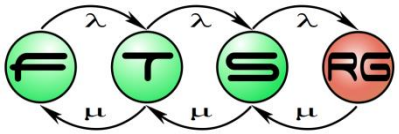
Teljesítménymodellezés

Budapest University of Technology and Economics
Fault Tolerant Systems Research Group



Emlékeztető

- **Egyensúlyi állapot:**
  - Átlagos értékekkel számolunk
  - $\lambda = X$ (érkezési ráta = átbocsátás)
- **Átbocsátóképesség (X^{max}):**
  - Az elérhető maximális átbocsátás
    T átlagos feldolgozási idő mellett
  - $X^{max} = \frac{K}{T}$ (K darab erőforráspéldány esetén)
- **Kihasználtság (U):**
  - Az átbocsátás és az átbocsátóképesség aránya
  - $U = \frac{X}{K} \times T$ (K erőforráspéldány esetén)

A Little-törvény

A Zipf-törvény

Terhelés változása

TARTALOM

- Alapfogalmak
- Terhelési diagram
- Erőforrásmodellezés
- **Folyamatmodellek elemzése**
- A Little-törvény

FOLYAMATMODELLEK ELEMZÉSE

Hogyan számoljuk ki egy összetett folyamat átbocsátóképességét?

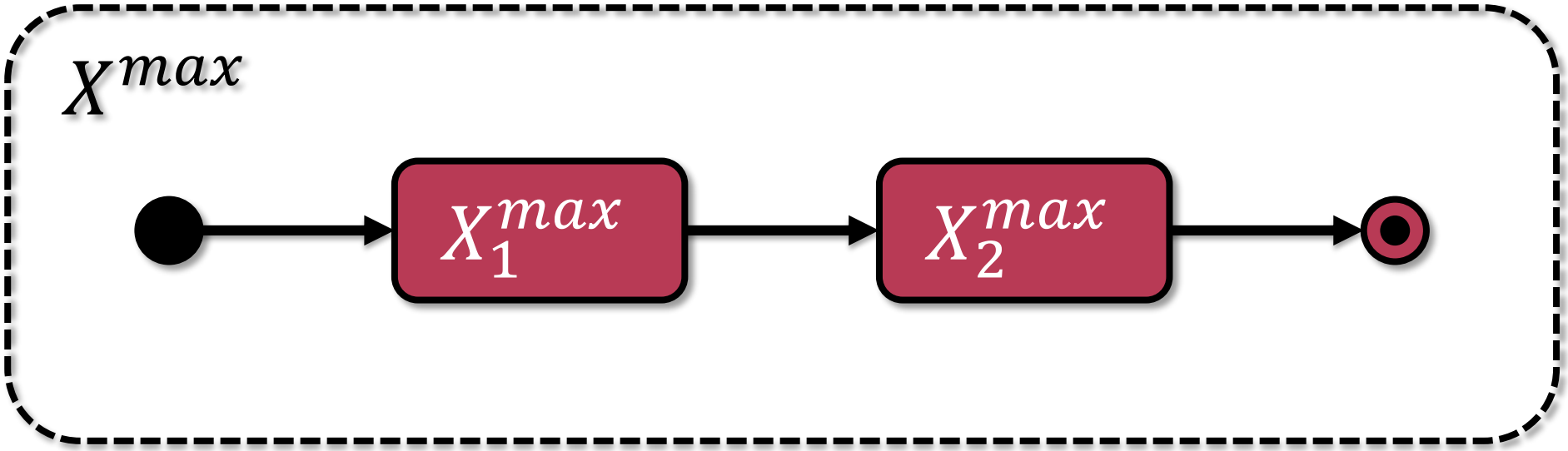
Mit tudunk tervezéskor?

- Általában tevékenységekhez rendelünk erőforrást
  - A tevékenység (átlagos) végrehajtási ideje is adott

    → X^{max} számítható a tevékenységre
- Pl. Neptunban a tárgyfelvétel az adatbázisszervert terheli 100 ms-ig
  - $\mathrm{T} = 100\ \mathrm{ms}$
  - $X^{max} = \frac{1}{T} = 10\ \frac{tárgyfelvétel}{másodperc}$

Adott folyamatmodell (pl. a felhasználói viselkedés) ismeretében mi a teljes rendszer átbocsátóképessége?

Szekvenciális komponálás

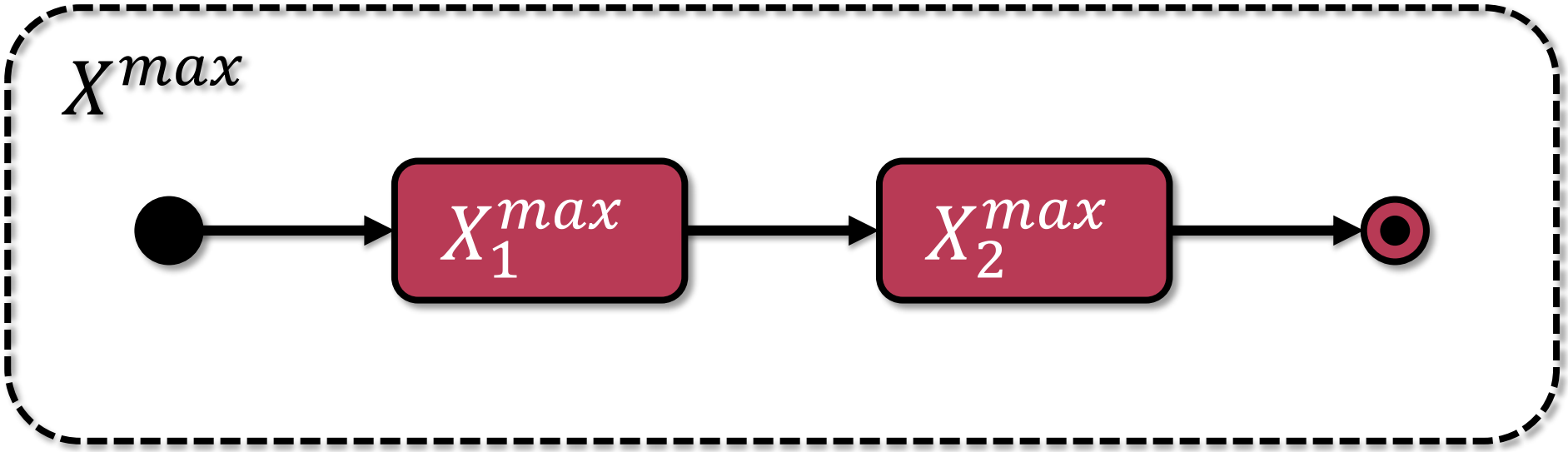


$$X^{max} = min(X_1^{max}, X_2^{max})$$

Hiába gyors az egyik tevékenység, a tokenek feltorlódnak a másik előtt

Pl. Okmányiroda
Sorszámhúzás (300 db/óra),
Ügyintézés (2 db/óra)

Szekvenciális komponálás

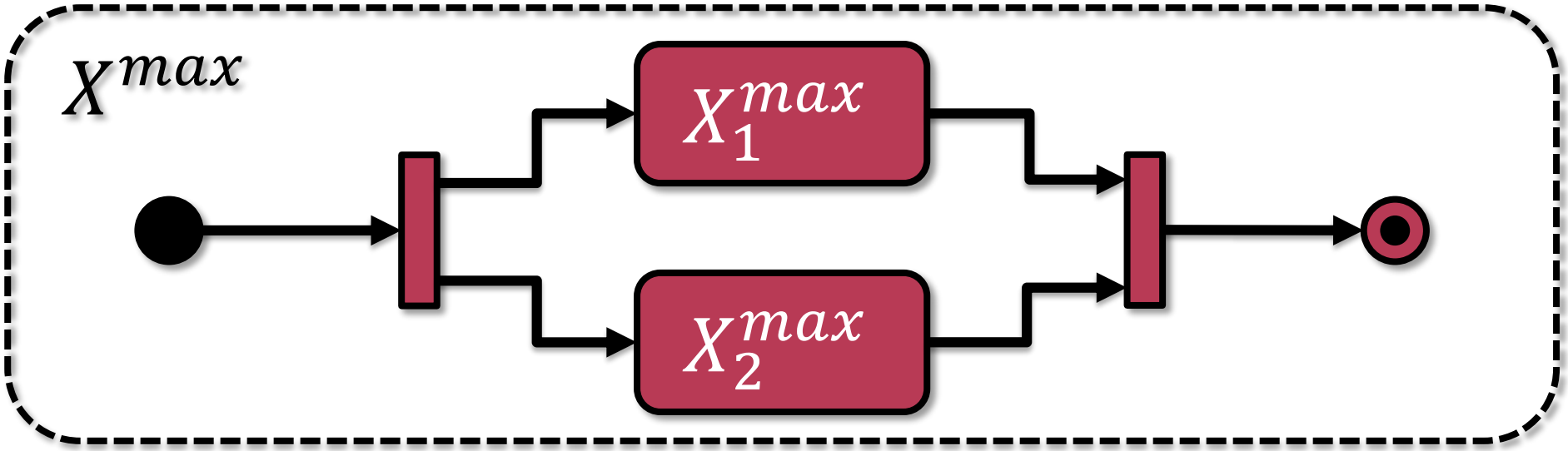


$$X^{max} = min(X_1^{max}, X_2^{max})$$

Szűk keresztmetszet:

A minimumhelyet adó tevékenység (vagy az ahhoz rendelt erőforrás).

Párhuzamos komponálás

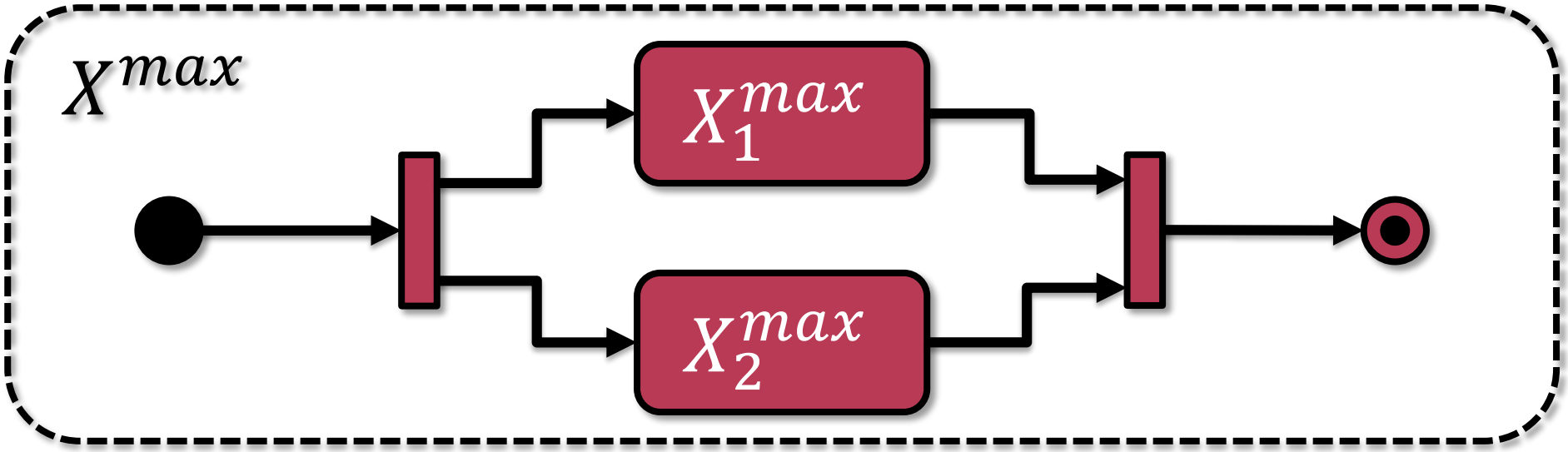


$$X^{max} = min(X_1^{max}, X_2^{max})$$

Hiába gyors az egyik tevékenység, a tokeneknek be kell várniuk egymást

Pl. ZH javítás:
Beugró (30 db/óra),
Nagyfeladat (12 db/óra)

Párhuzamos komponálás

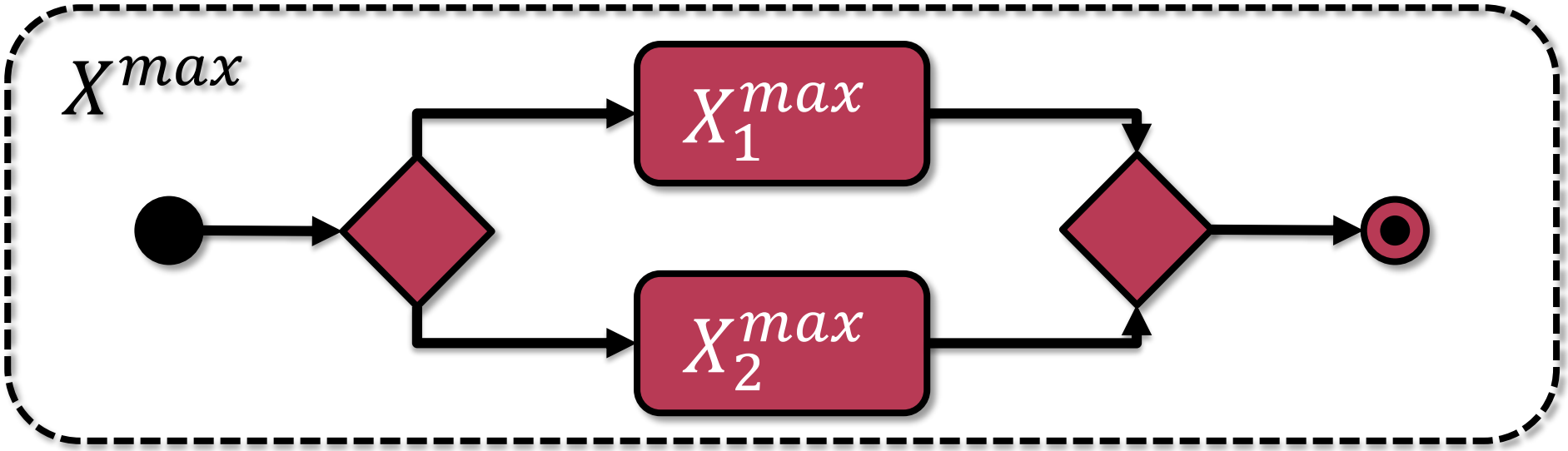


$$X^{max} = min(X_1^{max}, X_2^{max})$$

Szűk keresztmetszet:

A minimumhelyet adó tevékenység (vagy az ahhoz rendelt erőforrás).

Komponálás szabad választással

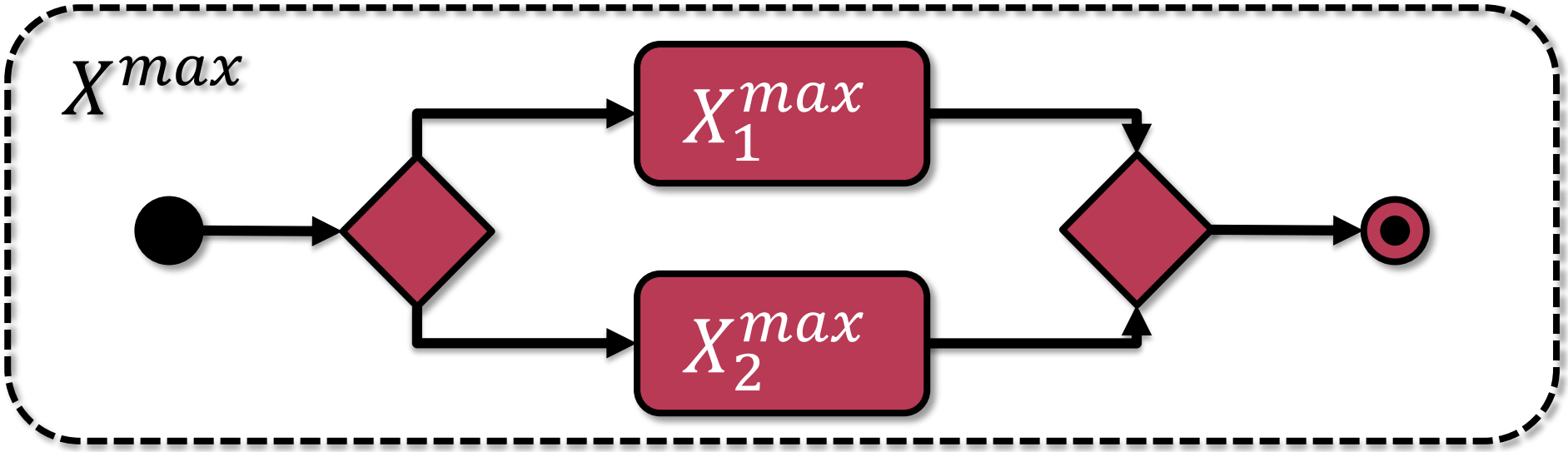


$$X^{max} = X_1^{max} + X_2^{max}$$

A tokenek mindkét irányba mehetnek: ha az egyik tevékenység telítésben van, a másik még fogadhat tokent.

Pl. Áruház:
K db pénztár,
mind 10 db/óra

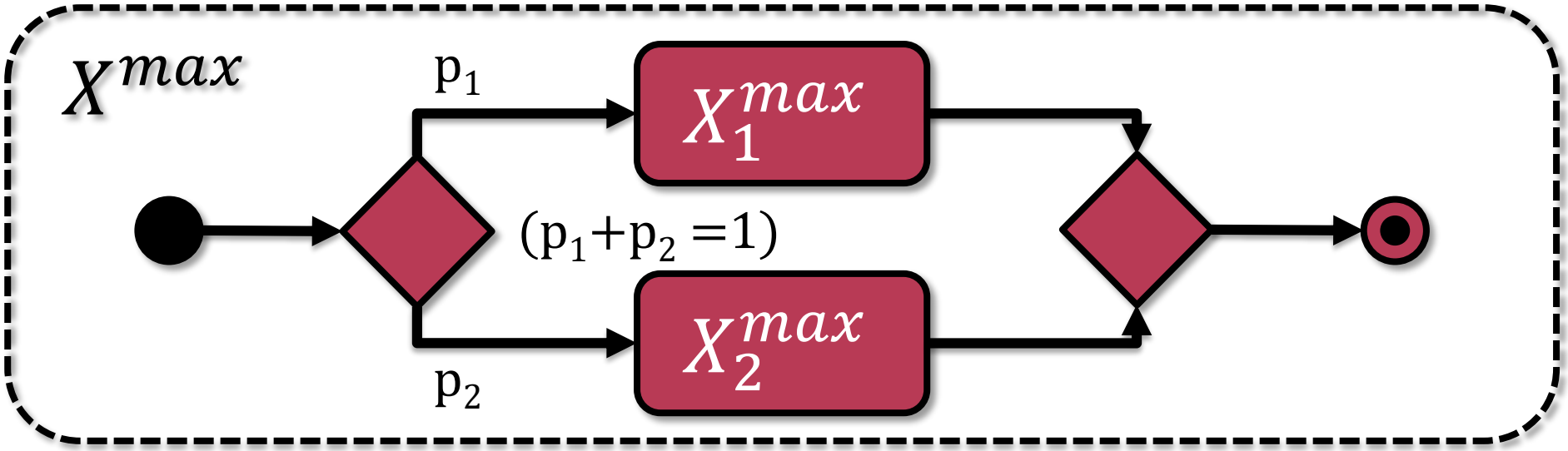
Komponálás szabad választással



$$X^{max} = X_1^{max} + X_2^{max}$$

Megj: van, amikor a lépéshez rendeljük
majd az erőforrások számát, és nem ugyanazt a logikai lépést
tüntetjük fel többször a modellben (ld. Szimuláció előadás).
Feltétel: minden erőforráson ugyanannyi ideig tartson

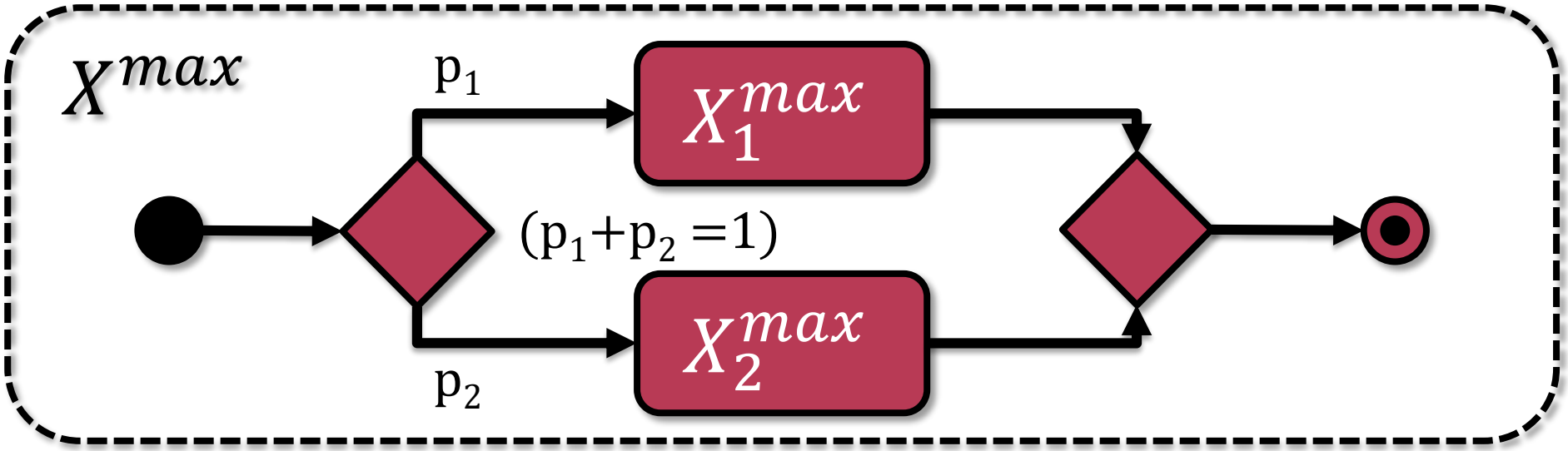
Komponálás kötött arányú választással



$$X^{max} = min(\frac{1}{p_1} \times X_1^{max}, \frac{1}{p_2} \times X_2^{max})$$

A tokenek p_1 és p_2 valósínűséggel választják az első ill. második tevékenységet. A teljes folyamatból tehát $\frac{1}{p_1}$ ill. $\frac{1}{p_2}$ tokenből egy jut az első ill. második tevékenységre.

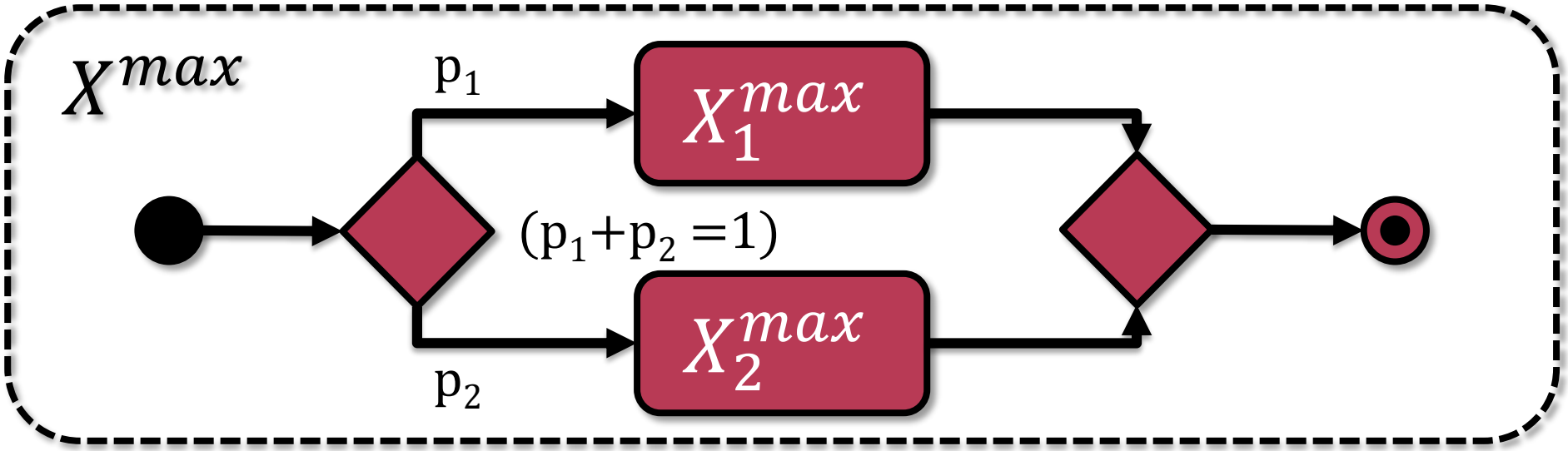
Komponálás kötött arányú választással



$$X^{max} = min(\frac{1}{p_1} \times X_1^{max}, \frac{1}{p_2} \times X_2^{max})$$

Pl. Felhasználó viselkedése egy weblapon:
20% eséllyel vásárol (20 db/s),
80% eséllyel elvet (200 db/s)

Komponálás kötött arányú választással

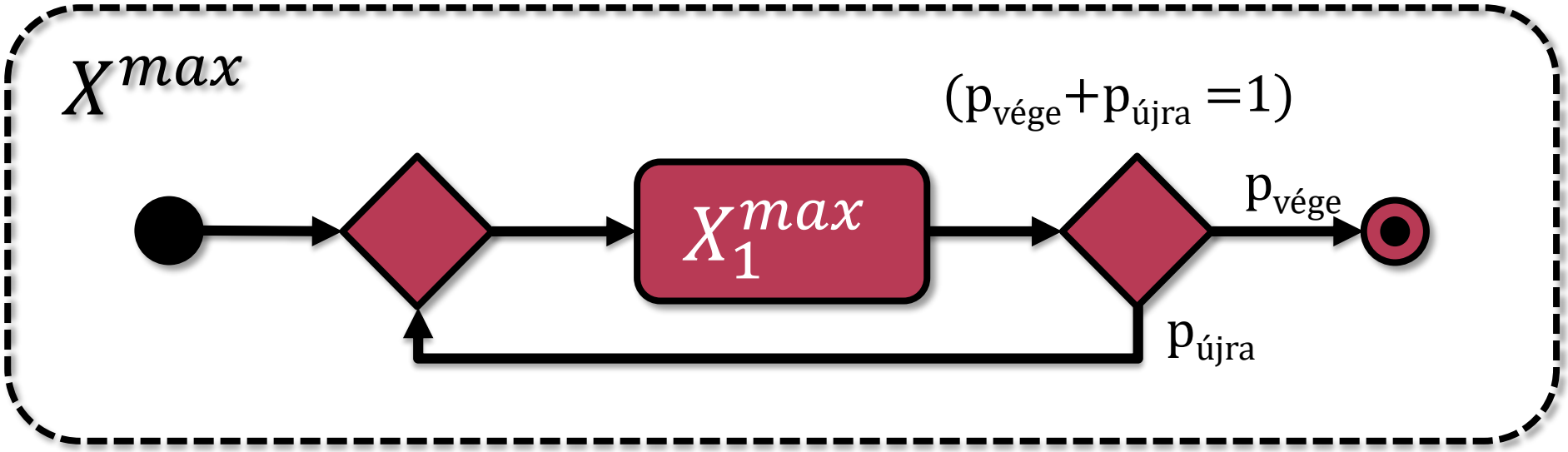


$$X^{max} = min\left(\frac{1}{p_1} \times X_1^{max}, \frac{1}{p_2} \times X_2^{max}\right)$$

Szűk keresztmetszet:

A minimumhelyet adó tevékenység (vagy az ahhoz rendelt erőforrás).

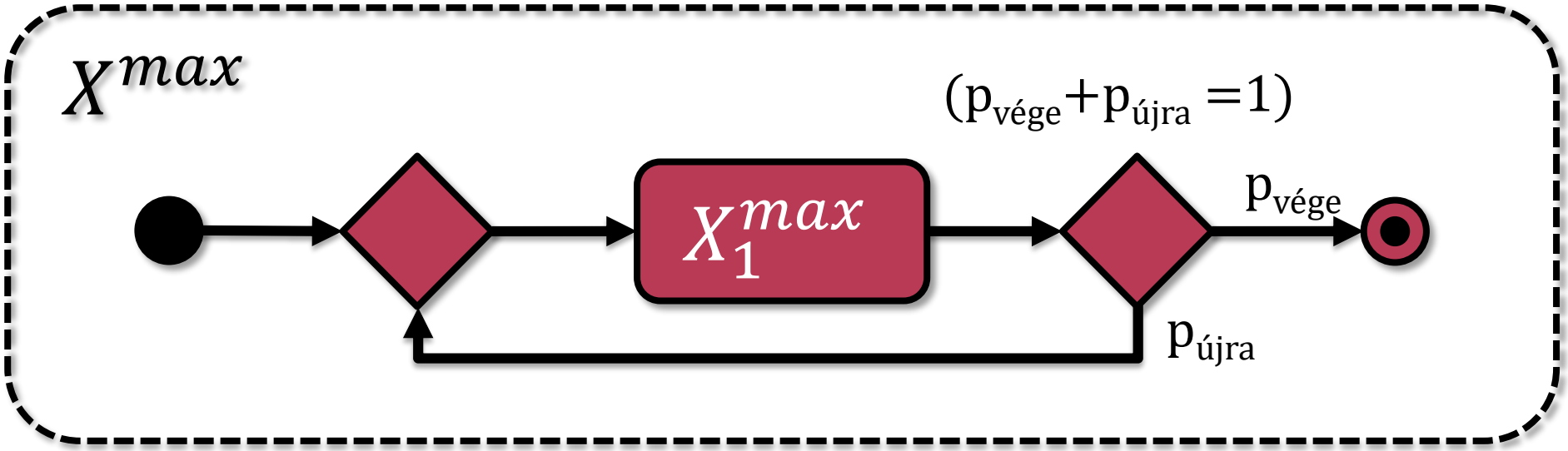
Komponálás ciklussal



$$X^{max} = \frac{1}{\frac{1}{p_{vége}}} \times X_1^{max} = p_{vége} \times X_1^{max}$$

Az $\frac{1}{p_{vége}}$ érték az iterációk várható száma
(lásd később a Valószínűségszámítás tantárgyban).

Komponálás ciklussal



$$X^{max} = \frac{1}{\frac{1}{p_{vége}}} \times X_1^{max} = p_{vége} \times X_1^{max}$$

Pl. Felhasználó a rendszerben:
10% eséllyel kilép, 90% eséllyel új kérés
15 kérés/s, átlagosan 10 kérés/munkamenet

Vizitációs szám

Választás: $$X^{max} = min(\frac{1}{p_1} \times X_1^{max}, \frac{1}{p_2} \times X_2^{max})$$

Ciklus: $$X^{max} = \frac{1}{\frac{1}{p_{vége}}} \times X_1^{max} = p_{vége} \times X_1^{max}$$

- **Vizitációs szám:** megmutatja, hogy a folyamat végrehajtása során átlagosan hányszor fut le az adott tevékenység/alfolyamat.
  - Választás esetén maga a döntési valószínűség
  - Ciklus esetén a várható iterációk száma

Vizitációs szám

Átbocsátóképesség a vizitációs szám ismeretében:

$$X^{max} = \frac{1}{v} \times X_1^{max}$$

Folyamat

Elemi tevékenység (taszk)

- **Vizitációs szám:** megmutatja, hogy a folyamat végrehajtása során átlagosan hányszor fut le az adott tevékenység/alfolyamat.
  - Választás esetén maga a döntési valószínűség
  - Ciklus esetén a várható iterációk száma

Vizitációs szám

Átbocsátóképesség a vizitációs szám ismeretében:

$$\frac{1}{X^{max}} = v \times \frac{1}{X_1^{max}}$$

- **Vizitációs szám:** megmutatja, hogy a folyamat végrehajtása során átlagosan hányszor fut le az adott tevékenység/alfolyamat.
  - Választás esetén maga a döntési valószínűség
  - Ciklus esetén a várható iterációk száma

Vizitációs szám

Végrehajtási idő a vizitációs szám ismeretében:

$$T_{folyamat} = v \times T_{taszk}$$

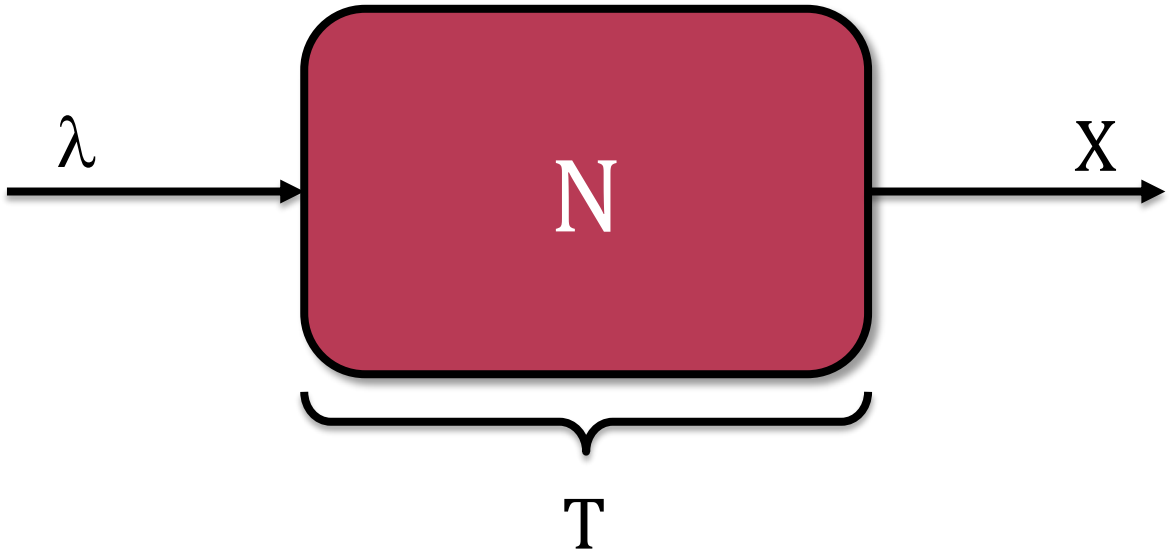
- **Vizitációs szám:** megmutatja, hogy a folyamat végrehajtása során átlagosan hányszor fut le az adott tevékenység/alfolyamat.
  - Választás esetén maga a döntési valószínűség
  - Ciklus esetén a várható iterációk száma

A LITTLE-TÖRVÉNY

Avagy az alapképlet

A Little-törvény

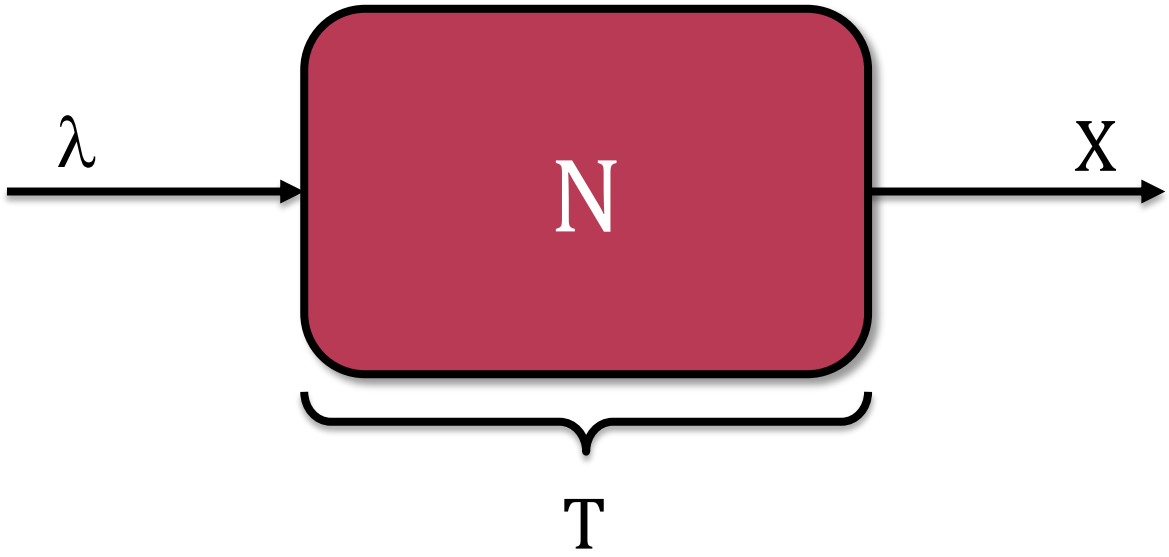
- λ: érkezési ráta $\left[\frac{darab}{s}\right]$
- X: átbocsátás $\left[\frac{darab}{s}\right]$
- T: rendszerben töltött idő $[s]$
- N: rendszerben lévő tokenek száma $[darab]$



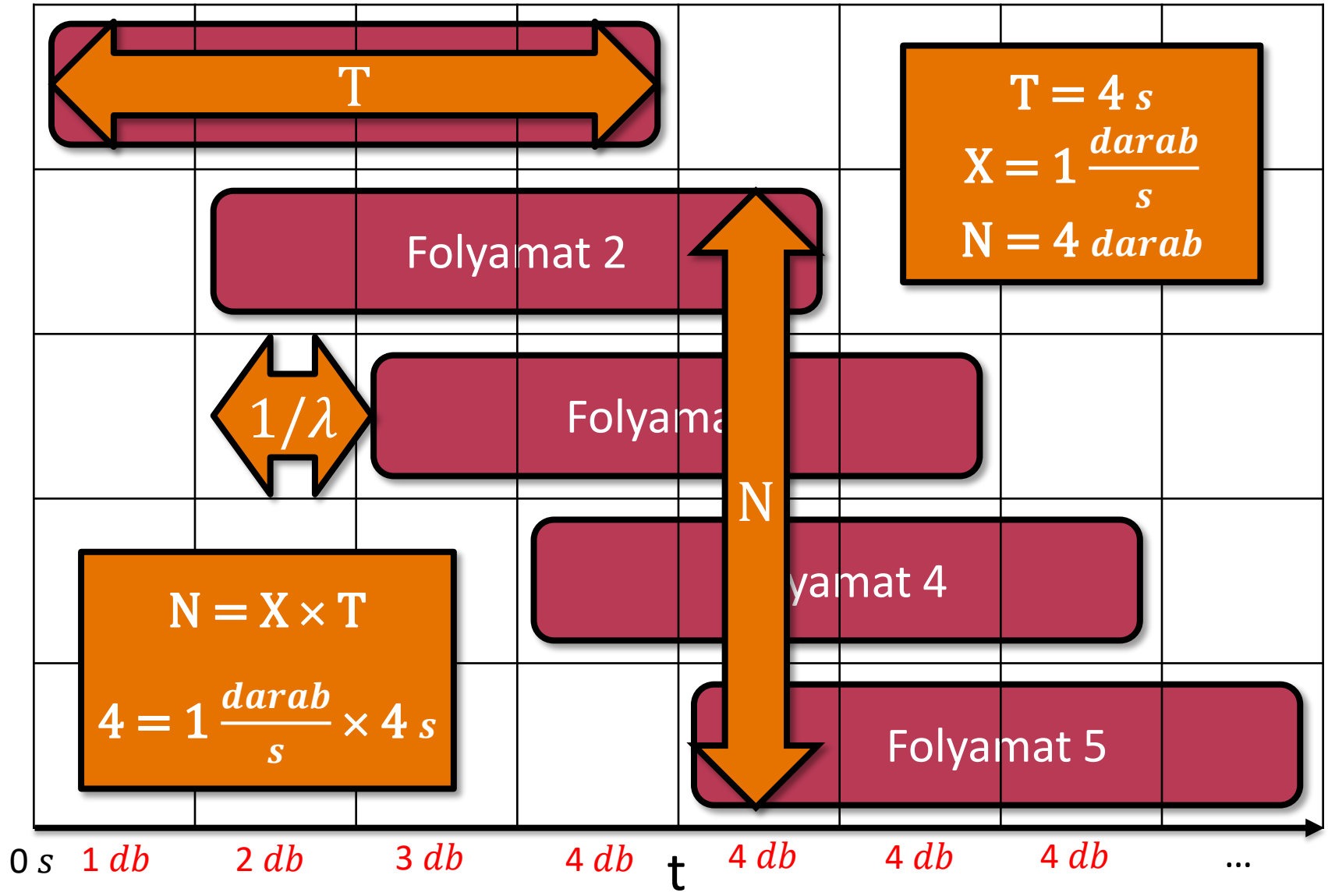
A Little-törvény

- **Egyensúlyi állapotban** ($\lambda = X$) igaz:

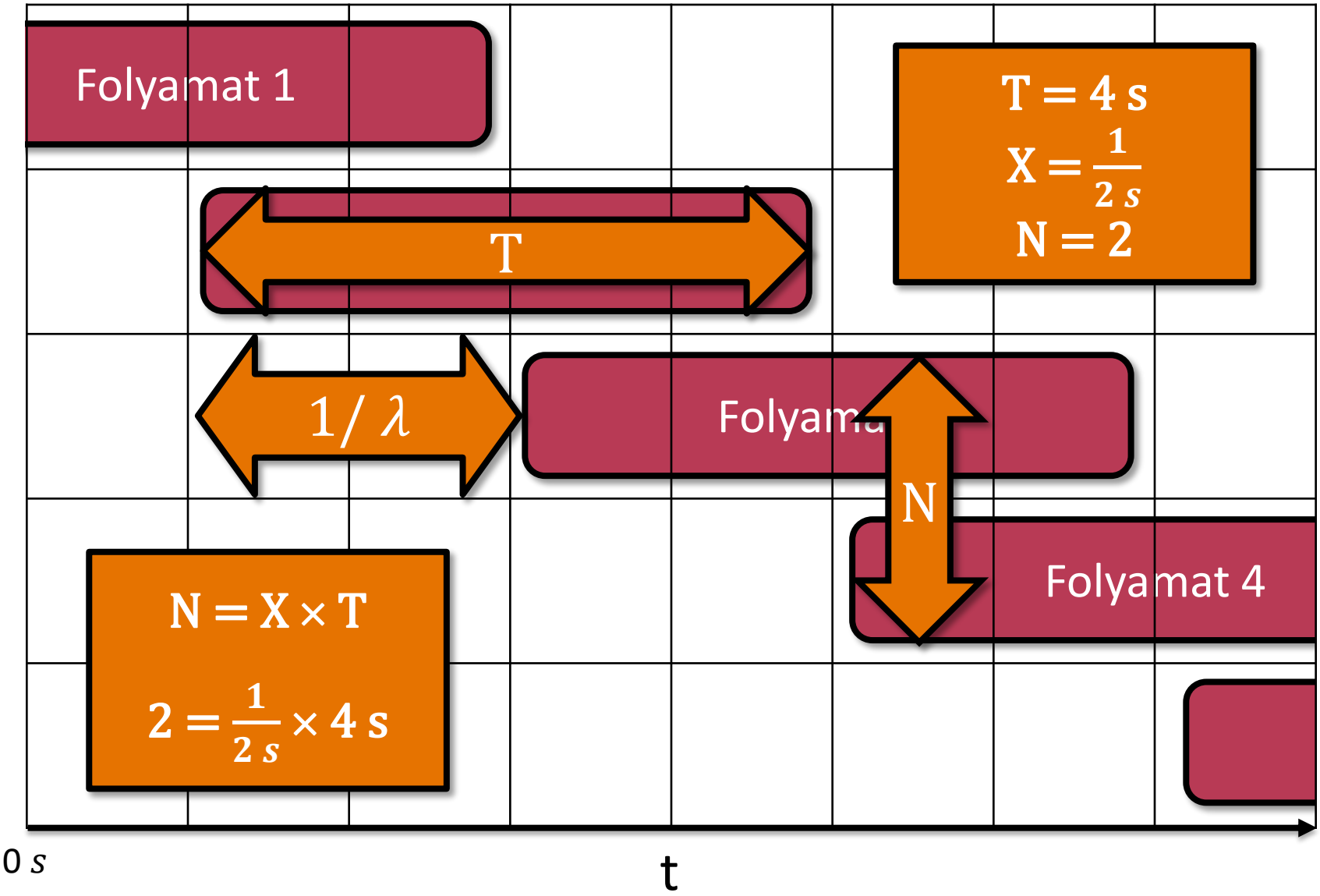
$$N = X \times T$$



A Little-törvény szemléltetése



A Little-törvény szemléltetése



Kihasználtság és a Little-törvény

- K darab erőforráspéldány:
  maximum K darab kérés végrehajtás alatt
- A Little-törvény:
  végrehajtás alatt álló kérések száma (N)?
- Levezethető az átlagos kihasználtság:

$$U = \frac{X}{K} \times T = \frac{X \times T}{K} = \frac{N}{K}$$

Kihasználtság K darab erőforráspéldányra

Little-törvény ($N = X \times T$)

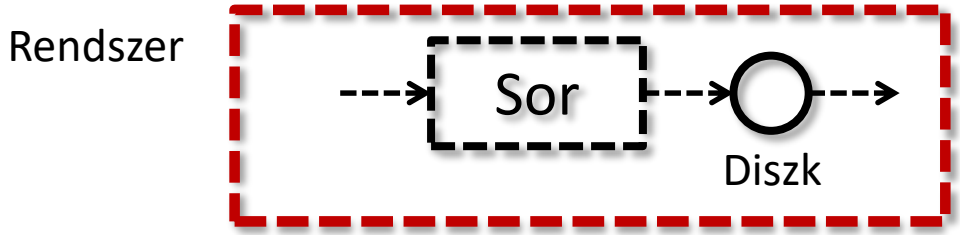
LITTLE TÖRVÉNY: GYAKORLATI PÉLDÁK

Példa

- Erőforrás: diszk
- 40 kérést szolgál ki másodpercenként (nincs átlapolódás)
- 1 kérés kiszolgálása átlagosan 0,0225 másodpercig tart
- Mekkora a kihasználtság?

$$U = X \times T_{diszk} = 40 \frac{kérés}{s} \times 0{,}0225\ s = 0{,}9 = 90\%$$

Példa

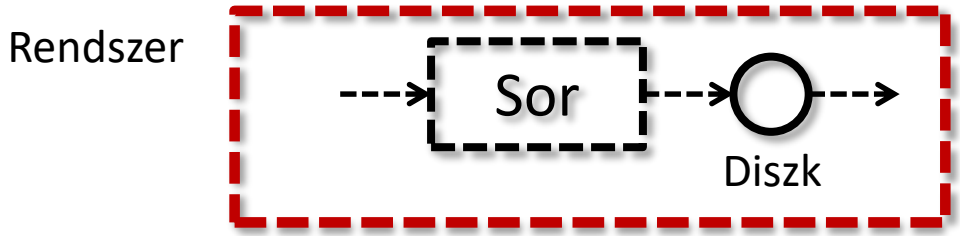


- Sorban állás is van a diszk előtt
- Diszk: 40 $kérés/s$
- Kérések átlagos száma a rendszerben: 4

Átlagos rendszerben tartózkodási idő? ($T_{rendszer}$)

Átlagos sorban állási idő? ($T_{várakozás}$)

Példa



- Sorban állás is van a diszk előtt
- Diszk: 40 $kérés/s$
- Kérések átlagos száma a rendszerbe

Sorbanállási és diszk kiszolgálási idő

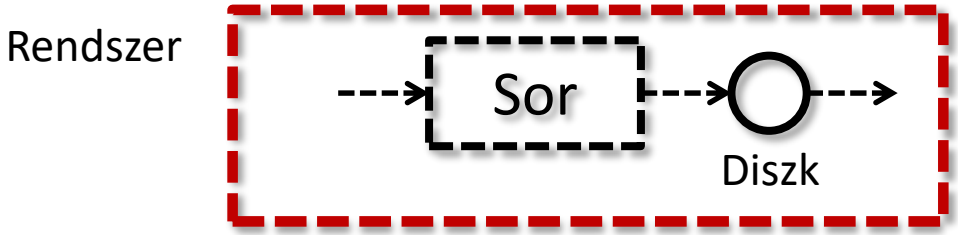
Rendszer

$$N = X \times T \rightarrow T_{rendszer} = 4\ kérés\ /\ 40\ \frac{kérés}{s} = 0{,}1\ s$$

Átlagos sorban állási idő?

$$(T_{rendszer} - T_{diszk}) = (0{,}1\ s - 0{,}0225\ s) = 0{,}0775\ s$$

Példa



- Sorban állás is van a diszk előtt
- Diszk: 40 $kérés/s$. Átlagosan 0,9 $darab$
- Kérések átlagos száma a rendszerben: 4 $darab$

Kérések átlagos száma a sorban?

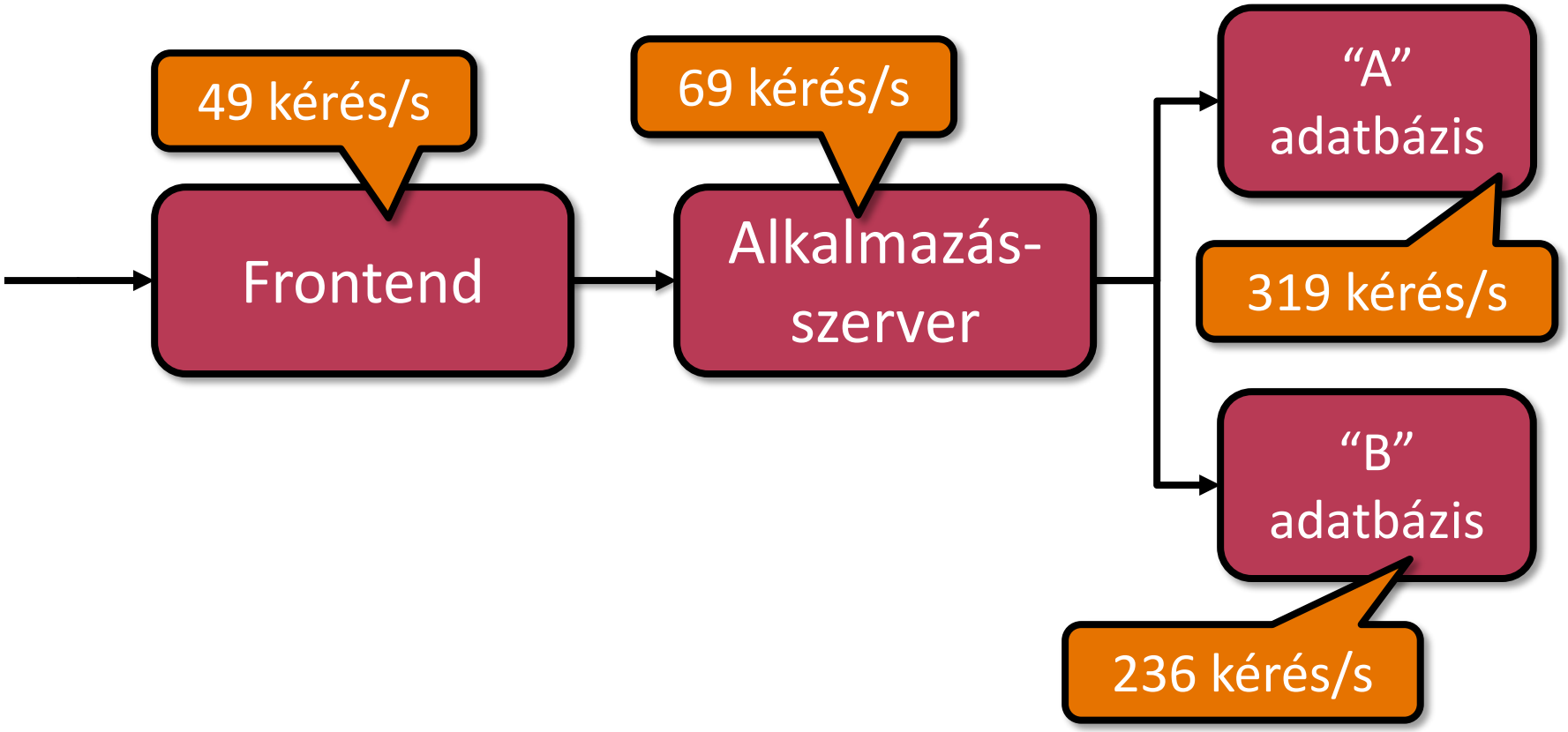
$(N_{rendszer} - N_{diszk})$

$4\ darab - 0{,}9\ darab = 3{,}1\ darab$

Little törvény a gyakorlatban

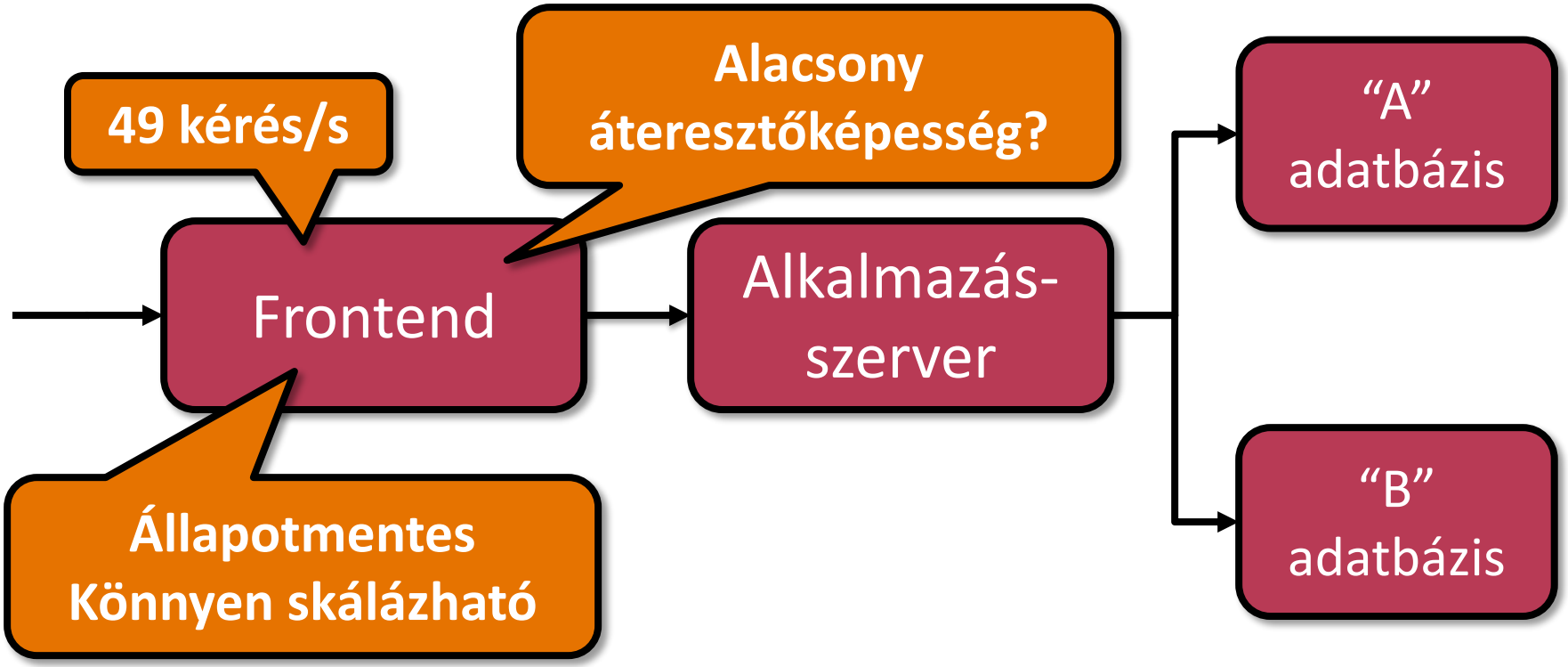
- Szimuláció
  - Dobson&Shumsky
  - https://youtu.be/UjzXQPGBaNA
- Miért oktatják(juk)
  - http://pubsonline.informs.org/doi/pdf/10.1287/ited.7.1.106
- Példák
  - http://web.mit.edu/sgraves/www/papers/Little's%20Law-Published.pdf

Teljesítmény 3 rétegű architektúrában



A mérőszámok itt az egész rendszerre érkező terhelésre utalnak! Például az „A. adatbázis” akkor válik szűk keresztmetszetté, ha a rendszerbe 319 lekérdezés érkezik másodpercenként.

Teljesítmény 3 rétegű architektúrában

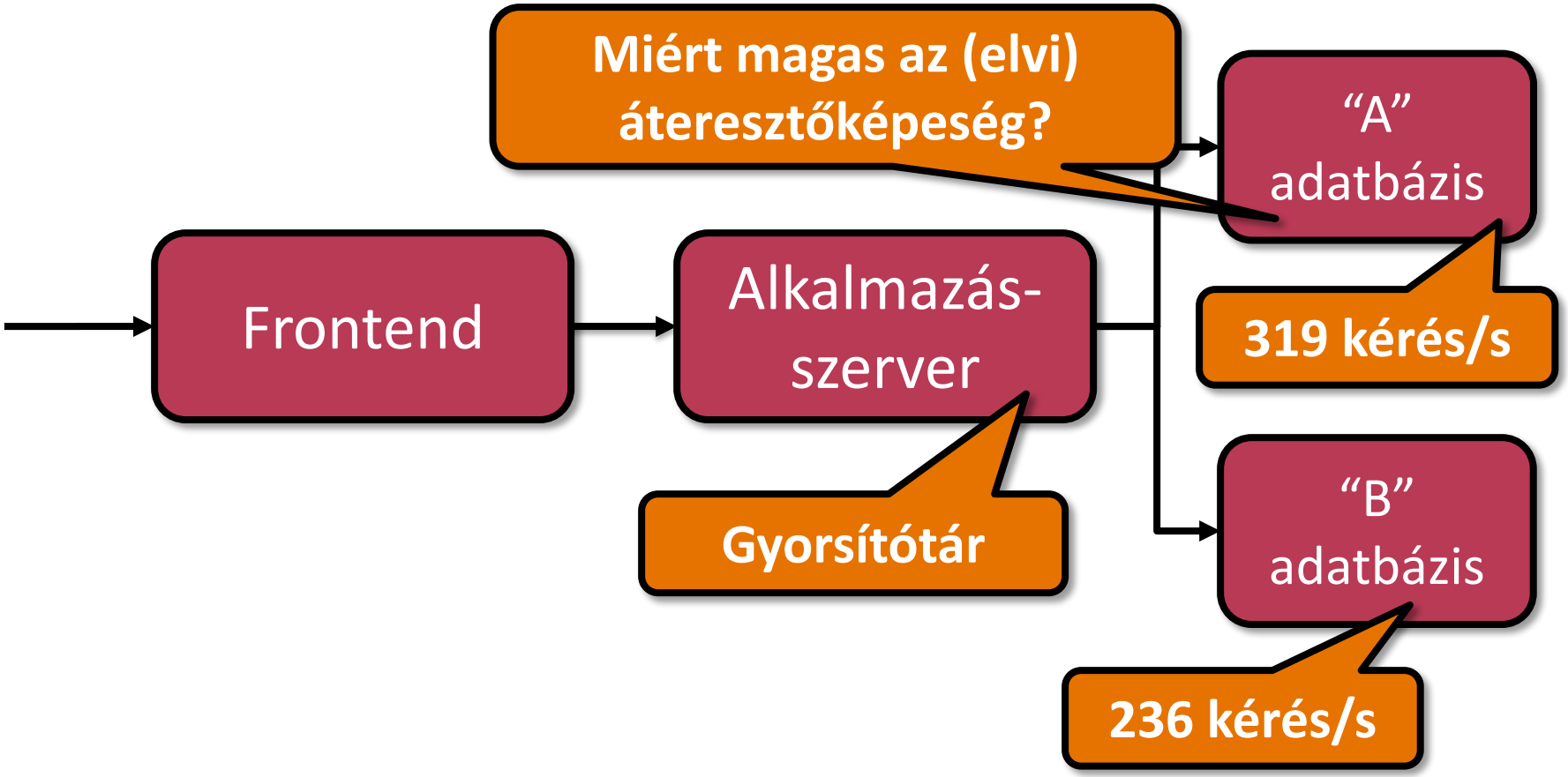


A mérőszámok itt az egész rendszerre érkező terhelésre utalnak! Például az „A. adatbázis” akkor válik szűk keresztmetszetté, ha a rendszerbe 319 lekérdezés érkezik másodpercenként.

Teljesítmény 3 rétegű architektúrában

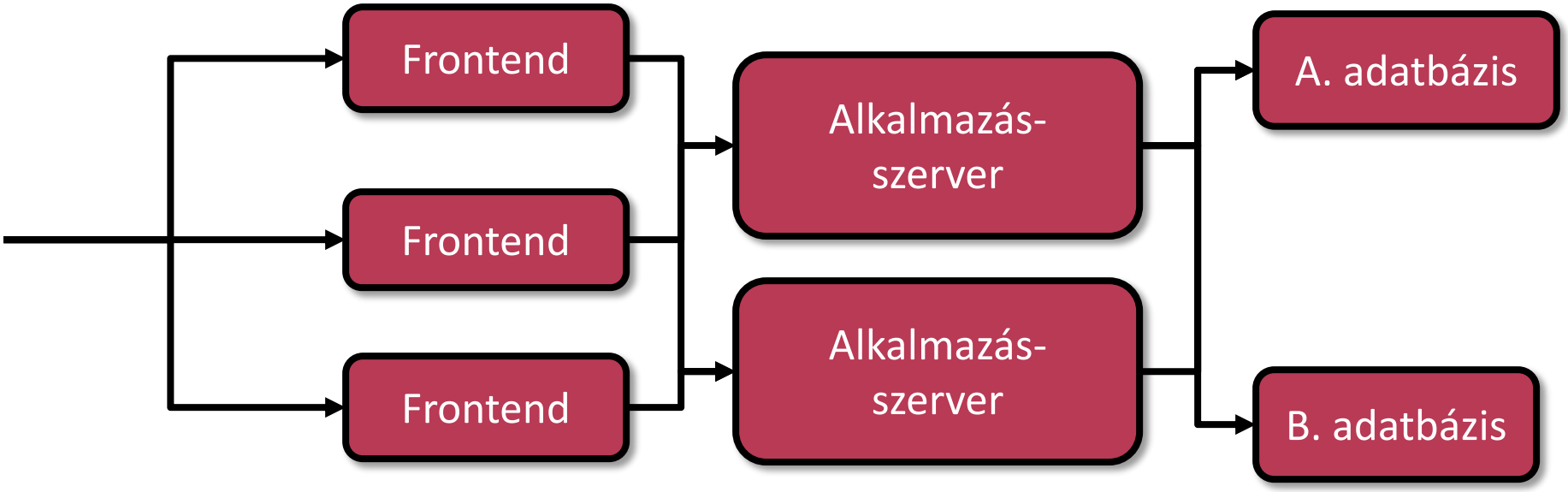
A mérőszámok itt az egész rendszerre érkező terhelésre utalnak! Például az „A. adatbázis" akkor válik szűk keresztmetszetté, ha a rendszerbe 319 lekérdezés érkezik másodpercenként.

Teljesítmény 3 rétegű architektúrában



A mérőszámok itt az egész rendszerre érkező terhelésre utalnak! Például az „A. adatbázis” akkor válik szűk keresztmetszetté, ha a rendszerbe 319 lekérdezés érkezik másodpercenként.

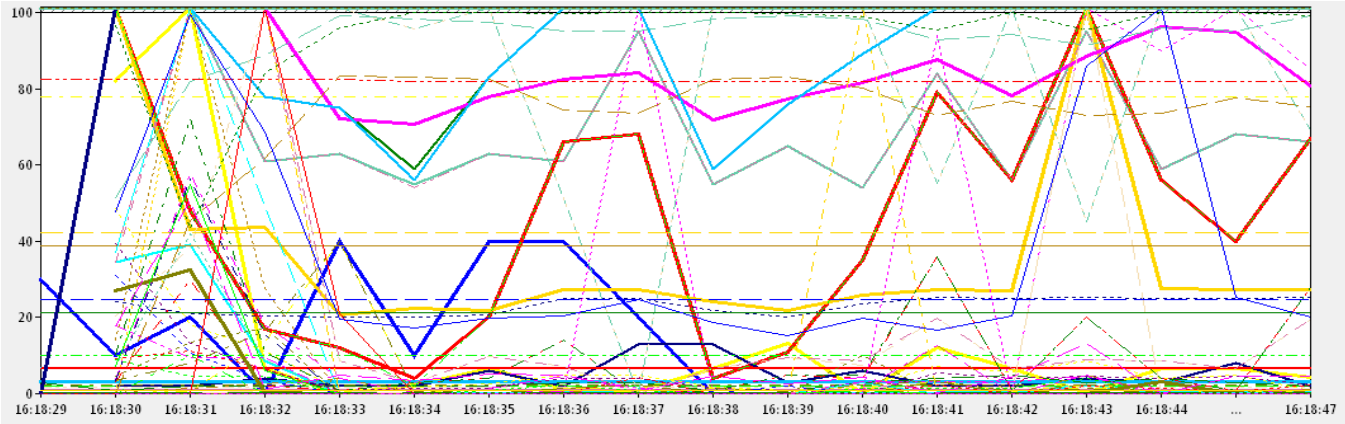
3 rétegű architektúra a valóságban



(Példa: Technológiai háttér érdeklődőknek:)
http://www.projectclearwater.org/wp-content/uploads/2013/05/Clearwater-Deployment-Sizing-10-Apr-13.xlsx
http://www.projectclearwater.org/technical/clearwater-performance/

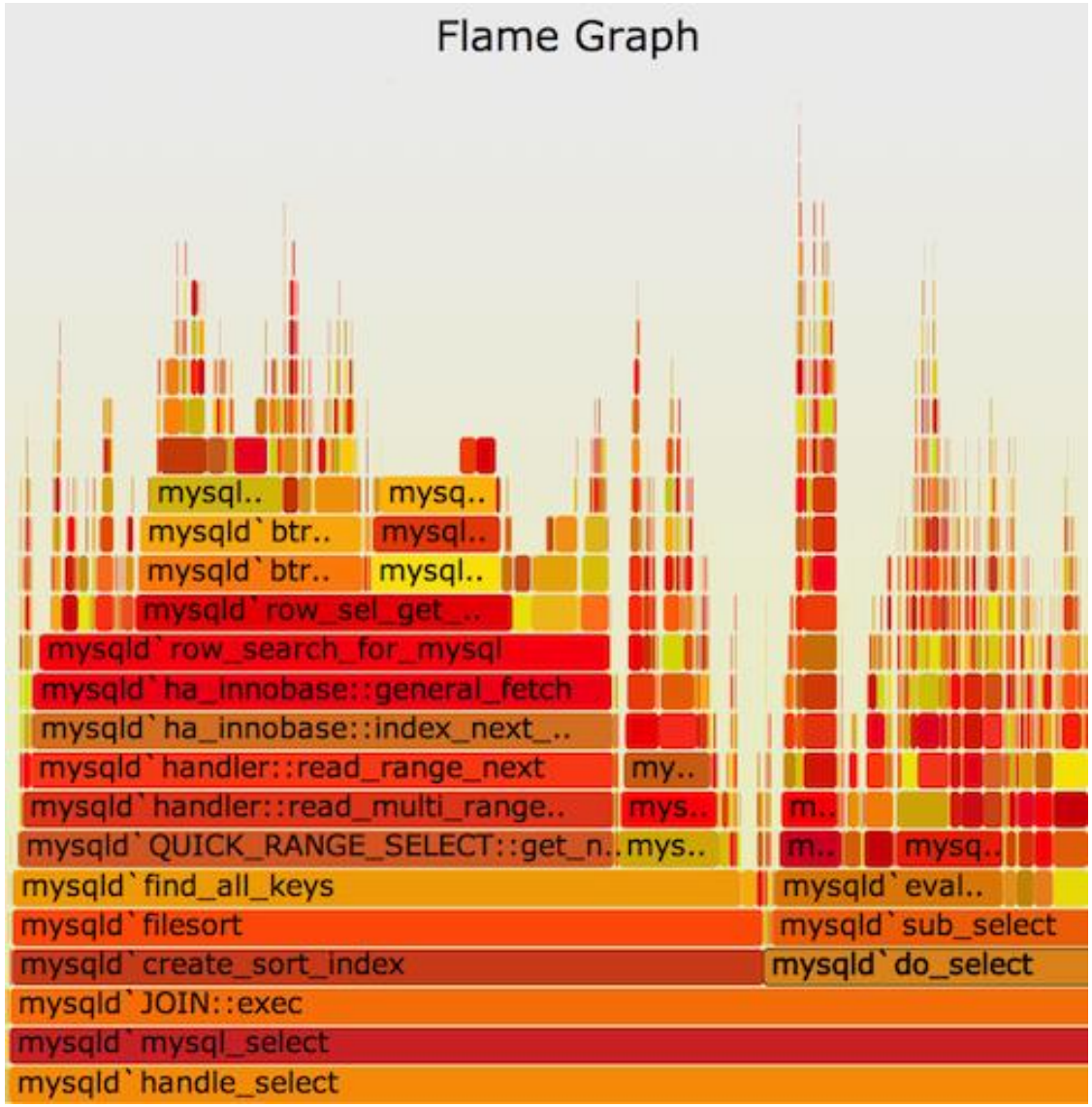
Mit mérjünk/mi a lényeges?

- Metrikák “kicsiben”
  - Pl. Task manager, Resource monitor, ugyanez szerver oldalon….



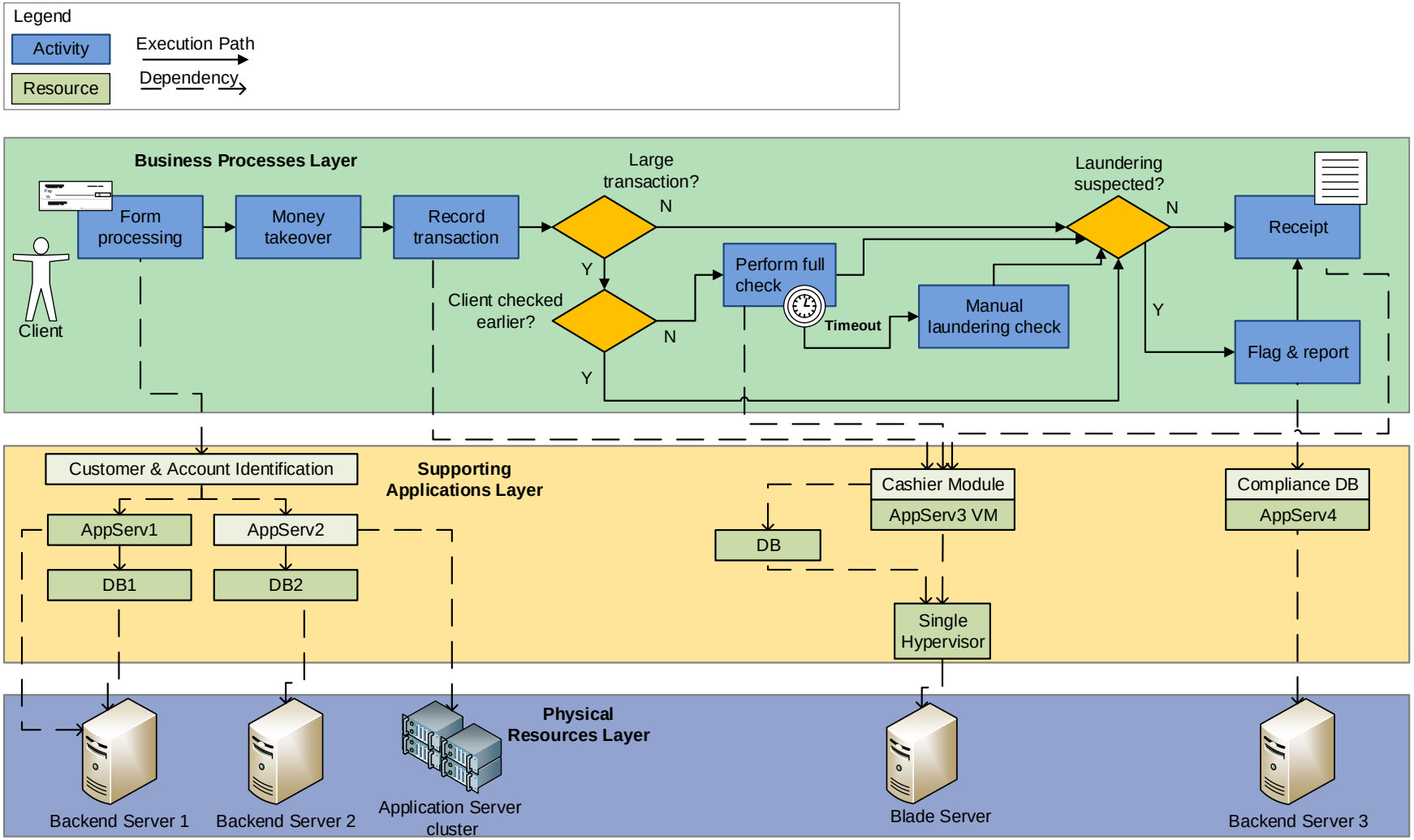
- Metrikák “nagyban”
  - Pl. virtualizált rendszer
- Melyik az érdekes?

Példa: mit számol ez a gép enyit ….?



http://www.brendangregg.com/flamegraphs.html

Példa: erőforrások és folyamatok együtt



Urbanics, G., Gönczy, L., Urbán, B., Hartwig, J., & Kocsis, I. (2014, October).
Combined Error Propagation Analysis and Runtime Event Detection in Process-Driven Systems.
In *International Workshop on Software Engineering for Resilient Systems* (pp. 169-183). Springer International Publishing.

Hol közelítünk?

- A gyakorlatban az értékek nehezen mérhetőek
  - (pl. válaszidő ingadozik, felpörgés, …)
- Az alkalmazások versengenek
  - $(2 * \lambda \neq \lambda + \lambda)$
- Erőforrások közt választani kell
  - →terheléselosztó is kritikus
  - →Pl. ugyanannak a felhasználónak a kérései ugyanoda
- Konkrét beérkezési sorrendtől/mintától eltekintünk
  - Pont ez a Little-törvény előnye…
- Egy feladat végrehajtása lehet adatfüggő
- A rendszer felépítése/paraméterei változhatnak

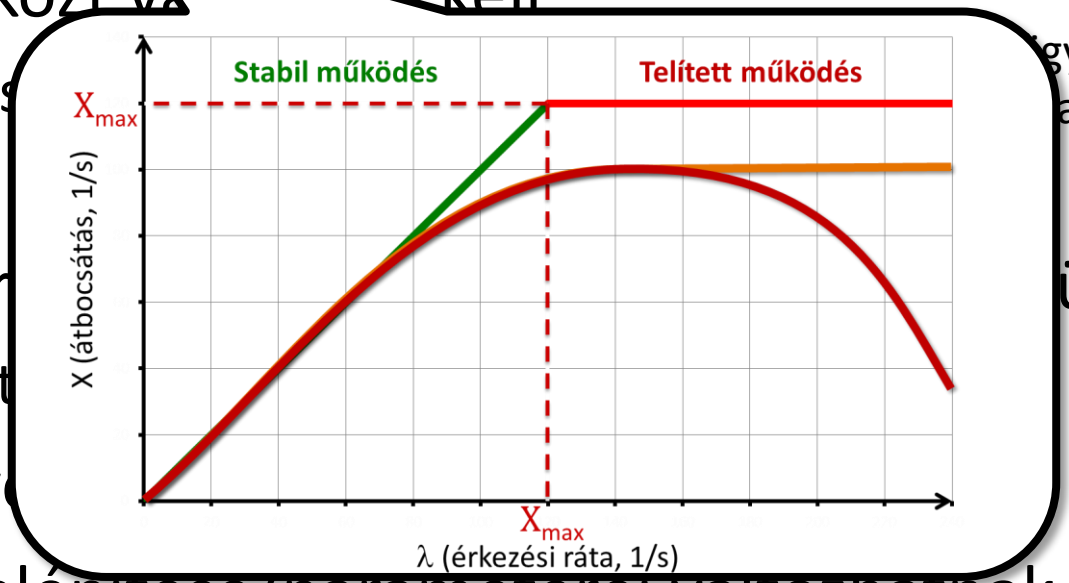
"A kezemet figyeljék, mert…"
(kép: wikipedia)

Hol közelítünk?

- A gyakorlatban az értékek nehezen mérhetőek
  - (pl. válaszidő ingadozik, felpörgés, …)
- Az alkalmazások versengenek
  - $(2 * \lambda \neq \lambda + \lambda)$
- Erőforrások közt v… ni kell
  - →terheléselos…
  - →Pl. ugyanan…
- Konkrét beér… ünk
  - Pont ez a Lit…
- Egy feladat v…
- A rendszer felépítése/paraméterei változhatnak

…gyeljék, mert…”

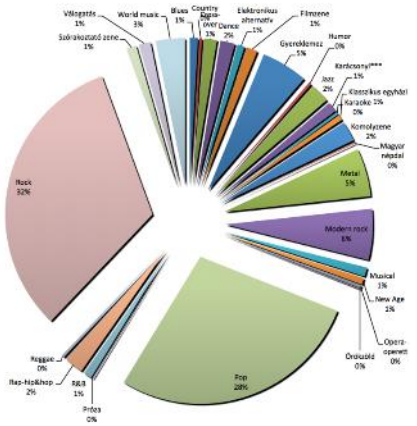
…a)



TERHELÉSMODELLEK: ZIPF TÖRVÉNY

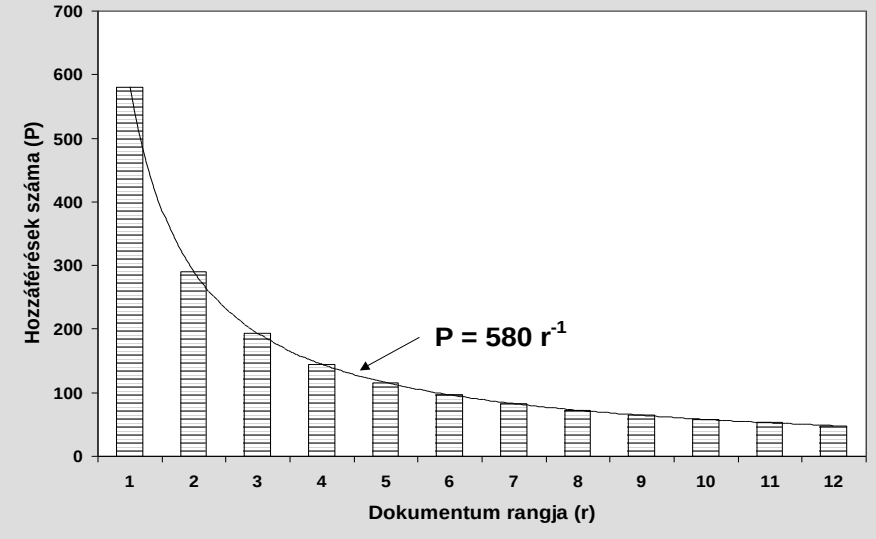
Mi a kérések tartalma?

- Eddig: minden kérés egyforma
  - “kérem egy könyv adatait”
- Valójában: a kéréseknek tartalmuk van
  - “Kérem az *Alapítány és Birodalom* adatait”
  - Ld. Pareto elv (80% -- 20%)
  - A kérések (többsége) az adatok (kis részére) irányul
- Lényeges, mert
  - Műszaki hatása van
    - Cache, pool size, statikus tár, …
  - A rendszermodellt is érinti
    - Gyakori kéréseket másképp kezeljük



Zipf törvénye

- Eredetileg: *korpuszokban* előforduló szavak népszerűségi rangsora és előfordulási gyakorisága jellegzetes eloszlása
  - Nemcsak nyelvi szövegekre igaz



George Kingsley Zipf (1902–1950) amerikai nyelvész, filológus

Zipf törvénye - Példák

- Slágerlisták
- Városok populációja rangsoruk szerint
- Internetes forgalom karakterisztikája
- Weboldalak aloldalainak népszerűsége
- Nyílt forrású rendszerek evolúciója

Zipf törvénye - Képlet

$$R_i \sim \frac{1}{i^{\alpha}} \qquad f \sim \frac{1}{p}$$

- R_i – az i. szó előfordulási gyakorisága
- α – a korpuszra jellemző 1 közeli érték
- Egyszerűsítve (α = 1):
  - f (frequency) gyakoriság
  - p (popularity): a szöveg ,,rangja'' (csökkenő sorrendben)

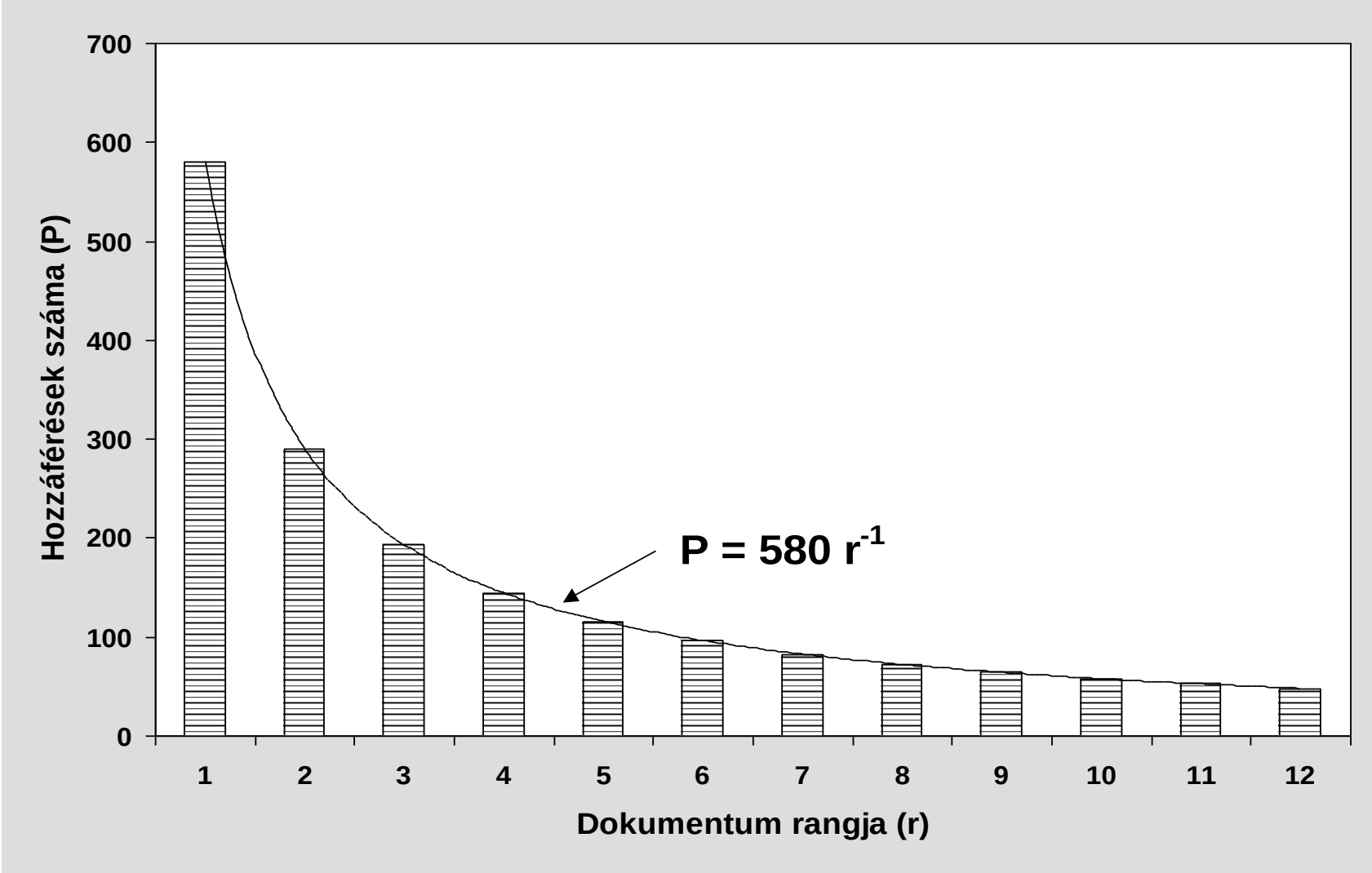
Zipf törvénye – pl. Web dokumentumokra

$$P = \frac{k}{r}$$

- P – hivatkozások (elérések)
- r – rang (1 = leggyakoribb)
- k – pozitív konstans

Bővebben: http://www.hpl.hp.com/research/idl/papers/ranking/adamicglottometrics.pdf

Zipf – Példa (1)

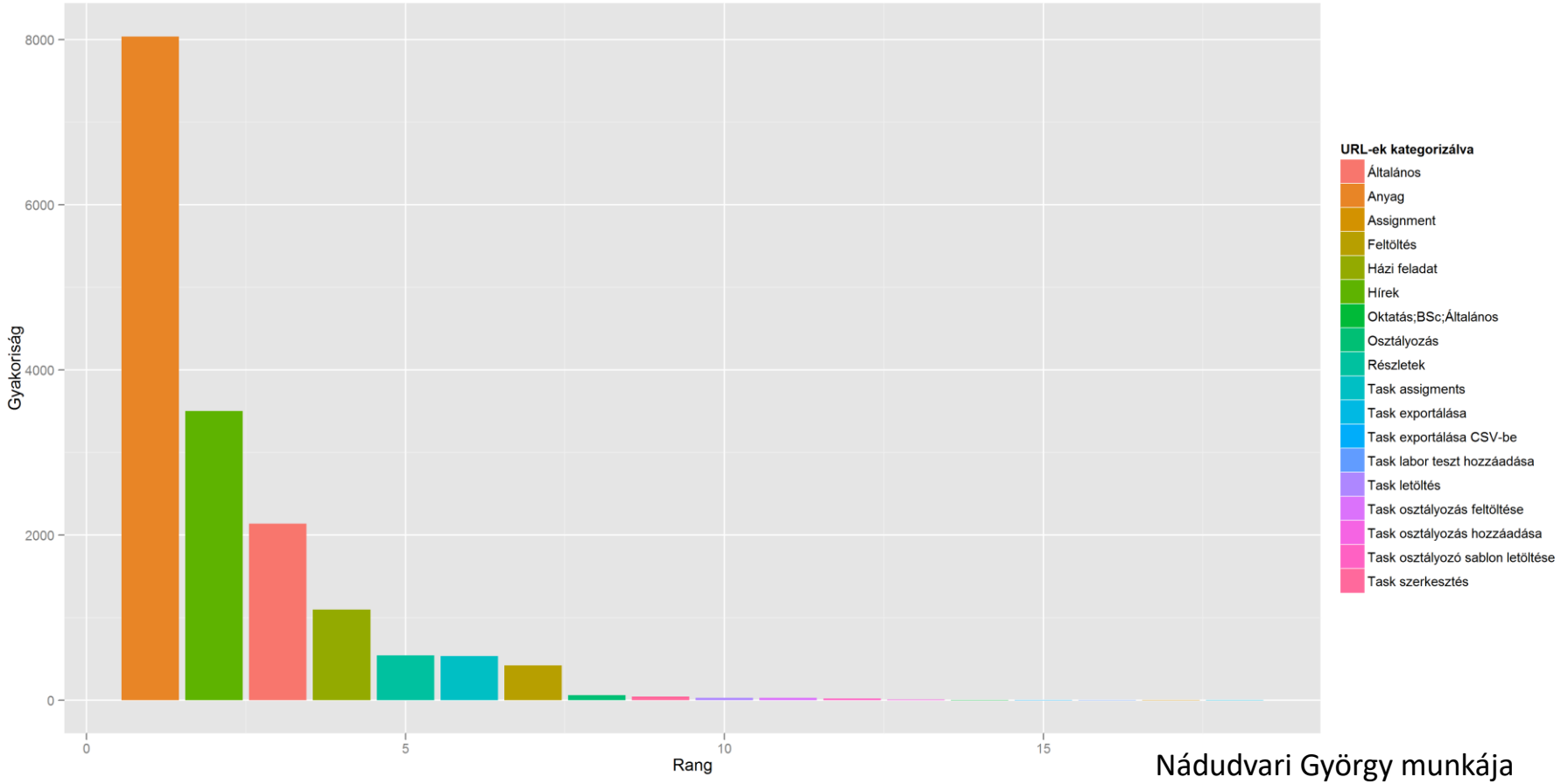

Hozzáférések száma (P)
700
600
500
400
300
200
100
0
1
2
3
4
5
6
7
8
9
10
11
12
Dokumentum rangja (r)
P = 580 r-1

Zipf – Példa: Tanszéki honlap (1)

- Aloldalak látogatottsága kategóriák szerint csoportosítva
- Megfigyelhető az ún. heavy-tailed eloszlás

Zipf – Példa: Tanszéki honlap (2)

- A Rendszermodellezés tárgy oldalainak látogatottsága



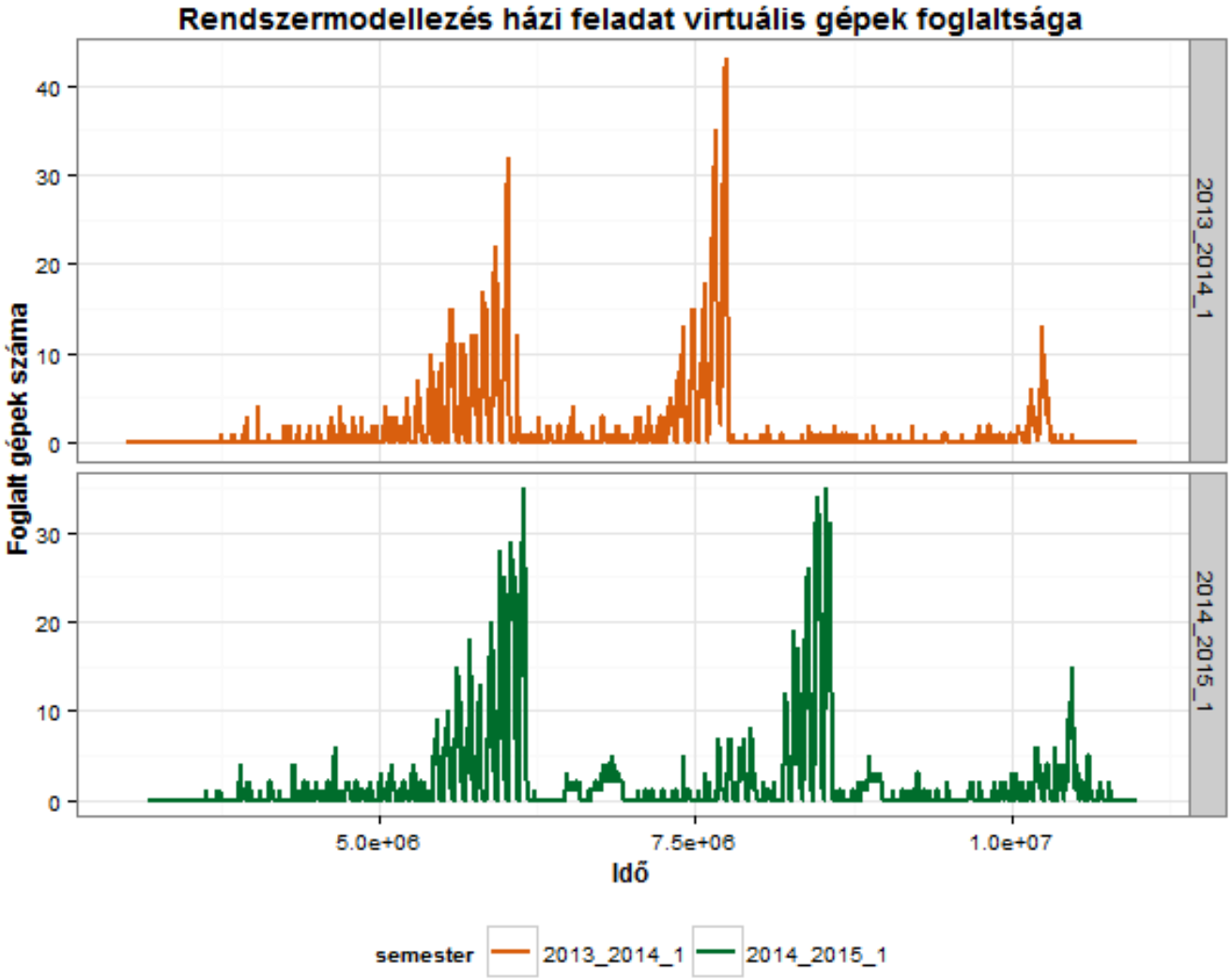
Nádudvari György munkája

TERHELÉS VÁLTOZÁSA

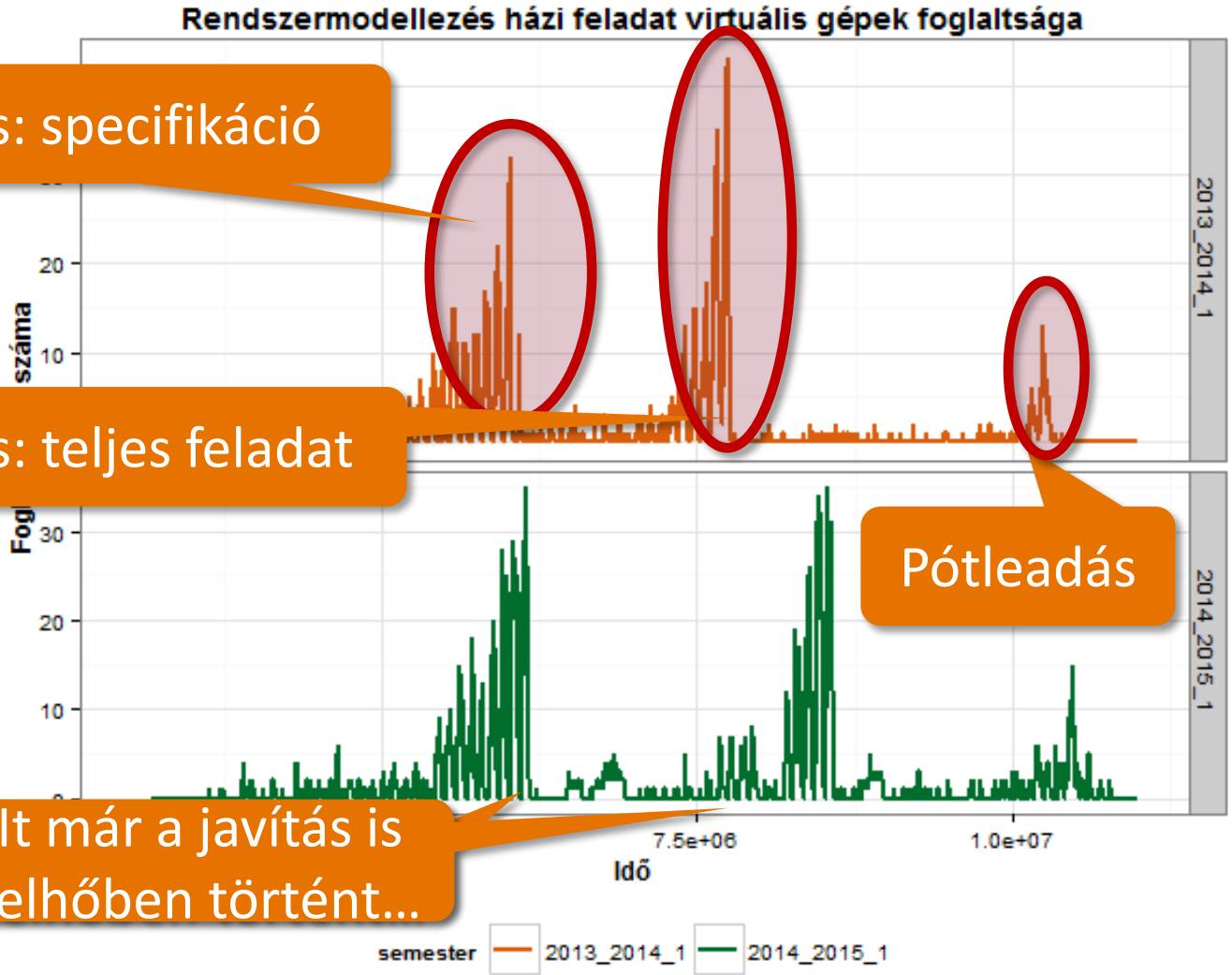
Milyen jellegű a terhelés?

- Eddig:
  - Átlagos értékekkel számoltunk
  - A rendszer viselkedését a *terhelés (intenzitás)* függvényében néztük
  - De: valójában nem (feltétlenül) előre kiszámíthatóan változik a terhelés
- Valójában
  - A rendszer viselkedése *időben* változik
  - Ennek műszaki hatásai vannak
    - Váltás feladatok közt, erőforrásfoglalás, stb. (pl. Operációs rendszerek)

Rendszermodellezés (7. félév) a felhőben



Rendszermodellezés (7. félév) a felhőben



Valós (történelmi) terhelés példa (iwiw)

Szakaszokra becslőfüggvény illeszthető

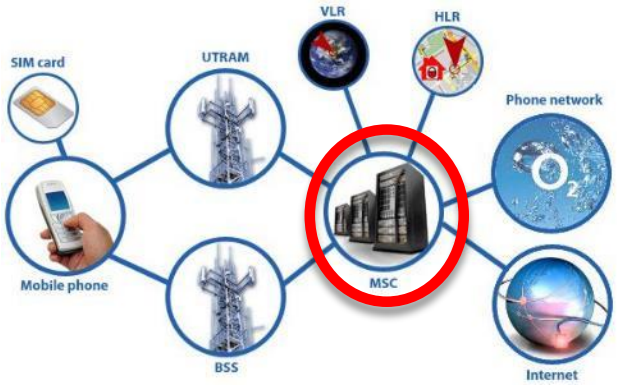
- Lineáris, exponenciális, logaritmikus
- Regresszió, Bővebben pl. Valószínűségszámítás

Forrás: http://www.sg.hu/cikkek/42924/

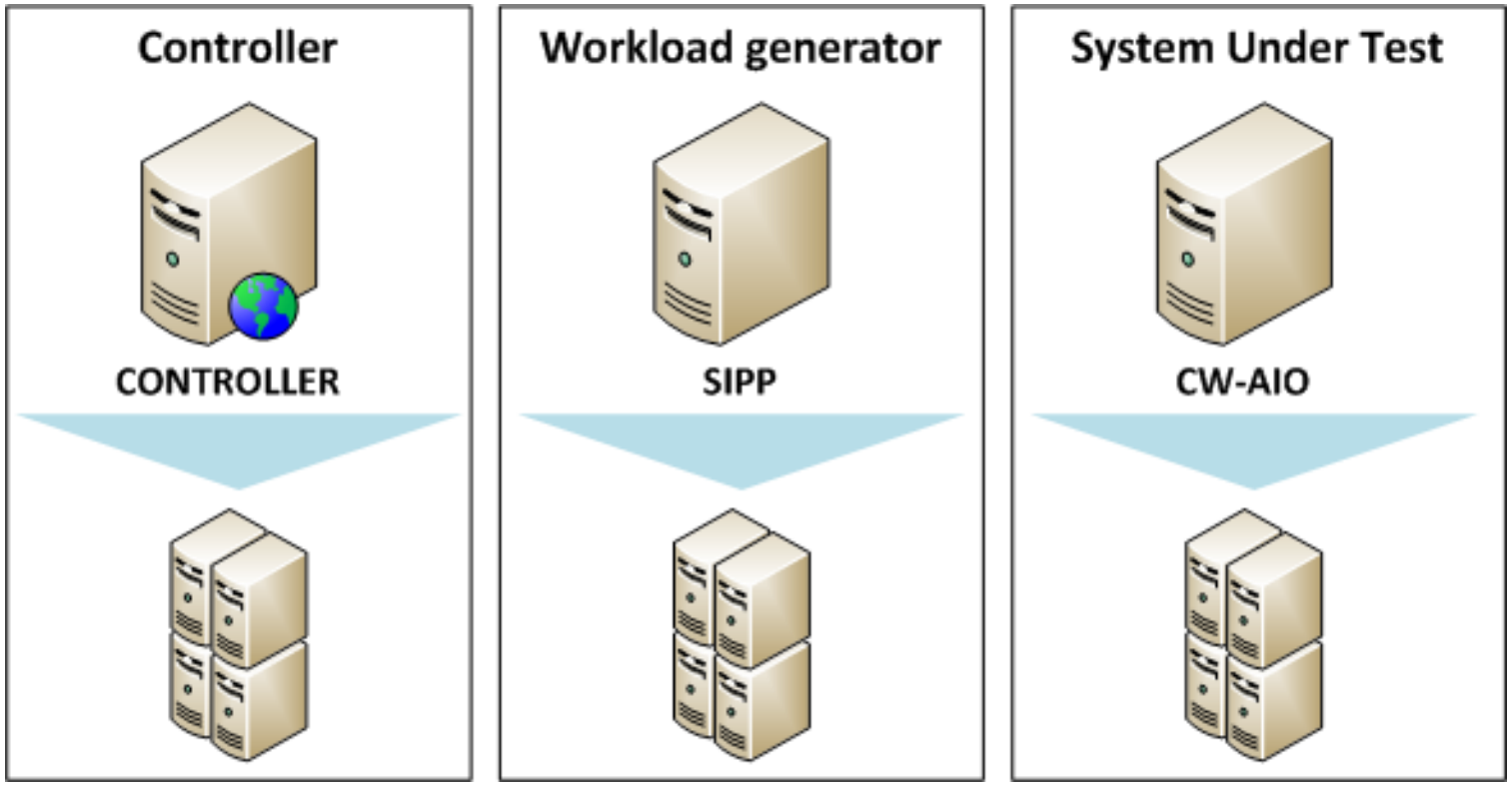
ESETTANULMÁNY: KRITIKUS SZOLGÁLTATÁS

Felhő alapú szolgáltatások

- Telekommunikációs szolgáltatások felhőben
- Az alkalmazásnak működnie kell…
  - Mit mérjünk? Mire számítsunk?
- Migrálás, hibatűrés támogatása
- Környezet változhat
- Mikor hibás a rendszer?
  - Kritikus szolgáltatásoknak működniük kell
- Ritka események…

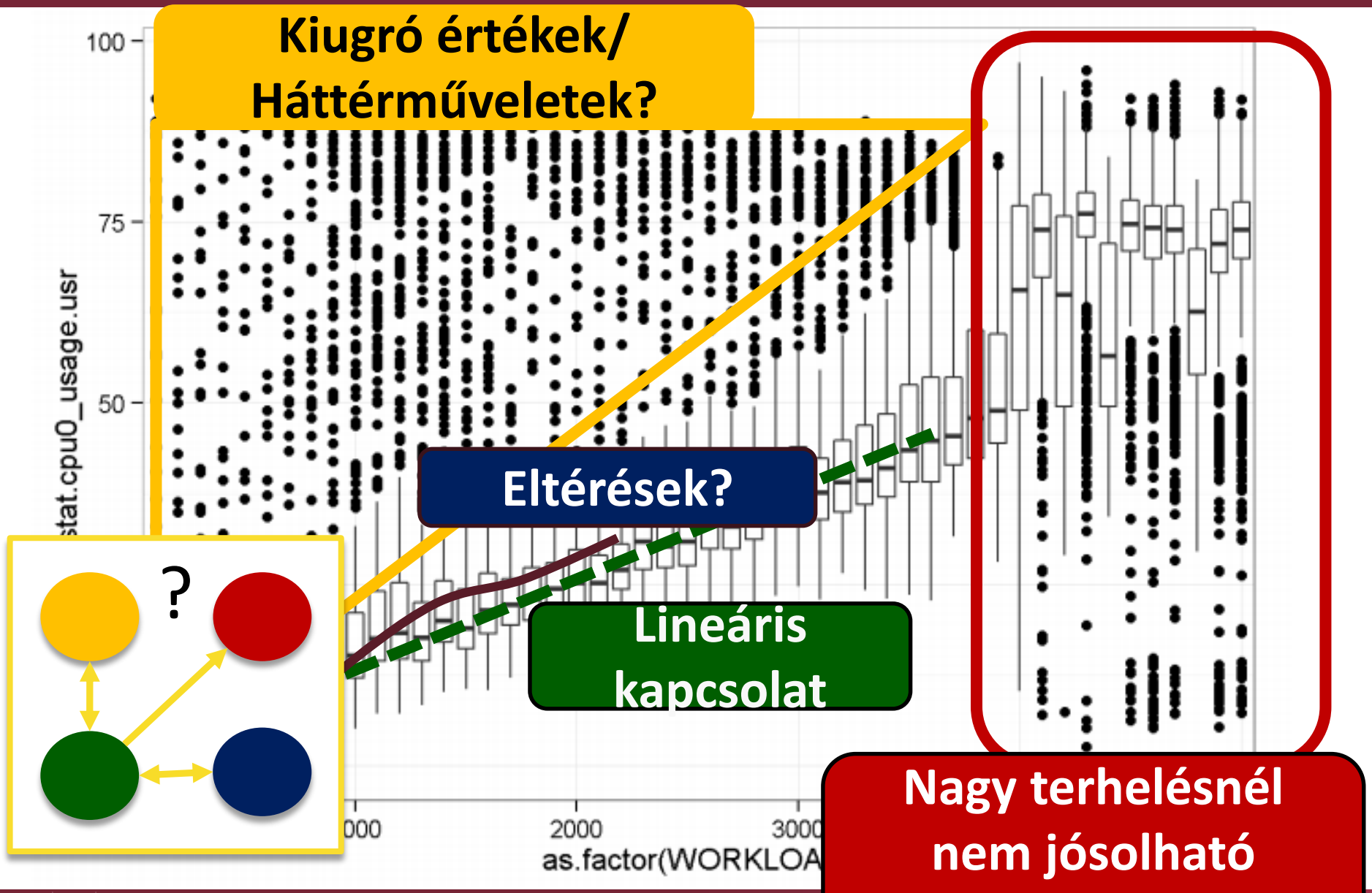


Példa: digitális “telefonközpont”

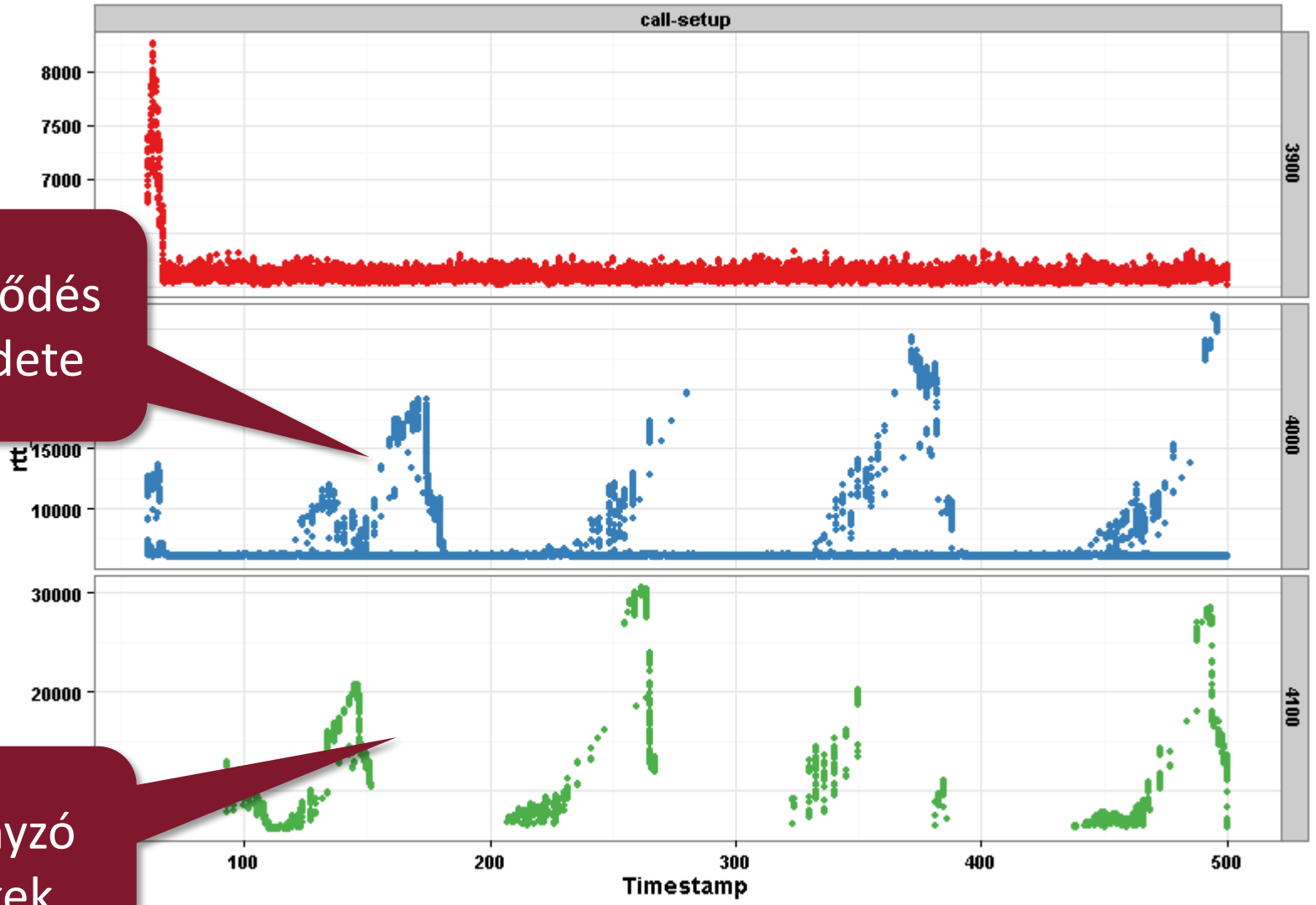


(Clearwater IMS)

Példa: terhelés vs. kihasználtság



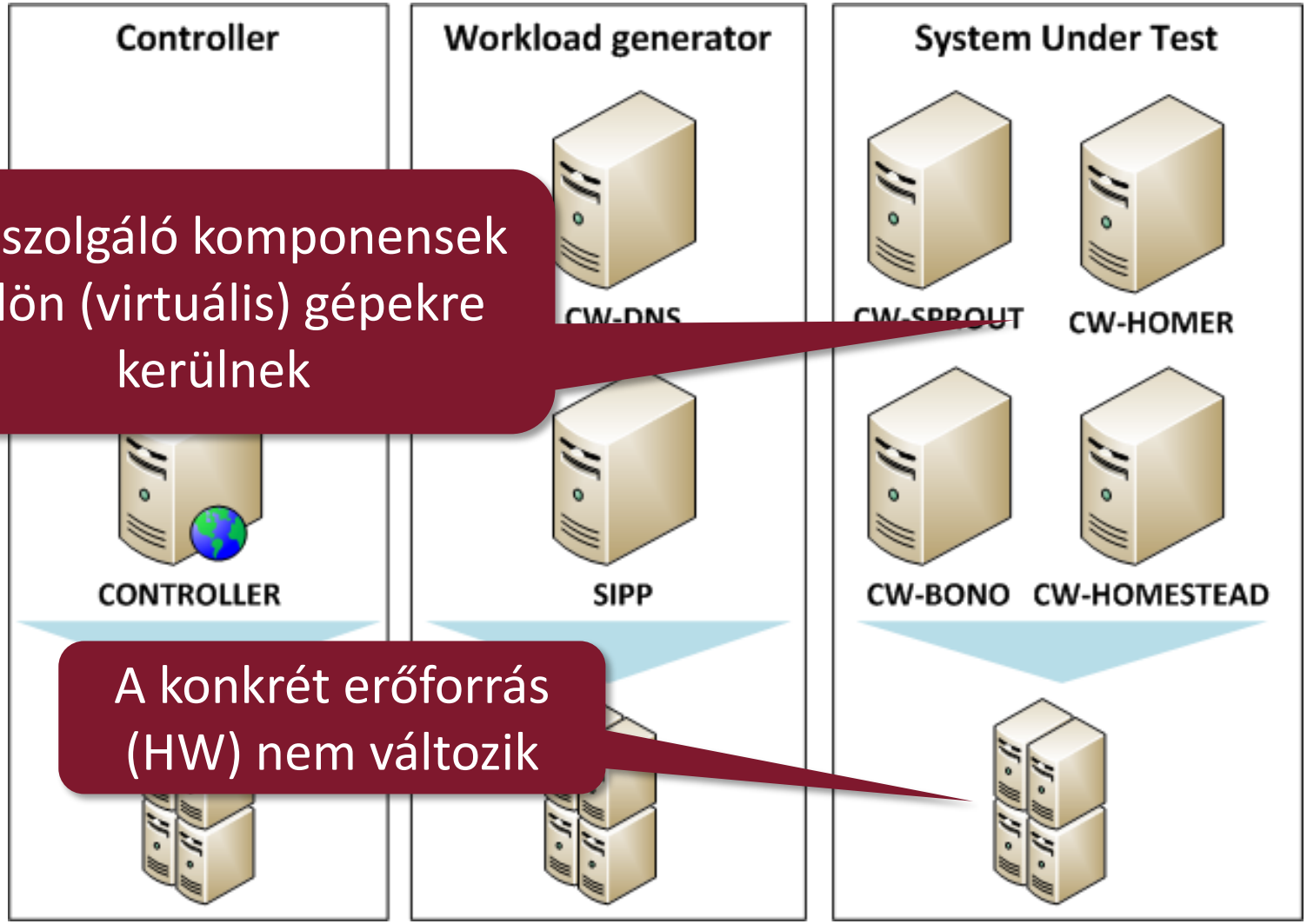
Idősorelemzés



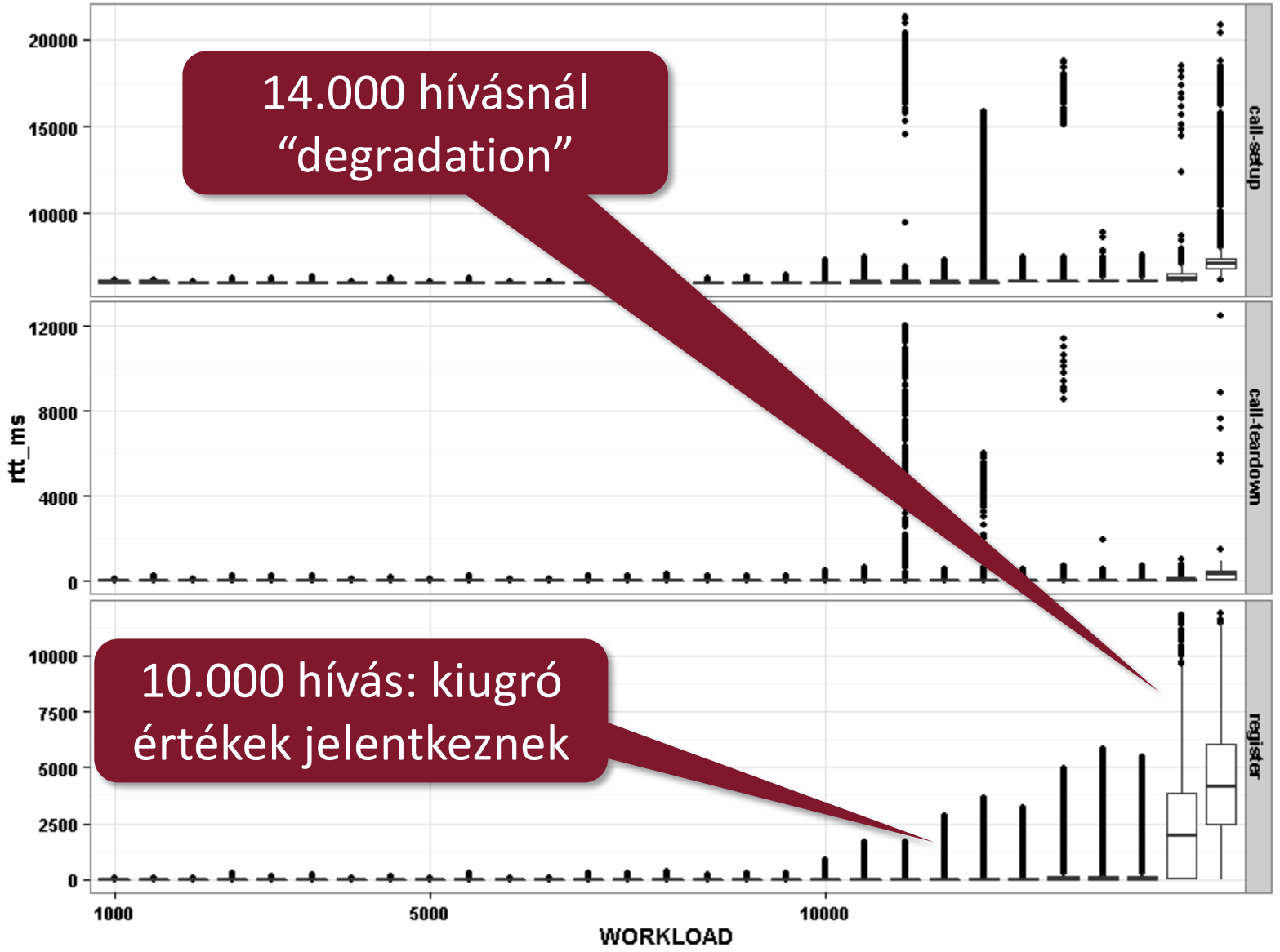
Vergődés kezdete

Hiányzó értékek…

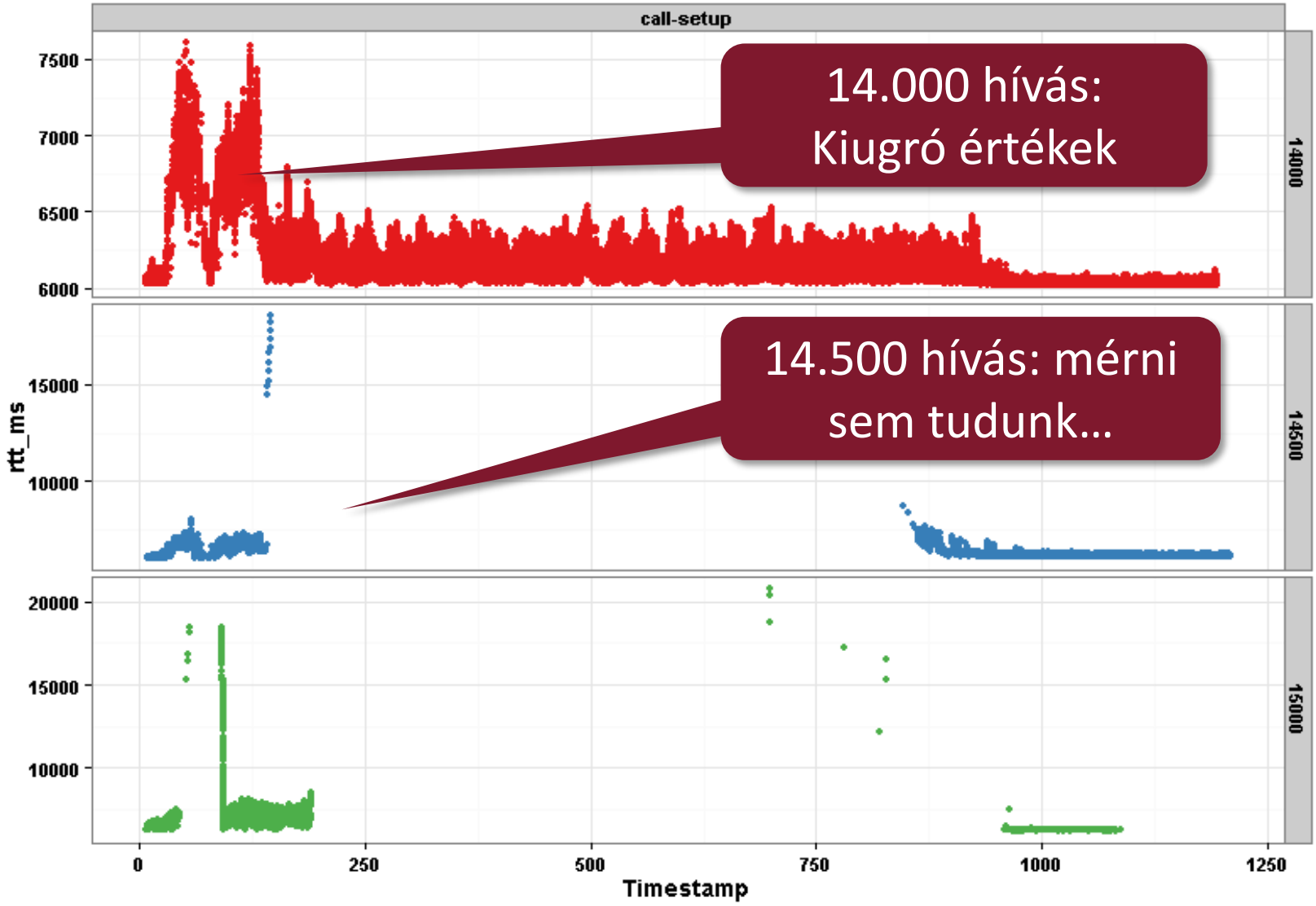
Osszuk szét a feladatokat!



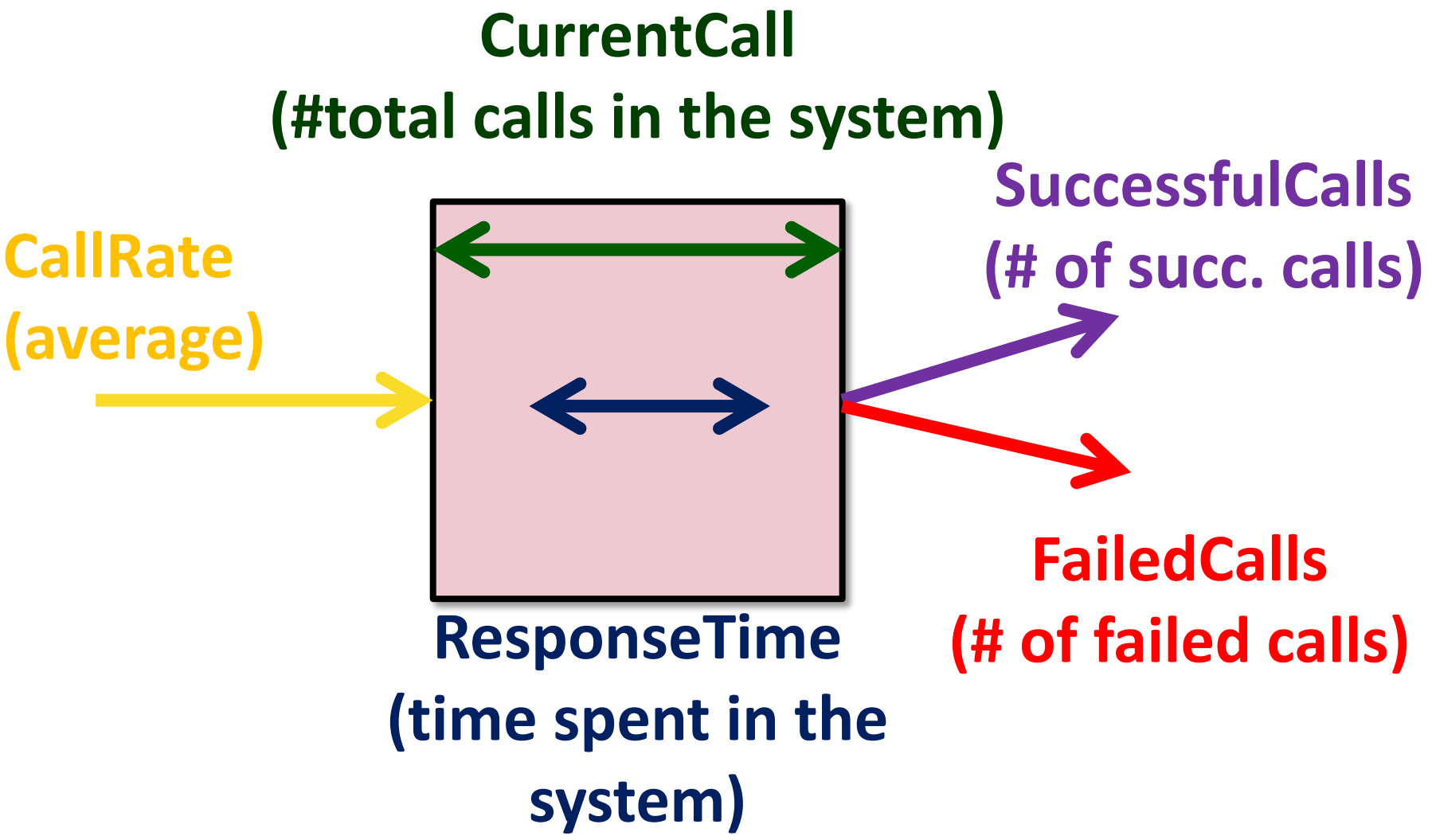
Válaszidő alakulása különböző műveletekre



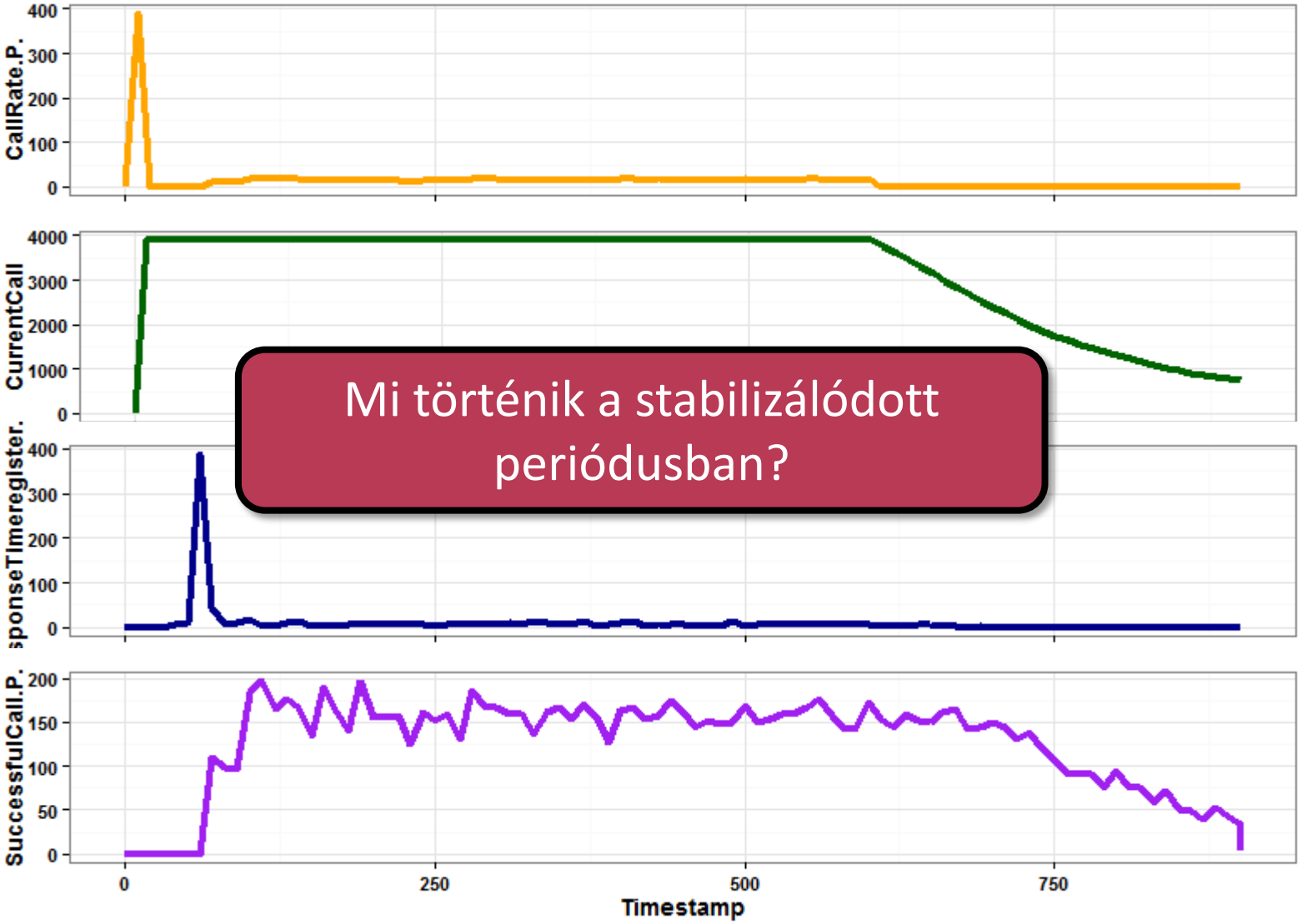
Idősor elemzés



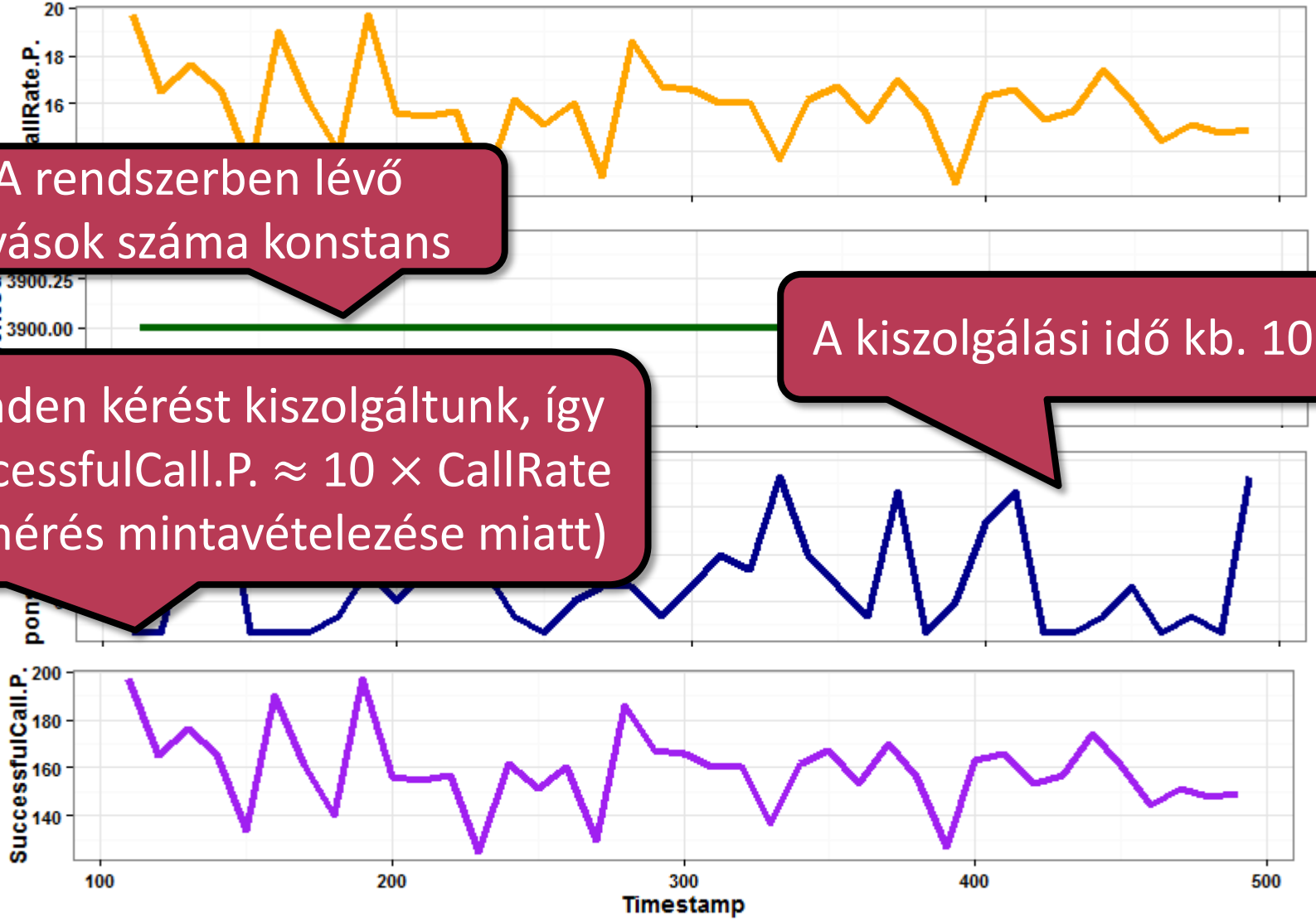
Little törvény példa



Visszacsatolásos terhelés



A mérés állandósult állapotban

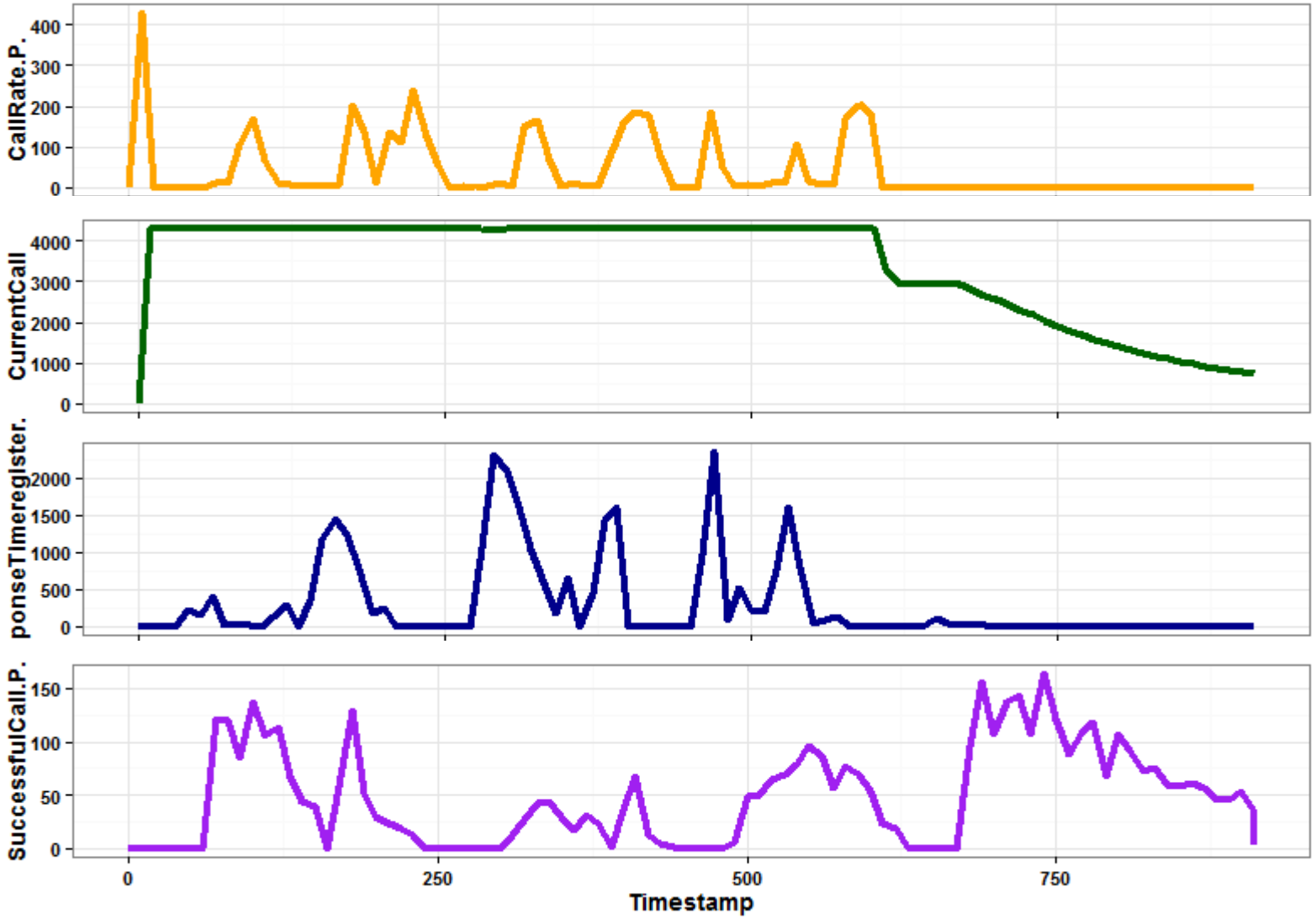


A rendszerben lévő hívások száma konstans

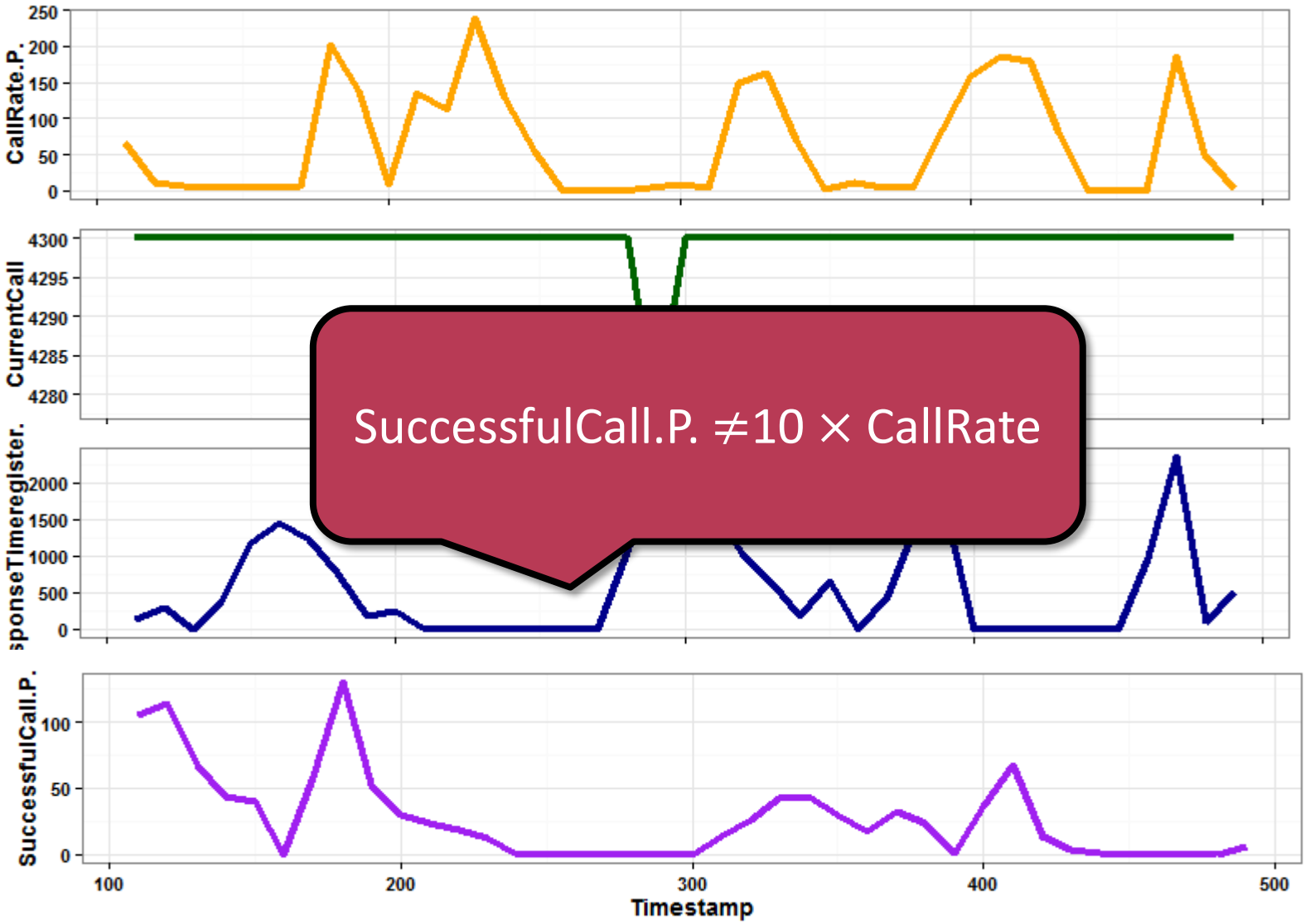
A kiszolgálási idő kb. 10 ms

Minden kérést kiszolgáltunk, így SuccessfulCall.P. ≈ 10 × CallRate (a mérés mintavételezése miatt)

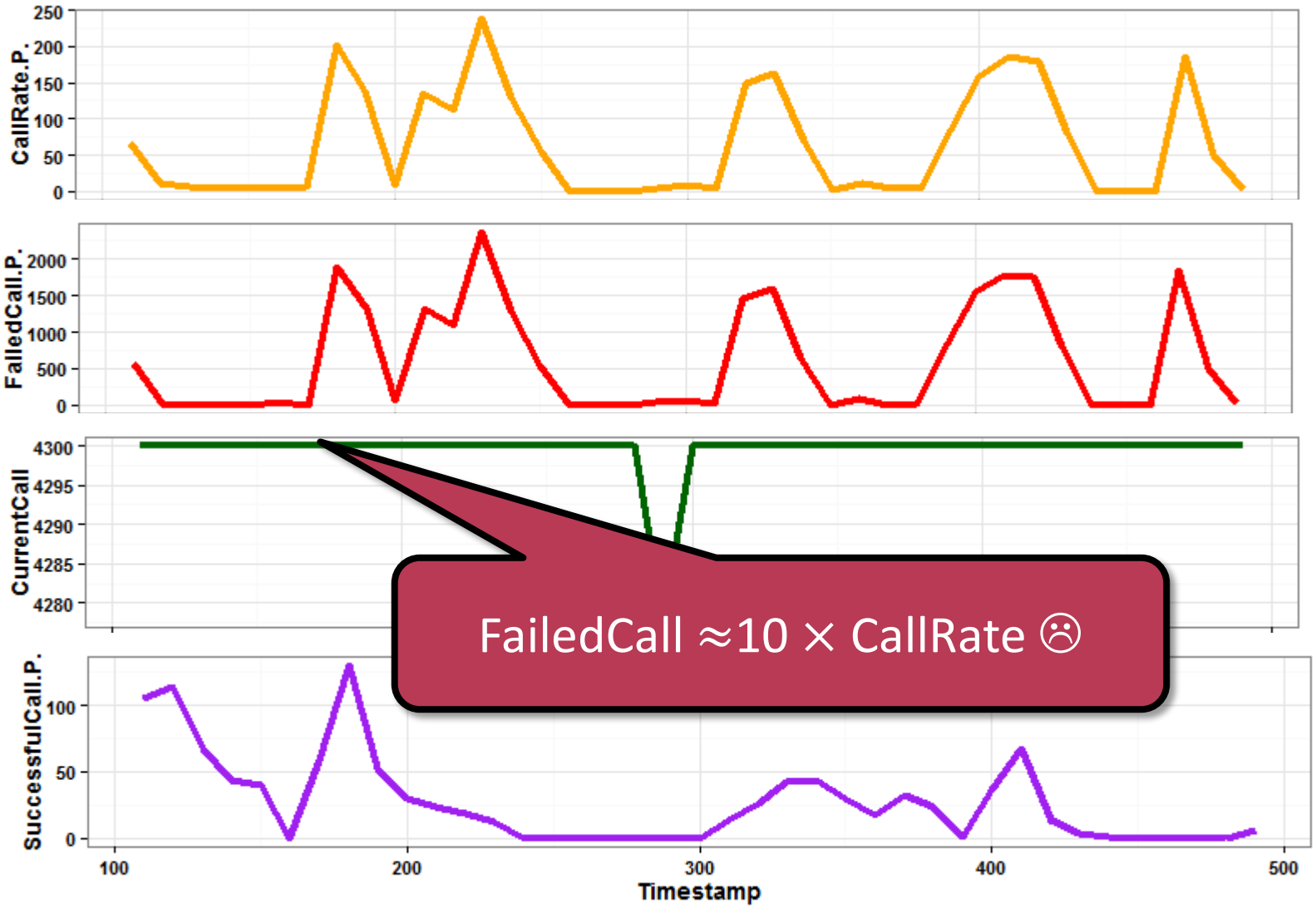
Visszacsatolás, túlterhelés



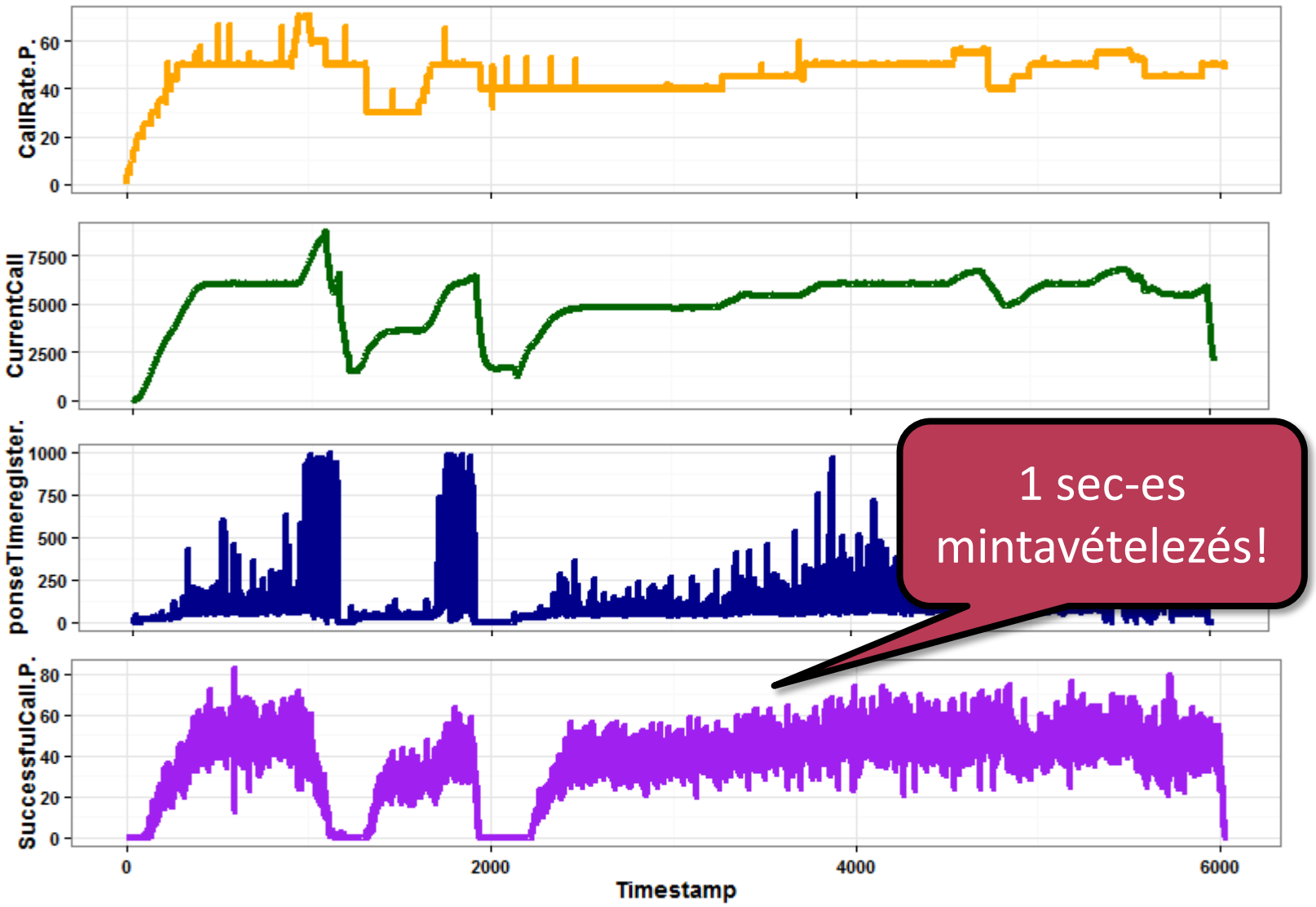
Hol sérül az állandósult állapot feltétele?



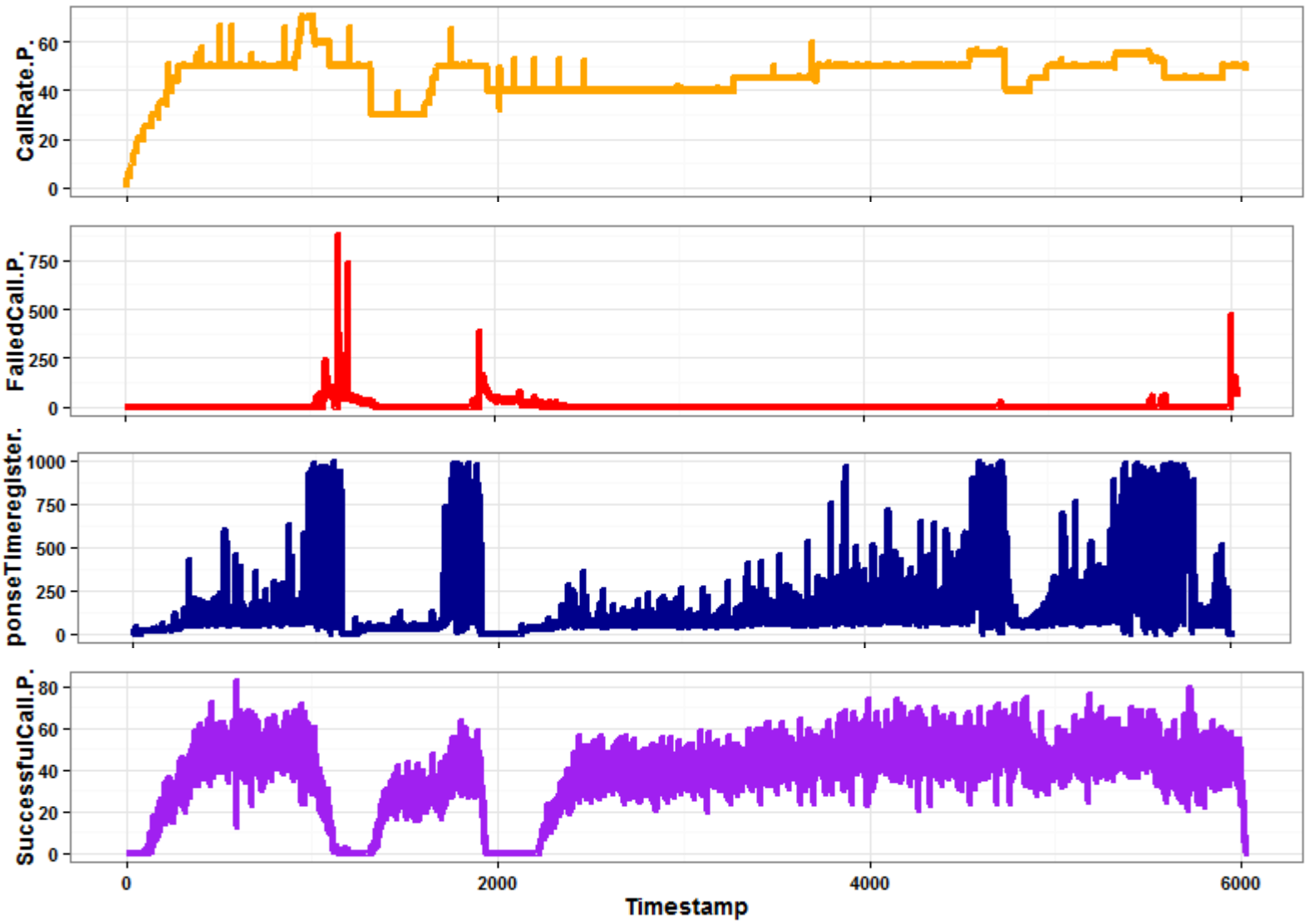
Túlterhelés: Szinte minden kérés hibás…



Nincs visszacsatolás, állandó terhelés



Finomabb mintavételezés



Tanulságok

- Az elvi teljesítménymodell elemeit a gyakorlatban nehéz meghatározni
- Az erőforrások skálázásának az alkalmazások is korlátot szabnak
- Az időbeli változás szerepe kritikus
- Sok változó, sok mérés → adatelemzés
  - Ld. később…