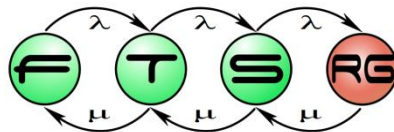


# Teljesítménymodellezés

**Budapest University of Technology and Economics  
Fault Tolerant Systems Research Group**



# Emlékeztető

## ■ Egyensúlyi állapot:

- Átlagos értékekkel számolunk
- $\lambda = X$  (érkezési ráta = átboocsátás)

## ■ Átboocsátóképesség ( $X_{\max}$ ):

- Az elérhető maximális átboocsátás  $T$  átlagos feldolgozási idő mellett
- $X_{\max} = \frac{K}{T}$  ( $K$  darab erőforráspéldány esetén)

## ■ Kihhasználtság ( $U$ ):

- Az átboocsátás és az átboocsátóképesség aránya
- $U = \frac{X}{K} \times T$  ( $K$  erőforráspéldány esetén)

A Little-törvény

A Zipf-törvény

Terhelés változása

# TARTALOM

A Little-törvény

A Zipf-törvény

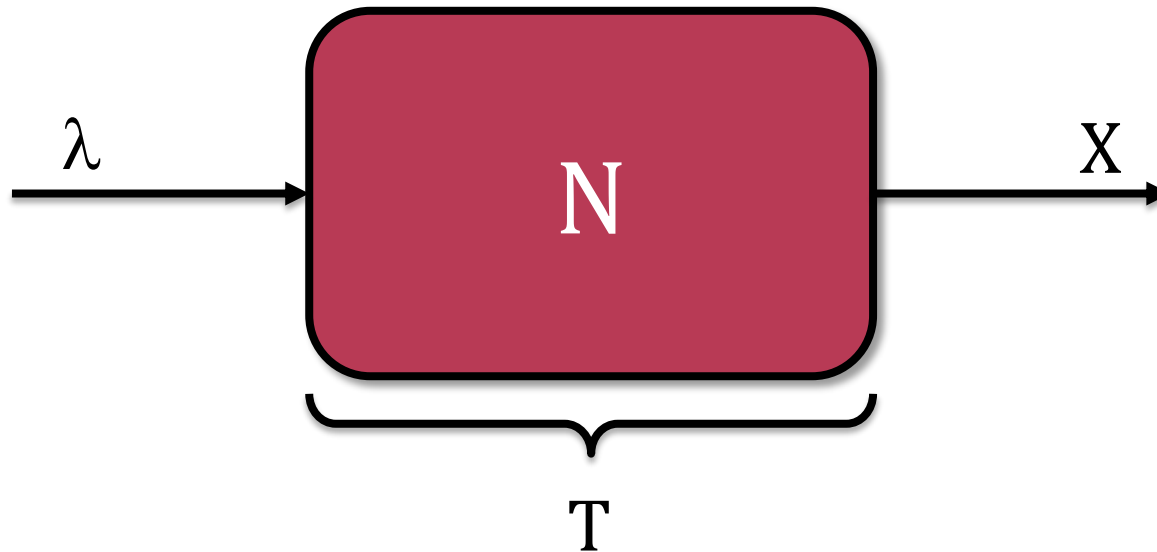
Terhelés változása

# A LITTLE-TÖRVÉNY

Avagy az alapképlet

# A Little-törvény

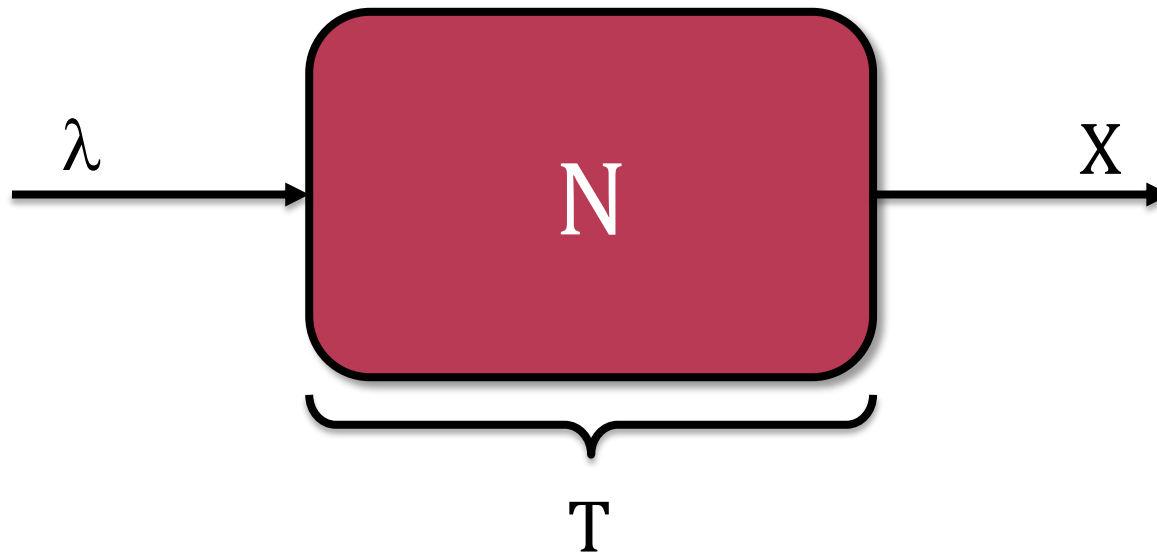
- $\lambda$ : érkezési ráta  $\left[\frac{\text{darab}}{s}\right]$
- $X$ : átbocsátás  $\left[\frac{\text{darab}}{s}\right]$
- $T$ : rendszerben töltött idő  $[s]$
- $N$ : rendszerben lévő tokenek száma  $[\text{darab}]$



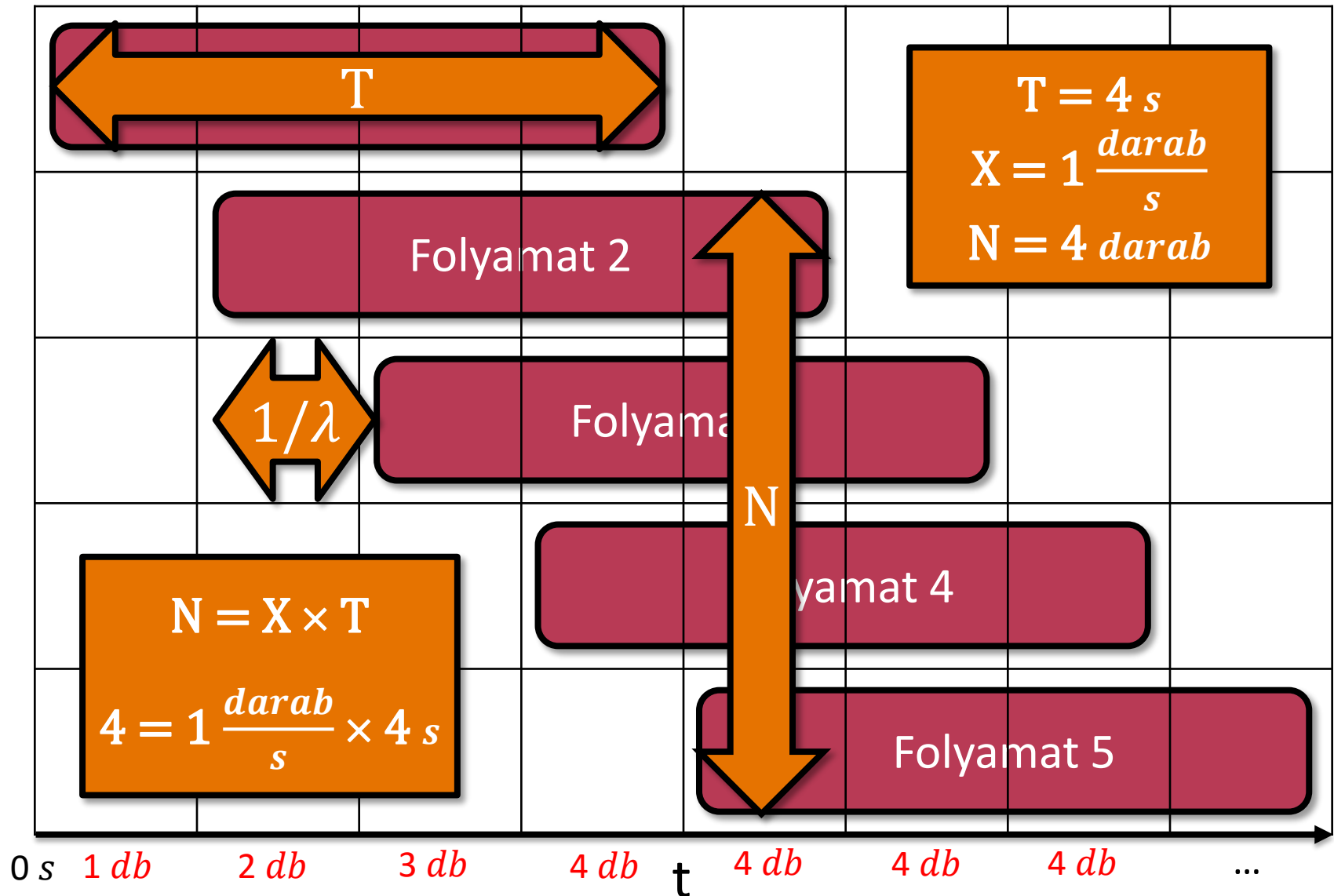
# A Little-törvény

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

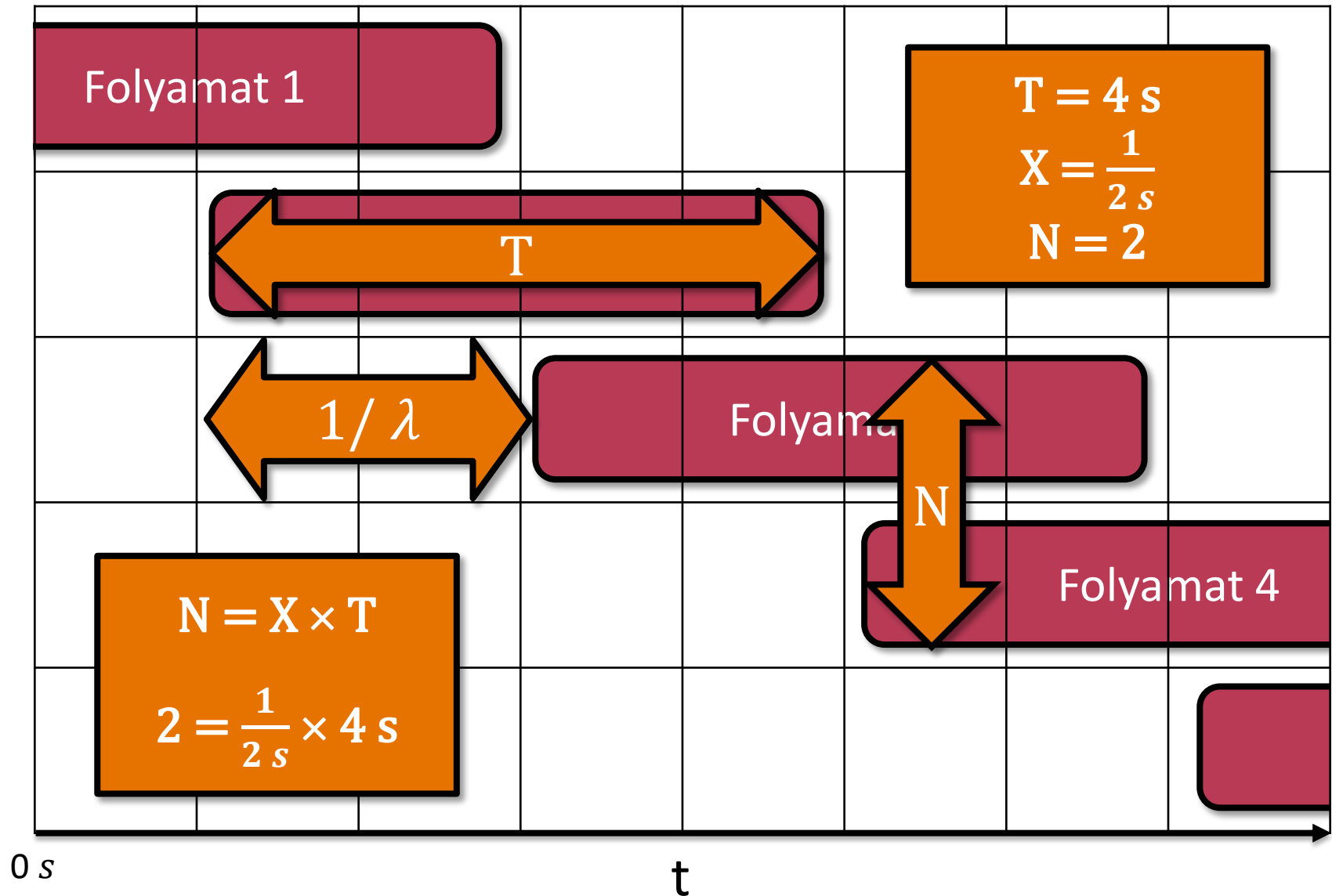
$$N = X \times T$$



# 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  
( $N = X \times T$ )

A Little-törvény

A Zipf-törvény

Terhelés változása

# LITTLE TÖRVÉNY: GYAKORLATI PÉLDÁK

# Példa

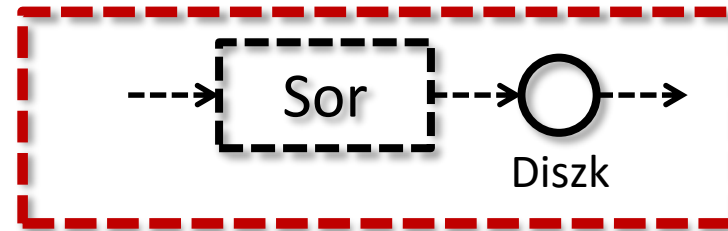


- Erőforrás: diszk
- 40 kérést szolgál ki másodpercenként (nincs átlapolódás)
- 1 kérés kiszolgálása átlagosan 0,0225 másodpercig tart
- Mekkora a kihasználtság?

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

# Példa

Rendszer



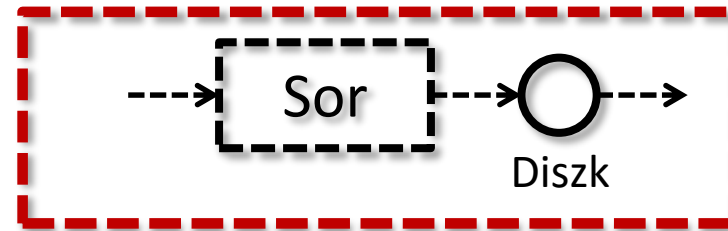
- Sorban állás is van a diszk előtt
- Diszk: 40 *kérés/s*
- Kérések átlagos száma a rendszerben: 4

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

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

# Példa

Rendszer



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

Sorbanállási és  
diszk  
kiszolgálási idő

Rendszer

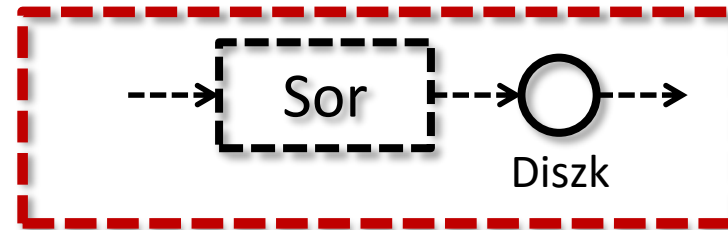
$$N = X \times T \rightarrow T_{rendszer} = 4 \text{ *kérés*} / 40 \frac{\text{*kérés*}}{\text{s}} = 0,1 \text{ s}$$

Átlagos sorban állási idő?

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

# Példa

Rendszer



- 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?

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

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

# Little törvény a gyakorlatban

## ■ Szimuláció

- Dobson&Shumsky
- <https://youtu.be/UjzXQPGBaNA>

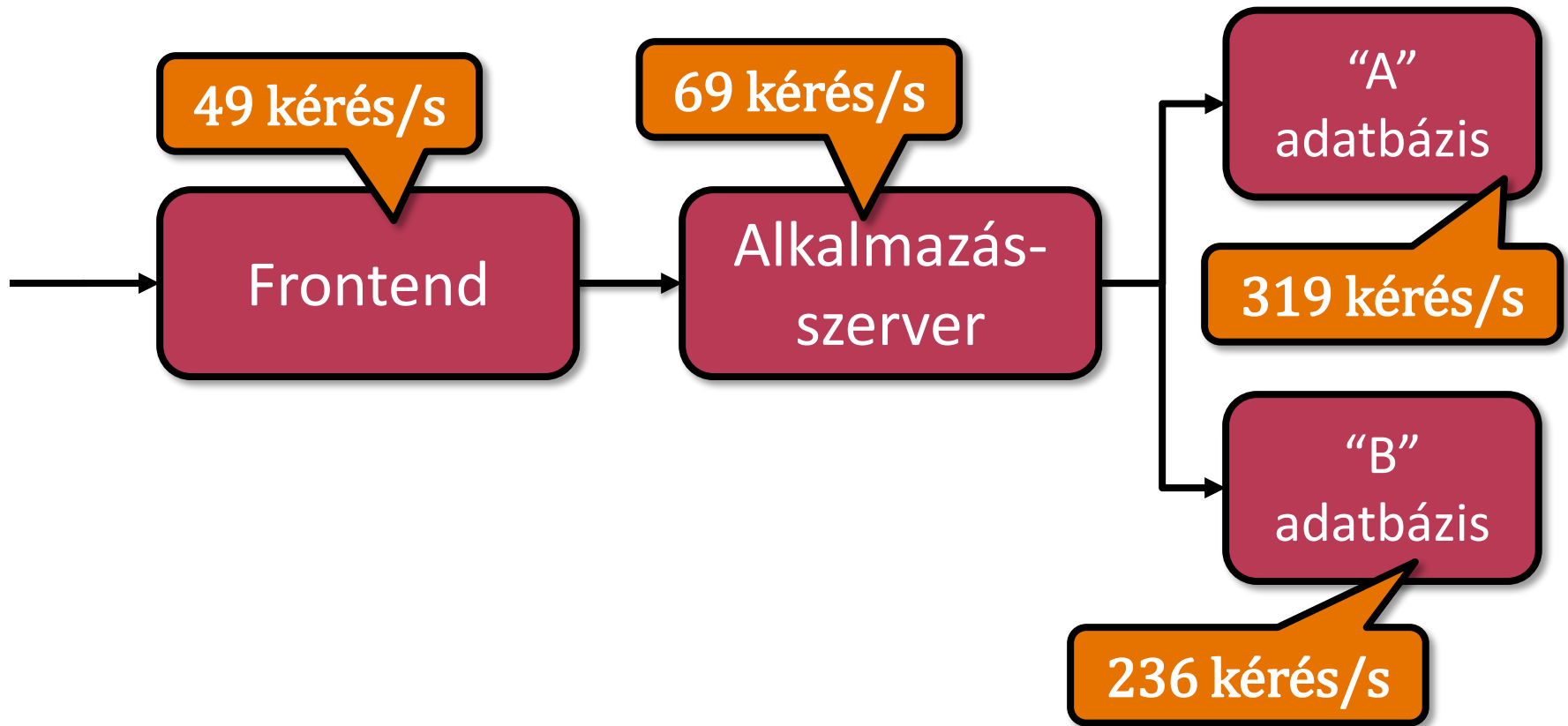
## ■ Miért oktatják(juk)

- <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>

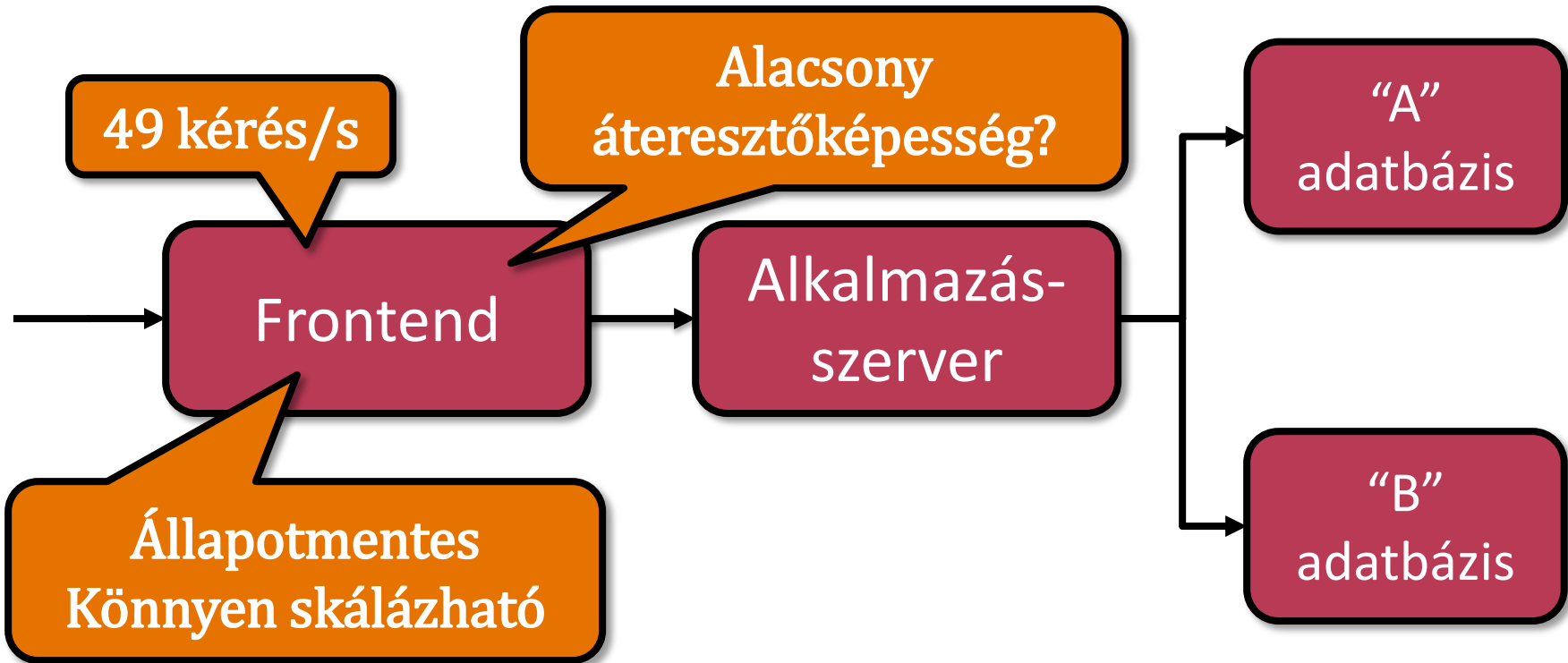
# Teljesítmény 3 rétegű architektúrában



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



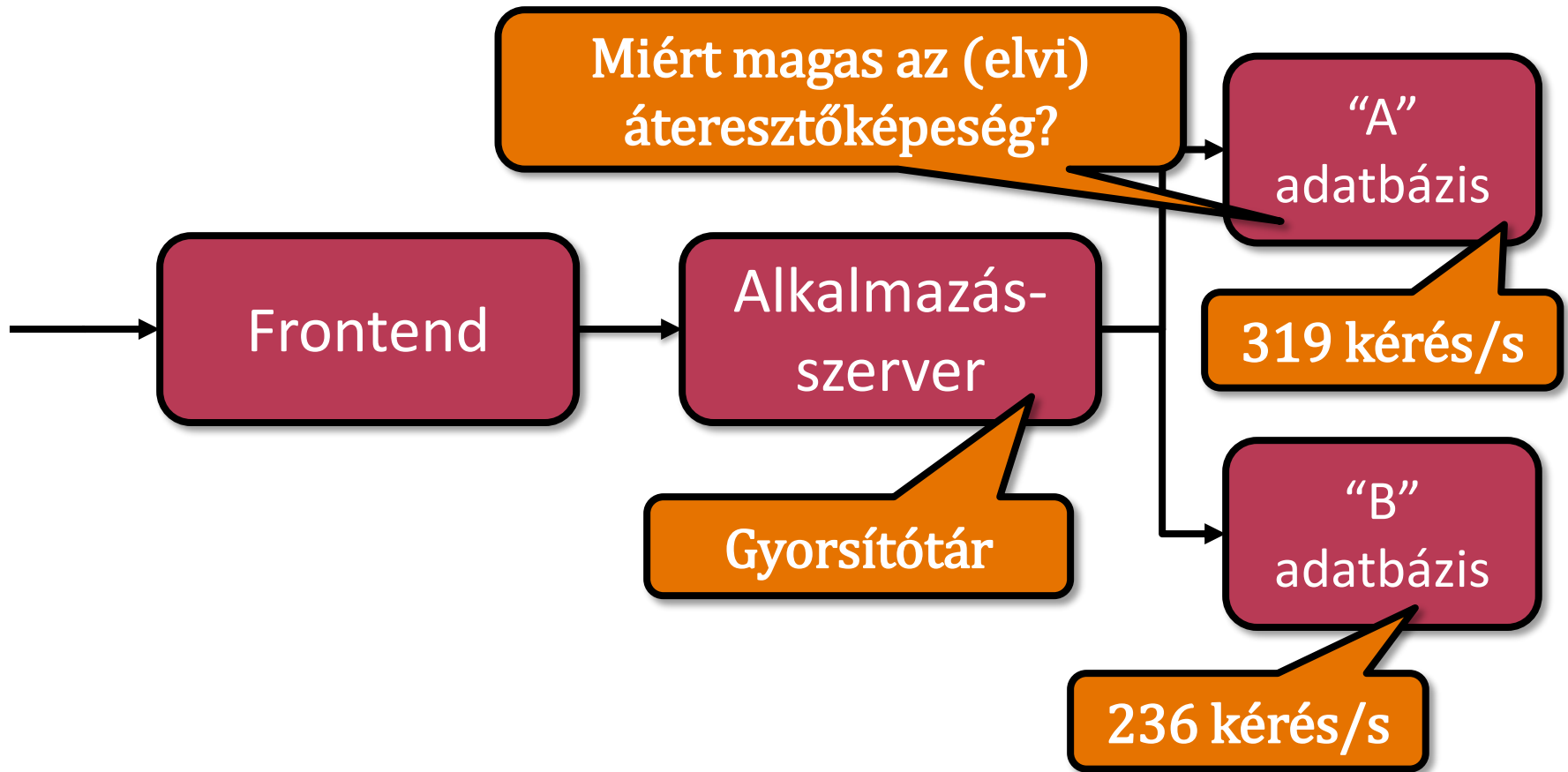
A mérőszámok itt az egész rendszerre érkező terhelésre utalnak! Például az „A. adatbázis” akkor válik szűk keresztmetszetté, ha a rendszerbe 319 lekérdezés érkezik másodpercenként.

# Teljesítmény 3 rétegű architektúrában



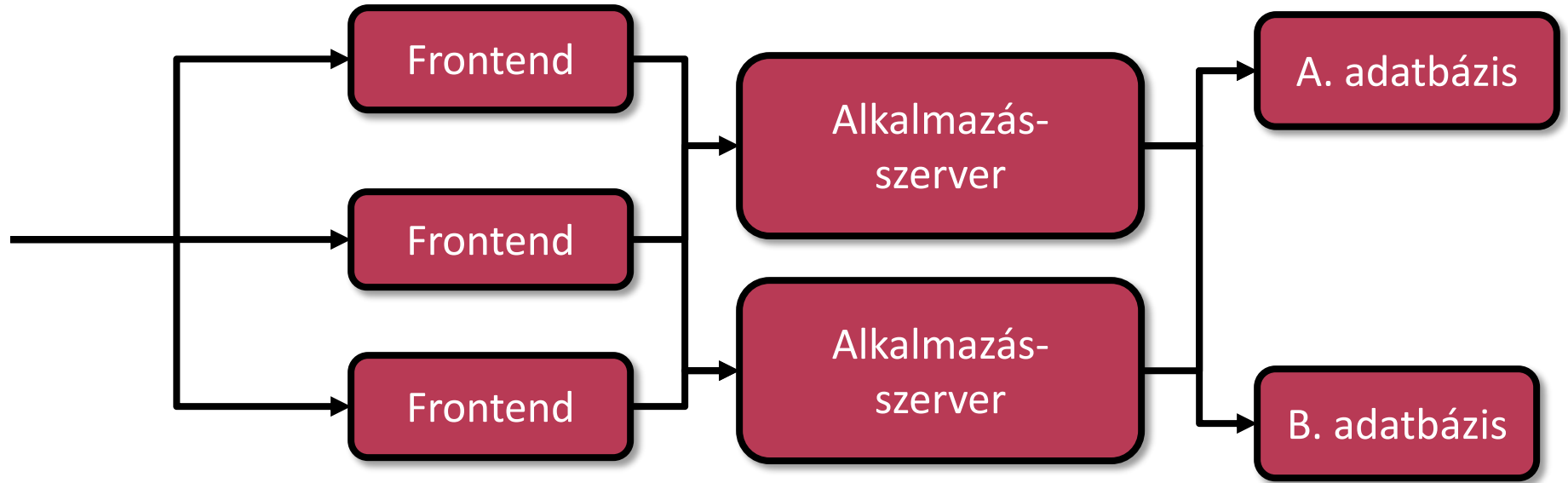
A mérőszámok itt az egész rendszerre érkező terhelésre utalnak! Például az „A. adatbázis” akkor válik szűk keresztmetszetté, ha a rendszerbe 319 lekérdezés érkezik másodpercenként.

# 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:)

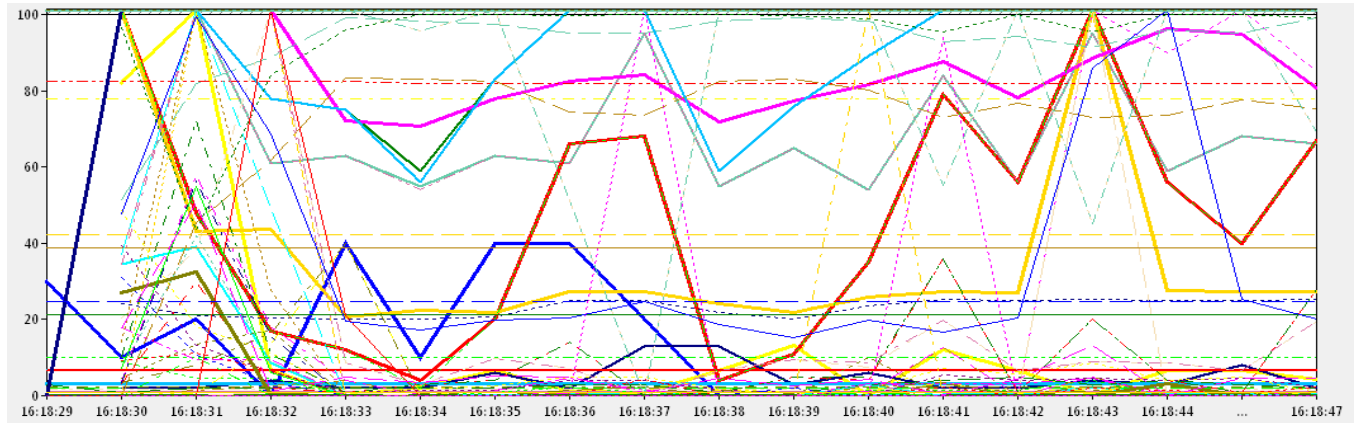
<http://www.projectclearwater.org/wp-content/uploads/2013/05/Clearwater-Deployment-Sizing-10-Apr-13.xlsx>

<http://www.projectclearwater.org/technical/clearwater-performance/>

# Mit mérjük/mi a lényeges?

- Metrikák “kicsiben”

- Pl. Task manager, Resource monitor, ugyanez szerver oldalon....

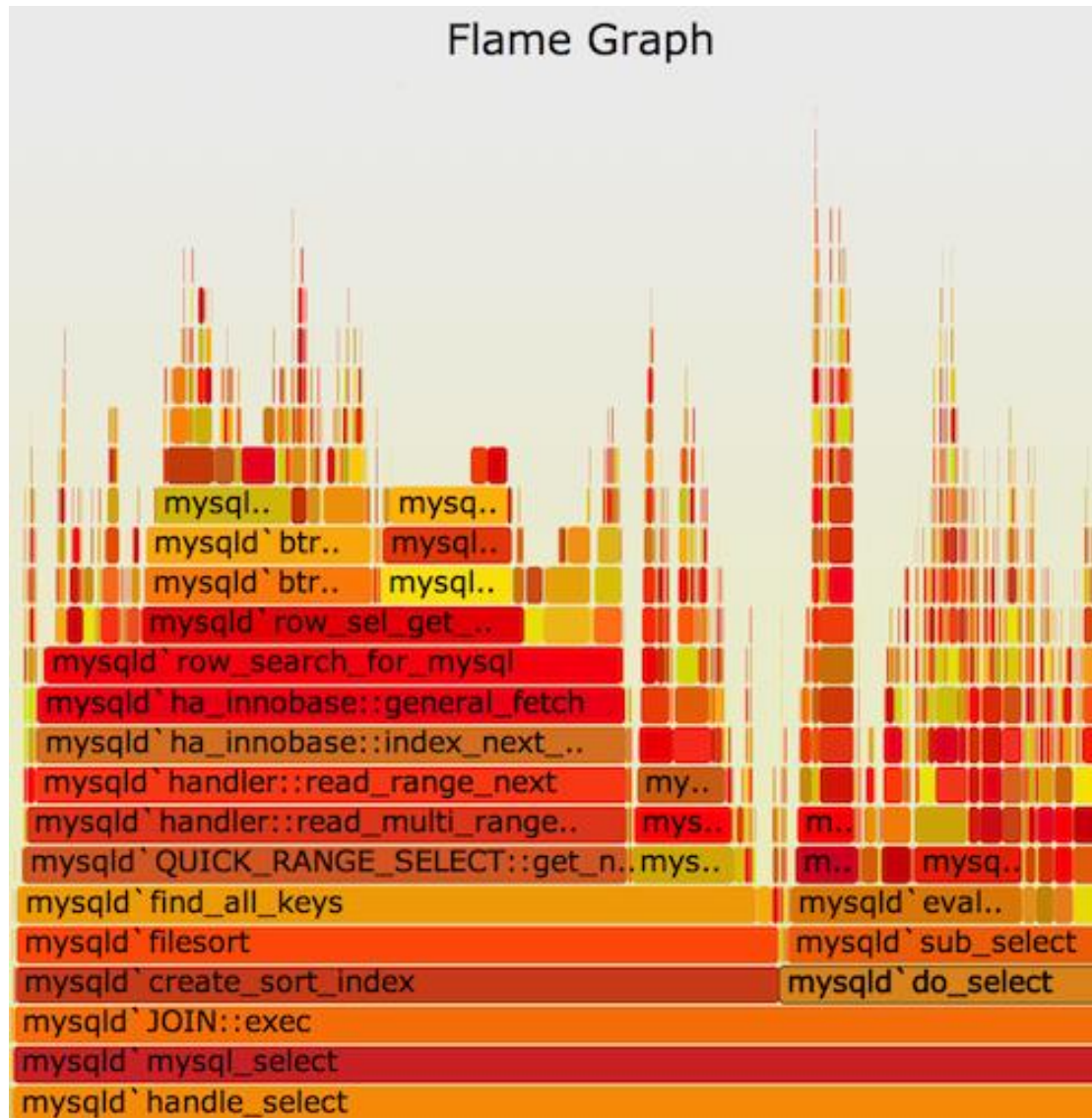


- Metrikák “nagyban”

- Pl. virtualizált rendszer

- Melyik az érdekes?

# Példa: mit számol ez a gép enyit ....?



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

# 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)

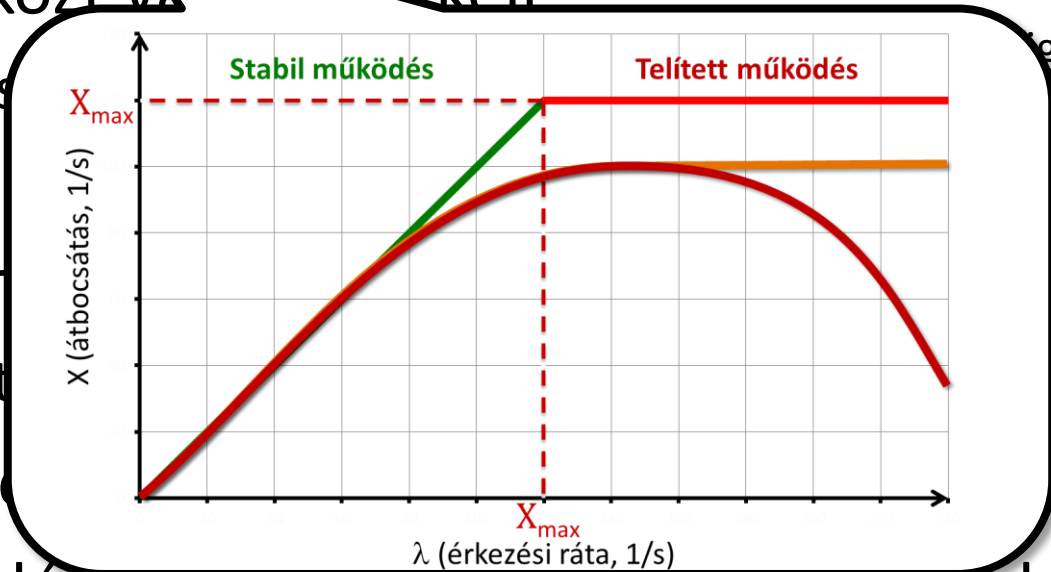


# Hol közelítünk?

- A gyakorlatban az értékek nehezen mérhetőek
  - (pl. válaszidő ingadozik, felpörgés, ...)
- Az alkalmazások versengenek
  - ( $2 * \lambda \neq \lambda + \lambda$ )
- Erőforrások közt van versengés, ami kell



- terheléselosítás
- Pl. ugyanannak a rendszernek figyeljük, mert...”
- Konkrét beépítés a) ünk
- Pont ez a Lit
- Egy feladat v
- A rendszer felépítése/parametererei változhatnak





A Little-törvény

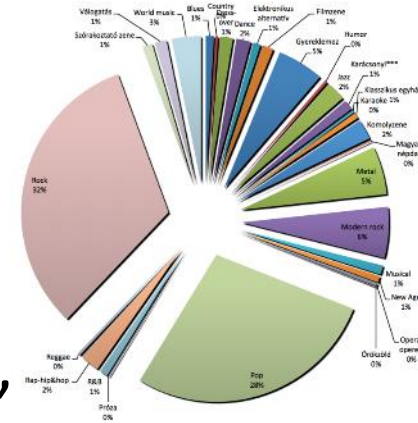
A Zipf-törvény

Terhelés változása

# TERHELÉSMODELLEK: ZIPF TÖRVÉNY

# Mi a kérések tartalma?

- Eddig: minden kérés egyforma
  - “kérem egy könyv adatait”
- Valójában: a kéréseknek tartalmuk van
  - “Kérem az *Alapítvány 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

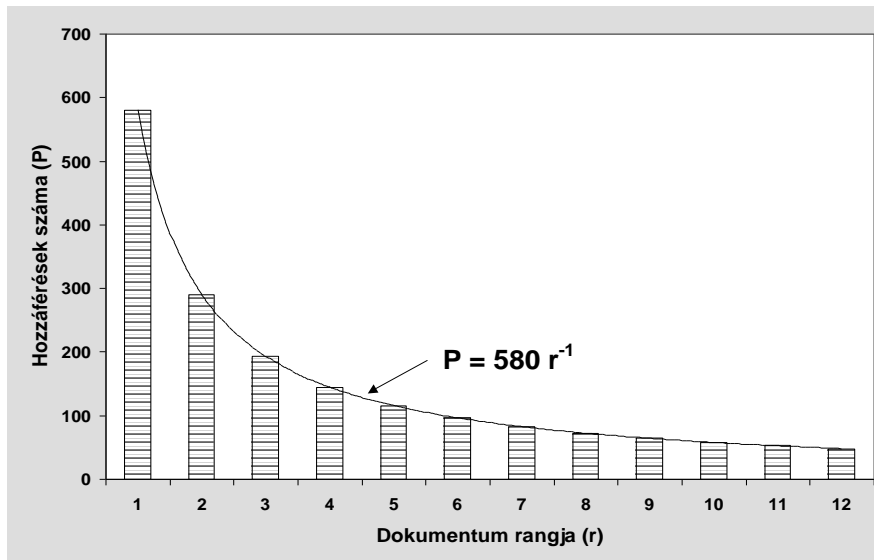


# Zipf törvénye

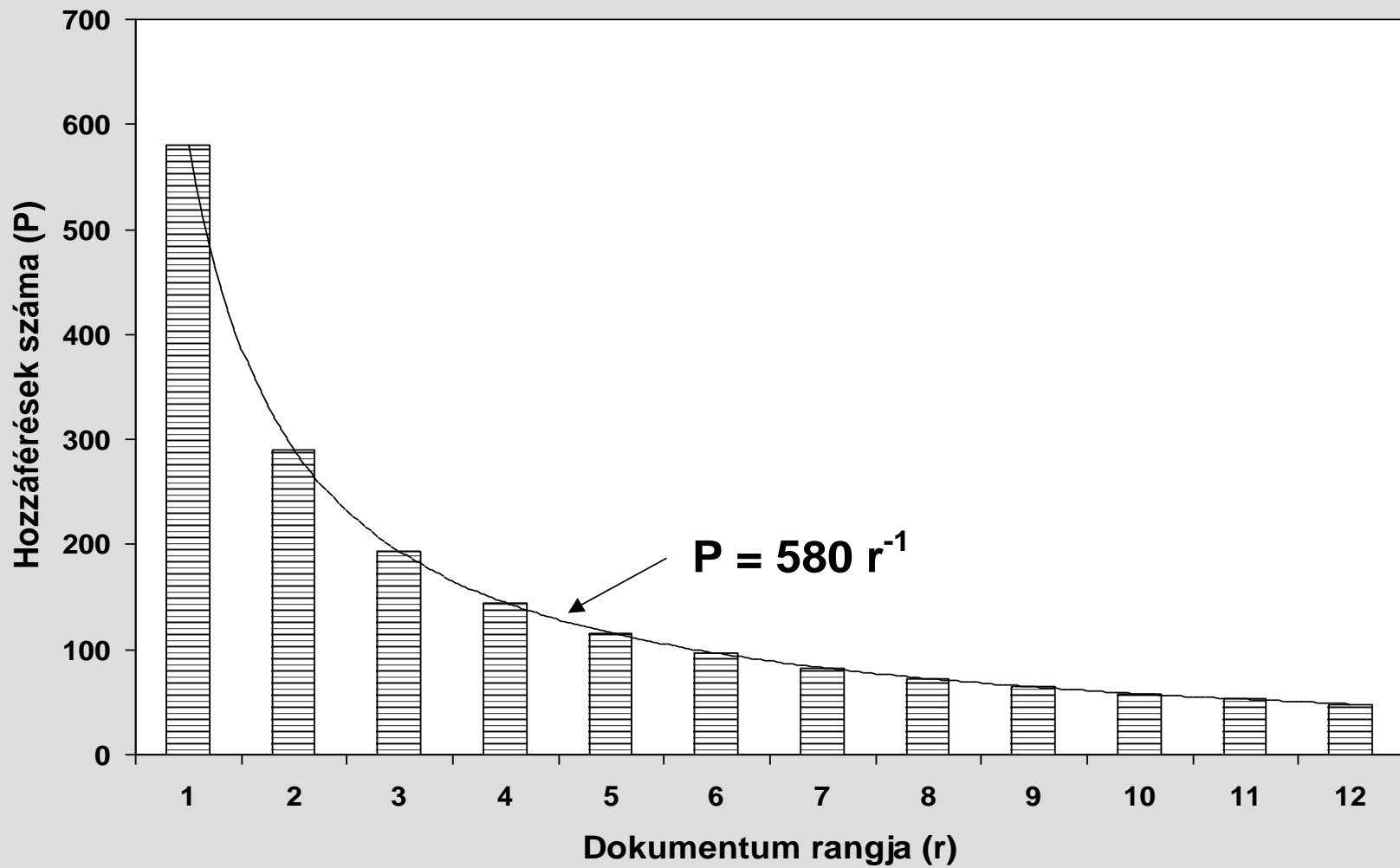
- Eredetileg: *körpuszokban* előforduló szavak népszerűségi rangsora és előfordulási gyakorisága jellegzetes eloszlása
  - Nemcsak nyelvi szövegekre igaz



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



# Zipf – Példa (1)



# 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
- $\alpha$  – a korpuszra jellemző 1 közeli érték
- Egyszerűsítve ( $\alpha = 1$ ):
  - $f$  (frequency) gyakoriság
  - $p$  (popularity): a szöveg „rangja” (csökkenő sorrendben)

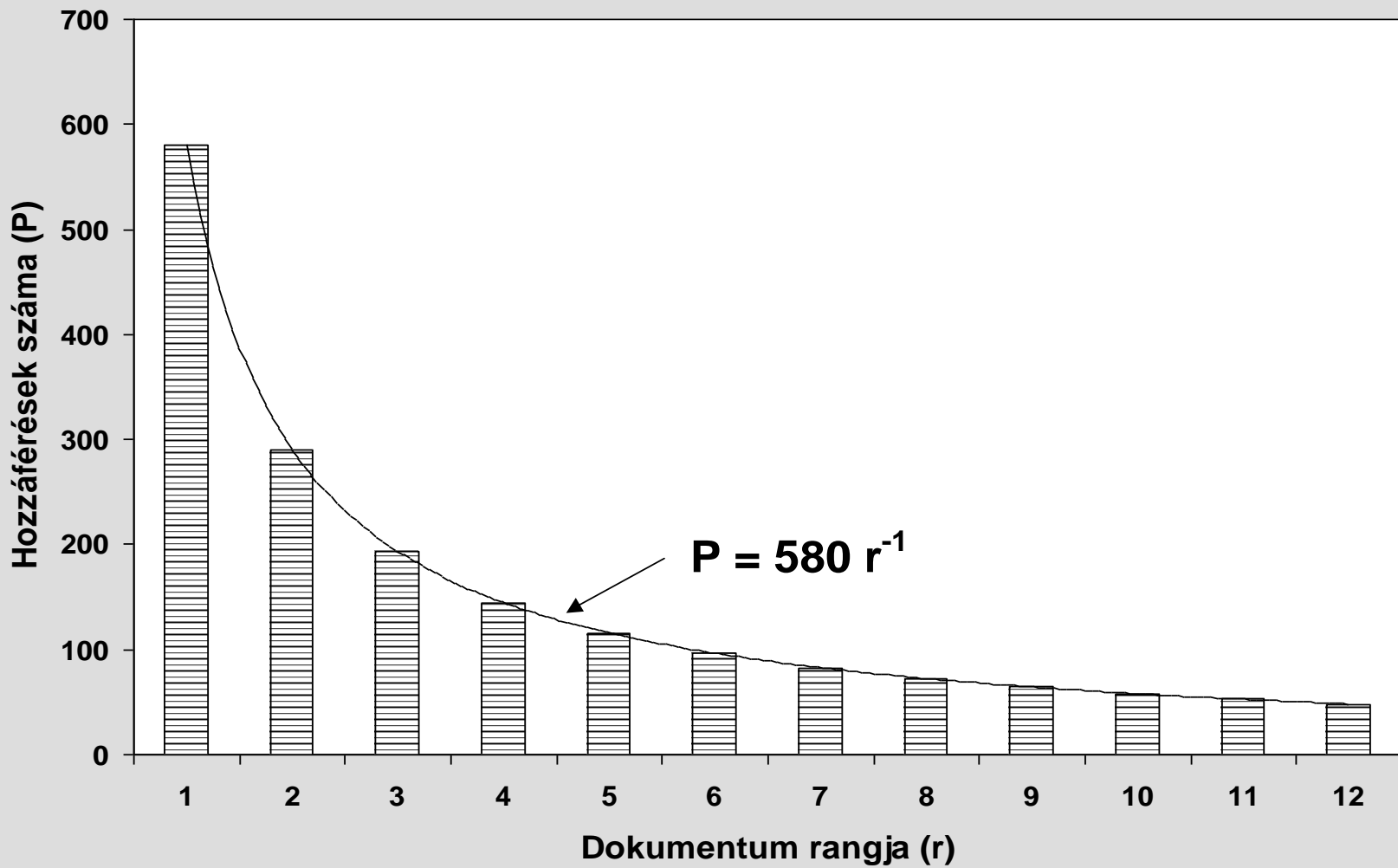
# Zipf törvénye – pl. Web dokumentumokra

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

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

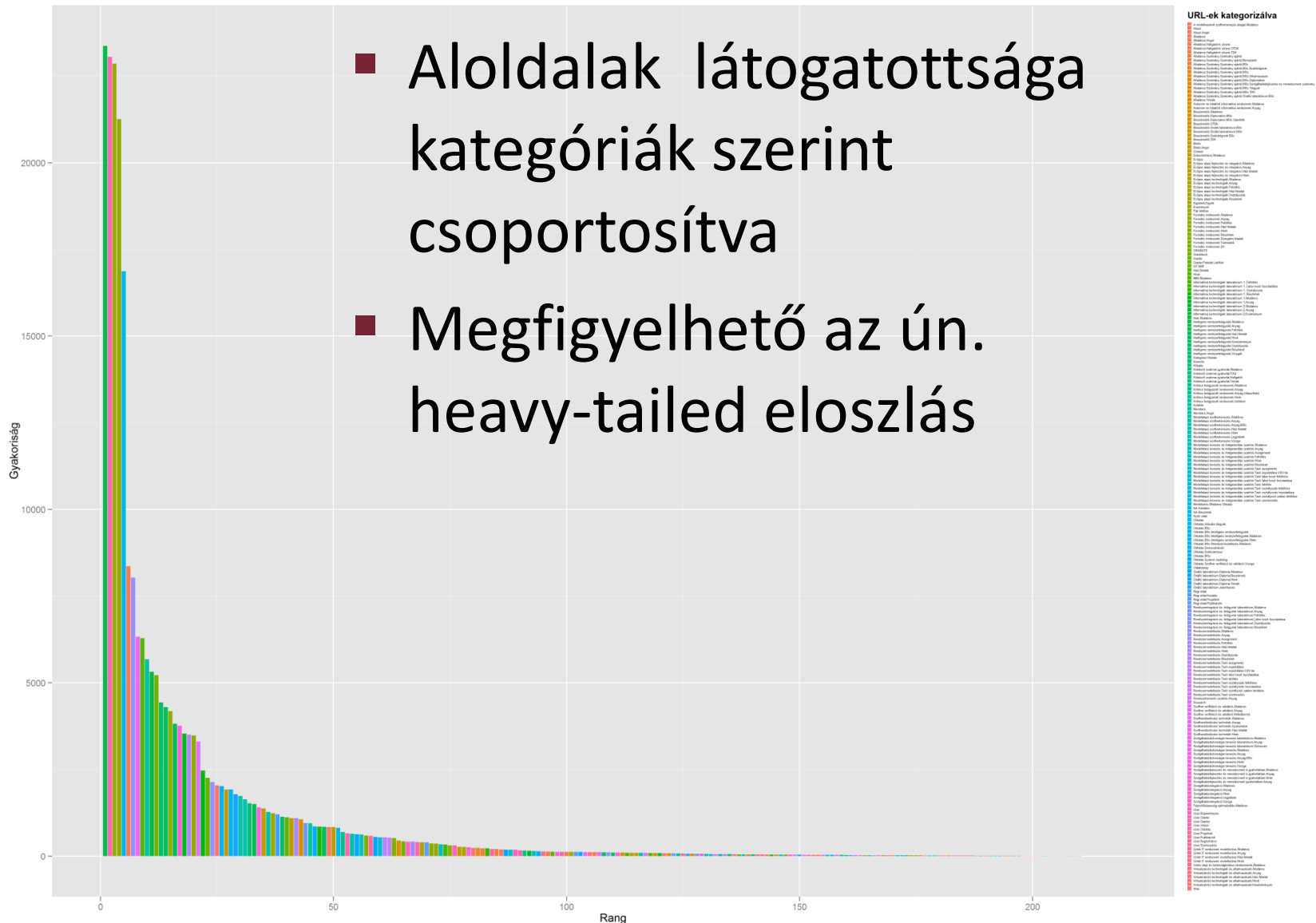
Bővebben: <http://www.hpl.hp.com/research/idl/papers/ranking/adamicglottometrics.pdf>

# Zipf – Példa (1)



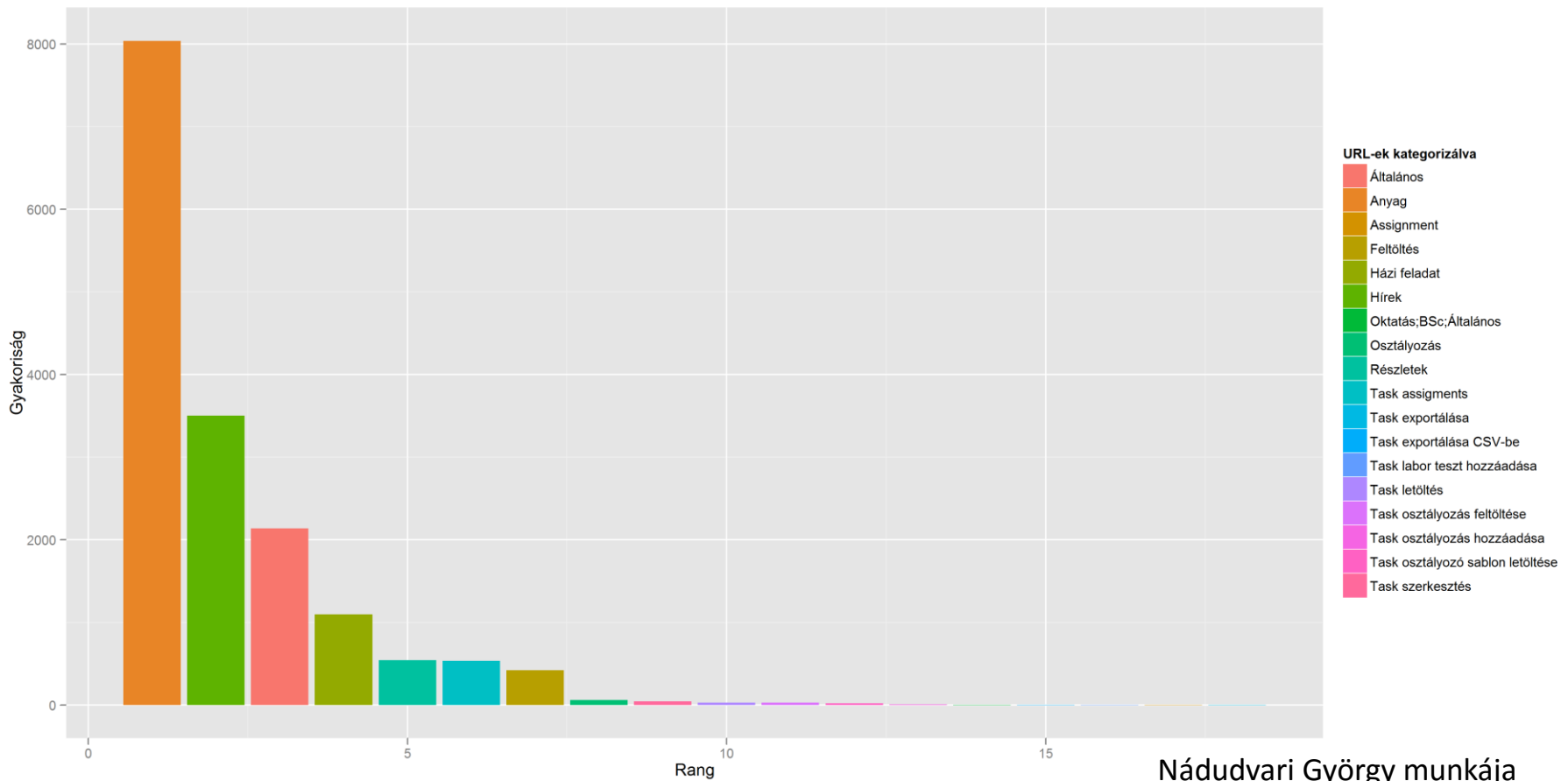


# Zipf – Példa: Tanszéki honlap (1)



# Zipf – Példa: Tanszéki honlap (2)

- A Rendszermodellezés tárgy oldalainak látogatottsága



Nádudvari György munkája

A Little-törvény

A Zipf-törvény

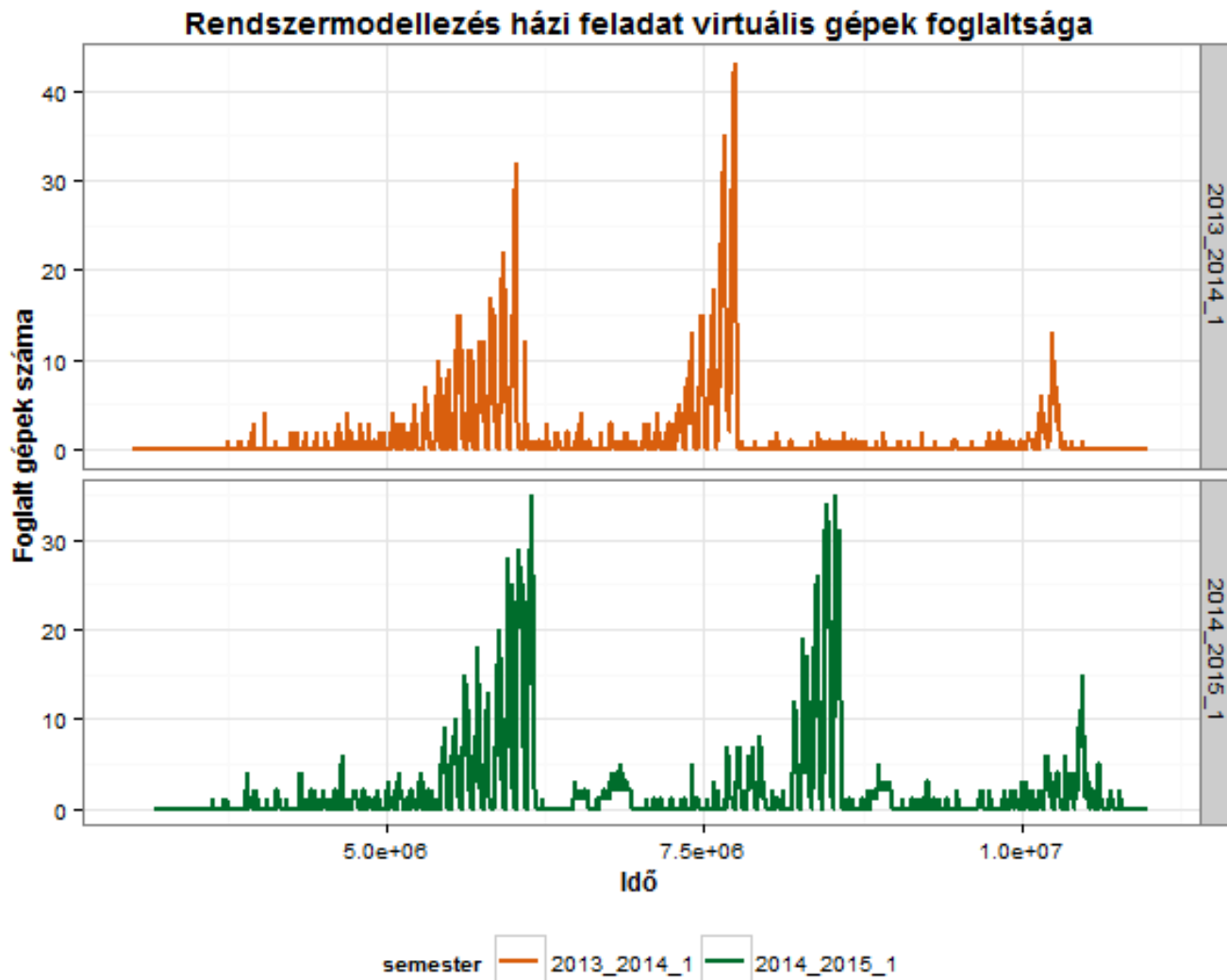
Terhelés változása

# 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

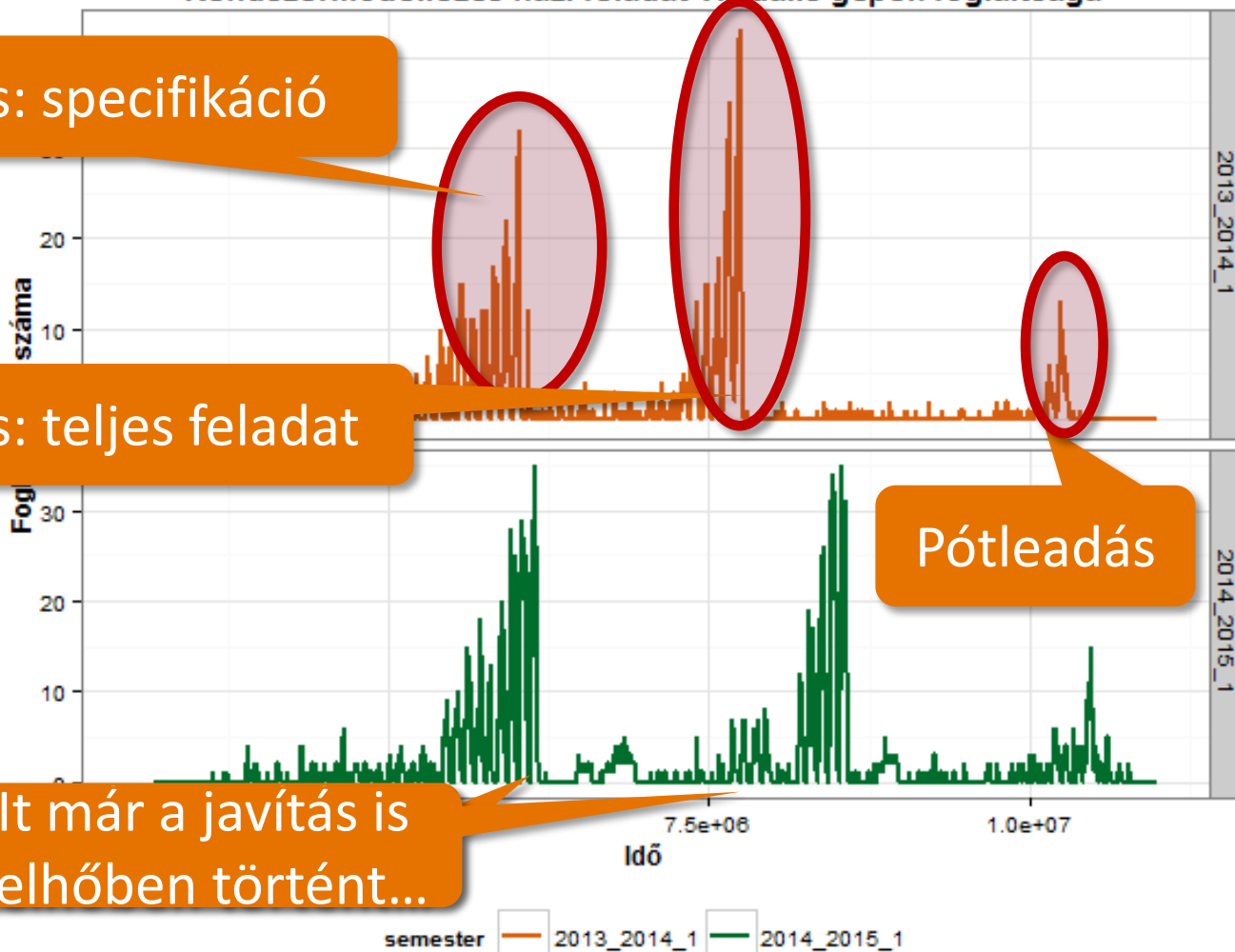
Rendszermodellezés házi feladat virtuális gépek foglaltsága

1. fázis: specifikáció

2. fázis: teljes feladat

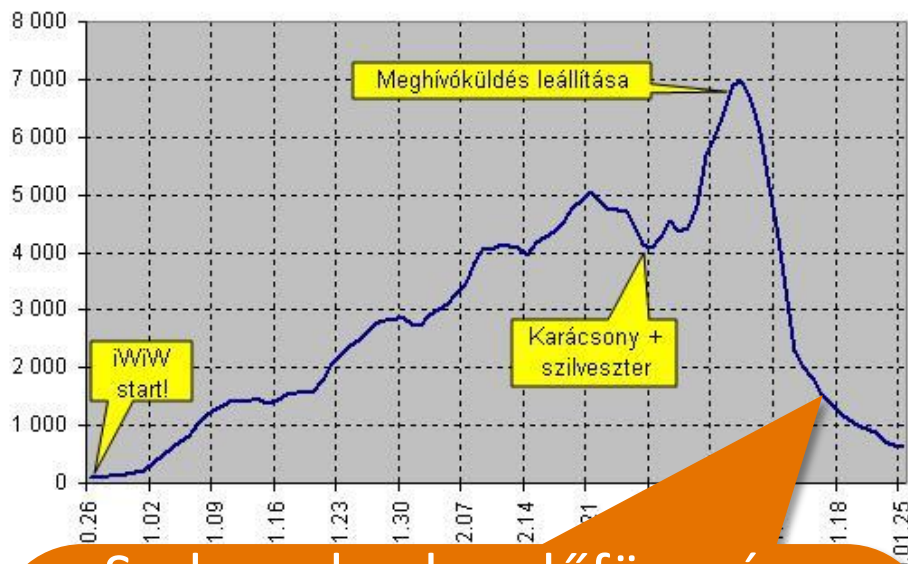
Pótleadás

It már a javítás is felhőben történt...



# Valós (történelmi) terhelés példa (iwiw)

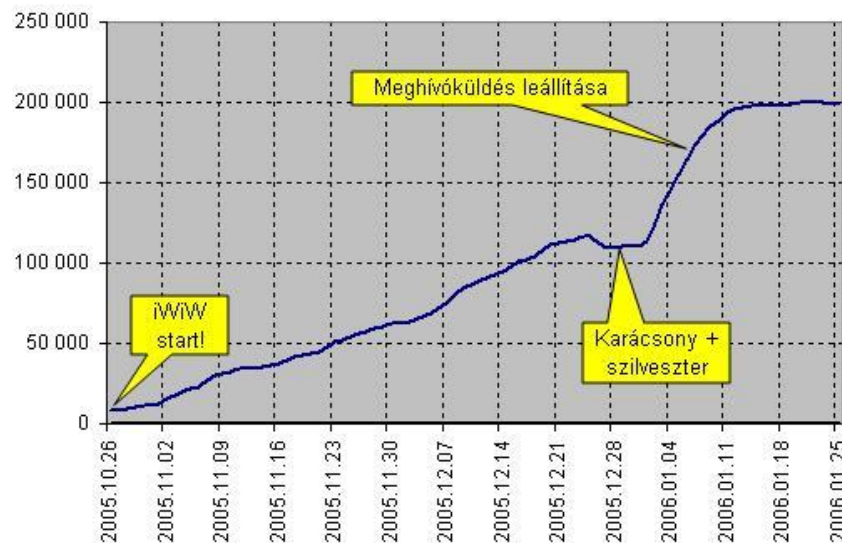
Napi regisztrációk (előző hét nap átlaga)



Szakaszokra becselőfüggvény  
illeszthető

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

Napi egyedi látogatók (előző hét nap átlaga)



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