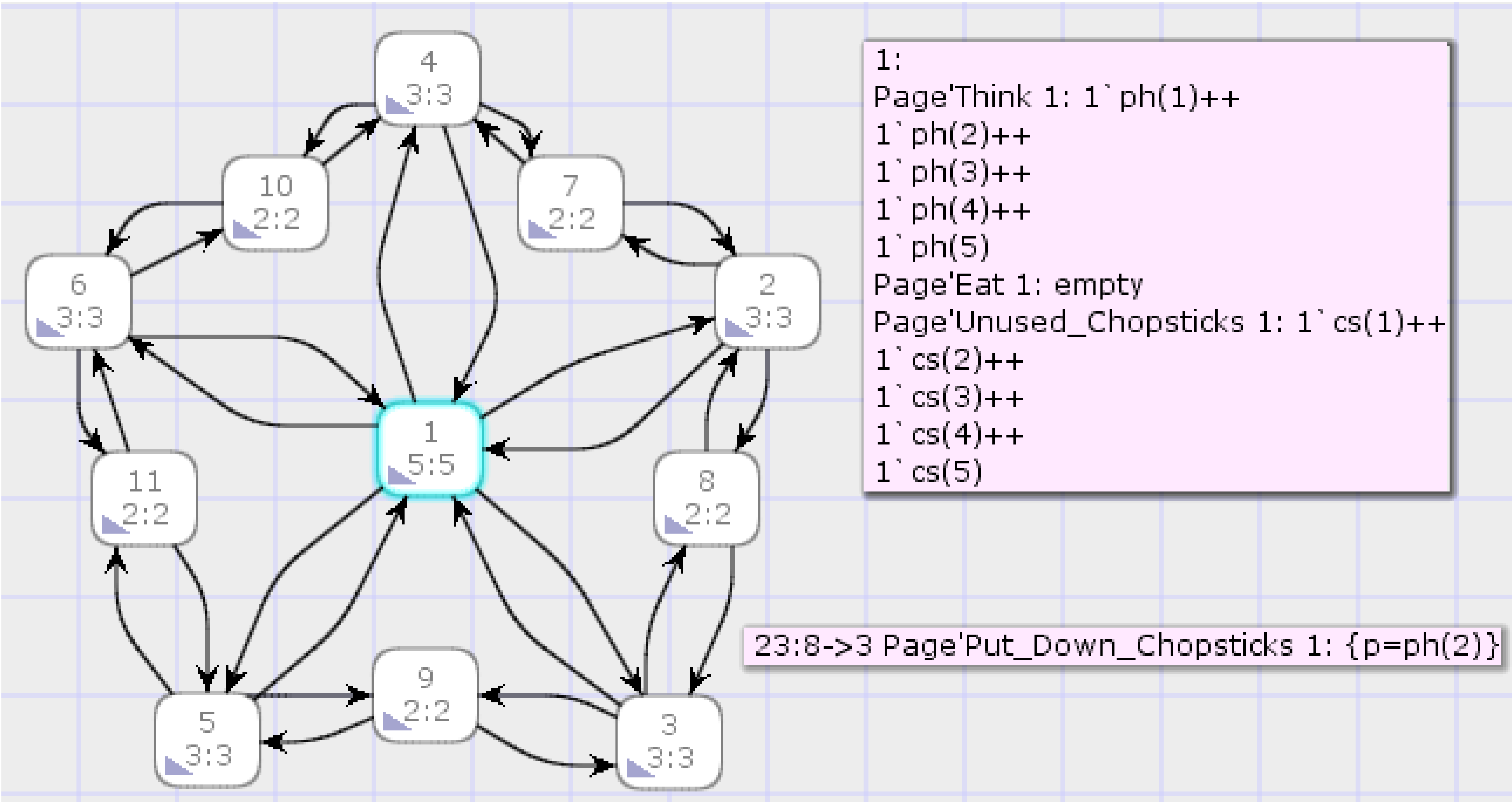
- Tüzelés menete:
  - Engedélyezett lekötések keresése
    - Meghatározzák a bemenő élkifejezések, őrfeltételek
  - Tranzíció engedélyezett adott lekötéssel $\rightarrow$ tüzelhet
  - Tüzelés: Színezett tokenek elvétele a bemenő helyekről, színezett tokenek odarakása a kimenő helyekre

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

  - Ekkor M' közvetlenül elérhető M-ből: $M \; [(t,b)\rangle \; M'$

Színezett Petri-hálók dinamikus tulajdonságai

Elérhetőségi gráf (ld. korábban)



- Csomópontok: Egy-egy jelölés (sorszámmal, elődök : utódok számával)
- Élek: Egy-egy tüzelő kötési elem (tranzíció és lekötés kifejthető)

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 egész korlát p-re, ha $\forall M \in \left[M_0 \right\rangle : \ \left| M(p) \right| < n$

  – m egy felső multihalmaz korlát p-re, ha $\forall M \in \left[M_0 \right\rangle : \ M(p) < m$

- Visszatérő tulajdonság

  Egy visszatérő állapotba mindig lehetséges visszajutni:

  M egy visszatérő állapot, ha $\forall M' \in \left[M_0 \right\rangle : \ M \in \left[M' \right\rangle$

Színezett Petri-hálók dinamikus tulajdonságai

- Élőség

  Az élőség garantálja, hogy a lekötési elemek egy része aktív marad (azaz tranzíció valamilyen lekötésben tüzelni tud)

  - Halott állapot (deadlock): egy lekötési elem sem engedélyezett

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

  - Halott tranzíció: egyik lekötése sem válhat engedélyezetté

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

  - Élő tranzíció: minden elérhető állapotra igaz, hogy onnan valamely trajektórián tüzelni tud

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

Színezett Petri-hálók dinamikus tulajdonságai

- Fair tulajdonság tüzelési szekvenciákban

  Fairség megmutatja, hogy egy lekötési elem milyen gyakran tüzel

  – Elfogulatlan (impartial) tranzíció: végtelen sokszor tüzel
    (itt OC() az előfordulások száma értelemszerű paraméterekkel)

$$\forall b \in B(t),\ |\sigma| = \infty : \quad \mathrm{OC}_b(\sigma) = \infty$$

  – Fair tranzíció: végtelen sok engedélyezés ⇒ végtelen sok tüzelés

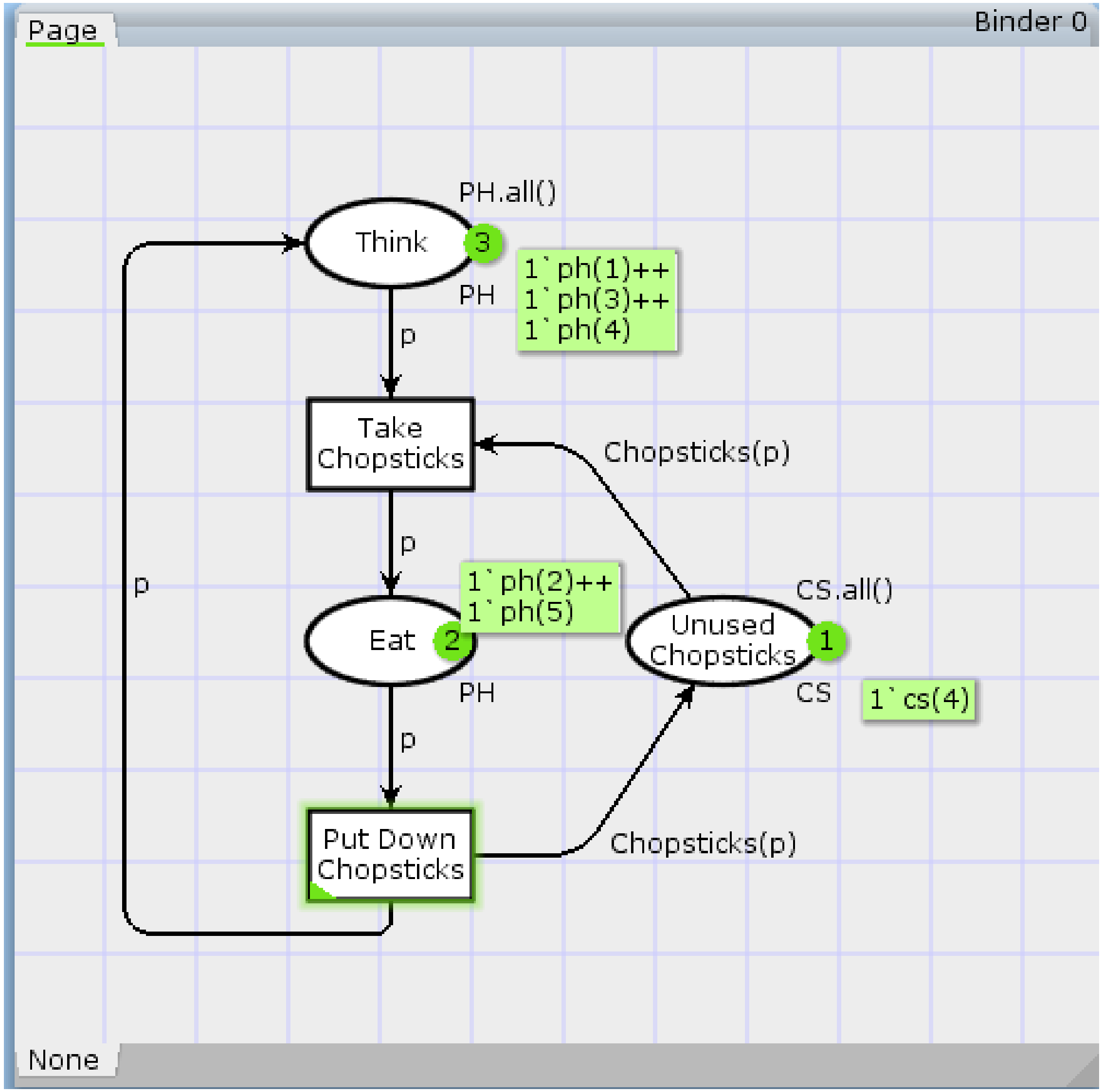
$$\forall b \in B(t),\ |\sigma| = \infty : \quad \mathrm{EN}_b(\sigma) = \infty \Rightarrow \mathrm{OC}_b(\sigma) = \infty$$

  – Igazságos (just) tranzíció: perzisztens engedélyezés ⇒ tüzelés
    (nincs perzisztens engedélyezés tüzelés nélkül)

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

CPN Tools demo

- Étkező filozófusok modellje
- Tulajdonságok: Jelentés generálása
- Lekérdezések (könyvtári ML függvények)



Színezett Petri-hálók strukturális tulajdonságai

T-invariáns színezett Petri-hálókban

- Tranzíció invariáns

Olyan σ tüzelési szekvencia, ami nem hat az állapotra:

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

ahol $M'(p) - M(p) = 0$ minden p-re

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

P-invariáns színezett Petri-hálókban

- Hely invariáns
  - Alapötlet: Súlyozott tokenösszeg képzése

$$\mathrm{W}_{p_1}\left(M(p_1)\right)+\mathrm{W}_{p_2}\left(M(p_2)\right)+\ldots\mathrm{W}_{p_n}\left(M(p_n)\right)$$

  - $\mathrm{W_P}$ súlyfüggvény: hely színkészletét egy közös multihalmazra képezi le
  - $\mathrm{W_P}$ egy P-invariáns, ha a súlyozott tokenösszeg állandó

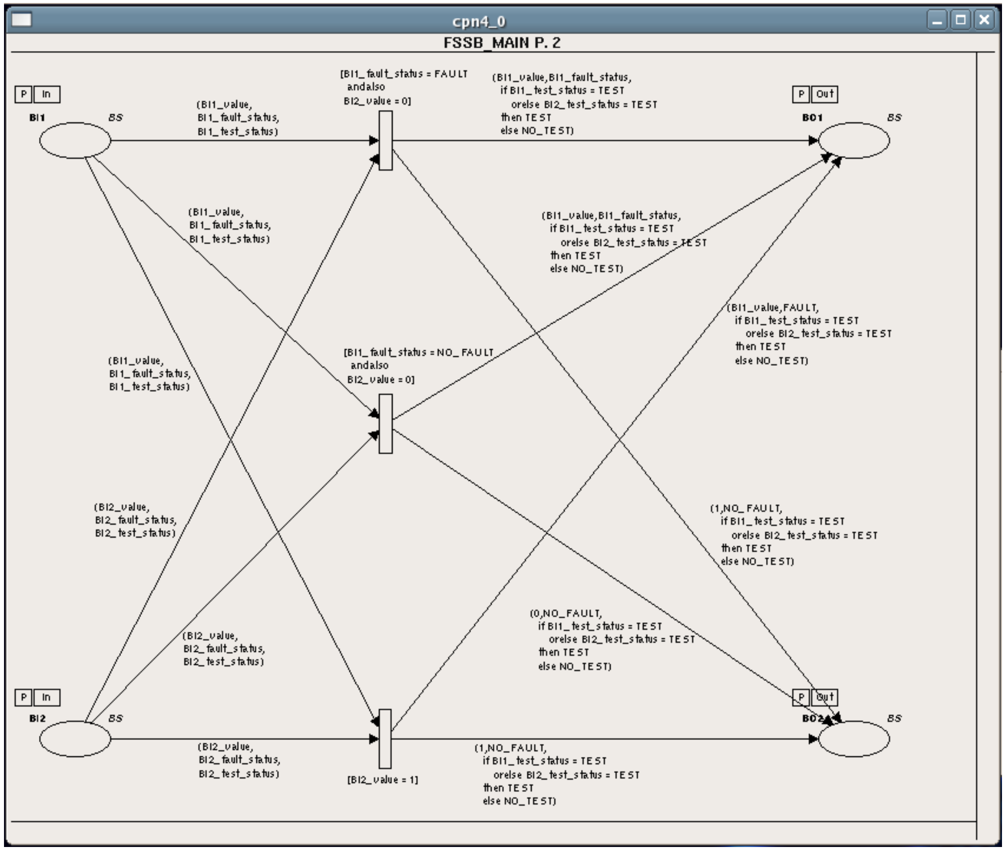
$$\forall M\in\left[M_0\right\rangle:\quad \sum_{p\in P}W_p\left(M(p)\right)=\sum_{p\in P}W_p\left(M_0(p)\right)$$

Színezett Petri-hálók széthajtogatása

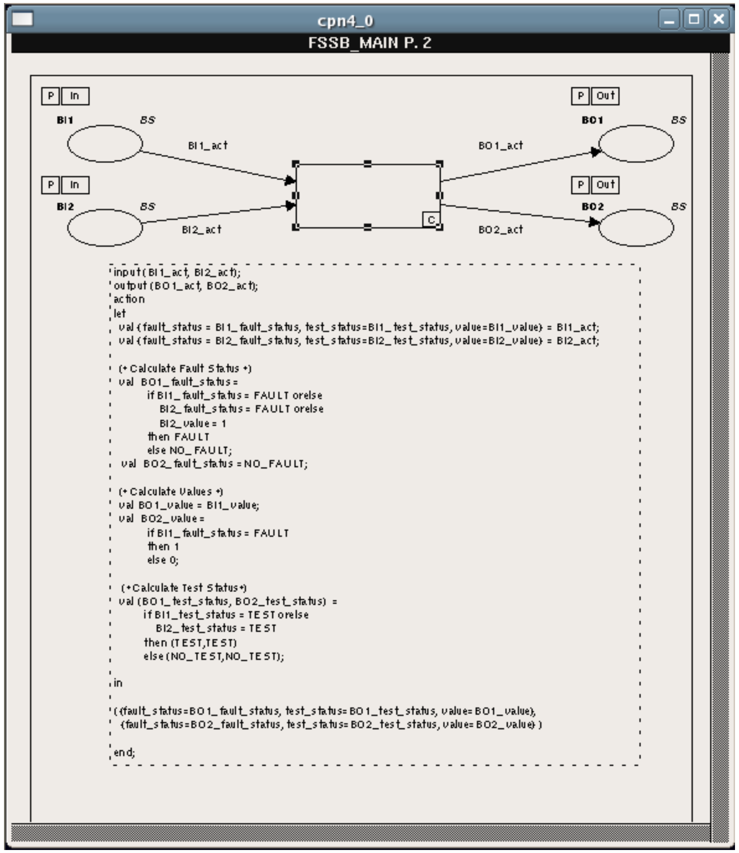
Színezett Petri-hálók felépítésének lehetőségei

- CPN hálók: struktúra és adattartalom is lehet
- Szélsőségek
  - Tisztán strukturális információ, nincs adattartalom:
    - Közönséges Petri-háló
  - Nincs struktúra, csak adattartalom (adat és vezérlési információ):
    - Egy hely + egy tranzíció, komplex színosztályok, függvények és élkifejezések
- Kompromisszumra van szükség
  - Érthető, áttekinthető struktúrájú CPN háló legyen

Példa: Modellezési lehetőségek



Vezérlési folyam struktúrában kifejezve

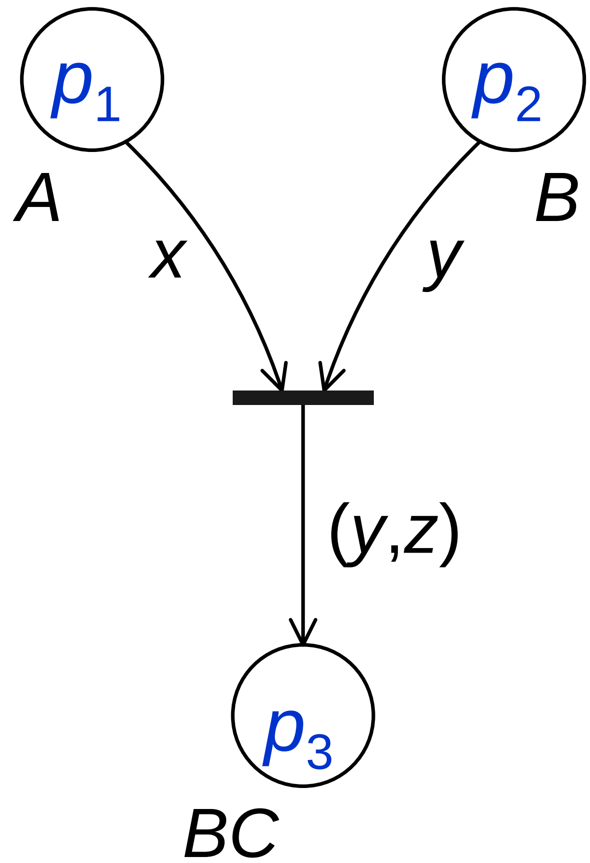


Ugyanez csak kódban („összehajtogatva”)

Széthajtogatás

- (Prioritásos) színezett hálók modellező ereje megfelel a tiltó éllel kiegészített (prioritásos) színezetlen hálókénak
  - Minden színezett hálónak megfeleltethető egy ekvivalens működésű színezetlen háló (automataelméleti értelemben: lépésekre van biszimuláció)
  - Ekvivalens színezetlen háló neve: ”széthajtogatott” háló
  - Széthajtogatás:
    - Tokenek adattartalmát struktúrában fejezzük ki
    - Minden eseménynek (tüzelésnek) a színezett hálóban megfelel egy és csak egy esemény (tüzelés) a széthajtogatott hálóban

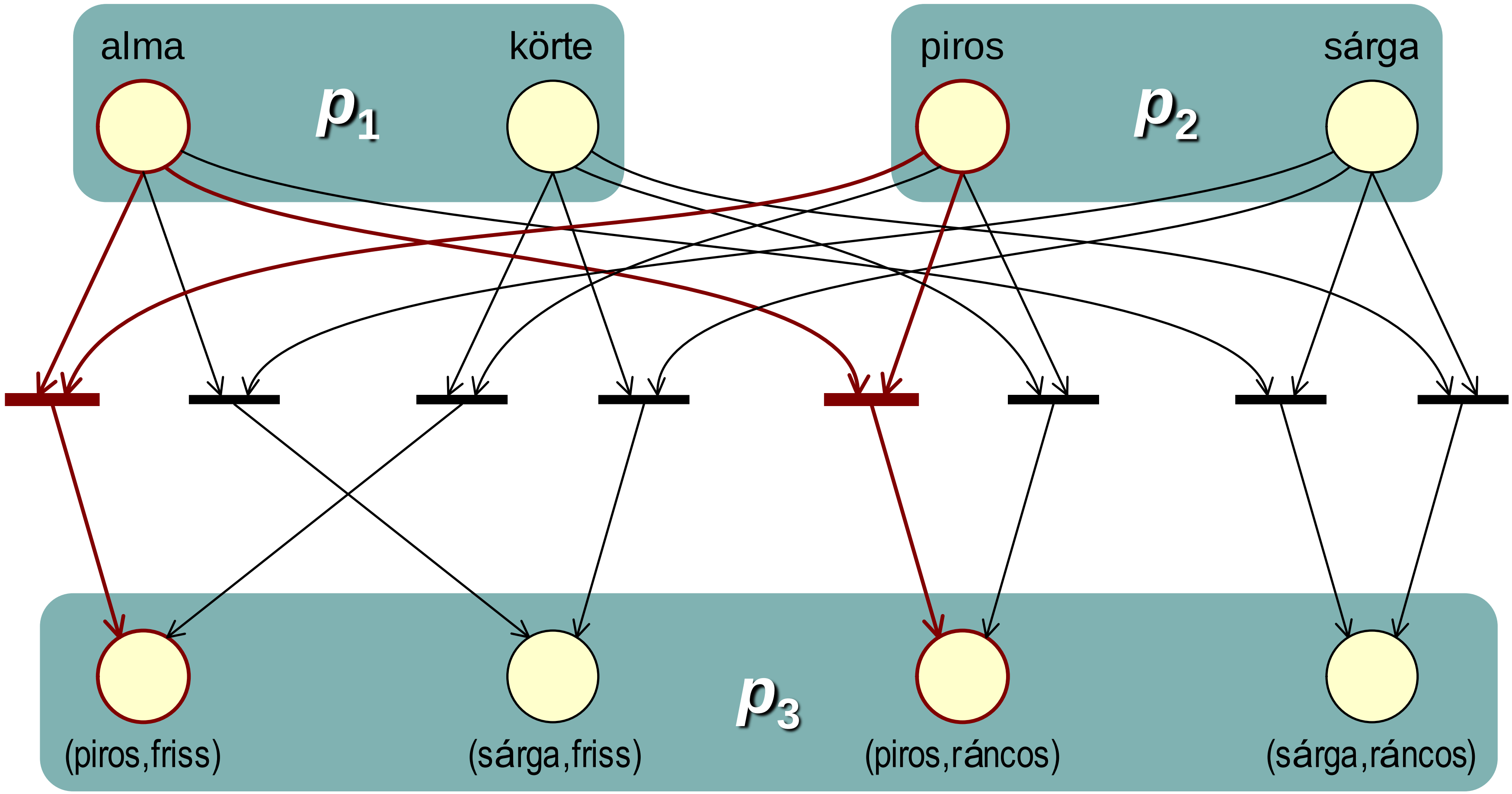
Egyszerű színezett háló



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;

- CPN hely → PN helyek a színosztály elemei szerint
- CPN tranzíció → PN tranzíciók a lekötések szerint

A széthajtogatott, színezetlen háló



- CPN hely → PN helyek a színosztály elemei szerint
- CPN tranzíció → PN tranzíciók a lekötések szerint

Hierarchikus színezett Petri-hálók

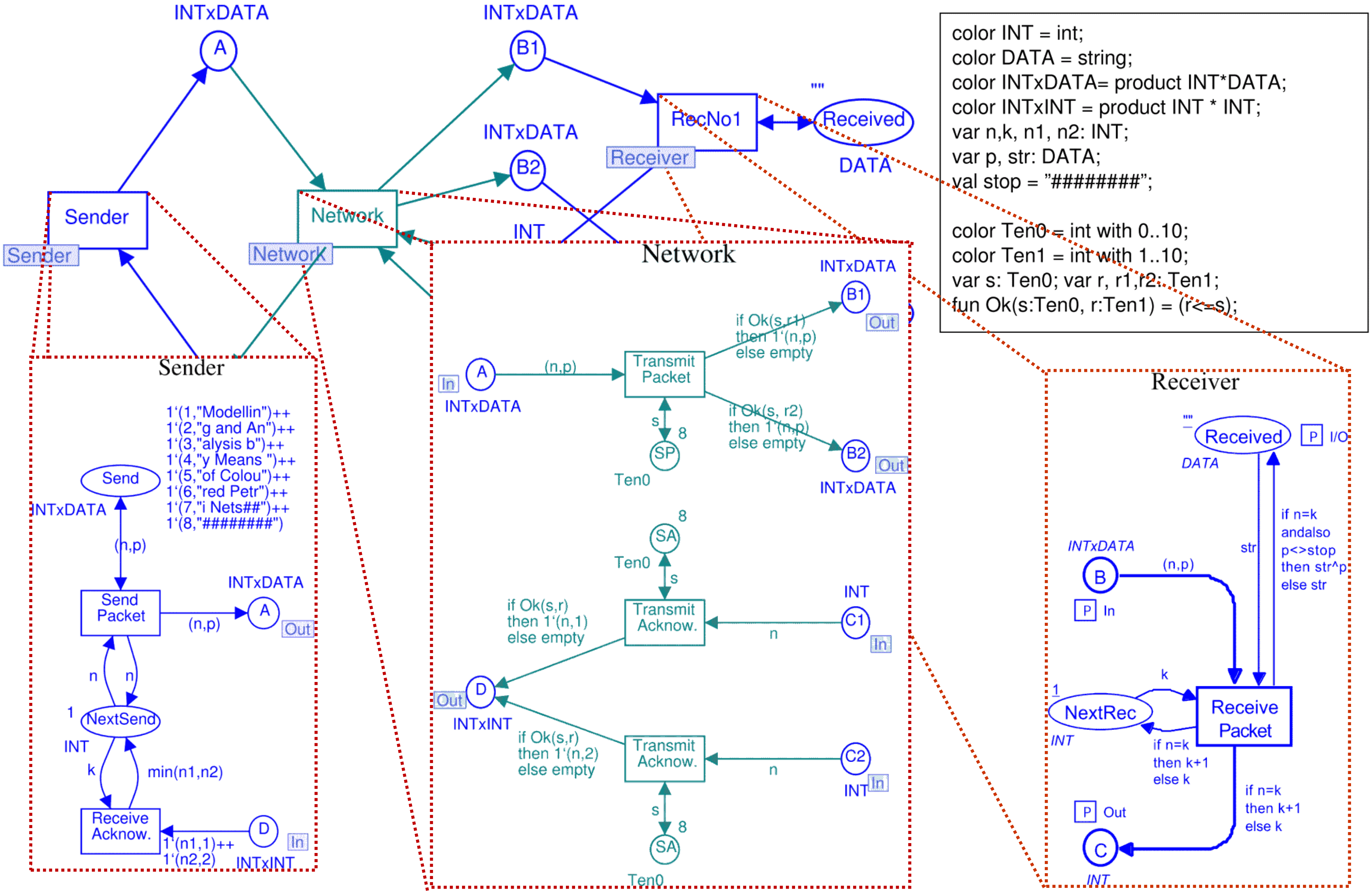
Hierarchikus színezett Petri-hálók

- Alhálók integrálása egyetlen összetett CPN hálóvá hierarchikus rendszerben
  - Színezett Petri-háló modellek (alhálók): Lapok
    - Hivatkozható a lap neve vagy száma
    - A lapok példányosíthatók (a hierarchia bármelyik szintjén)
    - A jelölés (tokeneloszlás) minden példányra egyedi
  - Hierarchia: Lapok struktúrája
    - Fő (prime, top) lap: legfelső szint
    - Másodlagos lap példányok (al-lapok)
      - Azonosítás: lap-példány azonosító szám
      - Laphierarchia gráf

Hierarchikus felépítés eszközei

1. Helyettesítő tranzíciók
   – Egy al-lap megjelenítése
   – Interfészek a lapok között: Helyek
     1. Főlapon: „Socket” helyek $\rightarrow$ az alhálók beillesztési pontjai
     2. Al-lapon: „Port” helyek $\rightarrow$ ezekkel kapcsolódik az alháló; port típus: bemenet, kimenet, I/O (kétirányú), általánosított
2. Fúziós helyek
   – Azonos névvel, több példányban létrehozott helyek, amik ugyanazon helyet jelölik a háló több pontján
   – A tokenek egyszerre kerülnek be / távolítódnak el egy adott fúziós helyhez tartozó helyhalmazba / helyhalmazból

Példa: Egy egyszerű protokoll hierarchikus változata



color INT = int;
color DATA = string;
color INTxDATA= product INT*DATA;
color INTxINT = product INT * INT;
var n,k, n1, n2: INT;
var p, str: DATA;
val stop = "########";

color Ten0 = int with 0..10;
color Ten1 = int with 1..10;
var s: Ten0; var r, r1,r2: Ten1;
fun Ok(s:Ten0, r:Ten1) = (r<=s);

Összefoglalás

- Színezett Petri-hálók felépítése
  - Színhalmazok, élkifejezések, őrfeltételek
- Színezett Petri-hálók működése
  - Engedélyezettség, lekötések, tüzelések
  - Elérhetőségi gráf
- Formális definíciók és szemantika
- Dinamikus tulajdonságok
- Strukturális tulajdonságok
- Széthajtogatás
- Hierarchikus hálók