Operációs rendszerek:

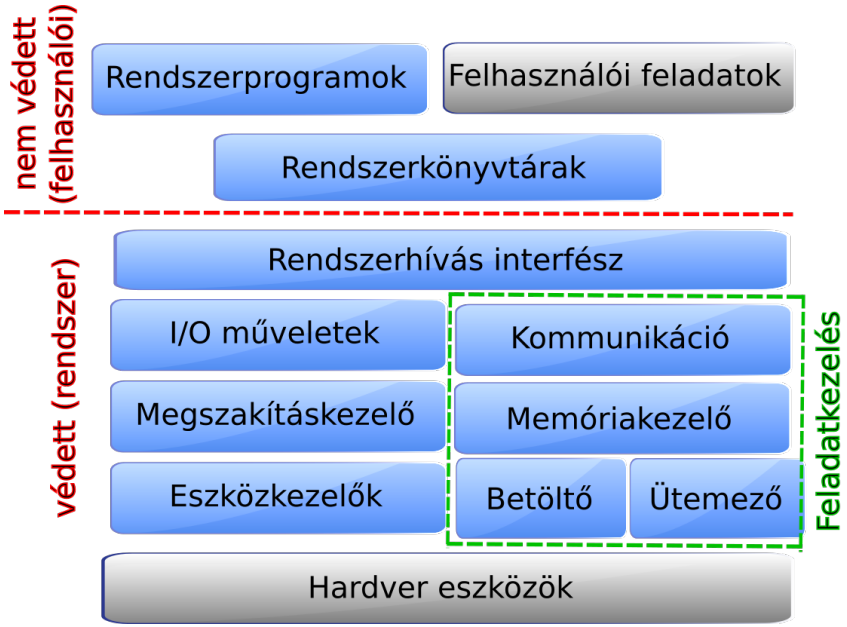
a virtualizáció alapjai

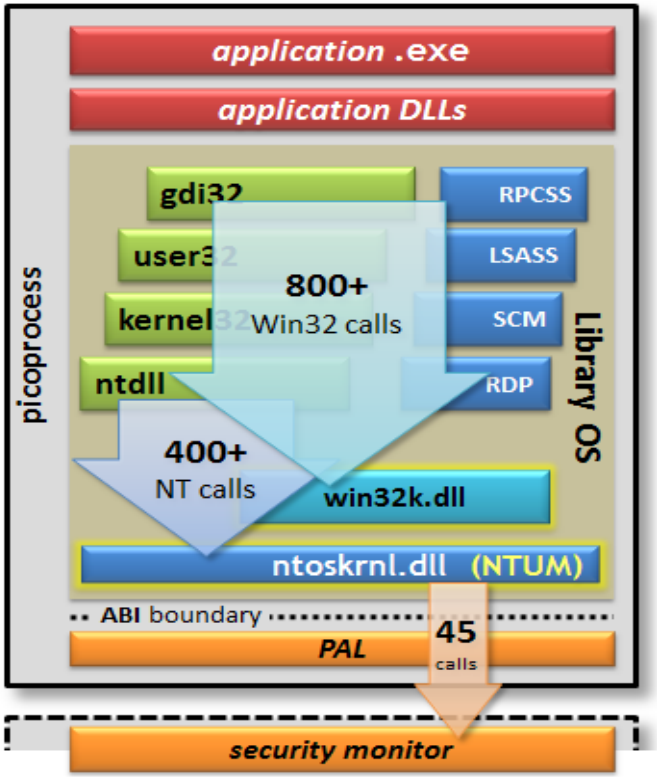
Mészáros Tamás
http://www.mit.bme.hu/~meszaros/

Budapesti Műszaki és Gazdaságtudományi Egyetem
Méréstechnika és Információs Rendszerek Tanszék

Az eddigiekben történt...

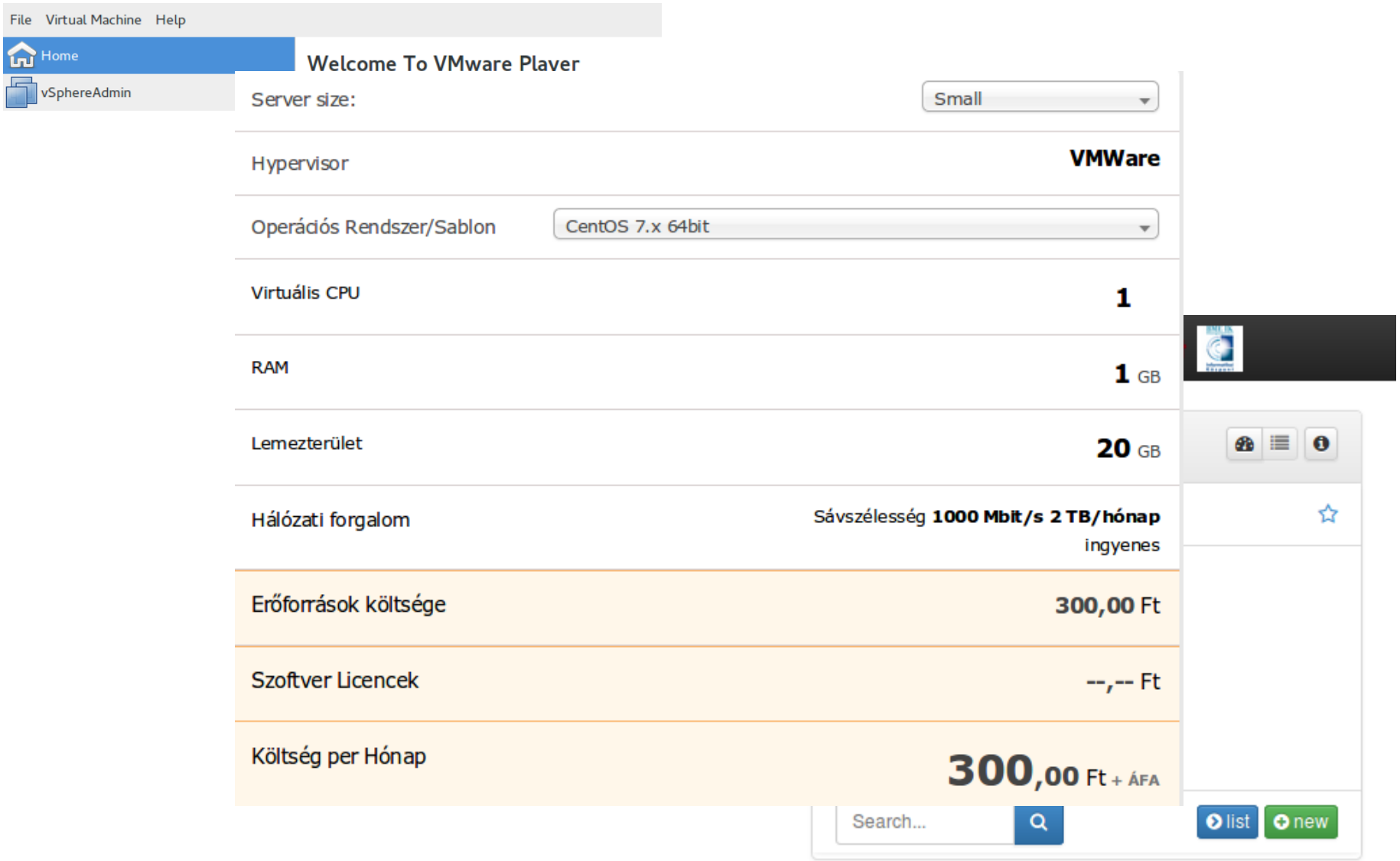
- Az OS multiprogramozott
  - szeparálja a taszkokat
  - absztrakt virtuális gép koncepció
- Erőforrások
  - CPU + védelmi szintek
  - memória + MMU + VMM
  - tárolórendszer virtualizáció



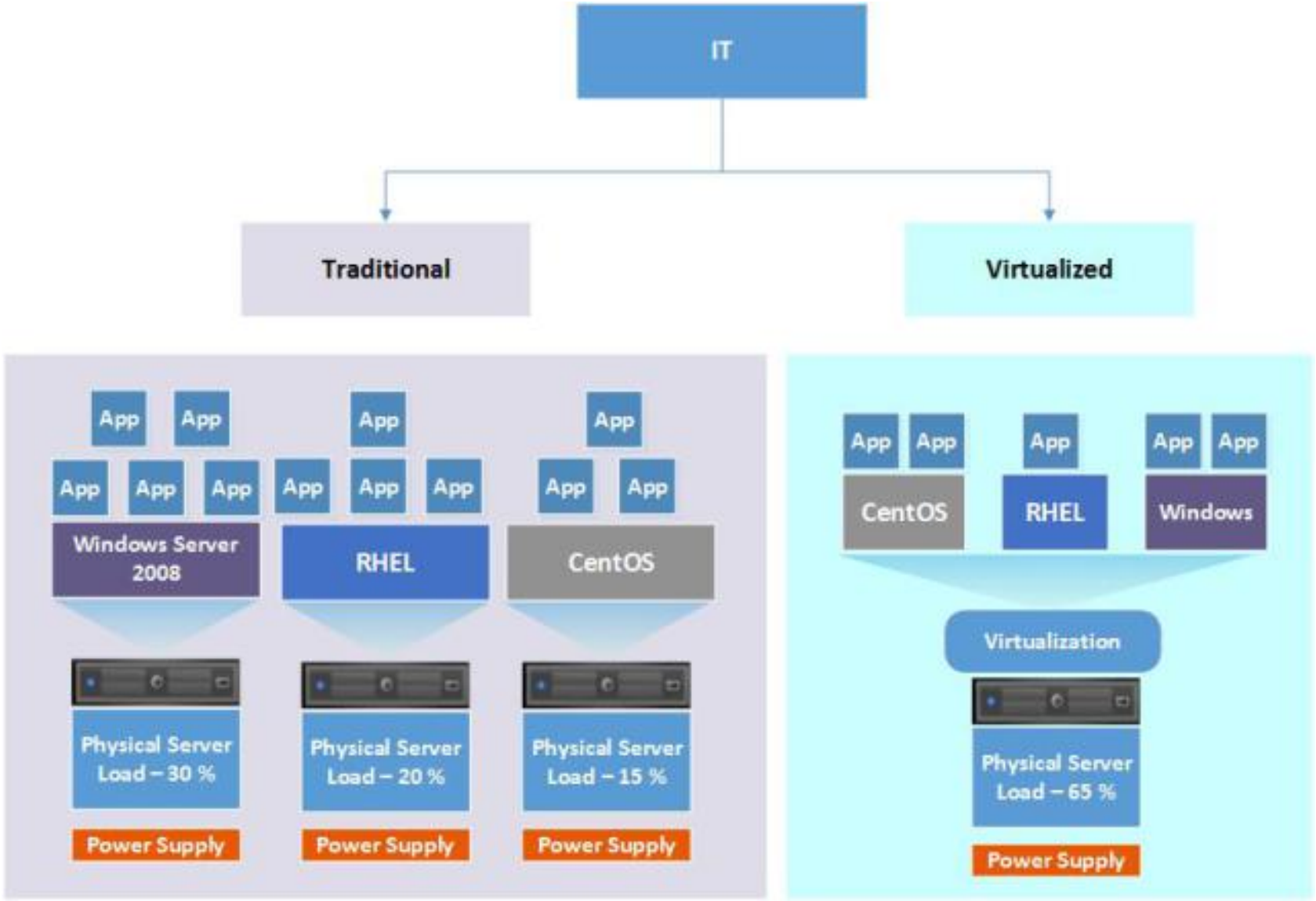


- Komplexitás → hibák, költségek
  - fejlesztés (l. architekturális részek)
  - üzemeltetés (l. laborok)

Mi a virtualizáció?



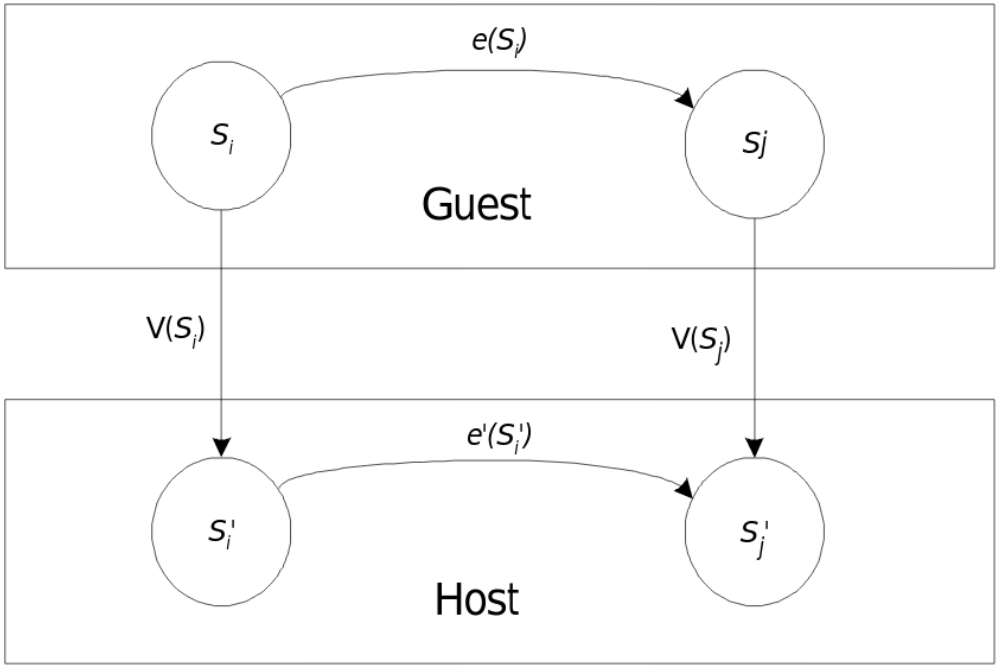
Mi a virtualizáció?



Mi a virtualizáció?

Erőforrás virtuális (szoftveres) változatának létrehozása, amely az eredetire támaszkodva, ahhoz hasonlóan, de attól elválasztott módon működik.

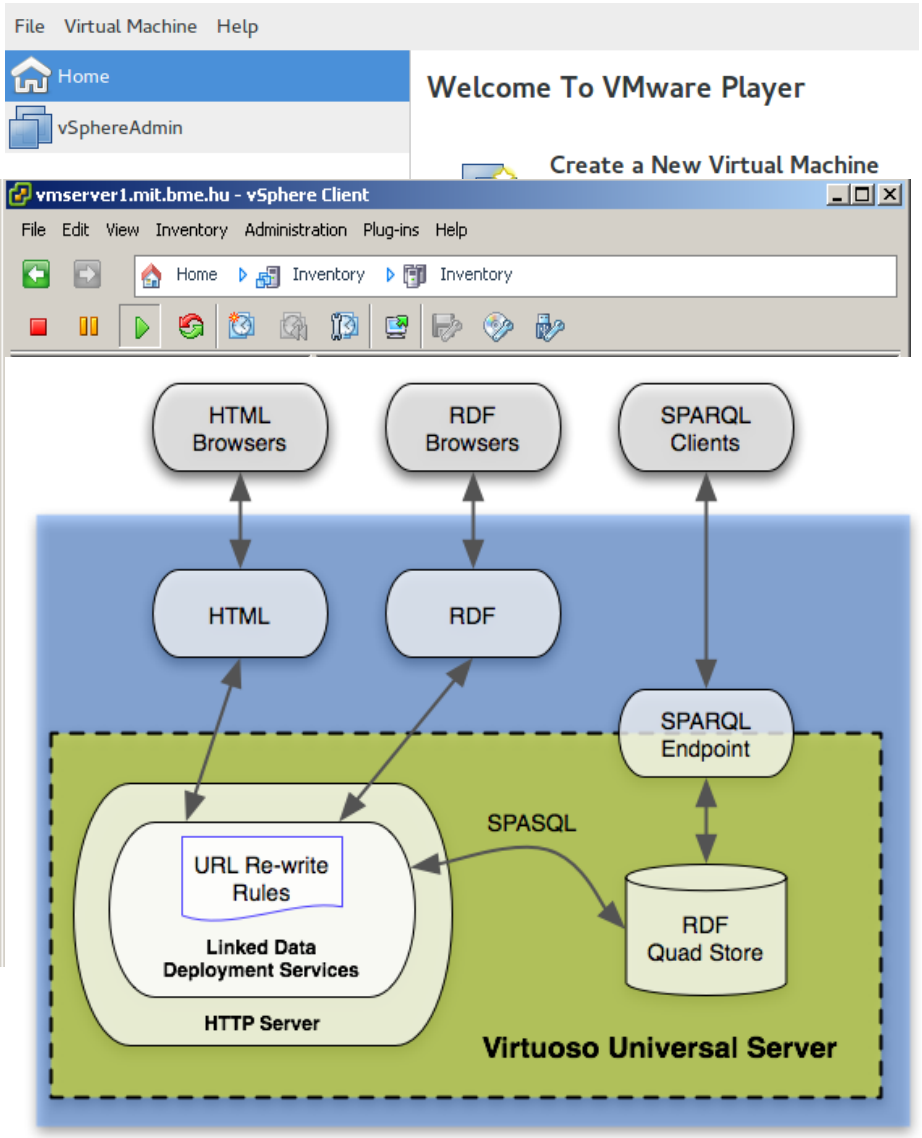
virtualizált erőforrás: számítógépes hardver vagy szoftver
host (gazda): a virtualizált erőforrást biztosítja
guest (vendég): az erőforrás felhasználója



Forrás: Smith, Nair: Introduction to Virtual Machines

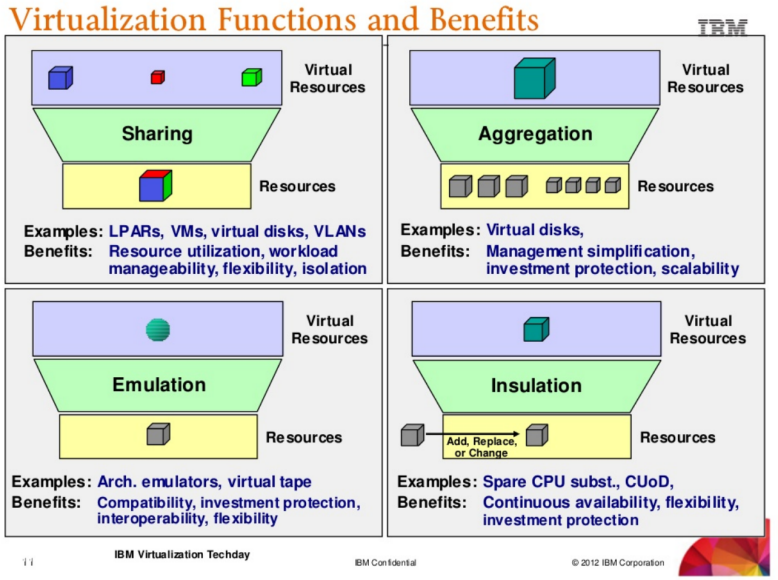
Mi virtualizálható?

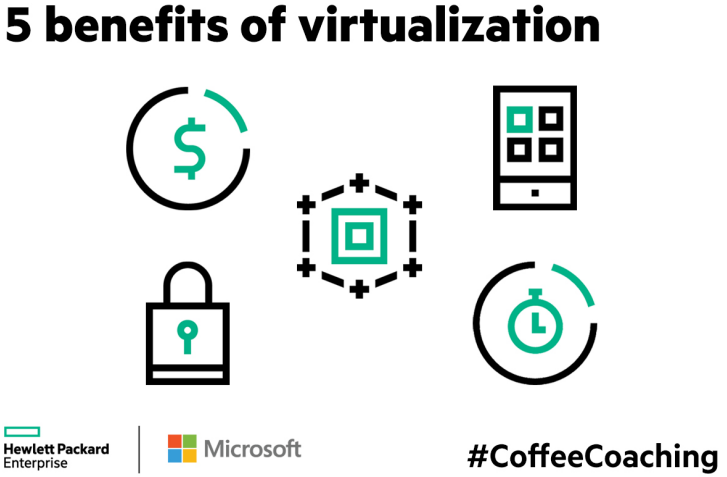
- Hardver
  - teljes számítógép
  - számítógépes hálózat
  - grafikus kártya
  - stb.
- Szoftver
  - egy szolgáltatáshalmaz, azaz API
  - lehet rendszerkönyvtár (pl. GUI), kernel (vagy egy része) is
- Adat
  - formátum- és elhelyezkedésfüggetlen
  - hozzáférés és módosítás
- Egy már virtualizált rendszer is



Mire jó a virtualizáció?

- Konkurens erőforrás-használat (→ multiprogramozott OS)
  - egyszerre többen használhatják az erőforrást
  - kezeli a versenyhelyzeteket
- Összeolvasztás
  - kapacitásbővítés (lásd még tárolórendszerek)
  - szolgáltatások fúziója
- Szolgáltatásbővítés és -szűkítés
  - csak ami kell, akár többfélét összegyúrva
  - újfajta aggregált szolgáltatásokat megvalósítva
- Felügyelet, menedzsment (→ OS)
  - szabályozott, automatizált
  - szereplők, jogosultságok
- Archiválás
  - „dobozba zárva” megőrizhető





Miért jó a virtualizáció?

- Erőforrás-kihasználtság (→ OS)
  - több használó, kevesebb „üresjárat”
- Csökken a gyártófüggőség
  - helyettesíthető erőforrások
- Csökken az erőforrások száma
  - kevesebb hiba és energiafelvétel
- Jobb menedzsment
  - automatizálható, egyszerűsíthető
- Nagyobb izoláció
  - kisebb, szeparált támadási felületek
- Hatékonyabb rendszerfejlesztés
  - (fél)kész komponensek polcról
  - egyszerűbb tesztelés és archiválás

Csökkenő költségek

TCO: total cost of ownership

beruházás

fenntartás (menedzsment)

Növekvő rendelkezésre állás

kezelhetőbb hibák

megbízhatóbb rendszerek

Növekvő flexibilitás

rugalmasabb specifikáció

skálázható, adaptív rendszerek

Kockázatok és mellékhatások

támadható és hibaforrás (SPOF)

van rezsiköltsége

komplex lehet a kezelése

A virtualizáció főbb fajtái

- Rendszer (system / platform / full)
  - teljes rendszer virtualizációja
  - teljes környezetet („élettér”) biztosítása feladatok végrehajtására
  - az erőforrás egy teljes rendszer
  - feladatok: OS és taszkok
  - pl. VMware Player

- Folyamat (process / software)
  - API / ABI virtualizációja
  - taszkok működéséhez biztosít felületet
  - az erőforrás egy működéshez szükséges (futtatási) felület
  - pl. Java VM

- Infrastruktúra (infrastructure)
  - (jellemzően fizikai) infrastrukturális elemek virtualizációja
  - az erőforrás egy hardver/szoftver elem
  - pl. számítógépes hálózat, adattároló rendszer stb.

Rendszer virtualizáció

- Cél: teljes virtualizált környezet felépítése
  - (jellemzően) **hardver virtualizáció**: teljes számítógép virtualizálása
  - **virtuális gép**: a virtualizált hardveren futó rendszer
  - a virtuális gépek használata a fizikai gépekhez hasonló módon történik
- Összetevői
  - **gazda számítógép** (host): a fizikai gép, amelyen a virtuális gépek futnak
  - **vendég gép** (guest): a gazdagépen futó virtuális gép
  - **virtuális gép monitor** (VMM): a virtuális gépeket felügyelő program
- Sokféle altípus
- Példák
  - Vmware Player, Xen, KVM, Hyper-V és ezernyi más
- Értékelés
  - egyszerű → nagyon elterjedt
  - hardvertámogatás? teljesítmény?

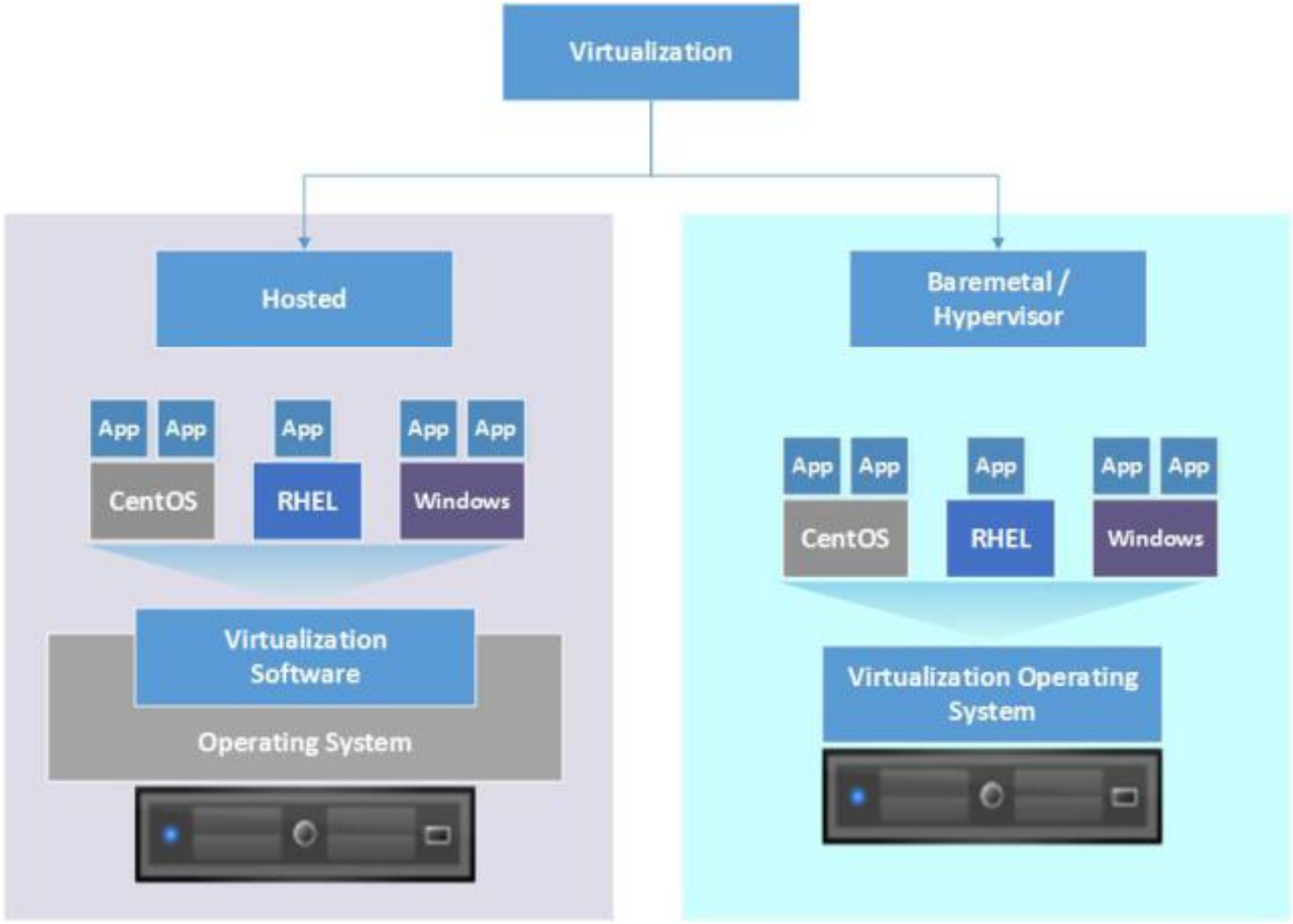
A hardver virtualizáció fajtái

- *Bare metal (1. típusú)*
  - a hardvert a VMM kezeli
  - a gazdagépen nem futnak más alkalmazások
  - a VMM neve ebben az esetben **hypervisor**
  - fizikai – virtuális hardver megfeleltetése
    - transzparens módon: **natív virtualizáció**
      - hardveres támogatással, vagy futásidejű bináris átírással
    - más hardver képében: **paravirtualizáció**
      - a fizikai hardverhez hasonló, de nem megegyező virtuális hardver

- *Hosted (2. típusú)*
  - a hardvert egy OS kezeli
  - a VMM egy alkalmazás a gazdagépen (pl. VMware Player)
  - a gazdagépen más alkalmazások is futhatnak (több VMM is)

- Hibrid megoldások
  - a hypervisor-ral egybeépítve is működik egy kernel, így
  - a VMM egyes funkcióit célszerű lehet az OS kernelre építve megvalósítani

Bare metal vs. hosted virtualizácó



A virtualizáció megvalósítása

A virtualizáció megvalósítása: elvárások

- Transzparencia
  - a vendég gép változtatás nélkül működjön
  - a programokat ne kelljen kézzel átírni
  - legyen automatikus és láthatatlan az utasítás-átírás

  → sok feladatot ró a virtualizációs rendszerre

- Védelem
  - vendég ↔ gazda, vendég ↔ vendég
  - pl. natív virtualizáció és a HALT utasítás → ne álljon le a gazdagép

  → felügyelet, jogosultságok megvalósítása

- Hatékonyság
  - a VMM rezsiköltsége legyen kicsi
  - az átírás minél kevésbé csökkentse a teljesítményt

  → a hardvertámogatás minél teljesebb kihasználása

A virtualizáció megvalósítása: CPU

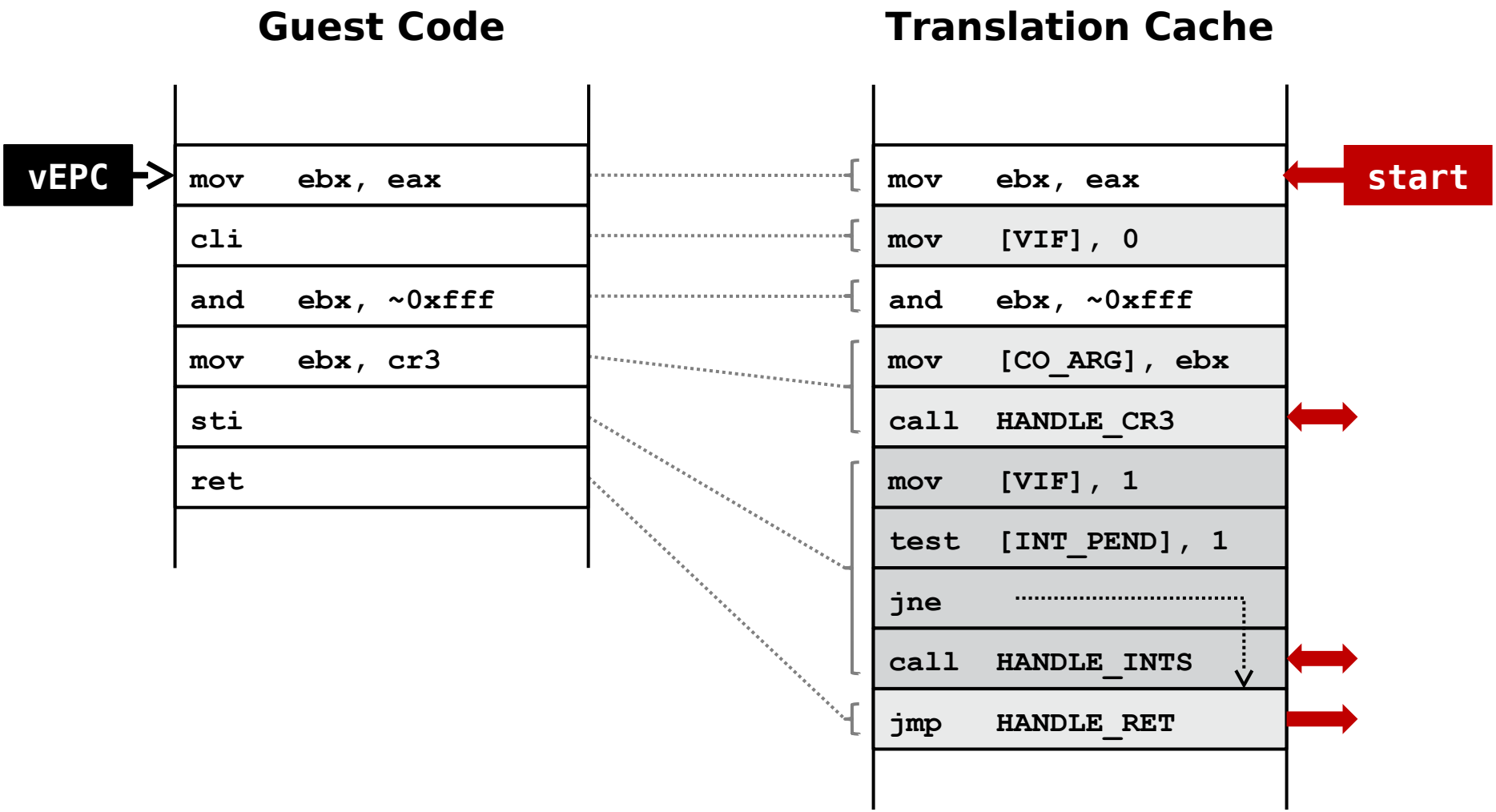
- Tiszta emuláció
  - virtuális hardveren (állapotgépen) hajtja végre az utasításokat
  - az utasításokat leképezi (lefordítja) a fizikai eszköze
  - **nem hatékony**

- Trap and emulate
  - utasítások válogatása **futásidőben** (végrehajtás közben)
    privilegizált: elkapja és átírja; nem védett: közvetlenül végrehajtja
  - hatékony, de **hardvertámogatást igényel**

- Bináris átírás
  - a VMM privilegizált utasításokat **végrehajtás előtt** (de futásidőben) átírja
  - a CPU már a biztonságos utasításokat hajtja végre
  - az átírás valamelyest csökkenti a hatékonyságot

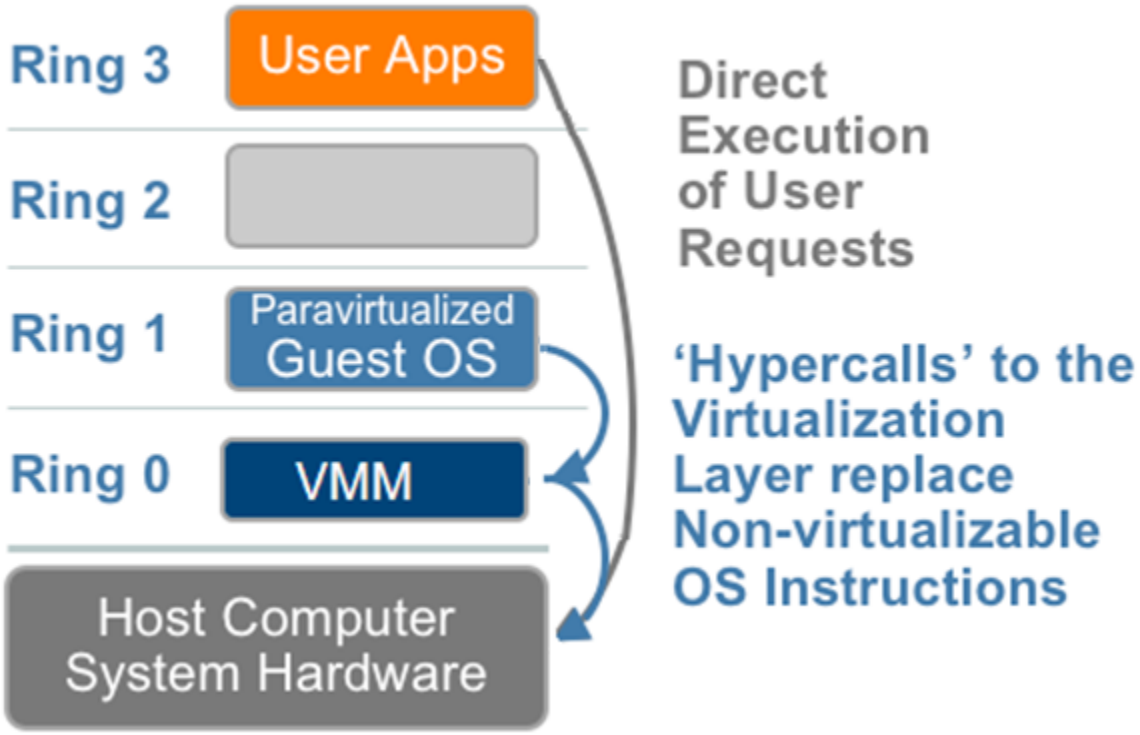
- Forráskód-átírás (paravirtualizáció)
  - a vendég OS forráskódját alakítják át fejlesztési időben
  - a privilegizált utasításokat VMM hívásokra cserélik
  - hatékony, de fejlesztői támogatást igényel (mai OS-ekben jellemző)

Példa a bináris átírásra

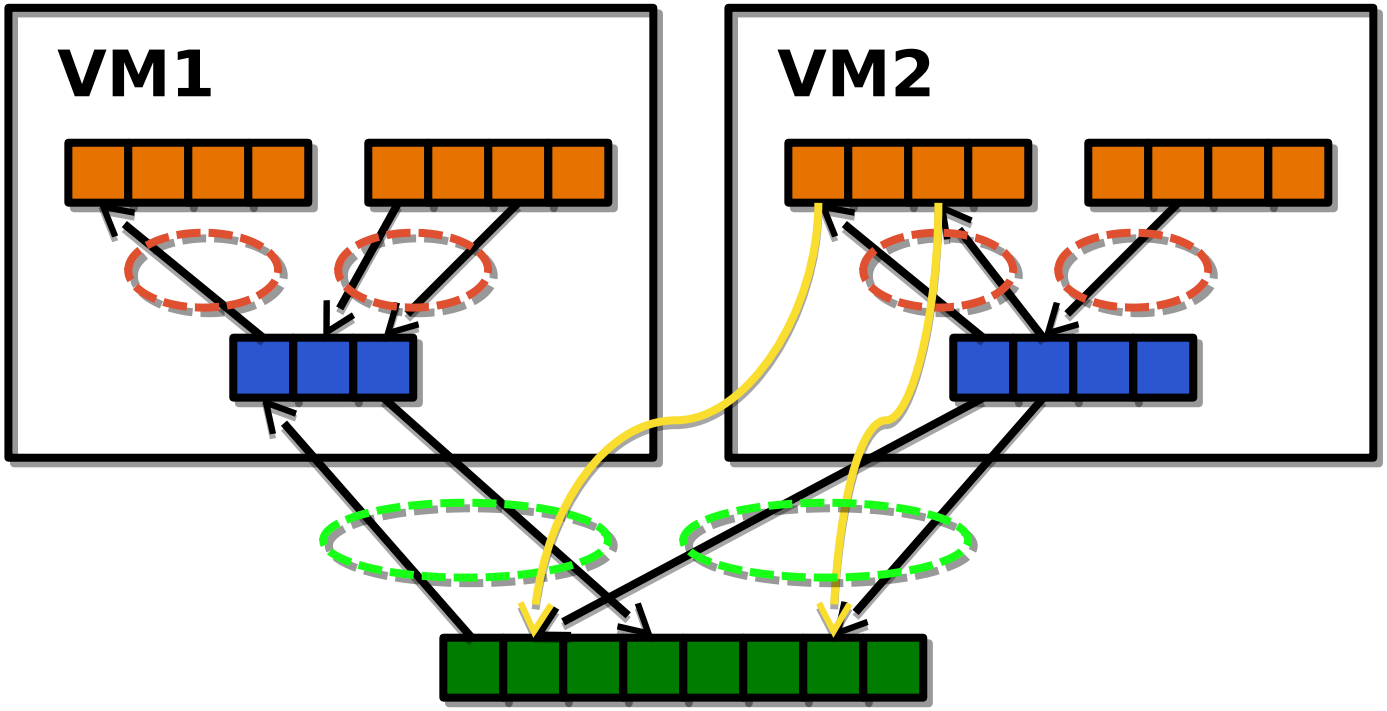


Forrás: Carl Waldspurger, Introduction to Virtual Machines

Paravirtualizáció



A virtualizáció megvalósítása: memória

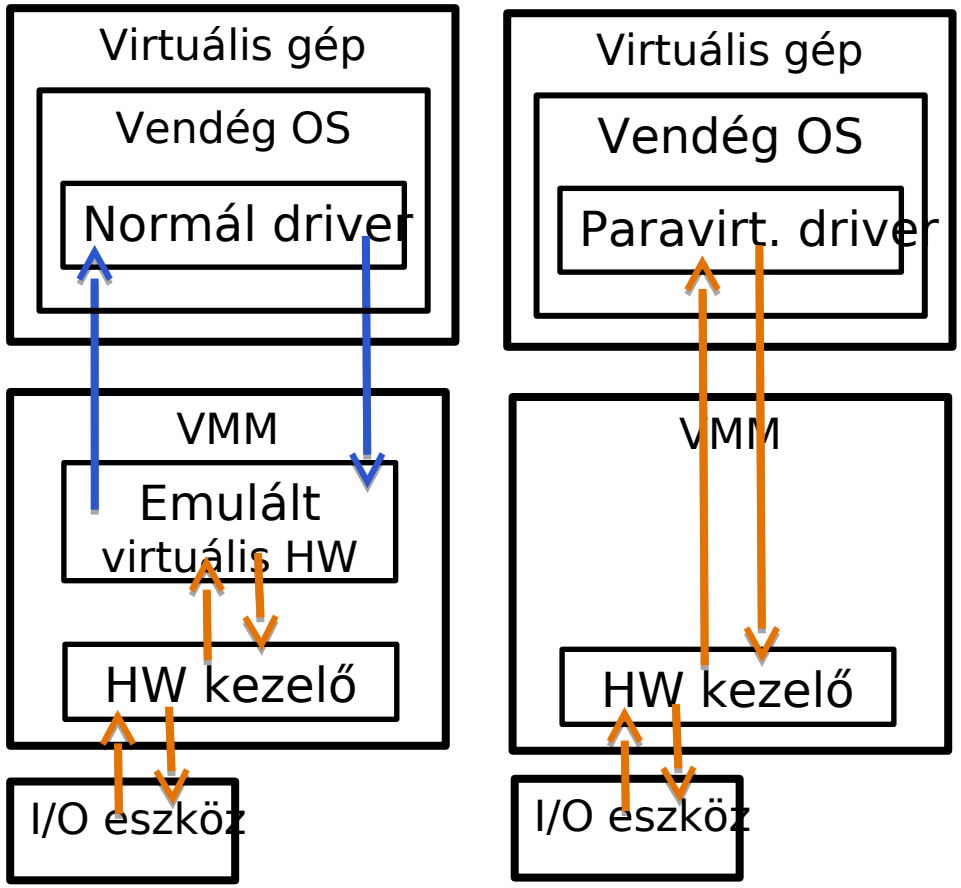


- Teljesítményérzékeny terület
- Kétszeres címfordítás
  - hardvertámogatás nélkül nagyon költséges
  - AMD Rapid Virtualization Indexing, Intel Extended Page Tables
    - beágyazott laptáblák és TLB címkézés

Az ábrát Micskei Zoltán, BME MIT készítette.

A virtualizáció megvalósítása: I/O

- Szoftveres emuláció
  - a teljes kommunikációt emulálja
  - egyszerű hardvert emulált
  - korlátozott képességek
  - transzparens, de **nem hatékony**
- Paravirtualizáció
  - a vendég a virtualizációs rendszer által felkínált eszközt használja
  - bizonyos hívások, adatmozgatások egyszerűsödnek a hardver felé
  - **speciális eszközmeghajtót kell telepíteni** a vendég OS-ben
  - nem annyira transzparens, de hatékonyabb az emulációnál
- hardveres virtualizáció
  - I/O eszközök megosztása
  - Intel VT-d, AMD IOMMU, PCI IOV



Az ábrát Micskei Zoltán, BME MIT készítette.

Termékek és szolgáltatások

Üzleti megoldások és piaci szereplők

- Saját kézben telepíthető rendszerek szállítói
  - VMware
  - XEN
  - Oracle Virtualbox
  - Microsoft Hyper-V, Virtual PC
  - Linux KVM
  - IBM PowerVM
  - Redhat EV
  - ...
- Szolgáltatók (felhő...)
  - Amazon EC2
  - Rackspace
  - Google Cloud Platform
  - Microsoft Azure
  - IBM Cloud
  - DigitalOcean

Felhőalapú szolgáltatások

- **IaaS**: infrastructure-as-a-service
  - teljes hardvert nyújt
  - operációs rendszert telepíthetünk
  - sokféle sablonnal
  - pl.: Amazon EC2, RackSpace, Microsoft Azure, Linode, DigitalOcean

- **PaaS**: platform-as-a-service
  - futtatókörnyezetet nyújt
  - saját alkalmazásainkat futtathatjuk
  - pl.: Amazon AWS, Microsoft Azure, Google AppEngine, Heroku

- **SaaS**: software-as-a-service
  - szoftverszolgáltatást nyújt
  - előre telepített alkalmazás (pl. adatbázis, dokumentumkezelő, email)
  - pl.: Microsoft Office365, Google Docs és Gmail

A virtualizáció kockázatai

- Támadások a virtualizációs rendszer ellen
  - a virtualizációs infrastruktúra lecserélésre (hyperjacking)
    - nagyon veszélyes, jelenleg inkább elvi lehetőség
  - támadás a virtualizációs mechanizmusok ellen
    - egy-egy mechanizmus (pl. hálózat, migráció) megfigyelése, megváltoztatása
  - a felügyelt rendszerek közötti adat- és kódszivárgás (VM jumping)
    - a szeparáció kijátszása megfigyelési vagy befolyásolási céllal
    - pl. vendég kitörése (guest breakout), a gyakorlatban is működik
- Auditálási nehézségek
  - az egyre bonyolultabb rendszer és annak nagyobb dinamizmusa miatt
- Bonyolultabb menedzsment
  - sokféle virtualizált erőforrás, összetett virtualizációs sémák
  - a rendszer telepítése és üzemeltetése esetenként nagyon összetett feladat
- Szakemberhiány
  - új és változó technológiák

Esettanulmányok: RHEV / oVirt (demo)