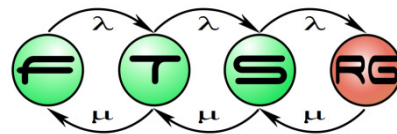
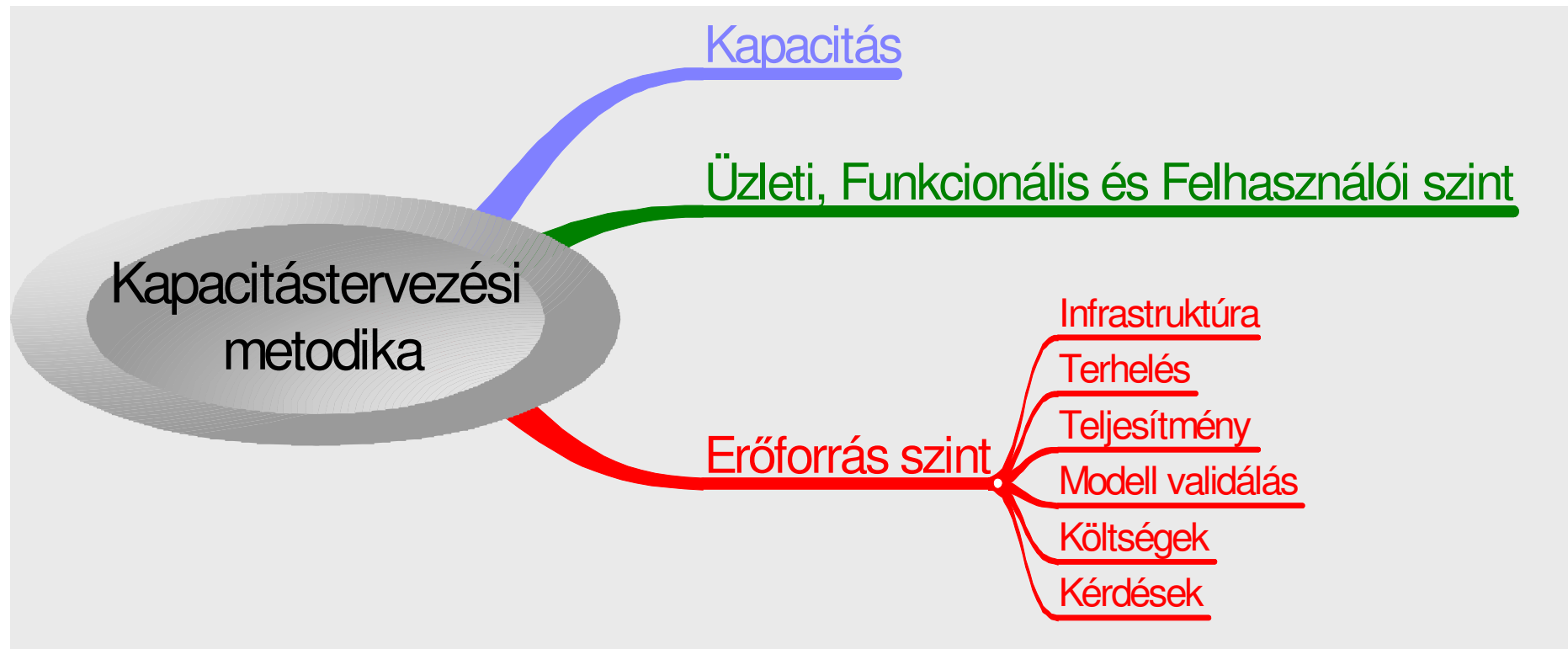


## Áttekintés: Kapacitástervezési metodika



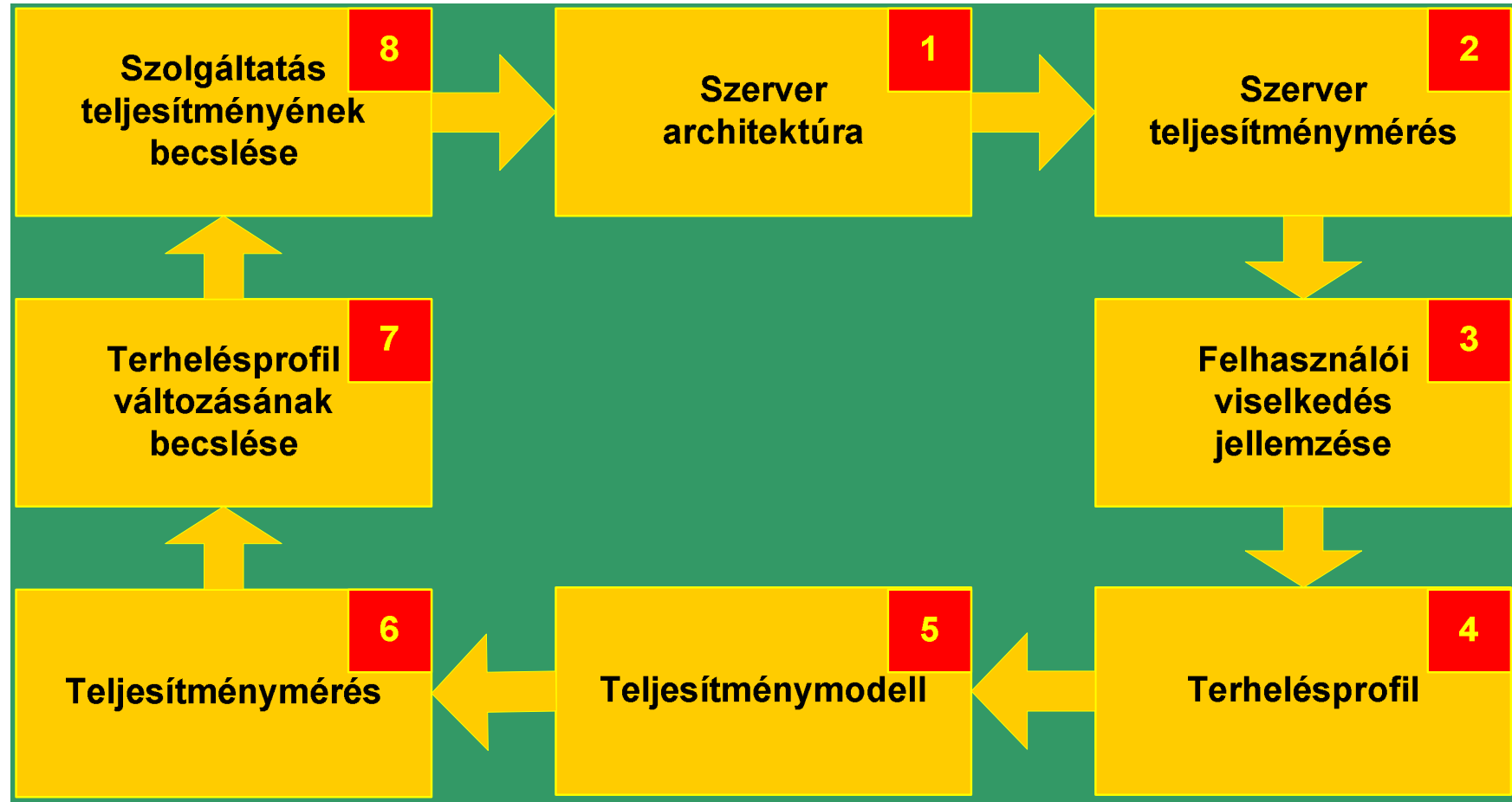
# Tartalom



# Kapacitástervezés

- „Annak **becslése**, hogy a rendszer mikor **telítődik** a terhelés hatására”
- „A leginkább **költséghatékony** módszer megtalálása, mellyel a rendszer **túlterhelése** a lehető legjobban **késleltethető**”
- Figyelembe veszi a nyújtani kívánt szolgáltatás szintjét

# A mennyiségi analízis lépései



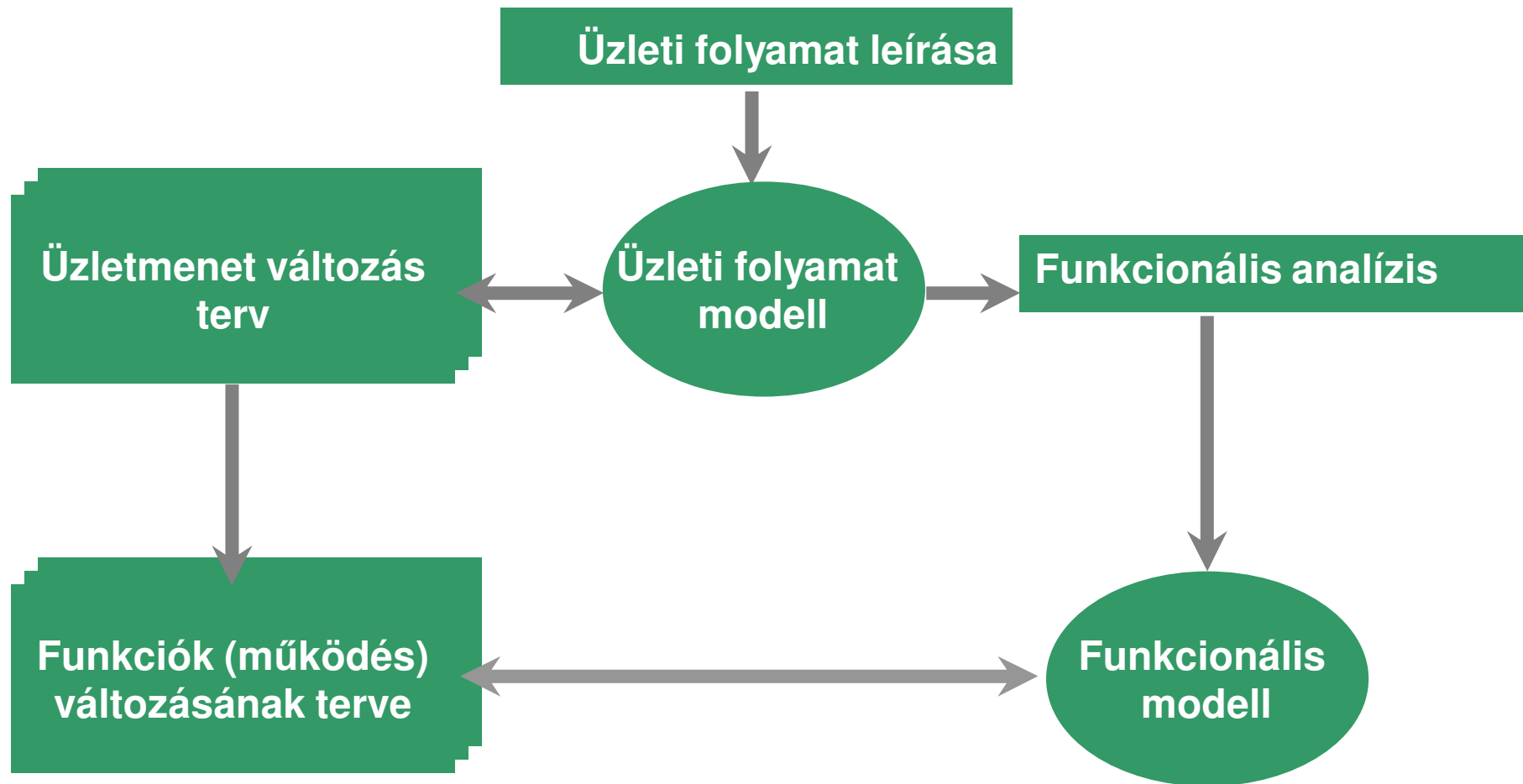
# Mitől függ az elvárt kapacitás?

- Service Level Agreement (SLA):
  - A vezetés által meghatározott mérőszámok (válaszidő, elérhetőség, átbecsátóképesség)
- Használt technológiák, szabványok
- Pénzügyi lehetőségek
- Kapacitástervezés:
  - **kvantitatív** szemlélet
  - cél: ne kelljen túl sűrűn változtatni a rendszeren

# Mitől nőhet a terhelés?

- A rendszer változatlan, nő az átlagos terhelés
  - pl. eddigi 10000 helyett 15000 látogató
- Új alkalmazások/szolgáltatások
  - pl. szemelvények a könyvekből
- Változik a felhasználók viselkedése
  - hirtelen változás az informatikai rendszeren kívül (pl. hirdetési kampány, 9.11.)
  - navigációs minták változnak (többen keresnek)

# Üzleti szintű leírás lépései



# Üzleti folyamat leírása

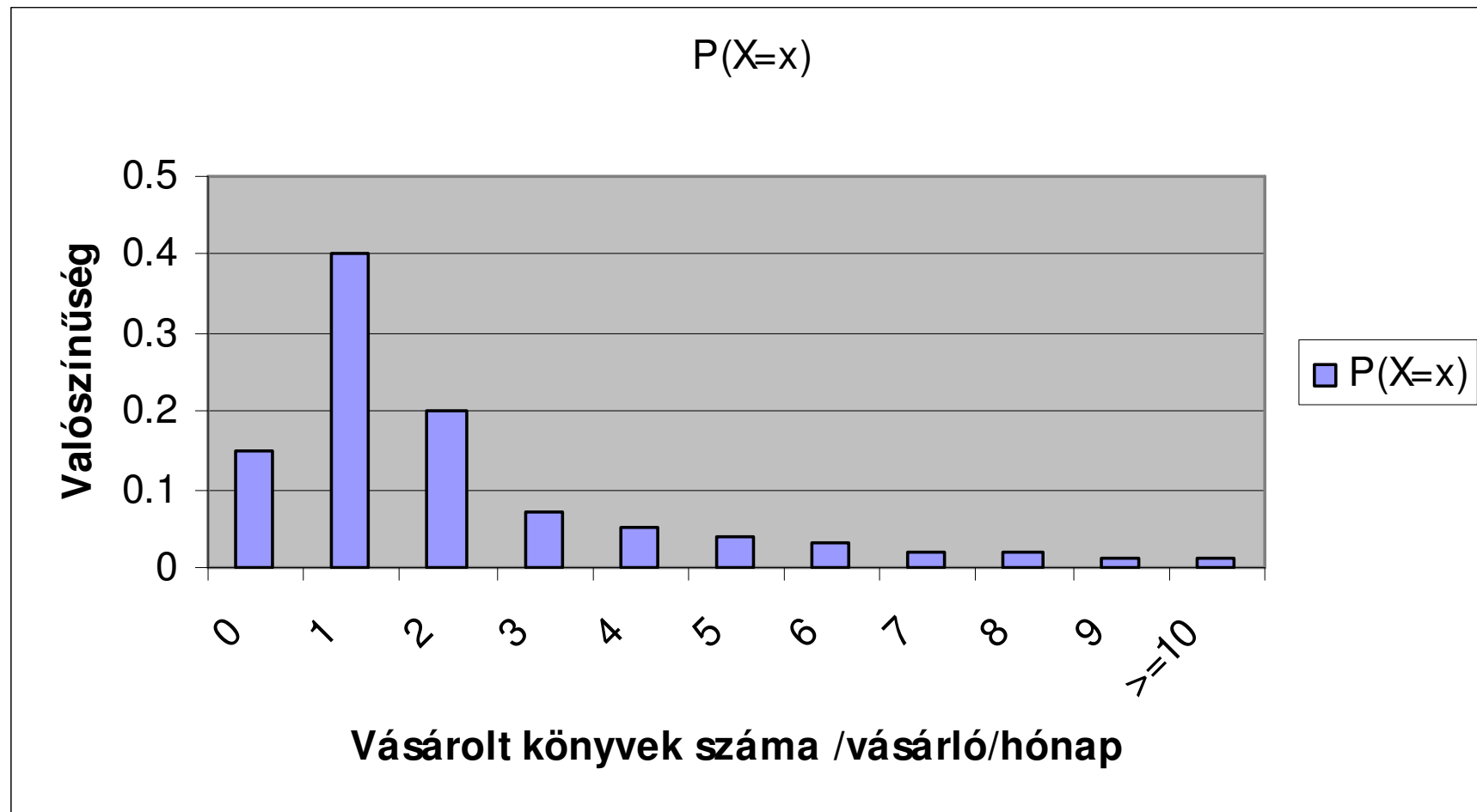
- Példa folyamat:
  - online könyvkereskedés
- Folyamat típusa
  - pl. B2C áruház, C2C aukció, B2C szolgáltatás
  - itt: B2C áruház
- Kiszállítás módja
  - azonnal letölthető (pl. cikk)
  - periodikusan küldik (upgrade)
  - fizikai kiszállítás
  - itt: fizikai áruküldés



# Üzleti folyamat leírása

- Külső szolgáltatások
  - hirdetés
  - fizetés külső szolgáltatás segítségével
  - itt: más cégek honlapjai (banner)
- Mennyiségi leírás
  - hány könyv van a rendszerben, most: 4000, cél : 10000
  - milyen árkategóriából mennyit akarunk eladni
  - vásárlási minta (milyen időszakban milyen valószínűséggel érkeznek bizonyos kérések)
  - egyéb eddigi statisztikák
  - készletek

# Online könyvesbolt vásárlási mintája

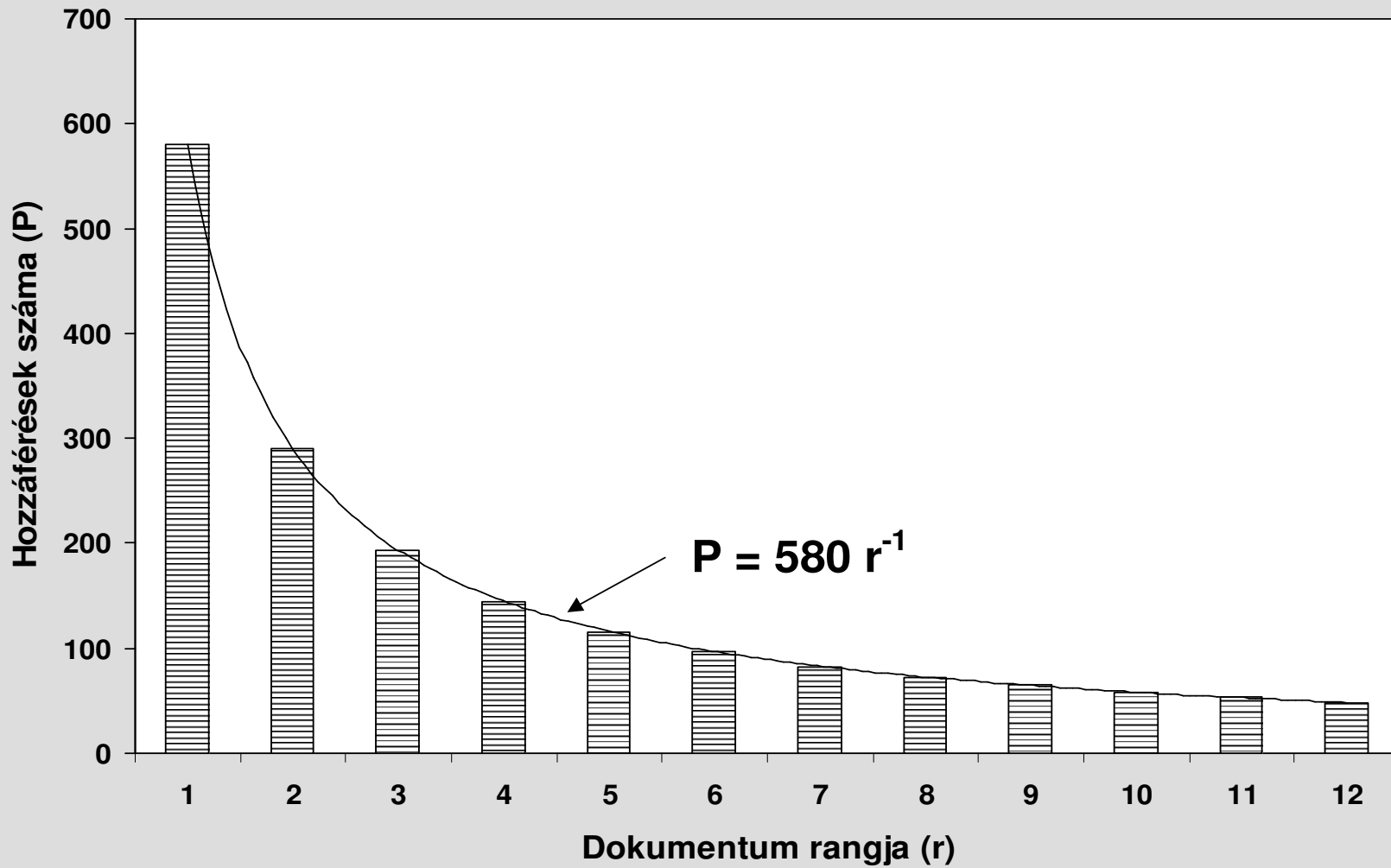


# Zipf törvénye

- ***p*** (popularity ):
  - szöveg „rangja” (csökkenő sorrendben)
- ***f*** (frequency):
  - gyakoriság  $f \sim \frac{1}{p}$
- Web dokumentumokra:
- ***P***: hivatkozások (elérések)
- ***r***: rang (1 = leggyakoribb)
- ***k***: pozitív konstans

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

# Zipf törvénye (példa)



# Üzleti folyamat leírása

- Nyitvatartás
  - lehet adott időszak (H-P/9-17)
  - lehet állandó (7\*24)
- Piac meghatározása
  - földrajzilag pl. 85 % USA - letöltések miatt is érdekes
- A leírás eredménye az üzleti (folyamat) modell - business modell

# Funkcionális analízis

- A leírás eredménye a funkcionális modell lesz
- Milyen szolgáltatások (funkciók) valósítják meg a folyamatot?
  - A Web oldal (portál) funkciói ~ az oldalak/menüpontok
  - Pl. Könyv kiválasztása, Személyes adatok bevitele, Rendelés lemondása, stb.
- Hogyan jellemezhetők ezek a funkciók?
- Interakciós modell
  - hogyan kommunikál a felhasználó a folyamattal (hogyan éri el az adott szolgáltatást az oldalon belül)
  - pl. kell X darab HTML form

# Felhasználói szint

- Felhasználó viselkedésének modellezése
- CBMG megalkotása
- Statikus: oldalak szerkezet alapján
- HTTP logok alapján
- Hasonló viselkedésű felhasználók csoportosítása
- Hogyan változik a modell az új funkciók bevezetése után?
  - Pl. a Home oldalról egyenesen mehet a Felolvasáshoz?

# Felhasználói, terhelés és erőforrás modellek

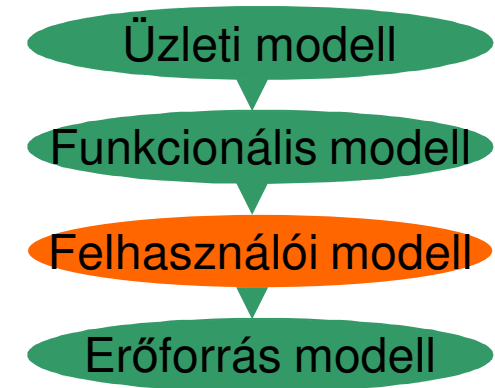
## ■ Felhasználói modell

- navigációs minták: felhasználhatók a későbbi terhelés előrejelzéséhez (mi történik, ha hirtelen több felhasználó jelenik meg, többet keresnek, stb.)
- eszköz: **Customer Behavior Model Graph** (CBMG) ,  
**Customer Behavior Model Statechart** (CBMS)
- terhelési paraméterek meghatározása: ha csak a jelenlegi terhelési modell felírása a cél, elég egy kevésbé részletes felírás
- eszköz: **Customer Visit Model** (CVM), kevésbé részletes, nem használható előrejelzésre
- mindig egy session-t vizsgálunk (azonos felhasználótól egy látogatás alatt érkezett kérések sorozata )



# Felhasználói modell – CBMG

- Irányított gráf az oldalak közti lehetséges átmenetek és valószínűségeik ábrázolására
- $n$  csúcs, ahol
  - 1. csúcs: **Entry** állapot, absztrakt belépési pont, minden felhasználó innen indul ide nem tér vissza
  - $n$ . csúcs: **Exit** állapot, absztrakt kilépési pont, a folyamat vége (nem mindig ábrázoljuk)
  - a többi csúcs megfelel a felhasználó által elérhető szolgáltatásoknak (oldalaknak)
- élek: az oldalak szerkezete határozza meg a csúcsok közt lehetséges átmeneteket



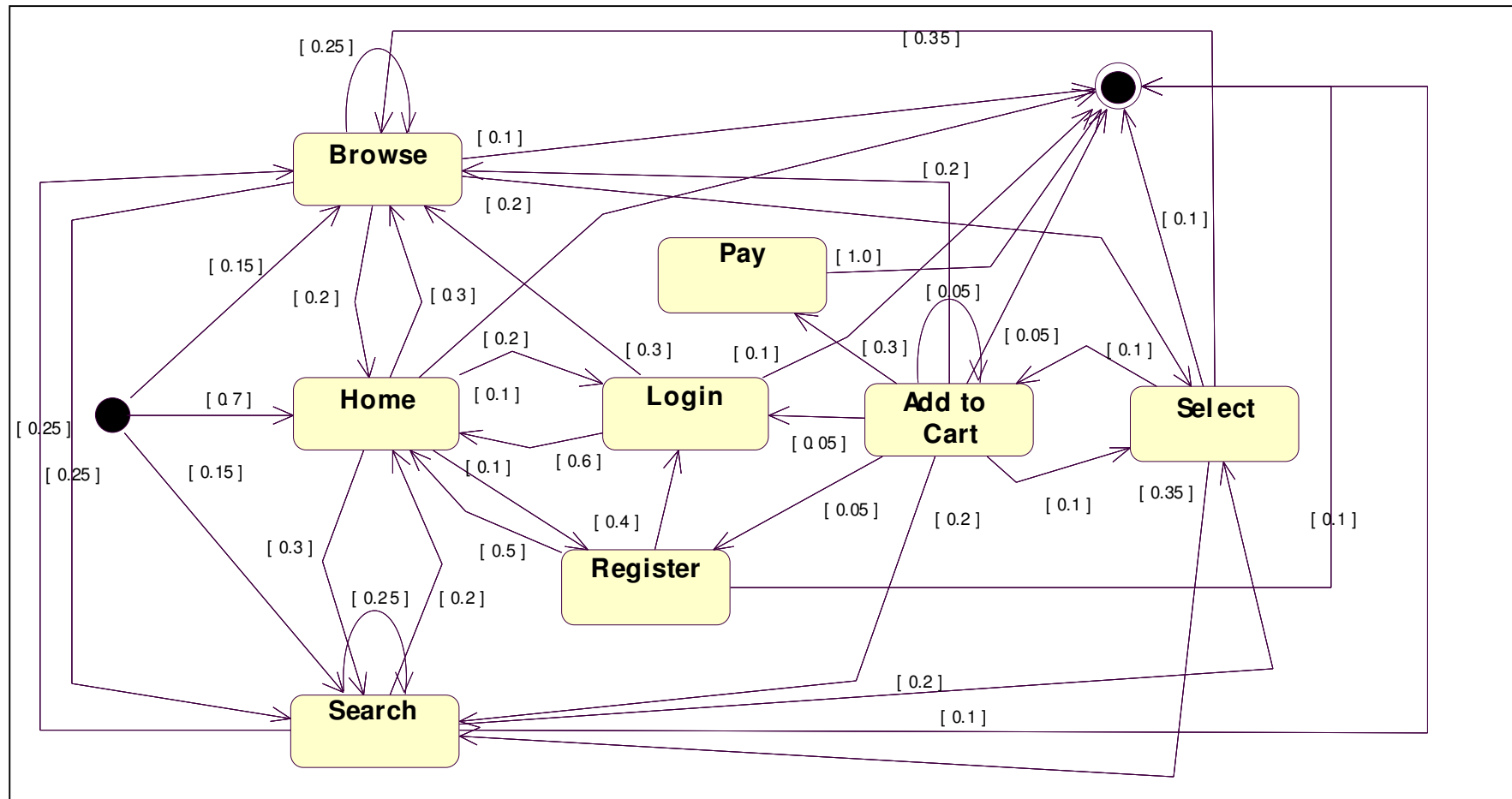
# CBMG meghatározása

- Statikus CBMG: oldalak és köztük lévő lehetséges átmenetek
  - az oldalak által nyújtott szolgáltatások meghatározása
    - (pl. Login, Register, Add Item, Remove Item, Pay, Get Quotes, Download, Subscribe, stb.)
  - a szolgáltatások halmazának finomítása – az infrastruktúrát különböző mértékben terhelő szolgáltatásokra
    - (pl. Download szétválasztása Download Audio, Download Video szolgáltatásokra)
  - lehetséges átmenetek meghatározása – az oldalak megjelenítésének vizsgálatával
    - (linkek, formok, menüpontok, stb.)

# Felhasználói modell: CBMS

- Customer Behavior Model Statechart
- UML állapotdiagram
- Hasonló információt hordoz, mint CBMG, de mindezt szabványos módon jeleníti meg
- Állapotok: az oldal funkciói (állapotai)
- Lehetséges átmenetek: rendszer modellje alapján
- Átmenetekhez valószínűség tartozik (dinamikus viselkedés)
  - meghatározás: „Terhelés” ea.

# Felhasználói viselkedés állapotdiagram modellje



# Felhasználói viselkedés vizsgálata

- Dinamikus CBMG ill. CBMS alapján különböző mérőszámokat határozhatunk meg:
  - *Hits/sec*: az oldalról letöltött objektumok száma (képek, bannerek is)
  - *Page View/Day*: adott oldalt hányszor nézték meg
  - *Click-throughs*: hányan néztek meg egy adott hirdetést
  - *Unique Visitors*: hány különböző látogató volt egy adott időszakban

# Felhasználói viselkedés vizsgálata

- További mérőszámok (felhasználói modell és egyéb statisztikák alapján):
  - *Revenue Throughput*: az oldal által elért átlagos bevétel
  - *Potential Loss Throughput*: mennyibe kerül a szolgáltatás kiesése egy adott időszakban
  - *Visit Ratio*: átlagosan hányszor vesznek igénybe egy adott szolgáltatást (egy session alatt)
  - *Buy to Visit Ratio*: átlagosan hányszor vásárolnak egy session alatt (tényleges eladási tanzakció)
  - *Average Session Length*: egy session átlagosan hány szolgáltatást vesz igénybe (nem időt mér)
  - mindezen mérőszámok változása, ha a bemenő paraméterek megváltoznak (pl. egyes átmenetek valószínűségei)

# Felhasználói viselkedés vizsgálata

- Az előbbi példából kinyerhető adatok :
  - *Buy to Visit Ratio (BV)*: **Pay** állapot előfordulásának várható értéke,  $\sum_k [\prod P(i,j)]$  az összes lehetséges **Entry-Pay** útra ( $k$  darab), *BV* értéke itt 0.058 (5.8 %)
  - *Végrehajtott eladási tranzakciók száma*: ha naponta 100000 látogató (session) van, akkor átlagosan  $100000 \cdot 0.058 = 5800$
  - *Átlagos session hossz* (average session length): az állapotok átlagos előfordulásainak összege, itt. 7.998 (**Exit** és **Entry** állapotok előfordulási gyakorisága mindig 1, itt nem számítanak)
  - mindezen adatok kiszámolhatóak más átmeneti valószínűségek esetén is (pl. ha megnő a keresés után vásárlók száma)

# Customer Visit Model (CVM)

- CVM: különböző típusú felhasználók viselkedését jellemző vektorok (melyik állapotban hányszor járt)
- A felhasználók csoportosítása cluster technikákkal történik, ld. „Terhelés” ea.
- Nem jelzi az egyes állapotok közti átmenetek gyakoriságát, kevésbé alkalmazható előrejelzésre

	Session1	Session2	Session3
Home	1	2	3
Browse	4	8	4
Search	5	5	3
Login	0	1	1
Pay	0	0	1
Register	0	0	1
Add to Cart	0	2	1
Select	3	3	2



# Session azonosítás

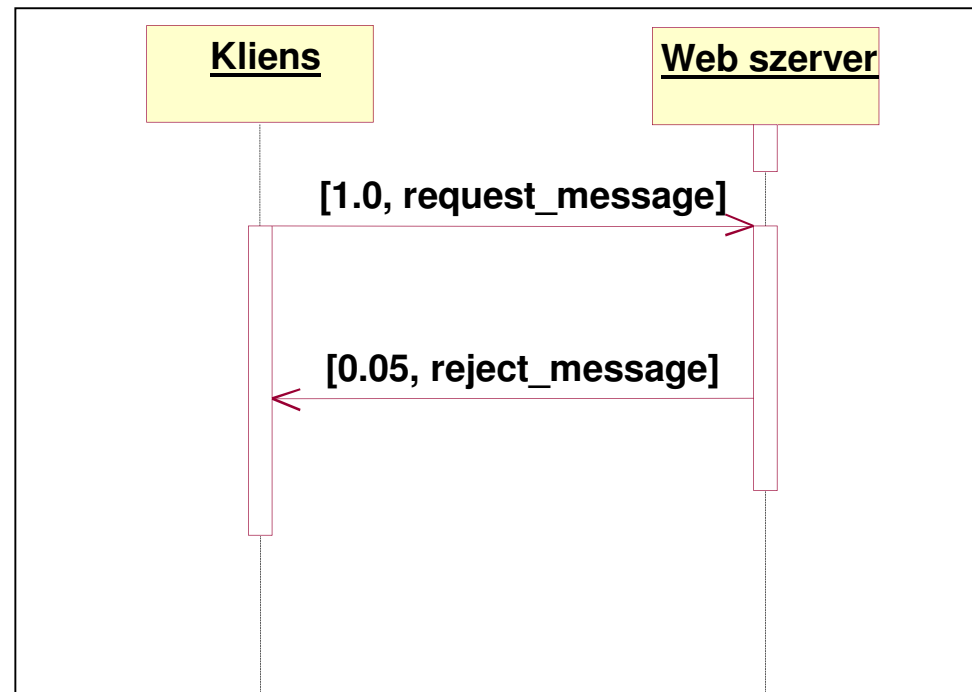
- Cookie-k használatával
  - szerver generál egy ID-t, amit a kliens minden kéréssel elküld, az kliens mellett az alkalmazás állapotát is tárolhatja (pl. bevásárlókocsi)
- Autentikációs mechanizmusok, rejtett mezők HTML oldalak formjaiban, dinamikus URL-ek, stb.
- Szerver logok alapján, várakozási küszöbérték (threshold) használatával
  - ha ennél hosszabb a szünet két kérés (request) elküldése közt, akkor két külön session

# C/S Interaction Sequence Diagram (CSISD)

- Szekvenciadiagramok minden lehetséges esetre
- Objektumok: kliens és a különböző szerverek (erőforrások)
- Nyilak: üzenetküldés
- Üzenetek:  $[p, m]$  párok, ahol  $p$  az üzenet küldésének valószínűsége,  $m$  a mérete (byte)
  - pl.  $[0.05, \text{reject\_message}]$ : 5% valószínűséggel reject\_message méretű üzenetet küldünk
- Minden lehetséges végrehajtási szekvencia kliensből indul ki és kliensben végződik
- CSID: C/S Interaction Diagram
  - a szekvenciák összessége

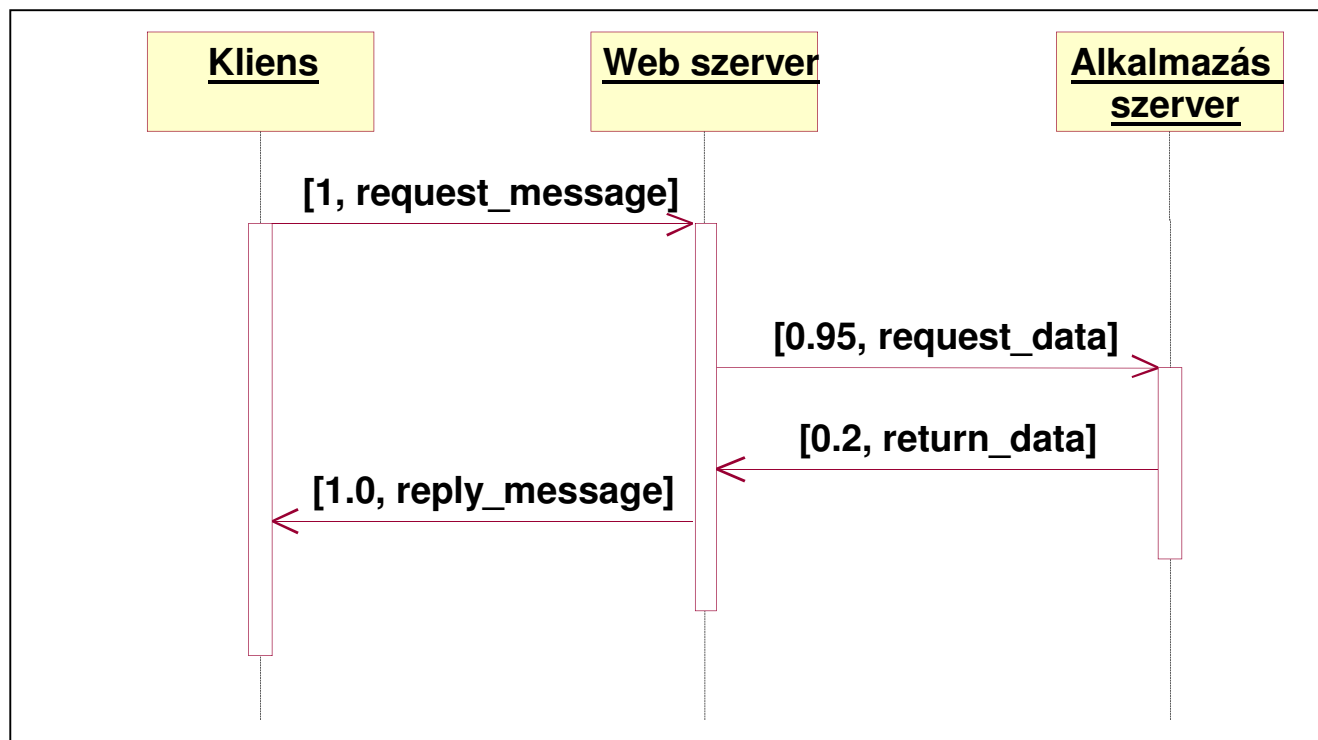
# CSISD példa 1.

- Kliens kér valamilyen adatot a Web szervertől, 3 lefutási lehetőség
- 1. eset: A Web szerver túlterhelt: elutasítja a kérést



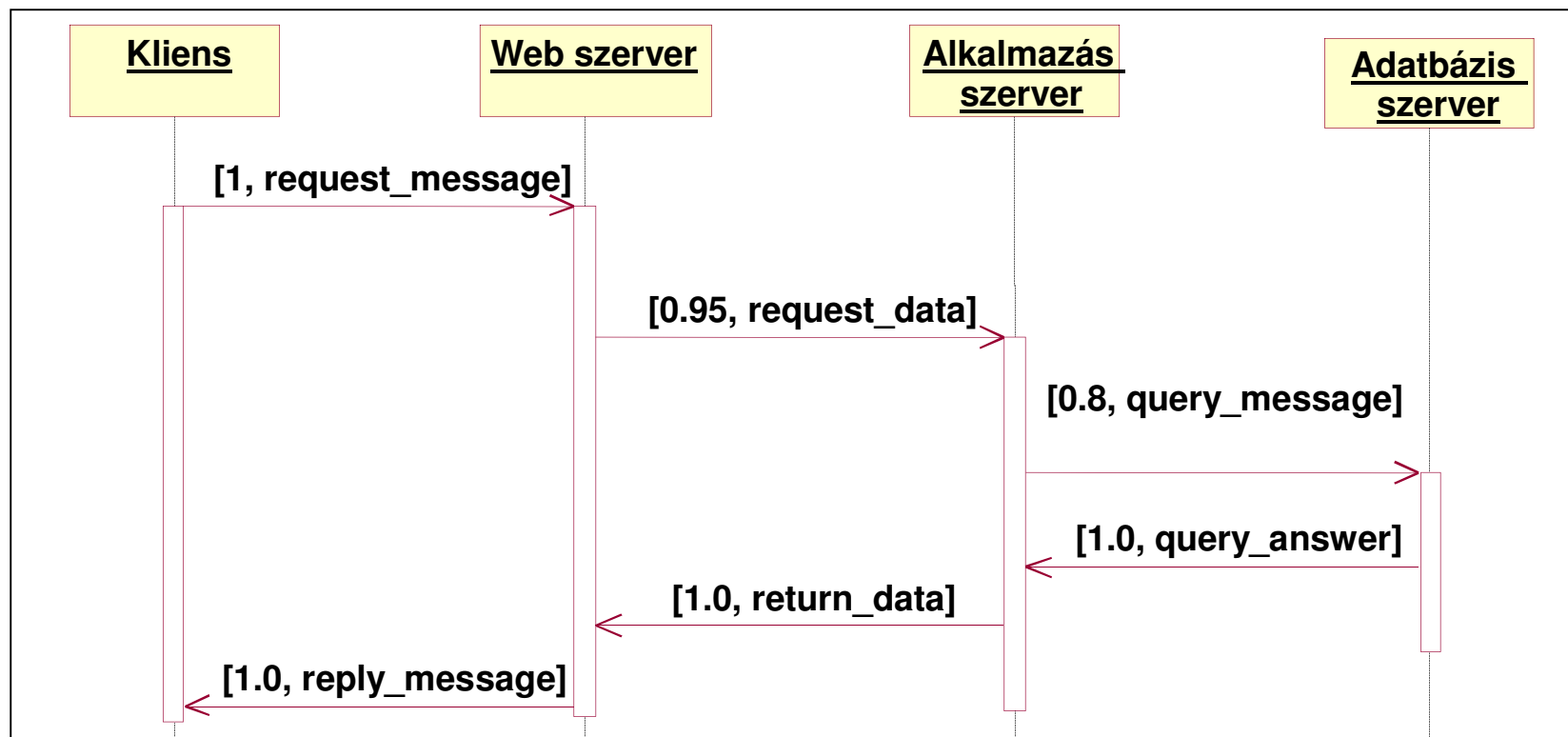
# CSID példa 2.

- 2. eset: A Web szerver továbbküldi a kérést valamilyen alkalmazásnak (alk. szerver), az alkalmazás cache-ből visszaadja az adatot



# CSID példa 3.

- 3. eset: A Web szerver továbbküldi a kérést, az alkalmazás lekérdezést hajt végre az adatbázisszerveren, majd visszaadja az adatot



# CSISD elemzése

- mennyi a lokális hálózat forgalma, ha minden szerver ugyanazon a LAN-on van?  $\Sigma (p * m)$  minden szerverek közti üzenetre
- Mi történik, ha a Web szerveret leválasztjuk?
  - (Web szerver:LAN1, többi LAN2)
  - (Szerver oldali) hálózati késleltetés meghatározása sávszélesség és üzenetméret alapján
- Milyen lesz a teljesítmény, ha minden szerver ugyanazon a gépen fut?

# Funkcionális analízis

- Felhasznált web technológiák
  - pl. ActiveX control, Java applet, stb.
- Autentikáció
  - használ-e autentikációs protokollt
  - ha igen, melyet, pl. SSL
- Példa: Személyes adatok bevitele
  - egy HTML formon keresztül történik
  - a felhasznált technológia HTML és CGI
  - autentikáció: SSL
- A kapott információk hasznosak a Client/Server Interaction Diagram felírásához

# Üzletmenet változásának vizsgálata

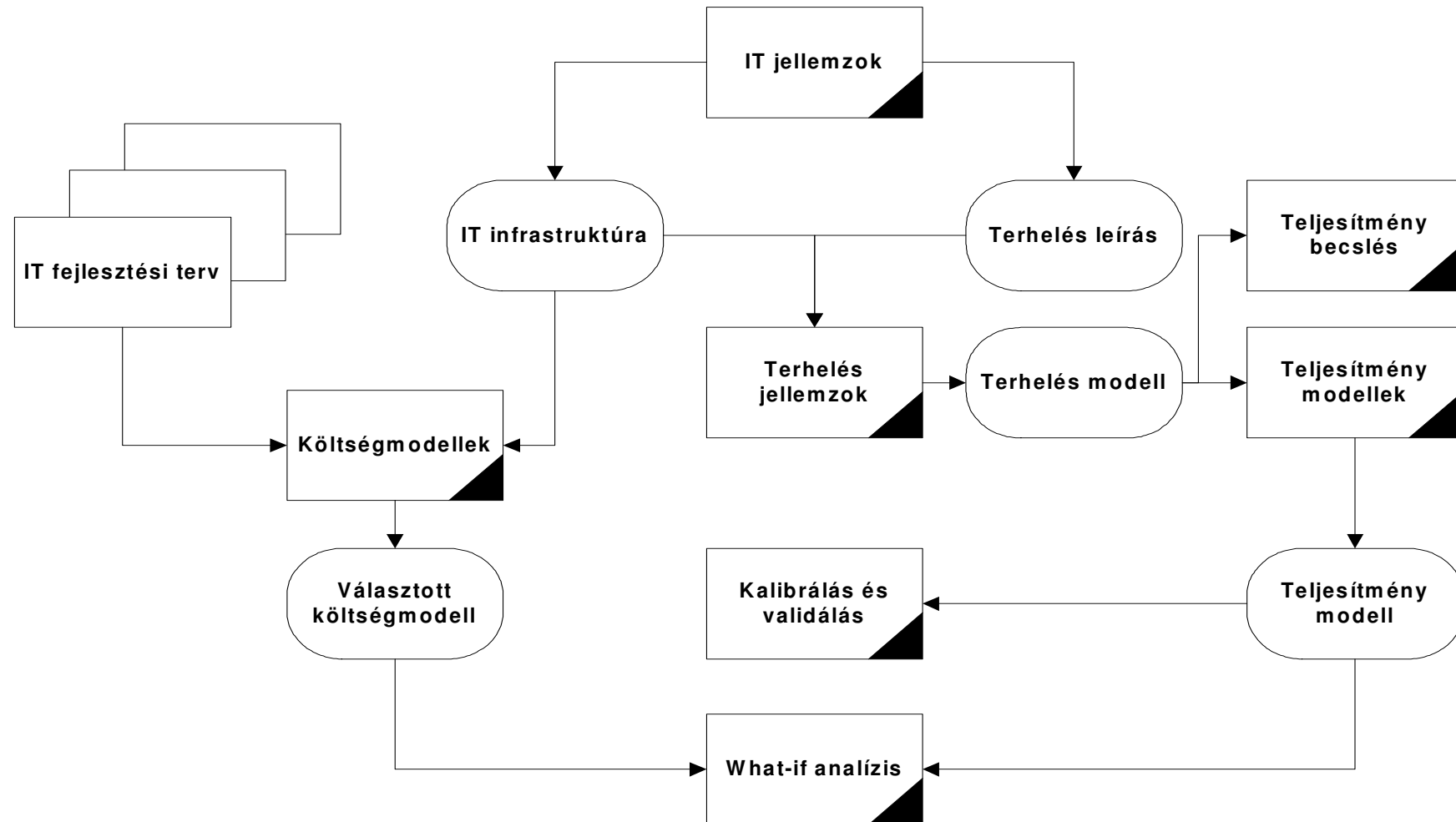
- Az üzletmenet (üzleti folyamat) várható (tervezett) változásai
- Új adatbázist építünk be a rendszerbe
  - árjegyzék használt könyvekre
  - új alkalmazás kell ennek eléréshez
  - megnő a terhelés
- Portált építünk (további szolgáltatások)
- Erős marketing kampány indul
  - meg kell növelni a kapacitást, különben az oldal lassú lesz és az eddig vásárlók is elpártolnak
  - 3 terv a hatásra: optimista, realista, pesszimista



# A funkciók változásának vizsgálata

- Az előbb vizsgált változások hatásai a funkcionális modell szintjén
  - nincs mindig funkcionális szintű változás, pl. ha a rendszerben elérhető kereskedők száma nő, attól nem változik a modell
- Példa: Multimédiás bővítés
  - új funkció: Részletek a könyvből (felolvasás); egy HTML form a felhasználói interakcióhoz, a használt technológia HTML és Quicktime, nincs autentikáció

# Erőforrás szintű kapacitástervezés lépései



# Erőforrás szint

- Felhasznált IT erőforrások konkrét meghatározása az előző modellek alapján
- Informatikai környezet meghatározása
  - infrastruktúra és
  - terhelés leírása
  - e-business funkciókhoz tartozó programok meghatározása
- Infrastruktúra:
  - hardver (szerver gépek, diszk farmok, routerek, tűzfalak...)
  - szerverek (Web ~, alkalmazás ~, adatbázis ~, DNS...)
  - szoftverek (OS, middleware, adatbáziskezelő...)
  - Hálózati kapcsolat, hálózati protokoll
  - Fizetési szolgáltatás (Payment service)

# Erőforrás szintű terhelés meghatározása

**R: adott erőforrás**

**F: funkciók halmaza**

$$E[\text{terhelés}_R] = \sum_{|F|} (\text{gyakoriság}_F * E[\text{terhelés}_{F,R}])$$

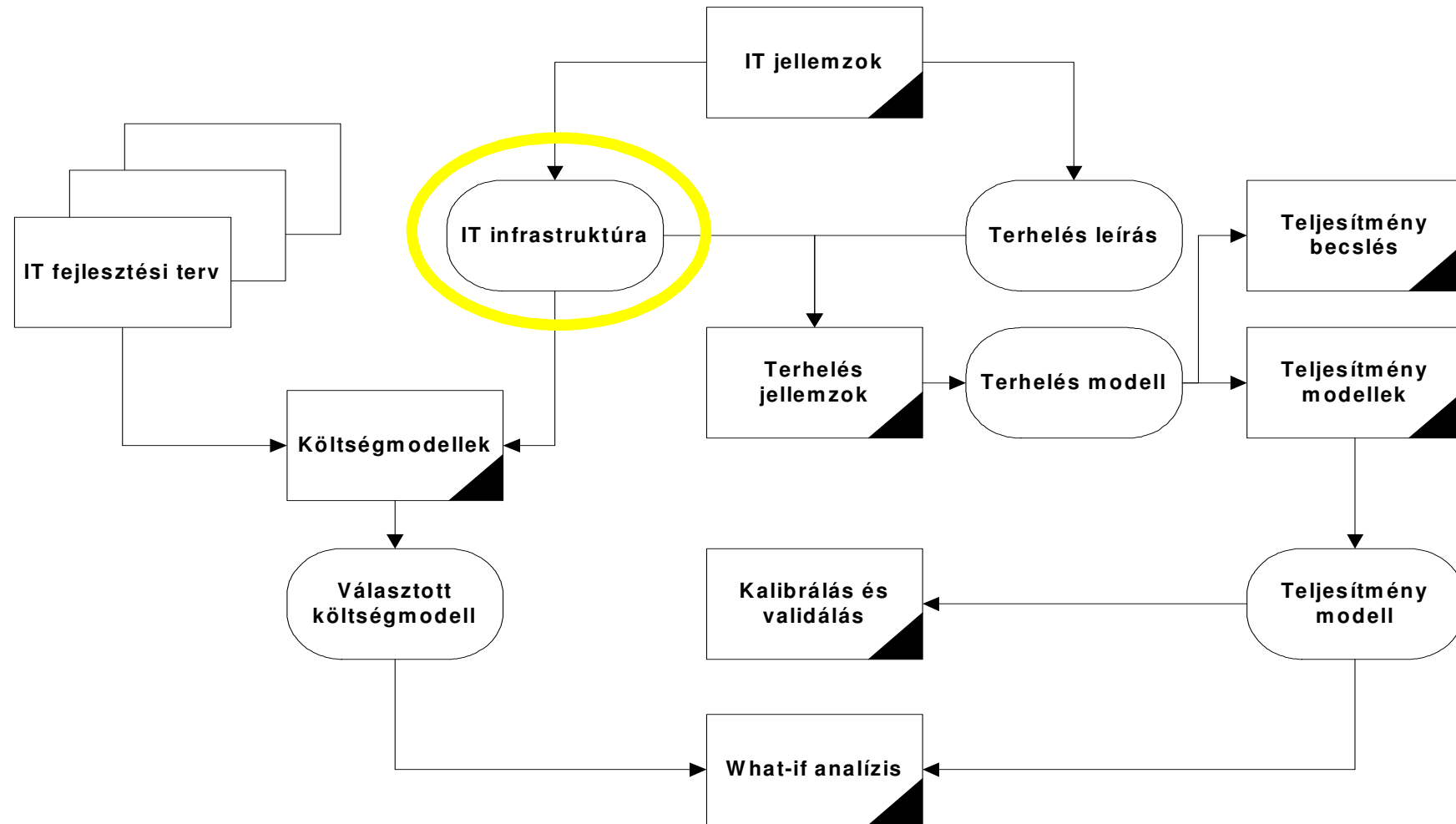
**közgazdasági  
(nem inf.) mérték**

... csal, mert

- nem additív a terhelés
  - Taszkváltás, cache frissítés, stb.
- lehetnek kiugró erőforrásigények

**informatikai  
(műszaki) mérték**

# Infrastruktúra leírása

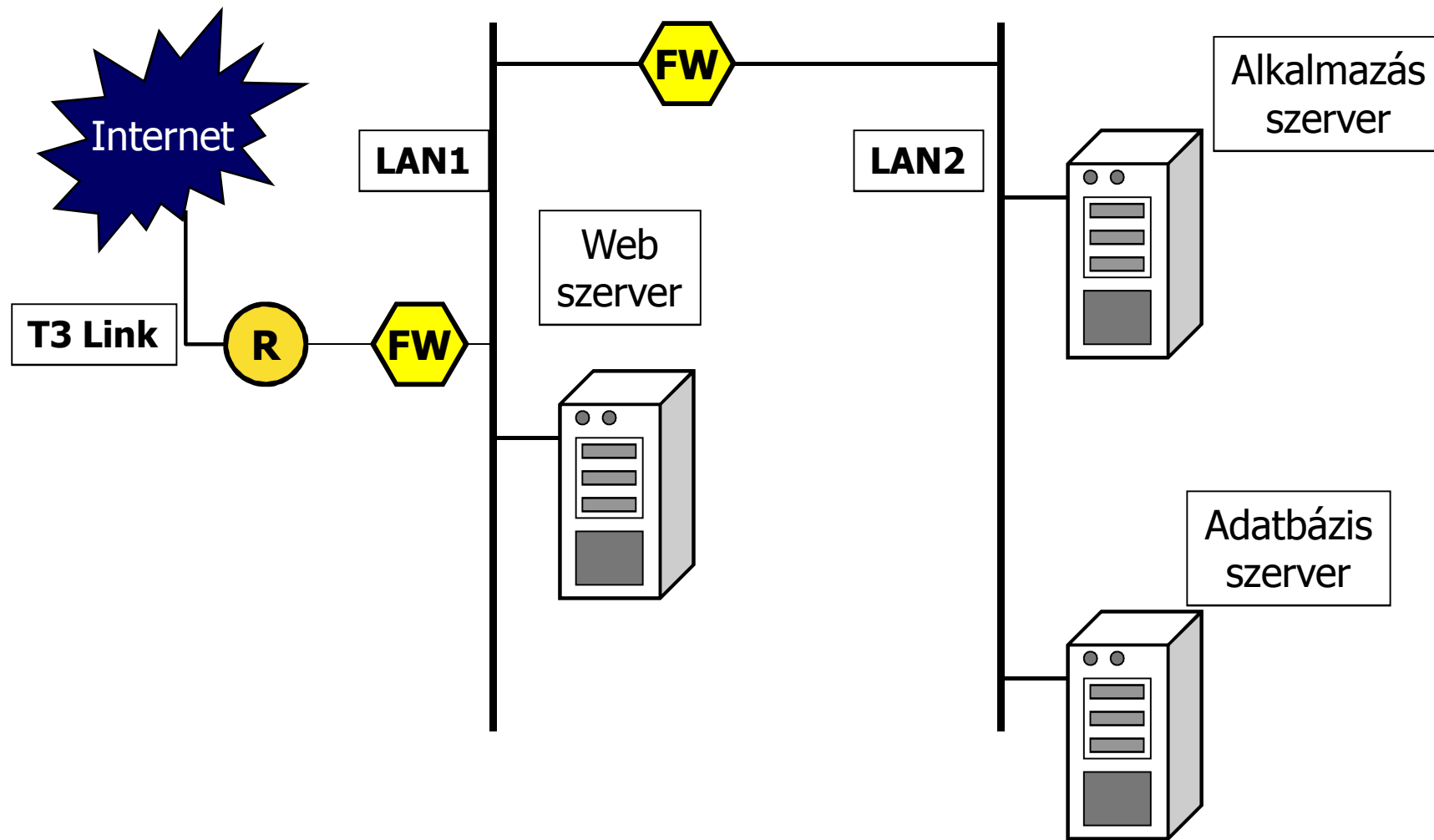


# Informatikai környezet meghatározása

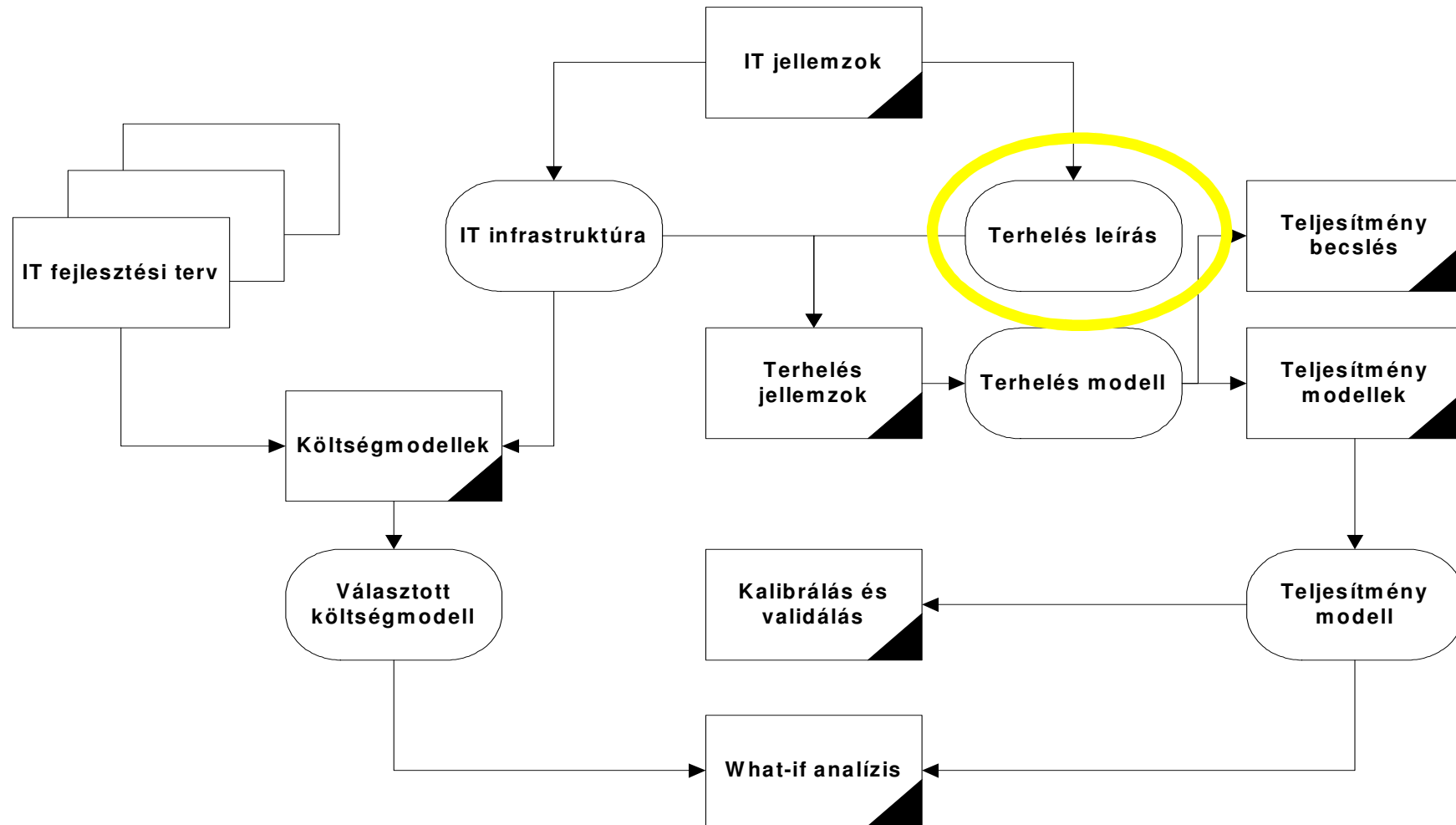
- Cél: meghatározni a funkciókhoz tartozó tranzakciókat (elemi lépéseket) és az ezek által használt erőforrásokat

E-Business szolgáltatás	Tranzakciók	Szerver
Könyv kiválasztás	ShowBooks	WS
	DisplayBooks	AS
	SearchBooks	DB
	LaunchShowBooks	WS
	DisplayBooksByAuthor	AS
	SearchBooksByAuthor	DB
	SendReply	WS
Option Selection	LaunchShowOptions	WS
	DisplayBookOptions	AS
	SearchBookOptions	DB
	SendReply	AS

# Példa: a kereskedés informatikai infrastruktúrája



# Terhelés leírása

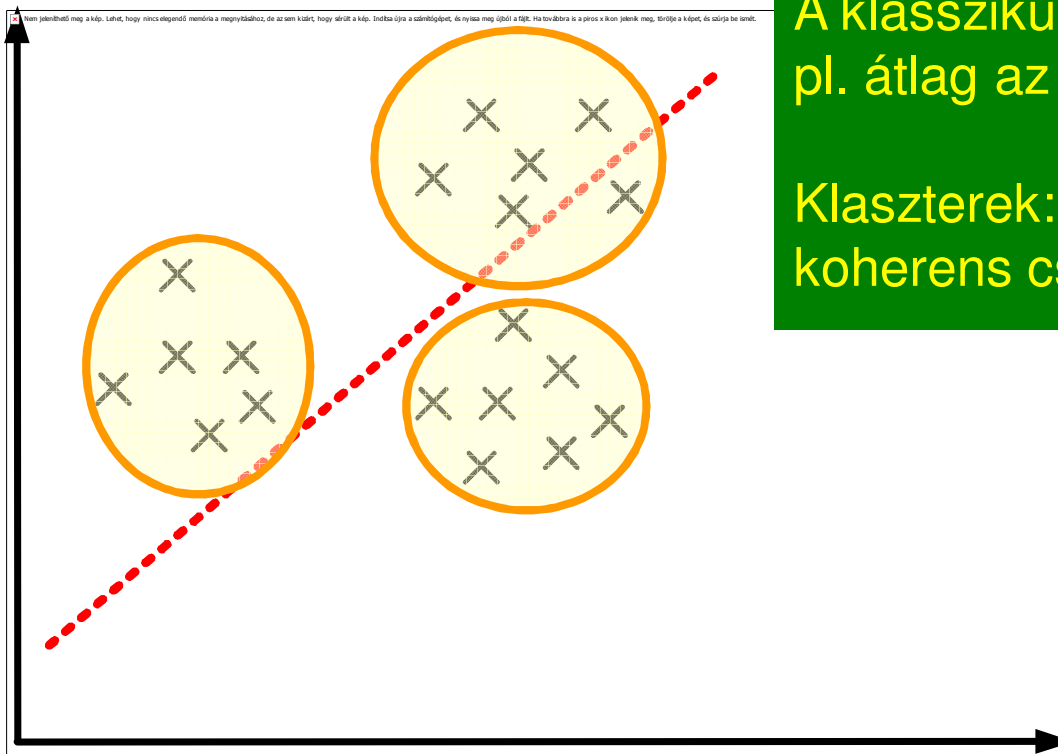




# Terhelés leírása

- Az előző lépésben meghatározott modellt veszi alapul
  - melyik szolgáltatás milyen erőforrásokat használ
- Alapvető komponensek: tranzakciók
- CSISD (szekvenciák) felírása, minden szerveroldali objektumhoz tranzakciónév
- Komponensekhez gyakoriság (érkezési ráta) és erőforrásigények meghatározása
- Cél: Szolgáltatások erőforrásigényeinek meghatározása

# Klaszter technikák



A klasszikus statisztikai jellemzők, pl. átlag az egész mintát jellemzik.

Klaszterek:  
koherens csoportok

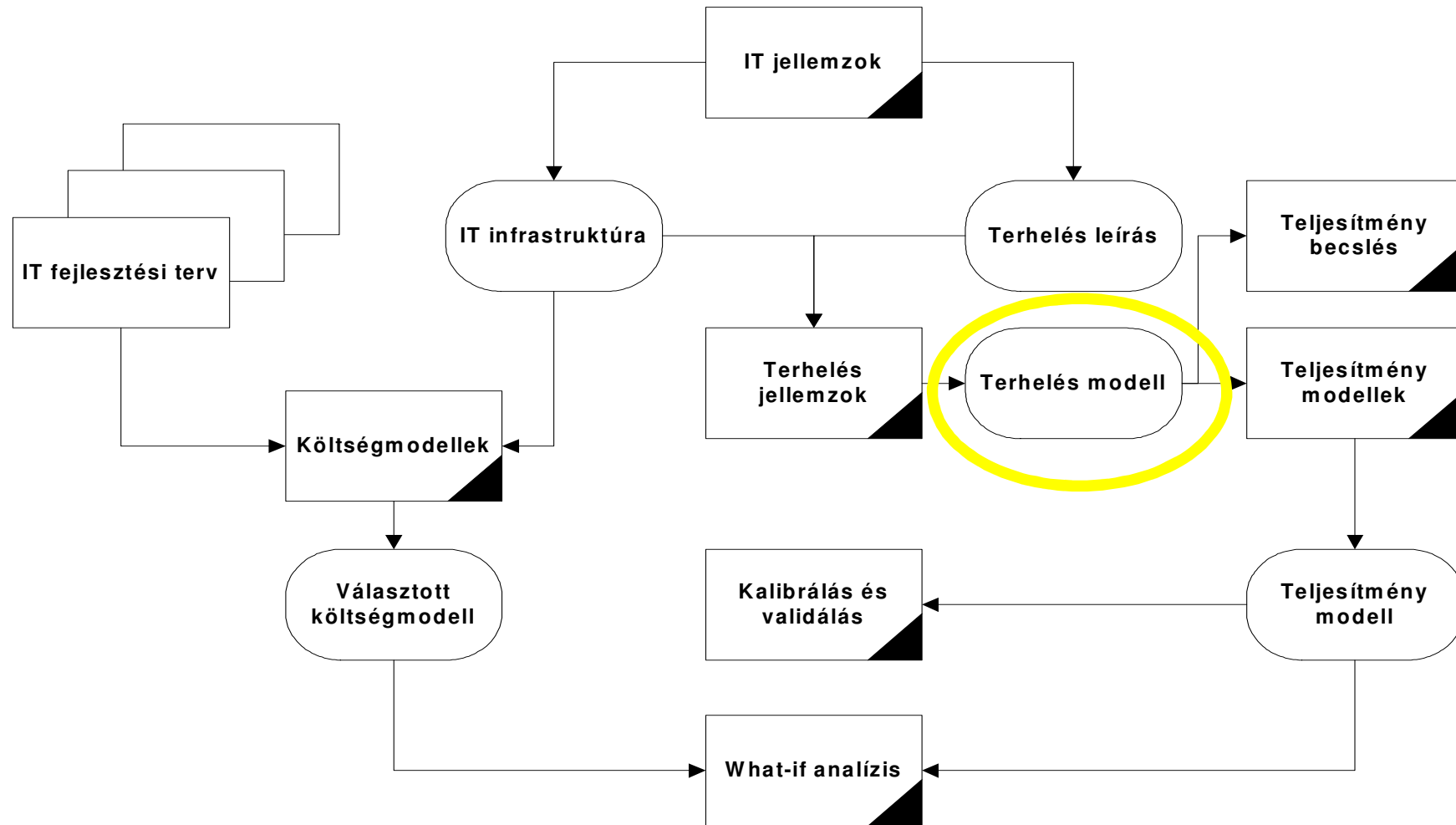
# Klaszter alapfogalmak

- Klaszter:
  - összetartozó, hasonló értékek csoportja
- Klaszter középpontja: centroid
  - ezzel helyettesíthető az adott csoport
- Különböző (n dimenziós) távolságfogalmak
  - nem csak euklideszi
- Különböző algoritmusok
- Algoritmus + távolság definíció
  - klaszter technika

# A komponensek jellemzése

<b>Komponensek és paraméterek</b>	<b>Paraméter típusa</b>
<i>"Eladás" tranzakció</i>	
Egy kliensre jutó tranzakciók száma	Intenzitás (I)
Kliensek száma	I
Adatbázis I/O műveletek száma	Szolgáltatás igény
CPU használat az adatbázis szerveren	Sz
Átlagos üzenetméret	Sz
<i>Multimédiás bemutató</i>	
Átl. Session szám/nap	I
Video fájlok átl. mérete	Sz
HTTP oldalak átl. mérete	Sz
Átl. letöltött képek / session	Sz
CPU használat a web szerveren	Sz

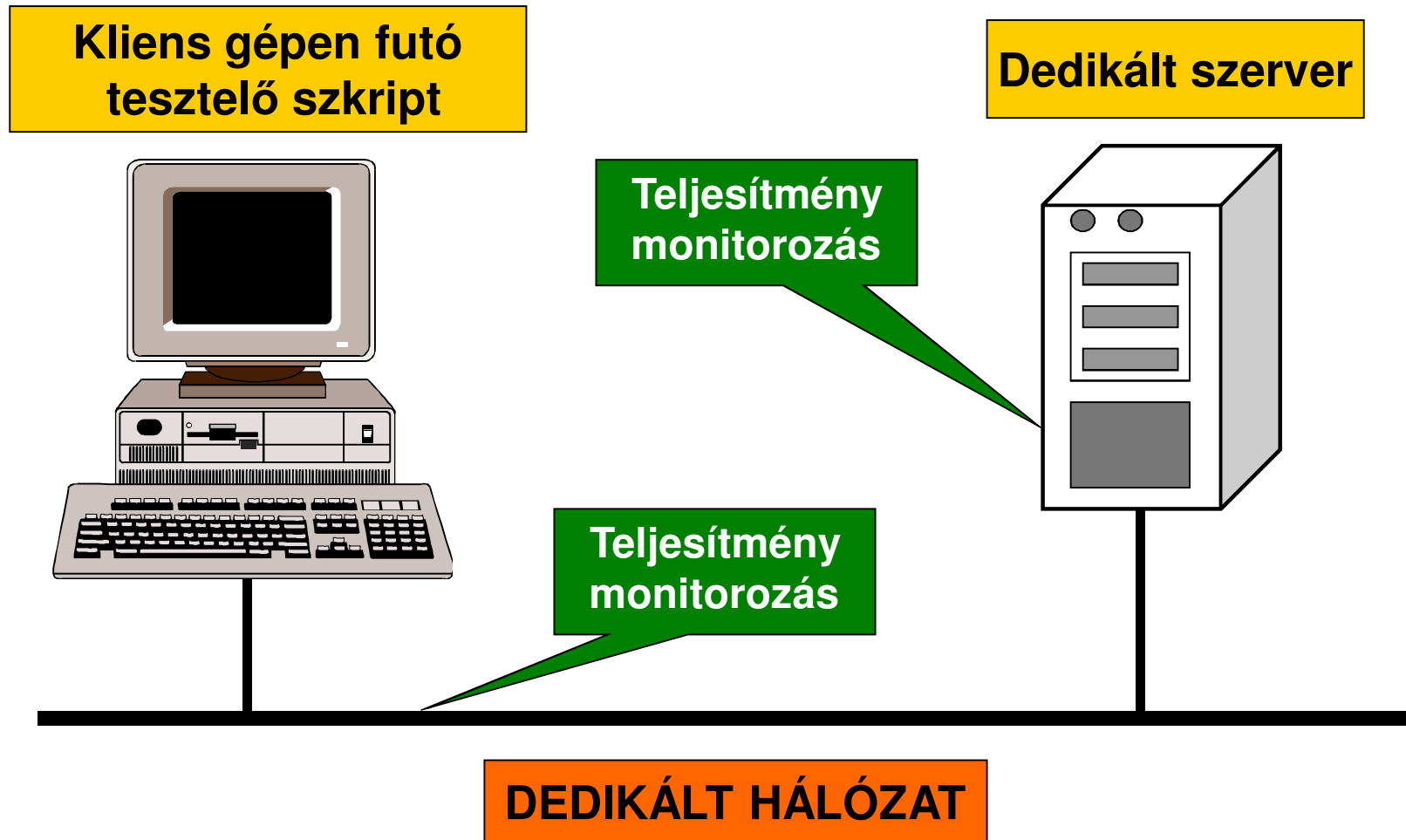
# Terhelés modell



# Terhelés modell meghatározása

- Adatgyűjtés
  - benchmarkok (Terhelés ea.)
  - ökölszabályok
  - „best practices”
  - mérések
- Monitorozás
  - belső (szerver)
  - külső (kliens, hálózat)
- Adatok rendszerezése
  - klaszter technikák

# Terhelés mérése ellenőrzött környezetben



# Terhelés benchmark példa

- Standard Performance Evaluation Corp.
- Alkalmazás szerver SPEC CINT2000=431
- Egy szolgáltatás CPU igénye 10 ms
- Új szerver, SPEC CINT2000=518
- Új CPU igény:  
 $10 / (518/431) = 8.3 \text{ ms}$
- Fontos: jó benchmarkot válasszunk!
- Pl. lebegőpontos számítás esetén  
SPEC CFP2000 kéne

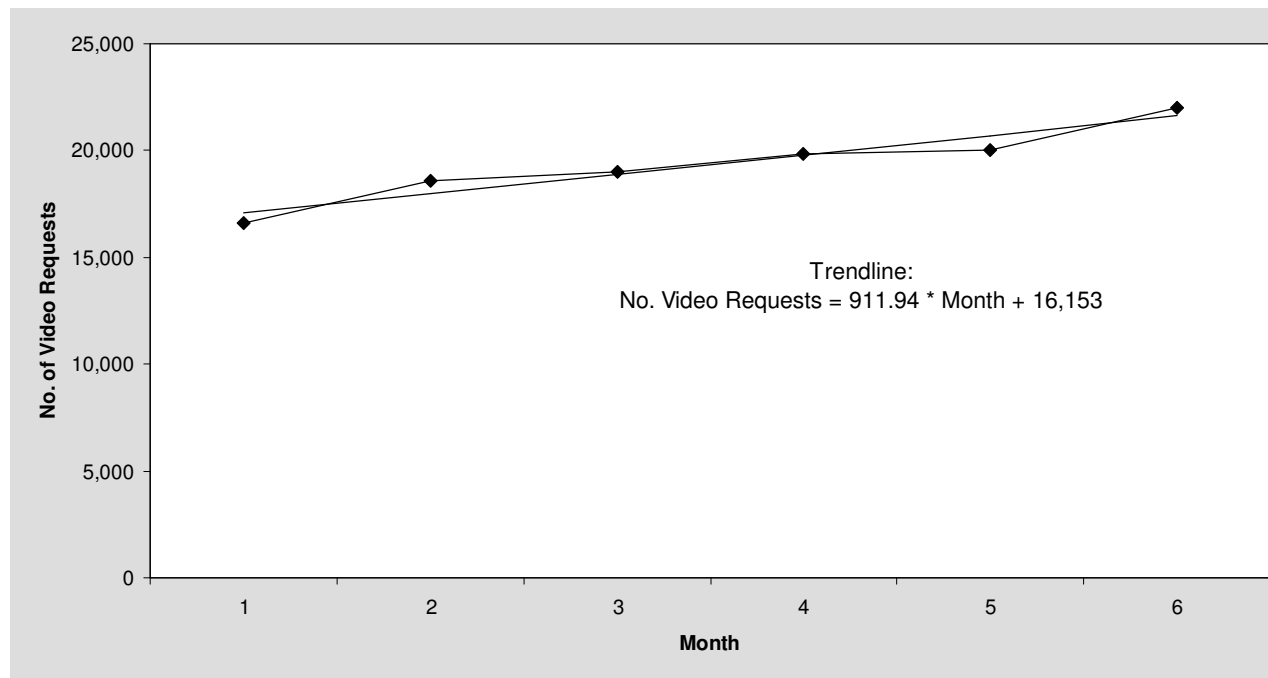


# Terhelés modell minta

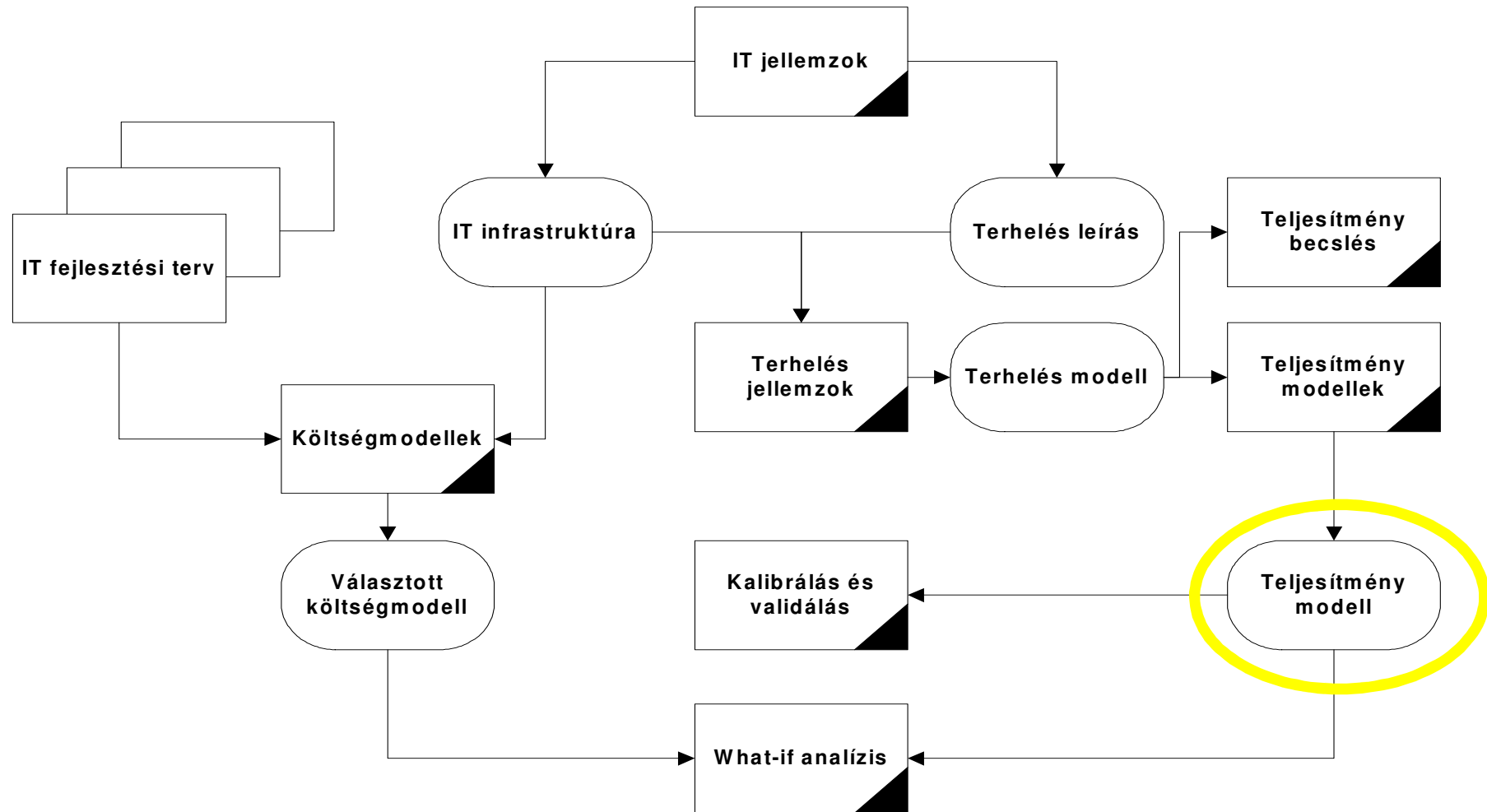
1 napi mérés adatai											
Érkezési ráta	20,000	kérés/óra	LAN 1	100	Mbps	LAN 2	100	Mbps	T3 Link	1.5	Mbps
Egy kérés szolgáltatás igénye											
E-business Funkció	Kérések száma	WS CPU (ms)	WS IO (ms)	AS CPU (ms)	AS IO (ms)	DB CPU (ms)	DB IO (ms)	LAN 1 (ms)	LAN 2 (ms)	T3 Link (ms)	Ossz. szolg. igény
általános info	20,000	5.2	9.5	25.0	15.0	10.0	20.0	0.492	0.532	16.4	132.1
könyv keresés	18,900	4.8	8.5	18.0	14.0	13.0	40.0	0.328	0.352	12.0	91.0
könyv böngészés	14,120	4.9	8.2	13.0	12.0	13.0	40.0	0.287	0.328	12.0	83.7
bejelentkezés	8,020	5.1	8.4	12.0	10.0	13.0	20.0	0.295	0.492	11.5	80.8
regisztráció	892	32.0	15.0	16.0	30.0	15.0	20.0	0.655	0.000	32.8	126.4
bevásárlókosár	670	32.0	14.0	18.0	24.0	0.0	0.0	0.410	0.000	19.1	107.5
fizetés	584	31.0	15.0	35.0	90.0	30.0	80.0	0.819	0.901	43.7	326.4

# Terhelés előrejelzés

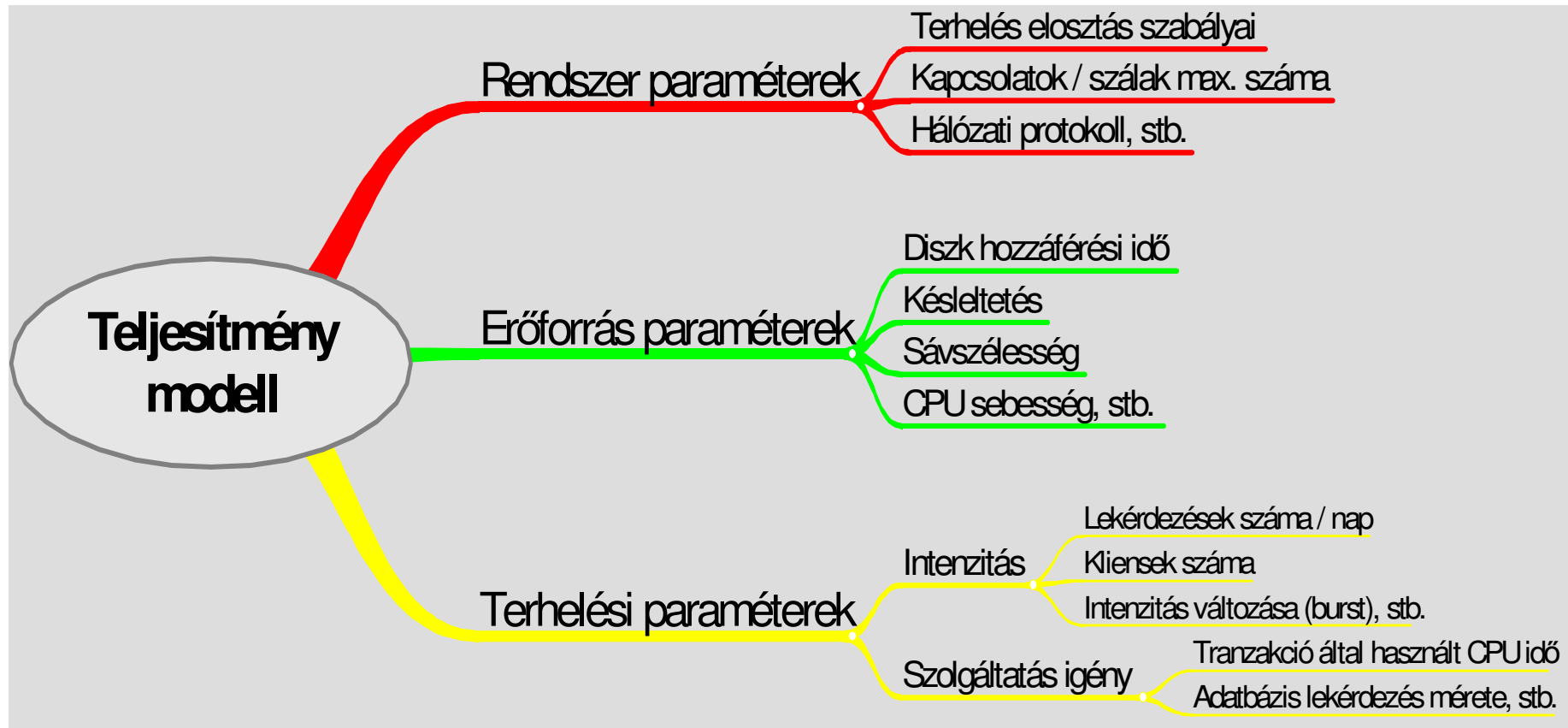
- A terhelési modell várható változása
- Többféle technika létezik („Terhelés” ea.)
- Példa: lineáris regresszió



# Teljesítmény modell



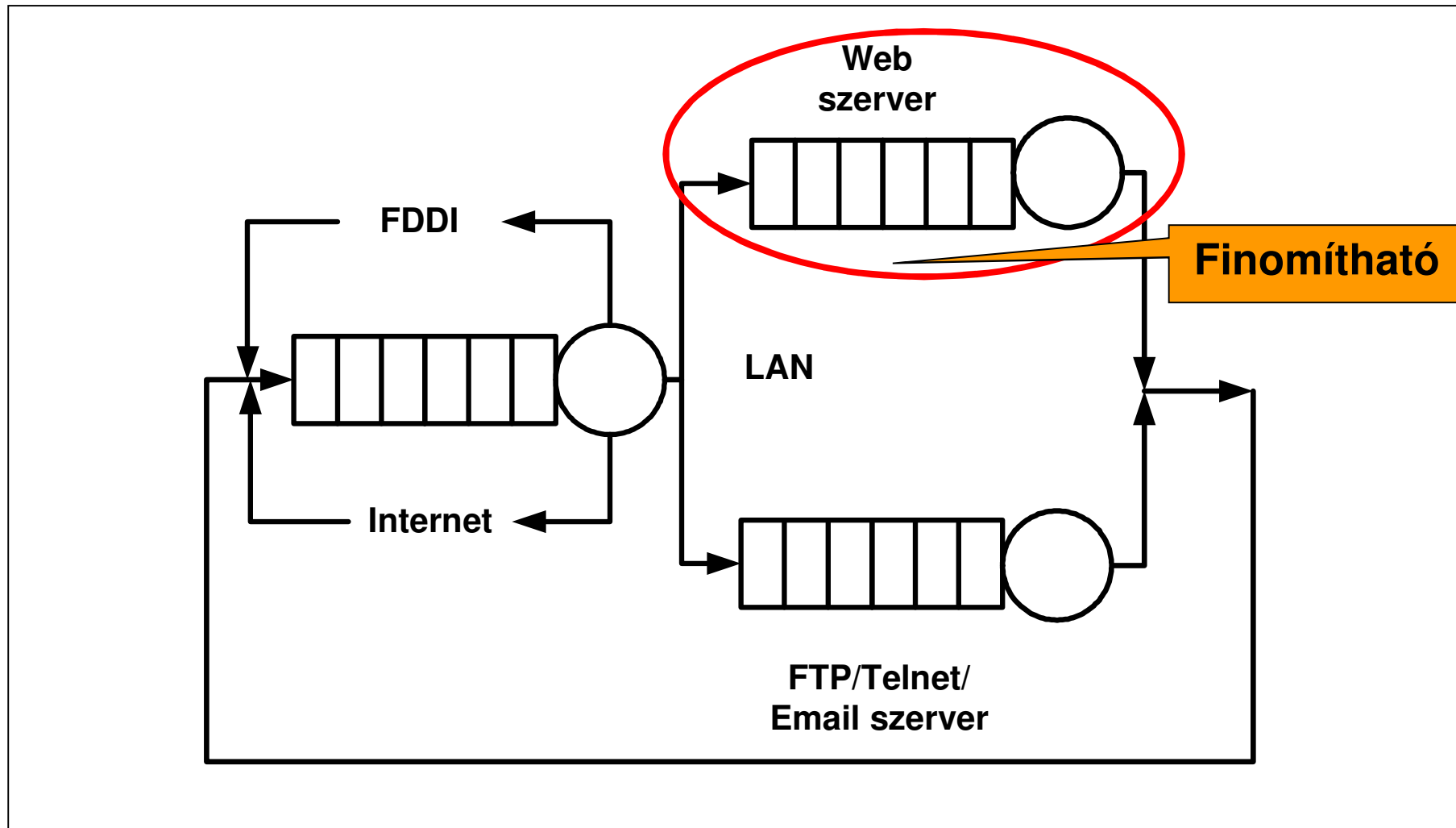
# Teljesítmény modell paramétere



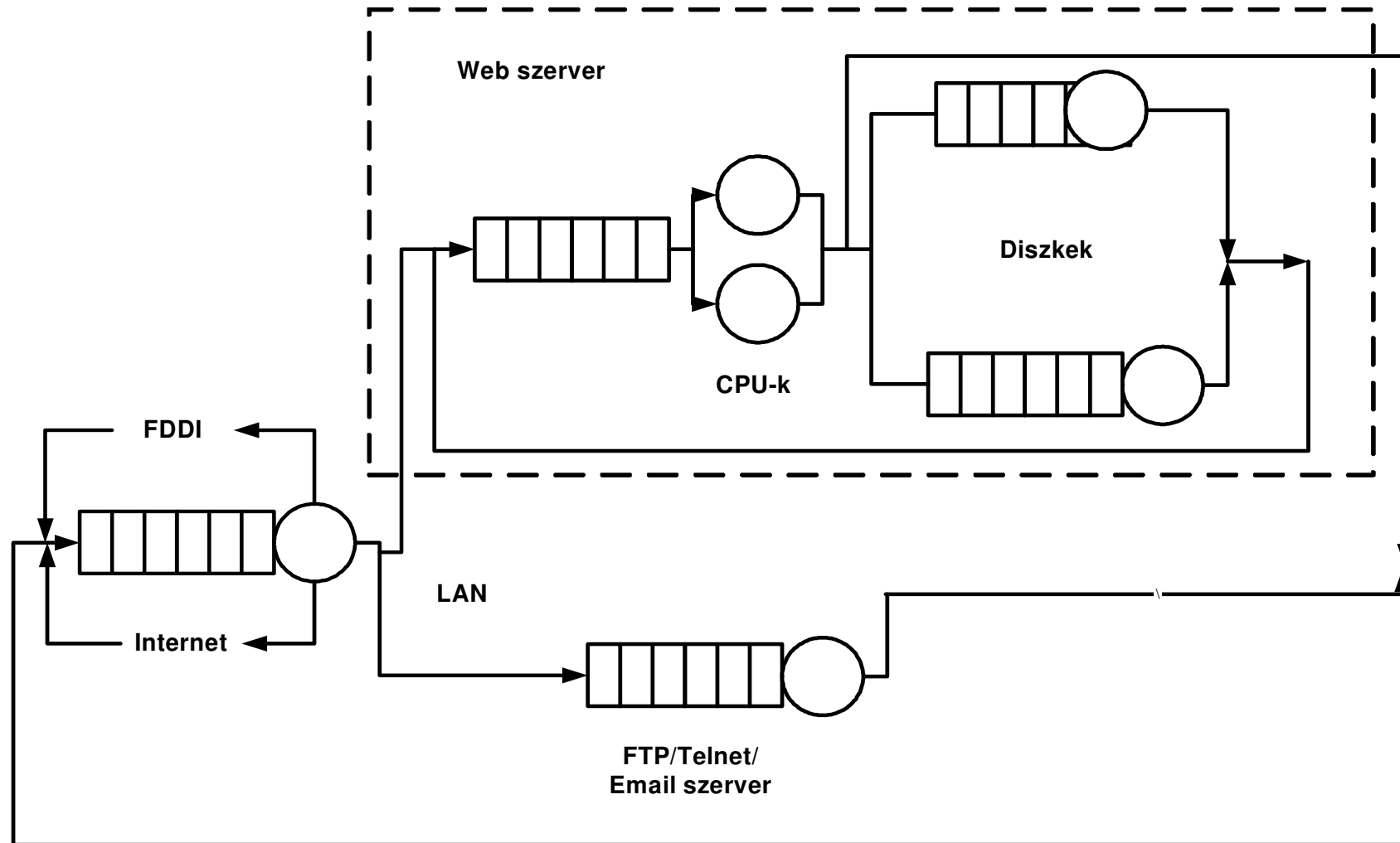
# Queuing Network modellezés

- Erőforrások
- Várakozási sorok
- Összeköttetések
- Hierarchikus (finomítható)
- Részletesebben „Teljesítmény” ea.

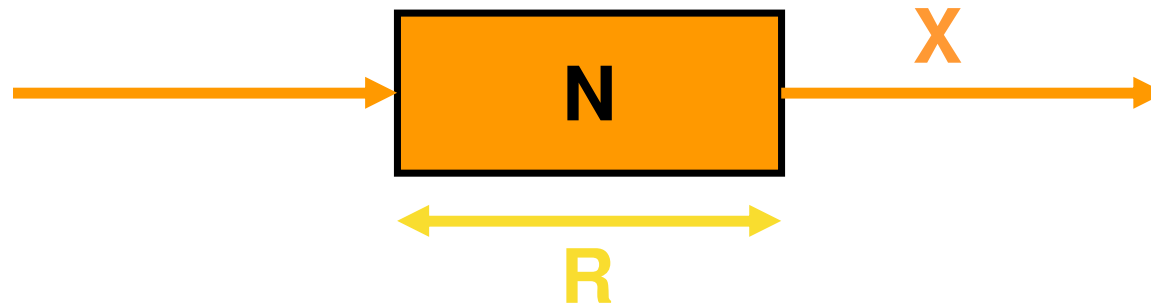
# Queuing Network példa



# Queuing Network példa 2.



# Little törvénye

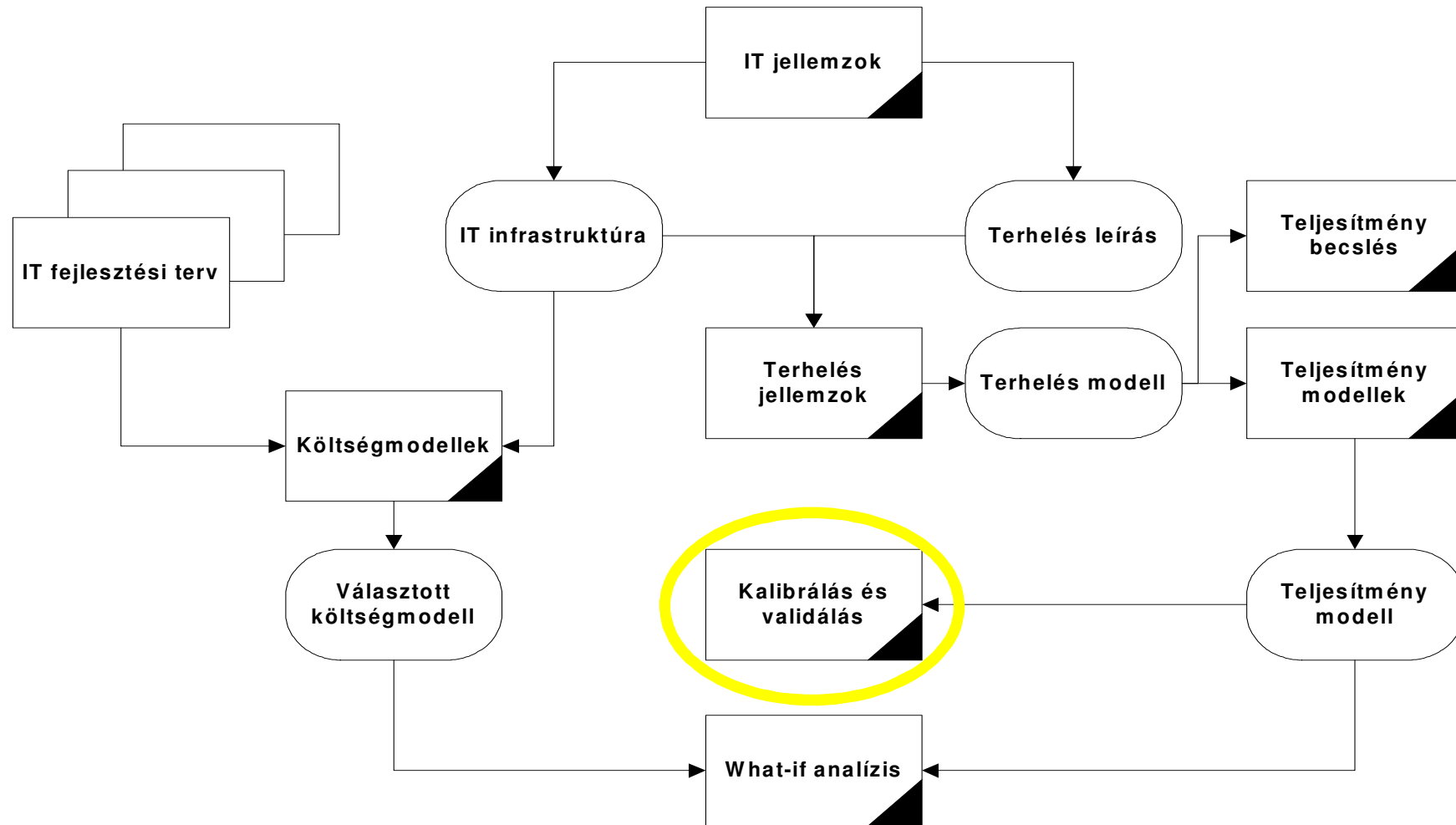


- $N$ : a rendszerben lévő kérések átlagos száma
- $X$ : átteresztőképesség
- $R$ : a rendszerben töltött átlagos idő

$$N = X \times R$$



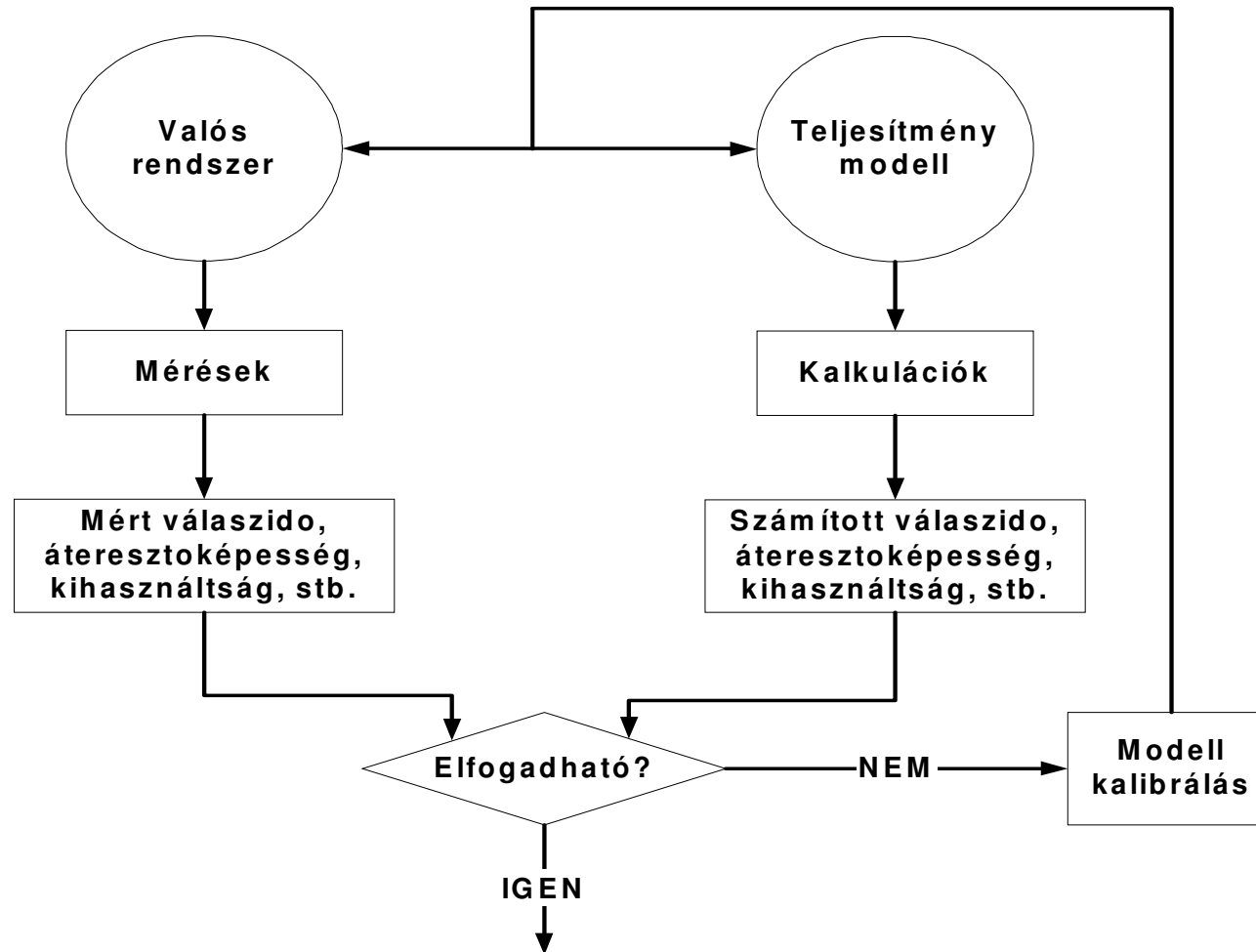
# Kalibrálás és validálás



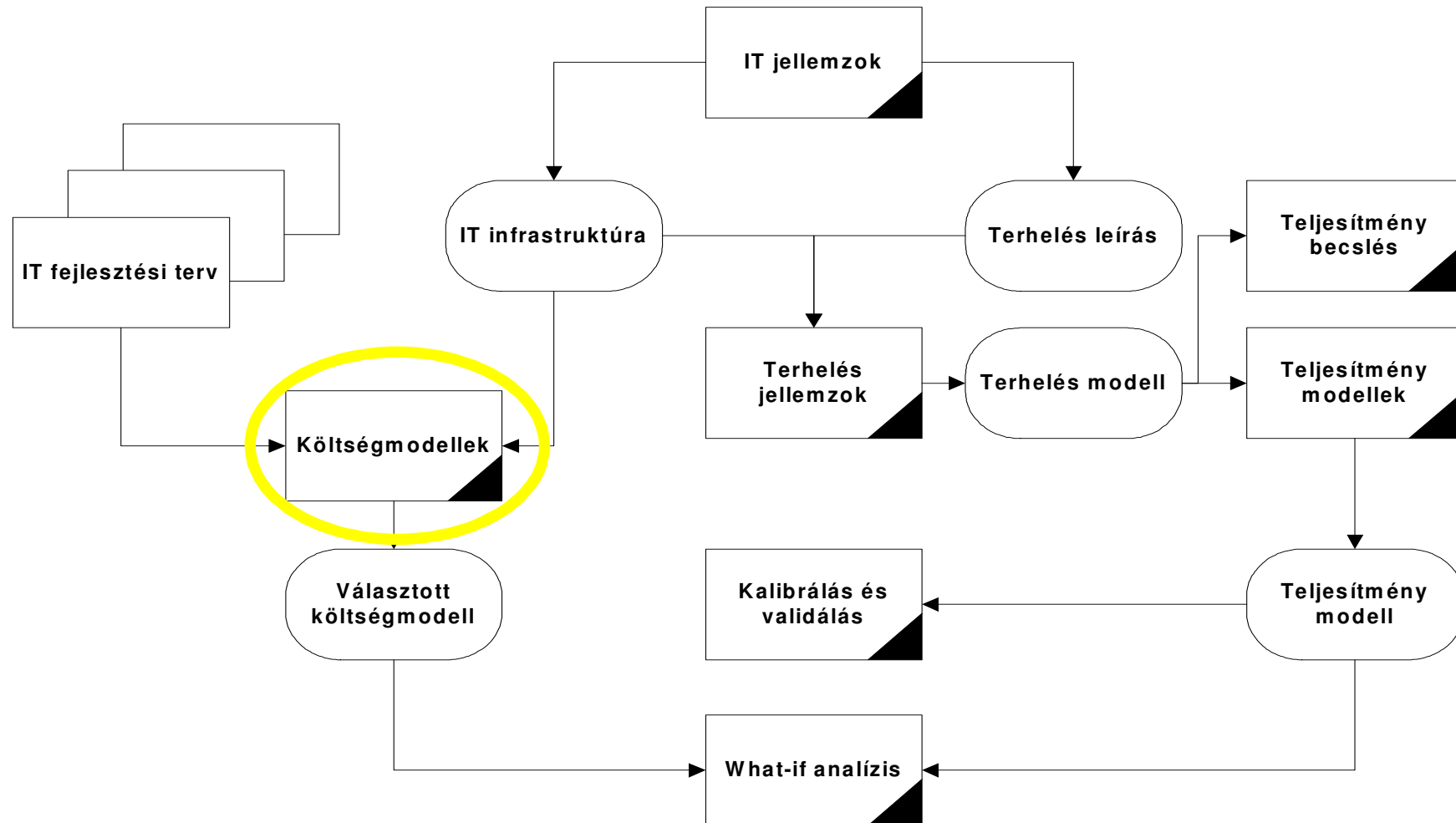
# Kalibrálás és validálás

- Modell torzítja a valóságot → ellenőrzés
- A terhelés modellt és az aktuálisan mért értéket összevetjük
- Ha elfogadható (10%, esetleg 30% hiba)
  - Validált modell
- Ha túl nagy a hiba
  - A modellt újra kell kalibrálni

# Kalibrálás és validálás folyamata



# Költség modellezés



# Informatikai költségek

## KÖLTSÉGEK

### Hardver

- diszk alrendszerek
- hálózati eszközök
- szünetmentes táp, kábelek
- tartalék eszközök, stb.

### Szoftver

- OS
- DBMS
- Middleware
- Alkalmazások
- Vírusvédelem, biztonság
- Fejlesztési költségek, stb.

### Telekommunikáció

- WAN
- ISP, stb.

### Beszállító költségei

- Web hosting
- Hirdetési szolgáltatások
- Fizetési (payment) szolg., stb.

### Fenntartási költségek

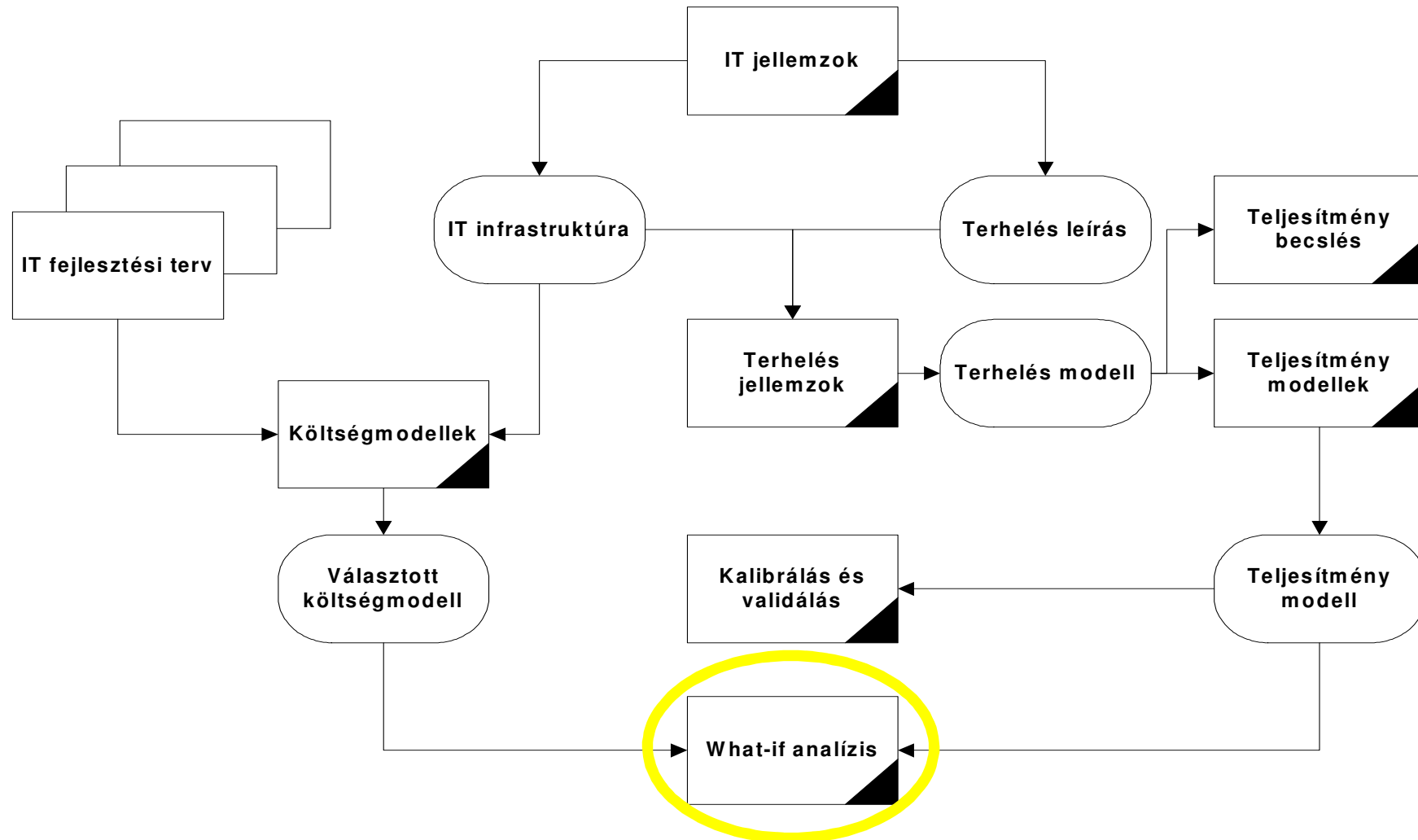
- Hálózati adminisztráció
- Support
- Betanítás, stb.

Részletesebben: ld. „Infrastruktúra” ea.

# Ökölszabályok („tradicionalis”)

1. A memória tárhely minden 3 évben megnégyszereződik (Moore)
2. A processzor sebessége minden 3 évben megduplázódik (Moore)
3. A háttértár kapacitása 10 évenként megszázsorozódik
4. A hálózati sávszélesség minden 3 évben megnégyszereződik (Gilder)
5. A RAM MB/MIPS arány 1-ről 4-re növekszik (Amdahl)

# What-if analízis



# What-if analízis

- Mi történik, ha berakunk egy új Web szervert?
  - teljesítmény nő 😊
  - költség is ☹️
- Megéri-e gyorsabb hálózatra váltani?
- Mi történik, ha replikáljuk az adatbázis szervert (eddig 1 helyett 2) ?
- ...
- A hatást visszacsatoljuk az üzleti modellbe