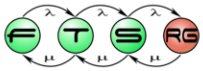
Intelligens rendszerfelügyelet

Rendszermonitorozás

Tóth Dániel, Kocsis Imre



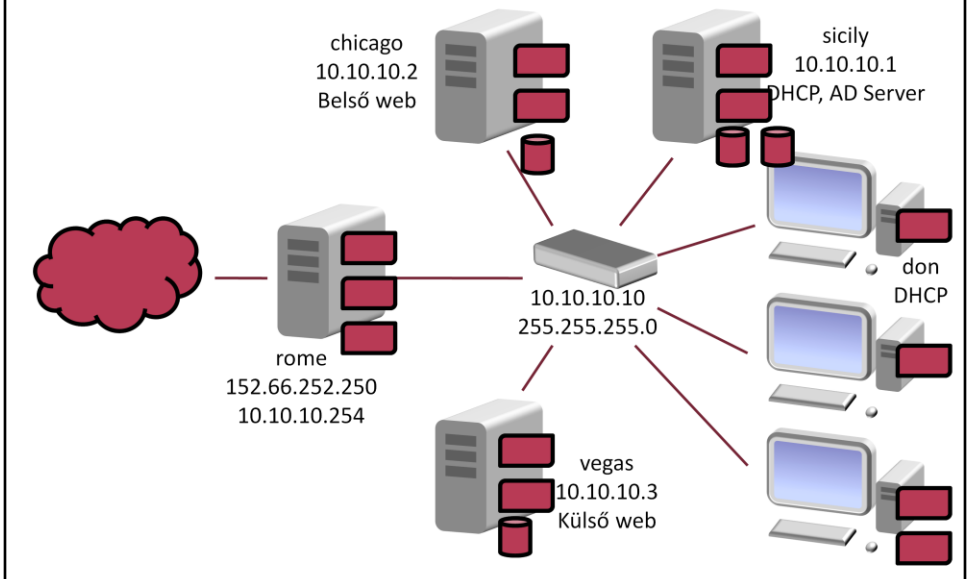
Budapesti Műszaki és Gazdaságtudományi Egyetem
Méréstechnika és Információs Rendszerek Tanszék

Utolsó módosítás: 2012. 04. 17.

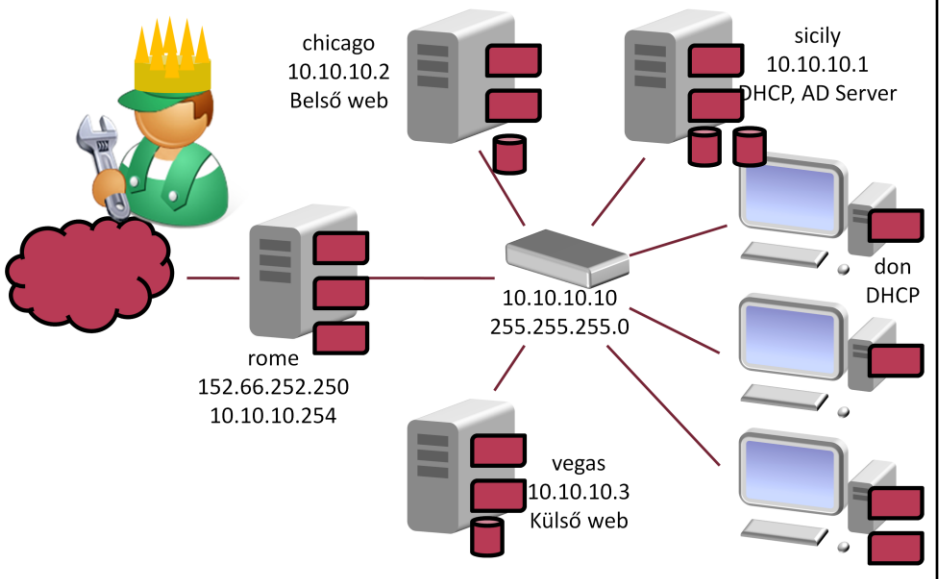
„When you can measure what you are speaking about, and express it in numbers, you know something about it; but when you cannot measure it, when you cannot express it in numbers, your knowledge of it is of a meager and unsatisfactory kind”

Lord Kelvin

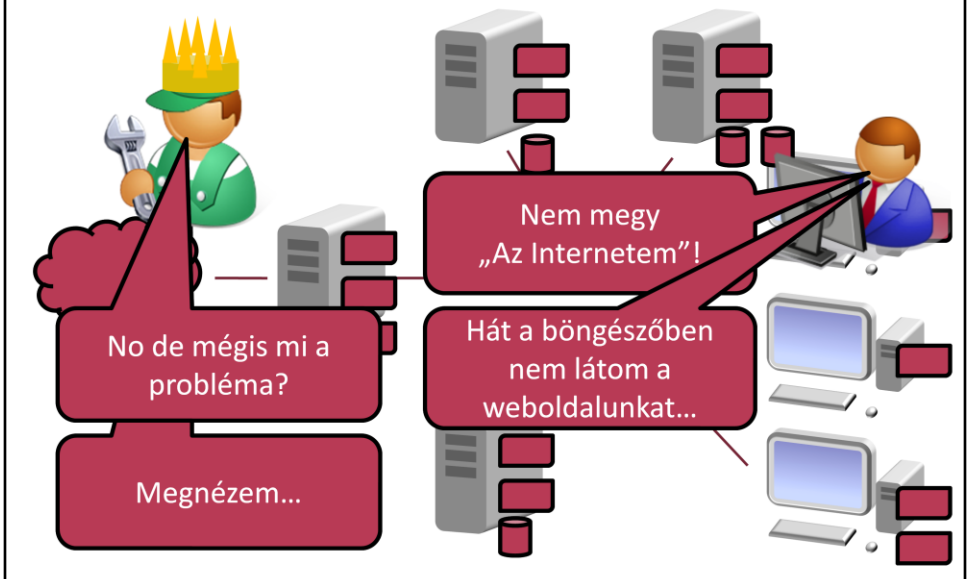
„Kézbentartott” rendszer



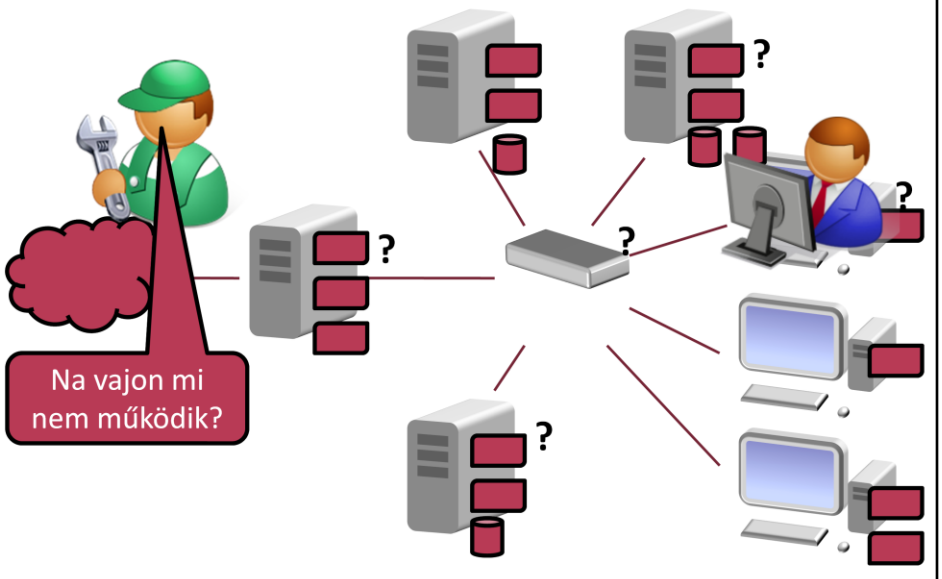
„Kézbentartott” rendszer



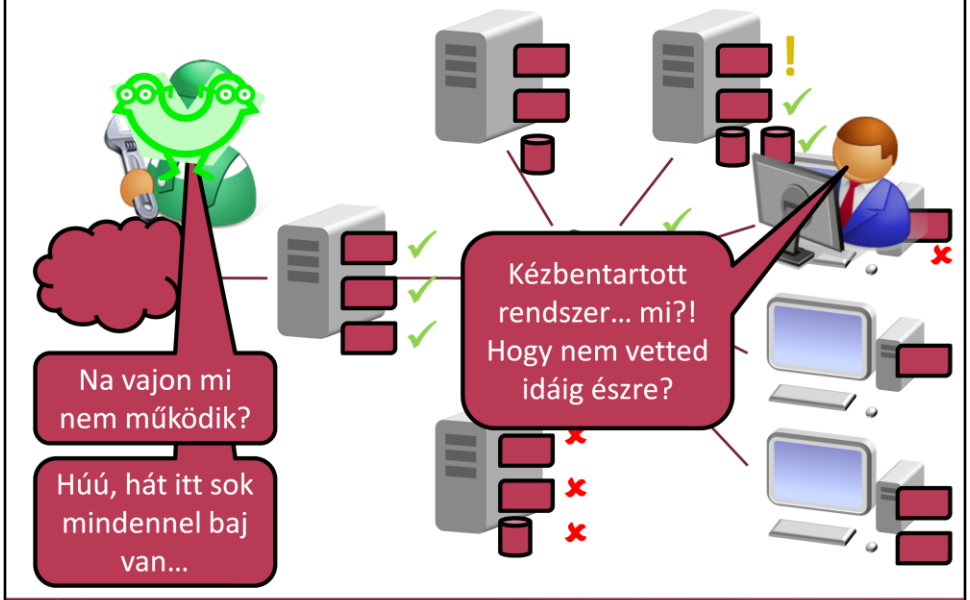
Káosz


Nem megy „Az Internetem”!
No de mégis mi a probléma?
Hát a böngészőben nem látom a weboldalunkat…
Megnézem…

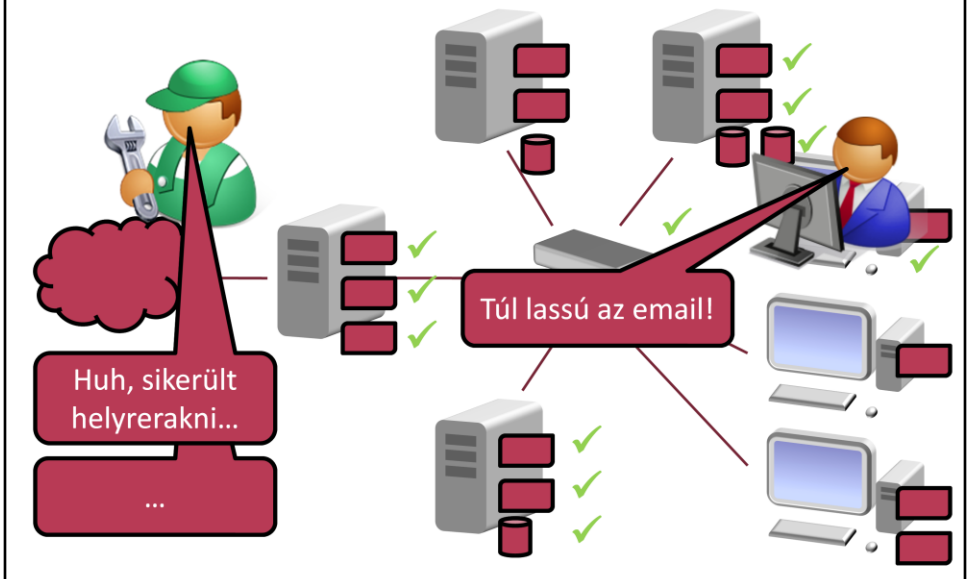
Káosz



Káosz


Na vajon mi nem működik?
Húú, hát itt sok mindennel baj van…
Kézbentartott rendszer… mi?! Hogy nem vetted idáig észre?

Káosz



Rendszermonitorozás

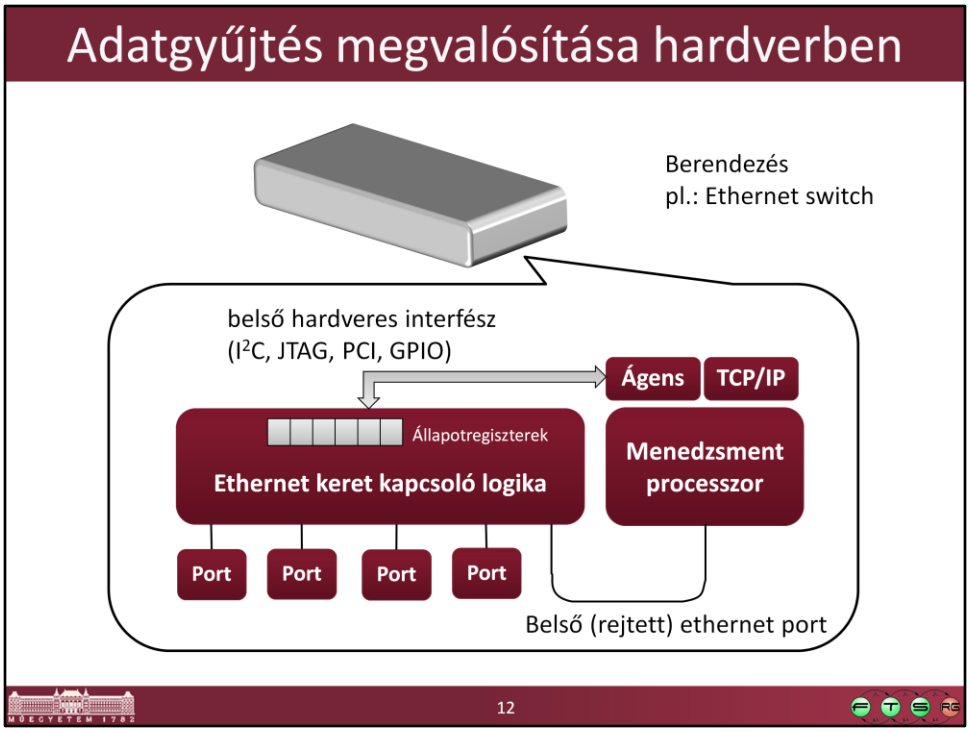
- A rendszer túl bonyolult
  - Ember nem látja át a teljes működését
  - Valami mindig történik benne…
  - Csak akkor értesülünk róla, ha a felhasználók nyaggatnak, hogy valami nem megy
    - ($$$!)
  - Csak akkor vesszük észre, hogy baj van, ha már tényleg nagy baj van (jó lett volna előbb preventív jelleggel)
  - A rendszer teljesítményéről, kihasználtságáról nincs elképzelésünk
    - Pedig ilyen adatok nélkül nehéz tervezni…

Rendszermonitorozás: állapotkép fenntartása

- Infrastrukturális komponensek és szolgáltatások működőképességéről
- Terhelésről, erőforrások kihasználtságáról
- Topológiáról, konfigurációról
  - Kapcsolat a konfiguráció-menedzsmenttel!
- Biztonságról

Adatgyűjtés megvalósítása

- Jellegzetes követelmény:
  - A rendszerünk nagy, sok különálló elemből áll
  - Az adatokat hálózaton keresztül olvassuk le
- A kulcselem az *ágens*
  - Kis beépülő komponens minden berendezésbe, aminek célja:
    - adatszolgáltatás valamilyen (hálózati) interfészen
    - értesítés különféle események bekövetkezéséről
    - egyszerű beavatkozások elvégzése

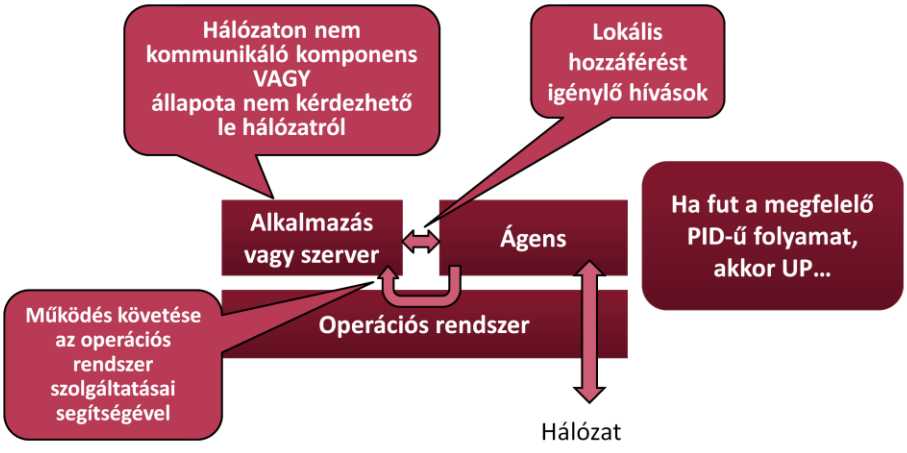


•Az Ethernet kapcsoló (switch) csak az Ethernet protokoll rétegében működik, nem ismeri a TCP/IP-t
•Hardver állapota belső állapotregiszterekből olvasható ki
•Lehetnek parancsregiszterek is beavatkozásra
•Mindez közvetlen elektromos kapcsolatot igényel, nem vezethető ki a készülékből, vagy legalábbis nagyon kényelmetlen lenne
•Megoldás: helyezzünk el egy kis beágyazott processzort a dobozba, ami közvetlenül össze van kötve a switch hardverrel
•A beágyazott processzoron futó szoftver támogatja TCP/IP protokollkészletet és tartalmazza az ágenst, aminek segítségével a hálózatról lekérdezhetjük a hardver állapotát

Adatgyűjtés megvalósítása szoftverben I.

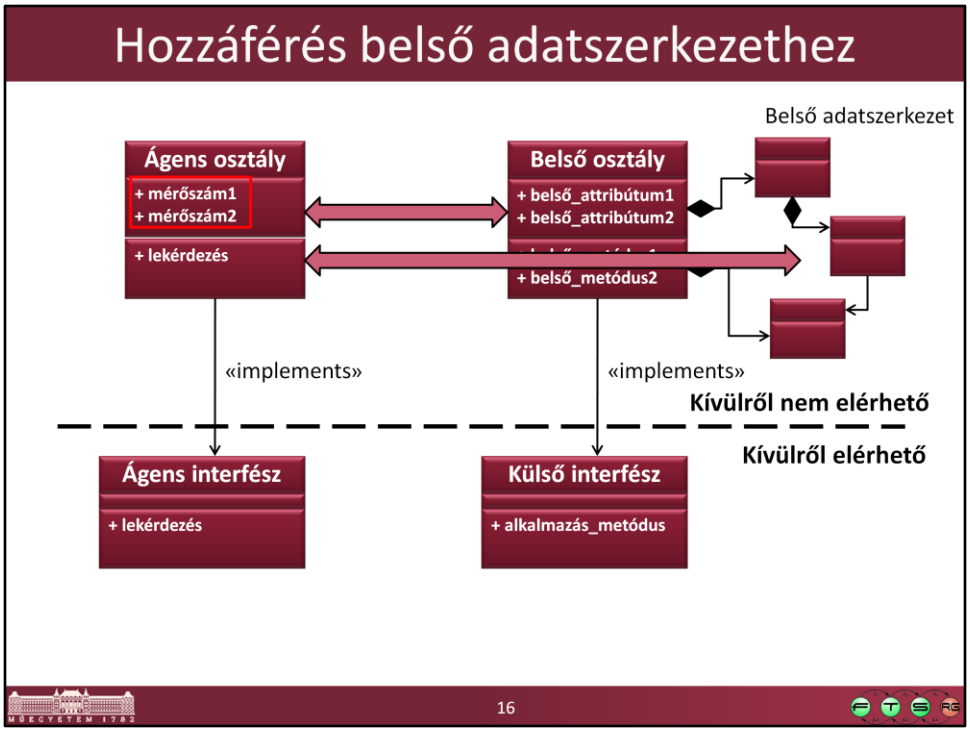
- Jellemző alapesetek:
  - **Olyan szoftver komponenst akarunk megfigyelni, ami nincs erre felkészítve**
    - Az ágens külön folyamat az operációs rendszeren
    - Olyan hívásokat végezhet el, ami csak egy gépen futó folyamatok között lehetséges (de a belső adatszerkezetekhez többnyire nem férünk hozzá)
    - Az operációs rendszer segítségével követi a megfigyelt folyamatot (futási állapot, létrehozott fájlok tartalma, erőforráshasználat, stb.)
  - Az ágens integrált része a szoftvernek

Adatgyűjtés megvalósítása szoftverben I.



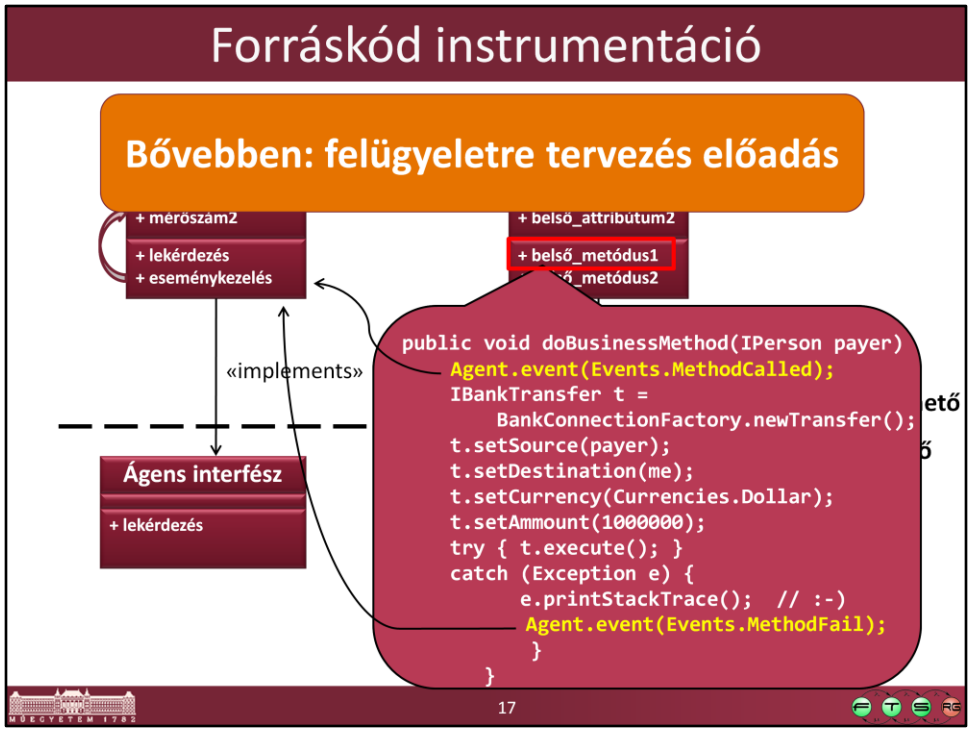
Adatgyűjtés megvalósítása szoftverben II.

- Jellemző alapesetek:
  - Olyan szoftver komponenst akarunk megfigyelni, ami nincs erre felkészítve
  - **Az ágens integrált része a szoftvernek**
    - Hozzáférünk a belső adatszerkezetekhez
    - Közvetlenül végezhetünk függvényhívásokat
    - Forráskód *instrumentálás* (mérő, adatgyűjtő hívások elhelyezése a forráskódban) lehetséges
    - A lényeg: a belső mérési lehetőségeket kívülről is elérhetővé kell tenni



(Demonstrációs ábra, ékezet és szóköz valódi modellben ne legyen osztály- és attribútumnévben!)

Forráskód instrumentáció



Ágens lekérdezési interfész

- Hogyan kérdezzük le az ágenstől a mért adatokat?
- Jó lenne…
  - hálózaton keresztül
  - szabványos interfész, protokoll
  - Egységesen: gyártók, készülékek, szoftver/hardver
    - Adatok széles skálájának támogatása
  - ha azt is le tudnánk kérdezni, hogy pontosan miket lehet lekérdezni az ágenstől

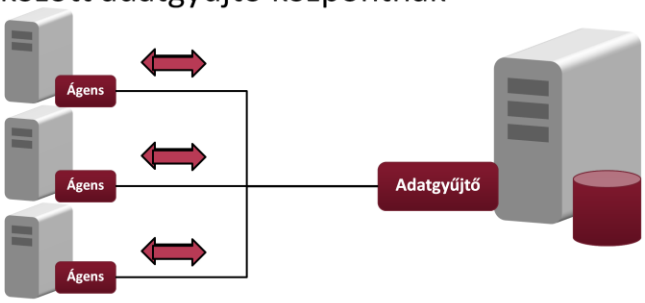
Konfigurációmenedzsment: hasonlóság!

Jellegzetes alapfunkciók

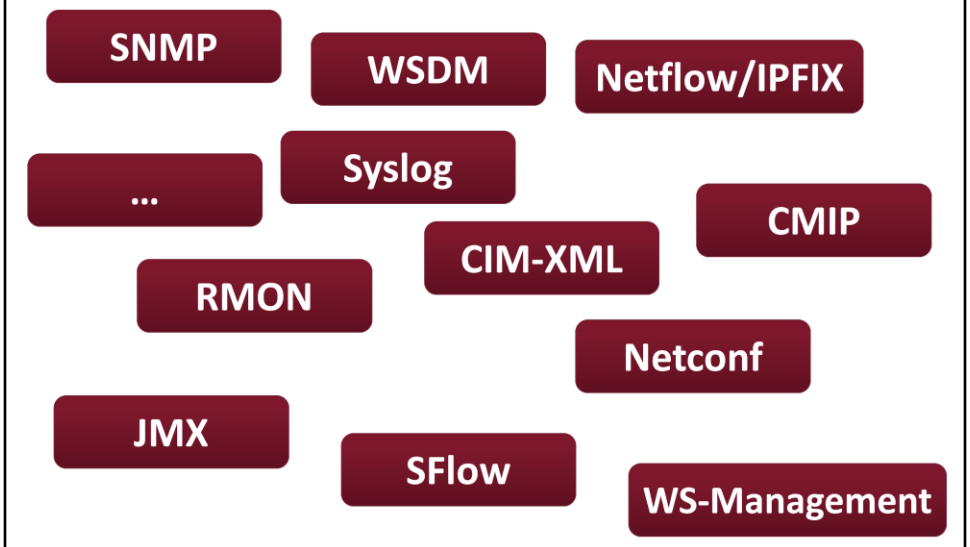
- Pillanatnyi értékek
  - Skalár mennyiség: CPU kihasználtság, RAM, tárhely telitettség, …
  - Diszkrét értékkészlet: Kiszolgáló-folyamat UP/DOWN/ERROR, …
- Összegyűjtött mérési adatok
  - Skalár mennyiség (pl. kumulatív hálózati forgalom)
  - Napló bejegyzések
- Értesítés eseményekről
  - Diszkrét állapotváltozás (ok→down)
  - Határérték túllépés (diszk telítettség >90%)

Ágens lekérdezési interfész

- Ágens interfészek működési elv szerint
  - Pull – a központi adatgyűjtő kezdeményezi az ágensek lekérdezést
  - Push – az ágens kezdeményezi az adatok elküldését a feliratkozott adatgyűjtő központnak

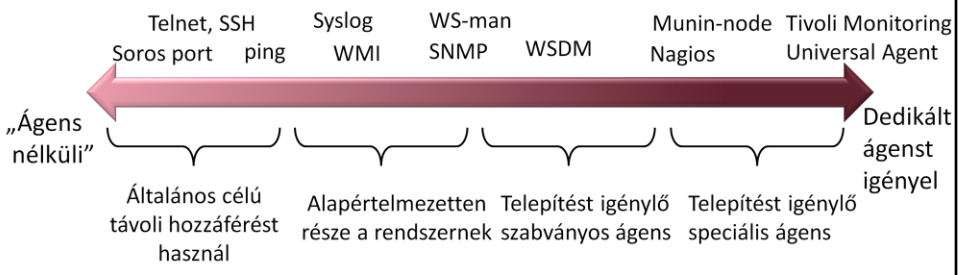


Szabványos protokollok



„Ágens alapú” és „ágens nélküli” technológiák

- Igazából nincs olyan, hogy ágens nélküli
  - Parancssoros belépés és értéklekérdezés: távoli hozzáférés kiszolgáló az „ágens”
  - Inkább: specializáltság alapján



Általános célú távoli hozzáférést használ

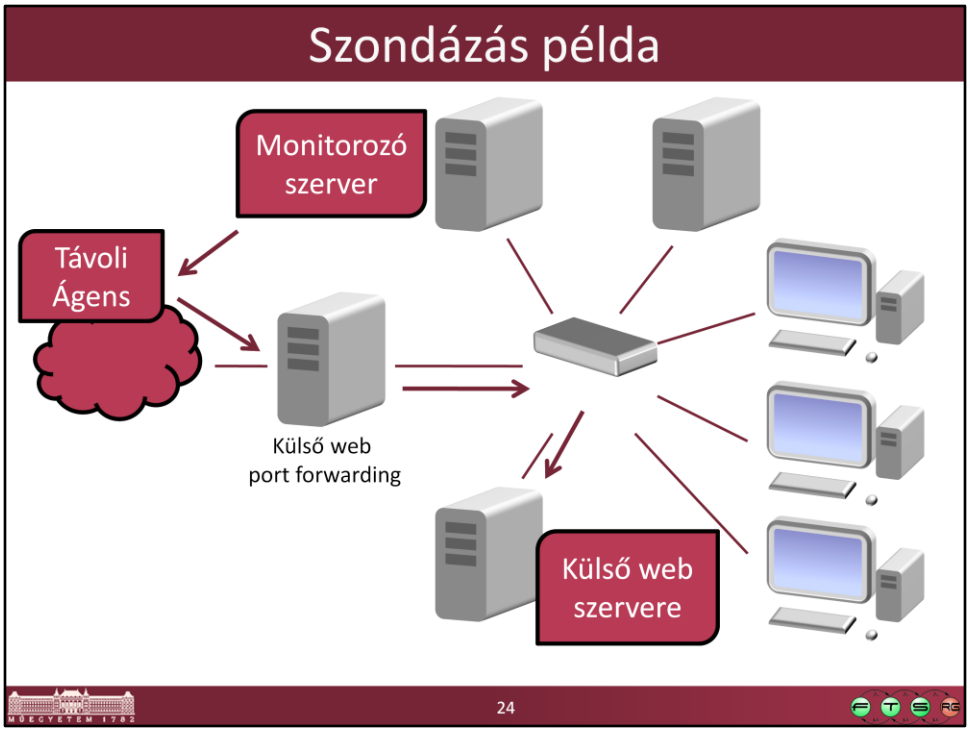
Alapértelmezetten része a rendszernek

Telepítést igénylő szabványos ágens

Telepítést igénylő speciális ágens

Szondázás

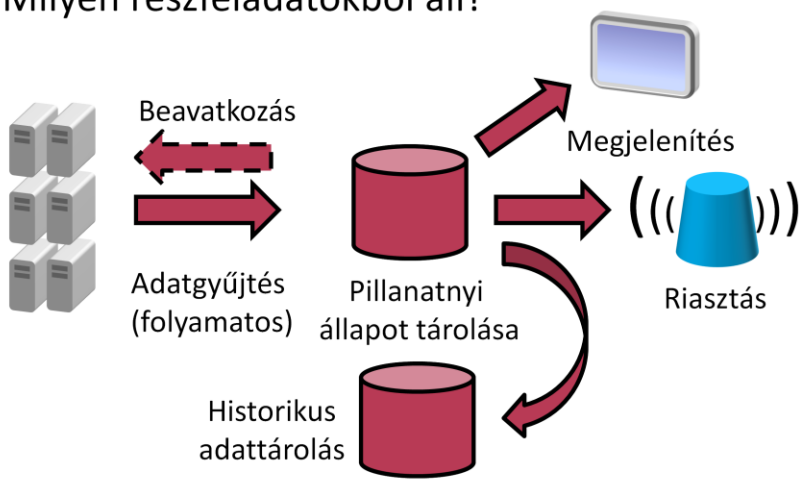
- Szondázás - probing
  - Tipikusan „ágens nélküli” megközelítés: ha nem belenézni akarunk a célrendszerbe, hanem a távolról elérhető szolgáltatását kipróbálni
  - Ilyenkor a monitorozó rendszer, mint hálózati kliens próbál igénybe venni egy szolgáltatást
  - Ilyenkor is szükség lehet ágensre
    - Meghatározott **szolgáltatás elérési pontról** (Service Access Point) nézve akarunk képet kapni a szolgáltatásról



A külső webnek, mint szolgáltatásnak a megcélzott szolgáltatás elérési pontja a tűzfalon kívül van -> akkor van „jó” állapotban a külső web, ha kívülről nézve működik. A monitorozó szerver viszont belül van, ezért kell egy távoli ágens, amit megkérhet, hogy kívülről is megnézze a szolgáltatást.

Rendszermonitorozás részei

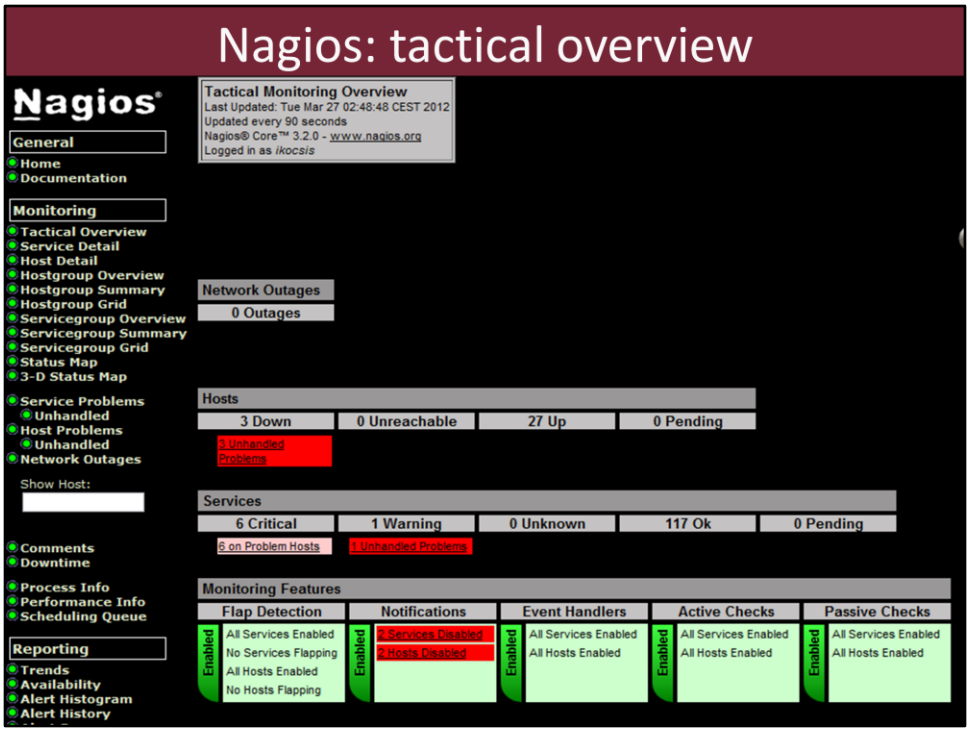
- Milyen részfeladatokból áll?



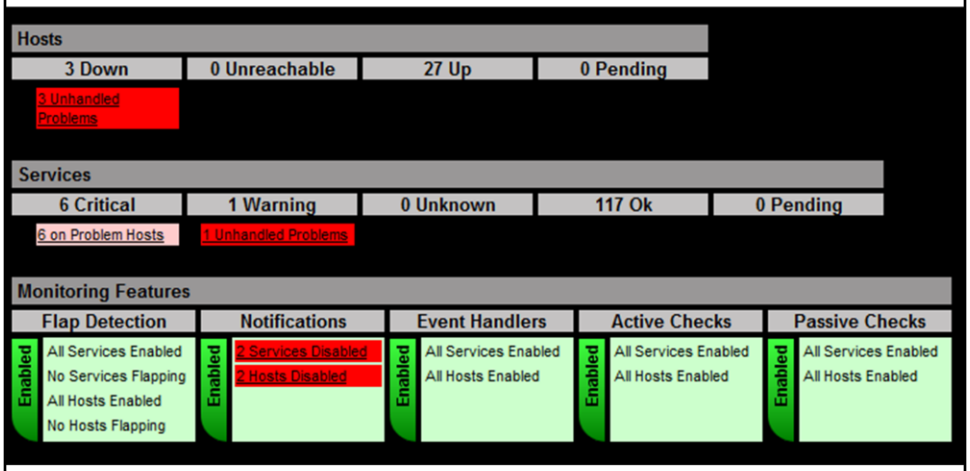
Monitorozó rendszer példa: Nagios

- Nagios
  - Free, open source
  - http://www.nagios.org/
  - Kevés (<100) gép megfigyelése esetén jó megoldás
  - Elsődlegesen a pillanatnyi állapot áttekintésére és automatikus riasztásra való
- Tactical overview
  - Monitorozott szolgáltatások
  - Grafikus megjelenítés
- Rendelkezésre állás és teljesítmény jelentés
- Naplók és riasztások
- Főleg aktív szondázásra alapoz, kézi konfigurálást igényel
- Saját ágens protokollja is van,
  - Egyszerű szöveges protokoll, könnyen bővíthető shell scriptekkel
  - Támogat szabványos protokollokat is

Nagios: tactical overview



Nagios: tactical overview



Nagios: service detail

[redacted]	NRPE PING	OK	2012-03-27 02:49:14	3d 9h 1m 47s	1/4	PING OK - Packet loss = 0%, RTA = 0.47 ms
	PING	OK	2012-03-27 02:49:15	31d 10h 37m 10s	1/4	PING OK - Packet loss = 0%, RTA = 0.23 ms
[redacted]	ESX Management Console	OK	2012-03-27 02:50:16	27d 10h 40m 46s	1/4	TCP OK - 0.000 second response time on p
	HTTP	OK	2012-03-27 02:46:30	55d 17h 27m 29s	1/4	HTTP OK: HTTP/1.1 301 Moved Permanently
	HTTPS	OK	2012-03-27 02:48:00	55d 17h 26m 35s	1/4	HTTP OK: HTTP/1.1 200 OK - 5293 bytes in
	PING	OK	2012-03-27 02:48:17	55d 17h 23m 12s	1/4	PING OK - Packet loss = 0%, RTA = 0.66 ms
[redacted]	ESX Management Console	CRITICAL	2012-03-27 02:49:17	128d 2h 29m 24s	1/4	No route to host
	HTTP	CRITICAL	2012-03-27 02:50:19	128d 2h 29m 24s	1/4	No route to host
	HTTPS	CRITICAL	2012-03-27 02:46:30	128d 2h 29m 24s	1/4	No route to host
	PING	CRITICAL	2012-03-27 02:48:00	128d 2h 29m 24s	1/4	CRITICAL - Host Unreachable (152.66.253.1
[redacted]	ITIM	OK	2012-03-27 02:48:19	53d 10h 46m 30s	1/4	HTTP OK: Status line output matched "HTTP/
	PING	OK	2012-03-27 02:49:24	51d 4h 40m 35s	1/4	PING OK - Packet loss = 0%, RTA = 0.27 ms
	SSH	OK	2012-03-27 02:46:30	23d 21h 19m 41s	1/4	SSH OK - OpenSSH_3.9p1 (protocol 1.99)

Adatgyűjtéstől a diagnosztikáig: szondázás

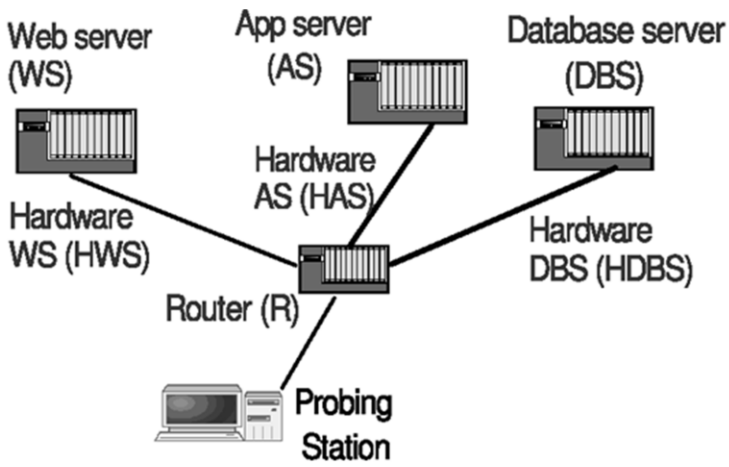
Diagnosztika

- Nem megy a webkiszolgáló. De *miért* nem?
  - Megfelelő megfigyelések kellenek
- Adott hibahatás okának felderítéséhez mit figyeljünk?
  - Pl. *egy* ESX hoszt több száz valósidejű metrikát definiál magán + VM-ek metrikái
  - Egy operációs rendszer még bonyolultabb lehet
- Hogyan következtessünk a hibaokra?

Diagnosztika

- Hibaok-detektálás (fault detection): van-e hibahatást (failure) okozó jelenség a rendszerben
- Hibaok-lokalizáció (fault localization): a hibahatást kiváltó pontos hibaokok meghatározása
- Szondázás: olyan teszttranzakció, melynek kimenetele több komponens állapotától is függhet
  - Gondoljuk végig: VM-ben futó Apache-re wget távolról
- I. Rish et al. (2005). Adaptive diagnosis in distributed systems. *IEEE transactions on neural networks*, *16*(5), 1088-1109.

Függőségek


Web server
(WS)
App server
(AS)
Database server
(DBS)
Hardware
AS (HAS)
Hardware
WS (HWS)
Hardware
DBS (HDBS)
Router (R)
Probing
Station

(Kiterjesztett) függőségi mátrix

Problem / Probe	WS	AS	DBS	R	HWS	HAS	HDBS	NF
pWS	1	1	1	1	1	1	1	0
pAS	0	1	1	1	0	1	1	0
pDBS	0	0	1	1	0	0	1	0
pingR	0	0	0	1	0	0	0	0
pingWS	0	0	0	1	1	0	0	0
pingAS	0	0	0	1	0	1	0	0
pingDBS	0	0	0	1	0	0	1	0

Egyszeres hibaok-feltételezésnél a hibaaktivációs kombinációk

Szonda futásának eredménye

Probes:

pWS - Web Page access, **pAS**-
pDBS - Database query, **pingR**
pingWS – ping Web Server, **pi**
server, **pingDBS** - ping Datab

!!! „Elkódolt” tudás:
- topológia-modell
- Szolgáltatás-függőségi modell
- (Egyszerű) hiba(terjedési) modell

Detektálás/lokalizálás

- Minimális hiba*detektáló* szondahalmaz választása?

Detektálás/lokalizálás

	WS	AS	DBS	R	HWS	HAS	HDBS	NF
pWS	1	1	1	1	1	1	1	0
pAS	0	1	1	1	0	1	1	0
pDBS	0	0	1	1	0	0	1	0
pingR	0	0	0	1	0	0	0	0
pingWS	0	0	0	1	1	0	0	0
pingAS	0	0	0	1	0	1	0	0
pingDBS	0	0	0	1	0	0	1	0

Detektálás/lokalizálás

	WS	AS	DBS	R	HWS	HAS	HDBS	NF
pWS	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**0**
pAS	**0**	**1**	**1**	**1**	**0**	**1**	**1**	**0**
pDBS	**0**	**0**	**1**	**1**	**0**	**0**	**1**	**0**
pingR	**0**	**0**	**0**	**1**	**0**	**0**	**0**	**0**
pingWS	**0**	**0**	**0**	**1**	**1**	**0**	**0**	**0**
pingAS	**0**	**0**	**0**	**1**	**0**	**1**	**0**	**0**
pingDBS	**0**	**0**	**0**	**1**	**0**	**0**	**1**	**0**

Detektálás/lokalizálás

- Minimális hiba*detektáló* szondahalmaz választása?
  - Az a minimális szondahalmaz, amire minden oszlopösszeg > 0
  - NP-nehéz ☹
  - == minimális halmazfedés („minimum set cover”)
  - De: igen jó heurisztikák

Detektálás/lokalizálás

- Minimális hiba*lokalizáló* szondahalmaz választása?

Detektálás/lokalizálás

	WS	AS	DBS	R	HWS	HAS	HDBS	NF
pWS	1	1	1	1	1	1	1	0
pAS	0	1	1	1	0	1	1	0
pDBS	0	0	1	1	0	0	1	0
pingR	0	0	0	1	0	0	0	0
pingWS	0	0	0	1	1	0	0	0
pingAS	0	0	0	1	0	1	0	0
pingDBS	0	0	0	1	0	0	1	0

Problémák

	WS	AS	DBS	R	HWS	HAS	HDBS	NF
pWS	1	1	1	1	1	1	1	0
pAS	0	1	1	1	0	1	1	0
pDBS	0	0	1	1	0	0	1	0
pingR	0	0	0	1	0	0	0	0
pingWS	0	0	0	1	1	0	0	0
pingAS	0	0	0	1	0	1	0	0
pingDBS	0	0	0	1	0	0	1	0

Detektálás/lokalizálás

	WS	AS	DBS	R	HWS	HAS	HDBS	NF
pWS	1	1	1	1	1	1	1	0
pAS	0	1	1	1	0	1	1	0
pDBS	0	0	1	1	0	0	1	0
pingR	0	0	0	1	0	0	0	0
pingWS	0	0	0	1	1	0	0	0
pingAS	0	0	0	1	0	1	0	0
pingDBS	0	0	0	1	0	0	1	0

Detektálás/lokalizálás

	WS	AS	DBS	R	HWS	HAS	HDBS	NF
pWS	1	1	1	1	1	1	1	0
pAS	0	1	1	1	0	1	1	0
pDBS	0	0	1	1	0	0	1	0
pingR	0	0	0	1	0	0	0	0
pingWS	0	0	0	1	1	0	0	0
pingAS	0	0	0	1	0	1	0	0
pingDBS	0	0	0	1	0	0	1	0

Detektálás/lokalizálás

	WS	AS	DBS	R	HWS	HAS	HDBS	NF
pWS	1	1	1	1	1	1	1	0
pAS	0	1	1	1	0	1	1	0
pDBS	0	0	1	1	0	0	1	0
pingR	0	0	0	1	0	0	0	0
pingWS	0	0	0	1	1	0	0	0
pingAS	0	0	0	1	0	1	0	0
pingDBS	0	0	0	1	0	0	1	0

Detektálás/lokalizálás

- Minimális hiba*lokalizáló* szondahalmaz választása?
  - Az a minimális szondahalmaz, ahol minden hibaok-párt meg tudunk még különböztetni → páronként különböző oszlopok
  - NP-nehéz ☹
  - Szintén jó heurisztikák

Historikus adatok gyűjtése

Historikus adatgyűjtés

- De jó lenne, ha…
  - Visszamenőleg látnánk, hogy mi történt
  - Látnánk a tendenciákat
  - Következtetéseket vonhatnánk le. Pl.:
    - Mi van túlterhelve, mi nincs kihasználva (bővítés tervezése)
    - Hogy néz ki, amikor 500 hallgató megrohanja a szervert ☺
    - Mennyi idő alatt sülnek meg a gépek, ha leáll a klímaberendezés (katasztrófa elhárítási terv)
    - Nem kezdett-e el valami elfogyni/elhasználódni, amit majd cserélni, pótolni kéne? (Proaktív beavatkozás) Pl. szabad tárhely, UPS akkumulátorok, merevlemezek, nyomtató toner stb.

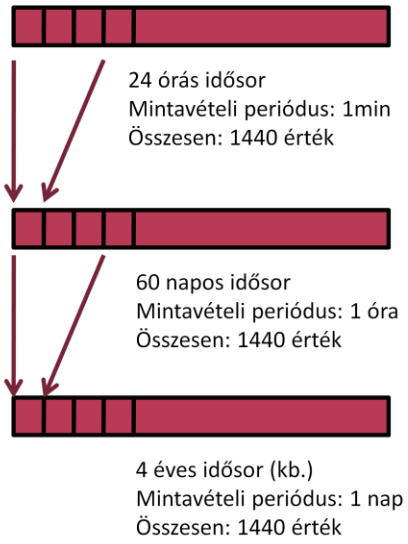
Historikus adatgyűjtés

- Megoldás
  - Periodikusan (mondjuk percenként mintavételezve) tároljuk el a mért értékeket
  - Mi ezzel a baj?
  - Számoljunk utána: belefulladunk az adathalmazba
  - Biztos, hogy tudni akarjuk, hogy pontosan mi történt 1 éve 5 hónapja, 13 napja, 8 óra 13 perce?
  - Attól függ:
    - Trend megállapításhoz: ilyen pontosan nem, de azért hozzávetőlegesen igen
    - Konkrét esemény dokumentálásához: kell a nagy pontosság

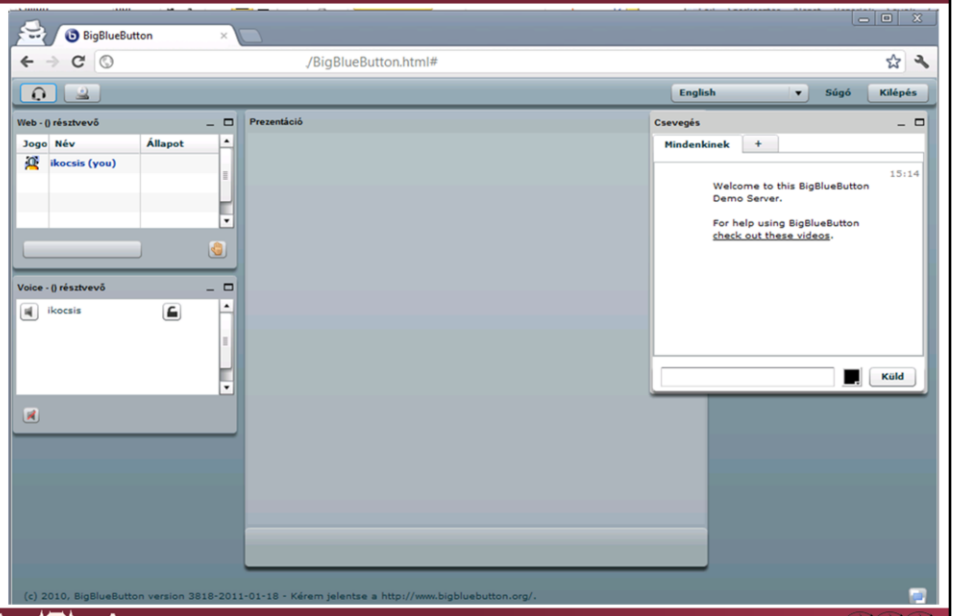
Van, amihez ez is kevés...

Historikus adatgyűjtés

- Aggregáció
  - „Adattárházas” fogalom
  - Több adatot vonunk össze egyetlen értékbe (felbontás rontás, pl. átlagolással)
  - Mit veszítünk vele?
    - Konkrét, rövid események lefutása
    - Börsztösség
  - Mit lehet tenni ellene?
    - külön archiválni kell az „érdekes” részeket -> eseménykorreláció
    - Összevont MIN/MAX/AVG értéket tárolni



BigBlueButton

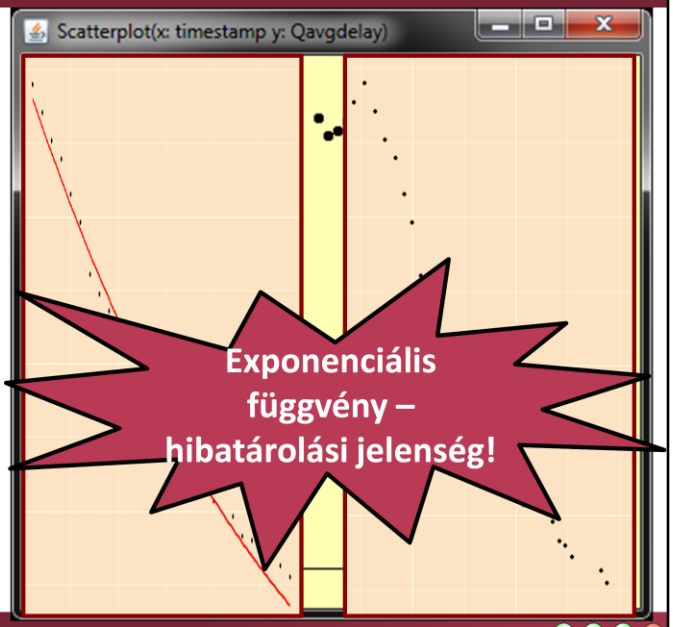


Rövid tranziens – hosszú kicsengés

Alacsony felh.szám

X: idő
Y: átlagos késleltetés

Tranziens CPU-túlterhelés

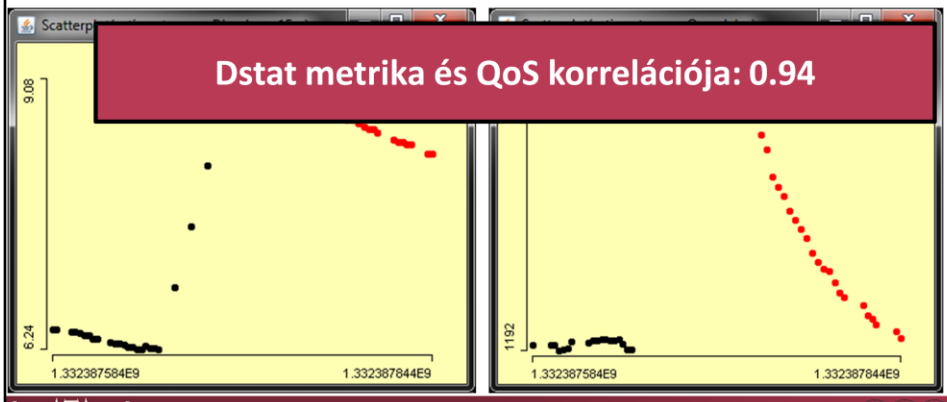


Erőforrásmetrikákkal korreláció

X: idő
Y: dstat load_avg_15min

X: tidő
Y: átlagos késleltetés

Dstat metrika és QoS korrelációja: 0.94



Metrikák

- dstat: Linux monitorozó eszköz
  - CPU, disk, paging, load, memory, network, processes, IO, swap, ...
- Unix load
  - „load number”: CPU-ra váró vagy azt használó folyamatok (ready queue/run queue)
  - 1/5/15 perces metrika: exponenciálisan súlyozott csúszóablakos átlag

Megfigyeléstől a menedzselésig

- Nade miket mérjünk?
  - dstat –Tcdglmnprsy: önmag
- Milyen felbontással?
- Mi a diagnosztikai logika?

Vizuális analízis + MI dimenzióredukció / változószelekció

Méréstechnika és méréselmélet

- Mi a cél?
  - Post-mortem analízis?
  - Hibaok-megelőzés?
  - Detektálás adott időablakon belül?
  - Proaktív javítás?

Inkább futásidejű, mint historikus