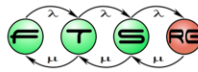


Kiszolgáló oldali virtualizáció

Tóth Dániel, Micskei Zoltán



Utolsó módosítás: 2010. 10. 21.

Tartalom

- **Követelmények**
- Szerver virtualizációs architektúrák
- Háttértárak virtualizációja
- VMware ESX és ESXi szerver architektúrája
- Microsoft Hyper-V architektúra
- Erőforrás gazdálkodás

Szerver virtualizáció

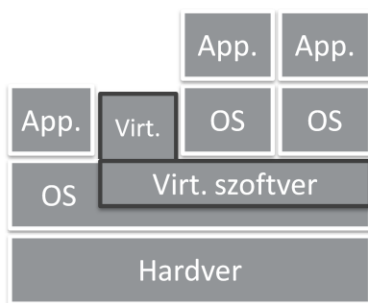
- Miben más, mint a desktop virtualizáció?
 - „Konszolidáció” – sok virtuális gép együttes futtatása, erőforrás hatékonyság, takarékoság
 - Hangsúlyos a távoli elérés, helyi hozzáférés teljesen háttérbe szorul
 - Az előző következménye: kevesebb fajta hardvert kell támogatni, nincs „multimédiás” igény
 - Karbantartási feladatok egyszerűsítése, automatizálhatóság
 - Menedzsment: monitorozás, távoli beavatkozás, terhelés és teljesítményadatok figyelése
 - „Integráció” – a virtualizált szerverek egy nagyobb infrastruktúra részei
 - Hozzáférés vezérlés, biztonság, a rendszernek sok (nem feltétlenül megbízható) felhasználója lehet egyszerre
 - Megbízható, hibatűrő működés



Most a vastagon kiemeltekkel foglalkozunk, a többi majd későbbi előadásokban kerül terítékre.

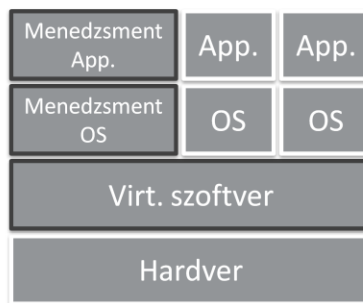
Emlékeztető – a két fő architektúra

▪ Kétféle megközelítés:



Hosted virtualizáció

Jellemzően kliens megoldások:
VMware Workstation, Server,
Player, Sun VirtualBox,
MS VirtualPC, KVM, UML



Bare-metal
virtualizáció

Jellemzően szerver megoldások:
VMware ESX Server, Xen
Enterprise, MS Hyper-V

Emlékeztető – a két fő architektúra

Hosted

- Interaktív alkalmazásnál előnyös
 - helyi hozzáférés, gyors grafika, hang stb.
 - OS szintű erőforrások biztosítása
 - Integráció asztali munkakörnyezetbe
 - Hardver meghajtókat a hoszt OS kezeli
- Jellemzően kevés virtuális gép fut egyszerre
 - sok virtuális gépnél már rossz skálázódás
 - Jellemzően az OS alkalmazásokhoz kitalált ütemezőit próbálja VM-ek erőforrásgazdálkodására használni

Bare-metal

- Interaktív alkalmazások nehézkesek
 - távoli hozzáférés kell (lokális gépen megjelenítésnél is!)
 - teljesen speciális saját környezet, csak virtuális gépek futtatására
 - nincsenek finomított OS erőforrások
 - Hardver támogatást külön meg kell oldani
- Jó skálázhatóság
 - Nincs hoszt OS, ami erőforrást fogyasztana
 - Saját, VM-ek számára optimalizált ütemezők, erőforrás-elosztók



Szerver virtualizáció

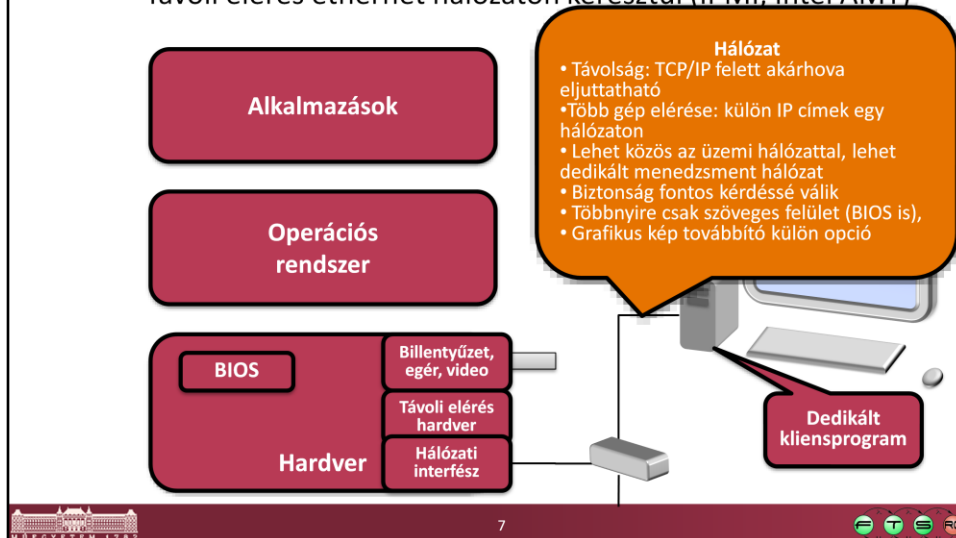
- Platform virtualizáción alapuló megoldások:
 - VMware ESX Server, ESXi – bare metal
 - VMware Server – hosted (de nem igazán komolyan vett termék...)
 - Xen – bare metal
 - Xen.org – bare metal
 - Citrix XenServer – bare metal
 - Oracle VM Server (Xen alapú) – bare metal
 - Microsoft Hyper-V – bare metal
 - IBM LPAR, DLPAR – bare metal
 - ...



Távoli hozzáférés megvalósítási lehetőségek

Beépített távoli menedzsment hardver

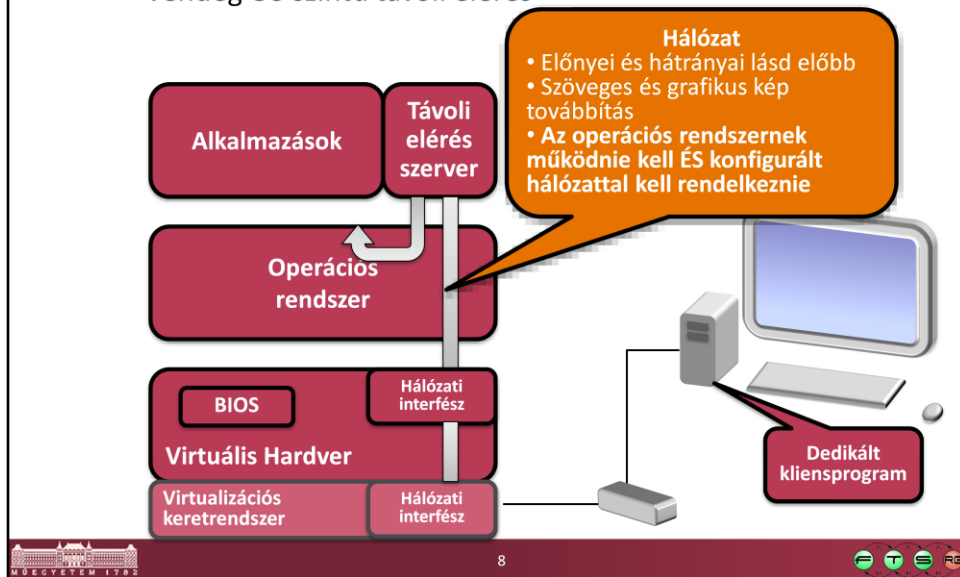
Távoli elérés ethernet hálózaton keresztül (IPMI, Intel AMT)



Szervereknél gyakori megoldás, hogy valamilyen dedikált hardver elemmel hálózatról elvégezhetővé tesznek olyan műveleteket, amik amúgy manuális hozzáférést igényelnének. Ez az igény a virtualizált környezetekkel szemben is megjelenik.

Operációs rendszer távoli elérése

Vendég OS szintű távoli elérés

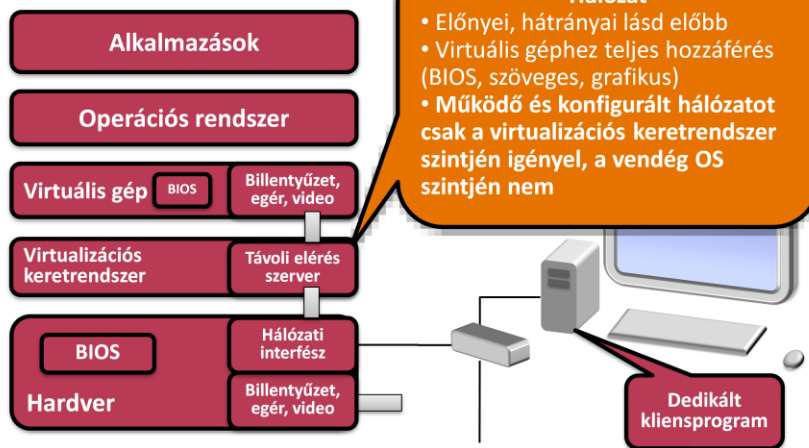


Ez ismerős, ha már az OS működik és konfigurált hálózata van, akkor lehet belépni SSH-val, VNC-vel, RDP-vel.

Az ábrát úgy kell nézni, hogy ez egy guest gép, ami magának biztosítja hozzáférést a beépített távoli elérés szerverével

Távoli hozzáférés megvalósítási lehetőségek

Virtuális gép virtuális konzolja



Bare-metal és hosted esetben is gyakran valamilyen távoli elérés szerver van a virtualizációs keretrendszerbe beépítve.

Nagy integrált rendszerek kiépítése

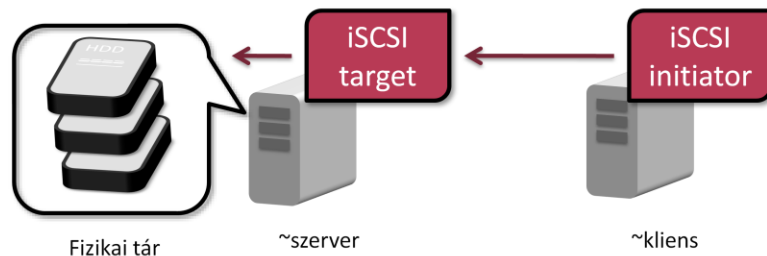
- Szerver virtualizációnál egy hoszt ritkán elég
- Jó, ha egyes erőforrások:
 - központilag kezeltek
 - átcsoportosíthatóak, hozzárendelések megváltoztathatóak
 - megoszthatóak, több helyről is egyformán elérhetőek
- Példák
 - háttértár: sok kicsi, gépekben elszórt merevlemez helyett egy-egy nagy storage box
 - hálózatok: minden hoszton egyformán elérhetőek legyenek
 - magasabb szinten: jogosultságkezelés, felhasználói fiókok címtára



Erről még később fogunk több szót ejteni.

Háttértár virtualizáció

- Elterjedt megoldások:
 - SAN – Storage Area Network, blokkos eszközt biztosít
 - NAS – Network Attached Storage, fájlrendszert biztosít
- iSCSI – egy elterjedt SAN megoldás: blokkos eszközök biztosítása TCP/IP hálózatok felett

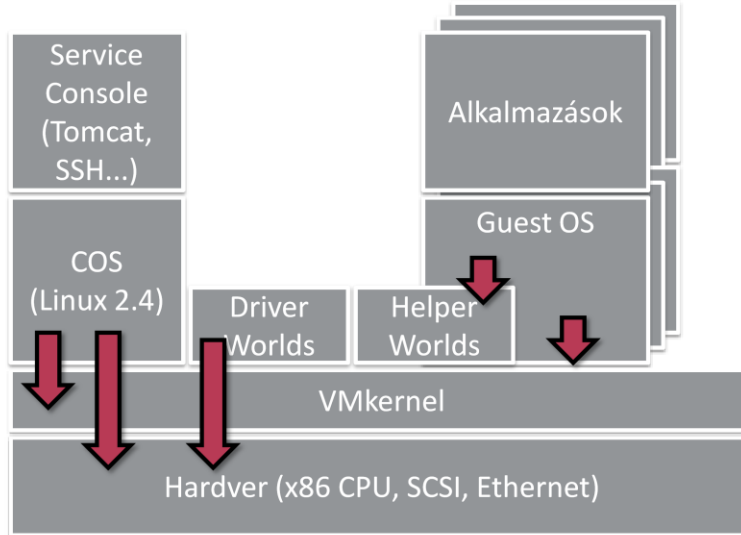


Az iscsi initatort futtató gép ugyanúgy tudja kezelni a virtuális tárat, mintha az egy lokális fizikai eszköz lenne. Egy target több különálló kötetet is szolgáltathat és egyszerre több initiatort is ki tud szolgálni (akár egy kötetet is többen használhatnak). iSCSI initiatorból van szoftveres implementáció és van hardveres is. Ez utóbbi bootolni is tud távoli eszközzel.

Tartalom

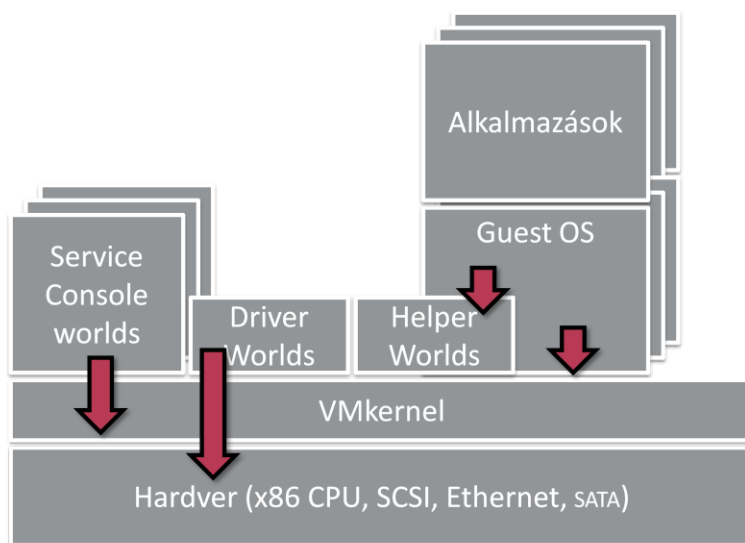
- Követelmények
- Szerver virtualizációs architektúrák
- Háttértárak virtualizációja
- **VMware ESX és ESXi szerver architektúrája**
- Microsoft Hyper-V architektúra
- Erőforrás gazdálkodás

VMware ESX Server architektúrája



Az ESX „hozthoz” való kapcsolódásnál valójában a service console-hoz kapcsolódunk, ami egy virtuális gép, bár kivételezett, mert sok menedzsment jellegű rendszerhívást tehet vmkernelbe, + néhány hardvert közvetlenül kezel (VGA, IDE diszkek, soros port). A vmkernel egy mikrokernel, driverek sincsenek benne, ezek is külön mintegy „virtuális gépben” futnak. A vmkernel felett futó tartományok neve „world”. Vannak segéd „helper” world-ök amik például a snapshot diszk image-ek kezelését végzik a vendég gépeknél.

VMware ESXi Server architektúrája



Az ESXi fő eltérése az ESX-hez képest: nincs külön Linux alapú console OS, a service console alkalmazásai (openwsman stb.) egy POSIX kompatibilitási könyvtáron keresztül a vmkernel, mint OS felett futnak. Ettől azért még továbbra is a bare-metal kategóriába sorolják, mert nem egy meglévő OS telepítés fölé kerül a virtualizációs keretrendszer, a service console world „appliance”-ként működik, nem lehet saját szoftvert telepíteni bele. Architektúráisan kicsit átmenetet képez a két megoldás között.

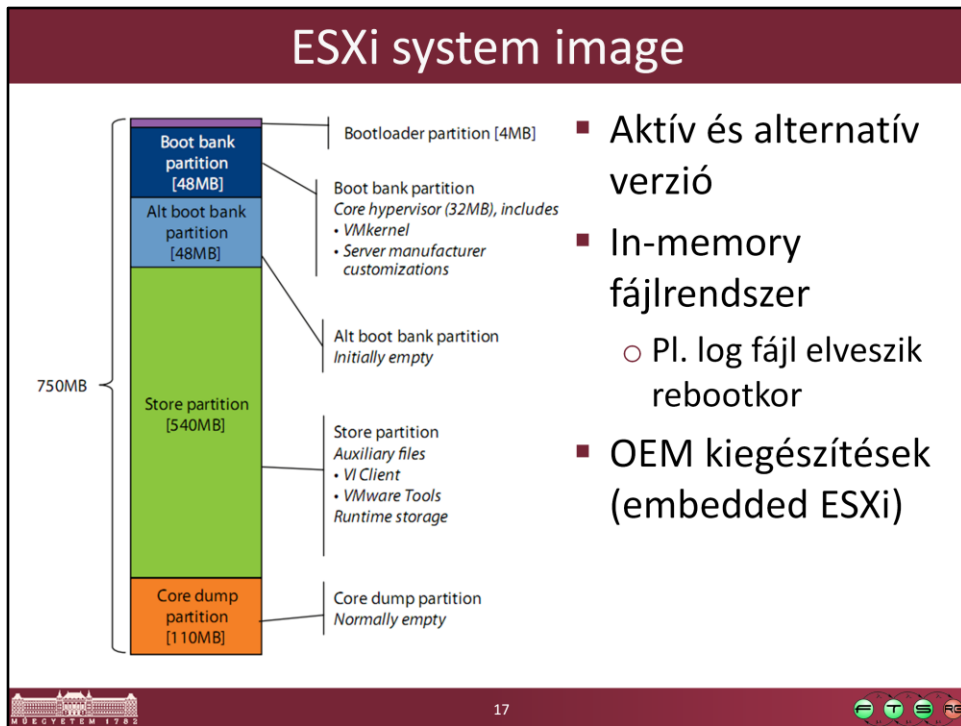
SSH hozzáférés megszűnt (még trükkösen engedélyezhető...). Most már nincs külön IP címe a Service Console-nak és a VMkernelnek. A Service console memóriaigénye kb. 32MB-ra csökkent a 256MB-ról, a diszkre telepített image 768MB, a korábbi ~5GB-hoz képest. Továbbá már nem a service console linux kernelje bootolja be „maga alá” a vmkernelt, így lehetővé vált a korábban problémás SATA eszközök használata is (hivatalosan ez még nem támogatott).

VMware ESX, ESXi – általános tulajdonságok

- Mindkét esetben saját eszközmeghajtókat igényel
 - emiatt elég szűk is a támogatott hardverek listája!
- A VMkernel egy mikrokernel, a driverek külön folyamatként futnak felette (ennek részben licenz okai is vannak, Linux kernelből vettek át meghajtókat)
 - Folyamat a VMware szóhasználatában: „world”
- Teljesen távoli elérésre alapozott rendszer
 - csak szöveges konzol van a lokális gépen,
 - de nem ezt ajánlott üzemszerűen használni, hanem a dedikált kliensprogramot (Virtual Infrastructure Client)
 - a vendégek konzoljához lokálisan egyáltalán nem is lehet hozzáférni
- A virtuális gépek formátuma (megkötésekkel) kompatibilis a Workstation/Server/Player formátumával
- Saját fájlrendszeren (VMFS3) tárolja a virtuális gépeket
 - Lehetőség van rá, hogy több ESX elosztottan használja ugyanazt a tárhelyet egyszerre!

VMware ESX, ESXi – eltérések

- Az ESX-ben Service Console egy teljes Linux-alapú rendszer
 - erősen módosított RedHat Enterprise Linux 3 ~ 4
 - Régi elavult, agyon módosított kernel (pl.: nincs SATA támogatás)
 - külön IP címmel
 - sok hardverhez dedikált közvetlen hozzáférése van
 - statikusan foglal sok memóriát
 - iszonyú bonyolult a boot folyamat, sérülékeny (előbb a Linux indul utána betölti a vmkernelt és végül paravirtualizáltan fut felette)
- ESXi-ben egy nagyon minimális service console van
 - nincs külön kernel, a folyamatok worldként futnak közvetlenül a vmkernel felett
 - kis memóriaigény
 - Nincs külön IP cím a Service Console és a VMkernel számára
 - Egyszerű bootolás, hálózatról is bootolható
- ESX-ben engedélyezett SSH hozzáférés, ESXi-ben nem
 - Helyette a Remote CLI-t javasolt használni a paracssori műveletekre



Forrás: VMware, The Architecture of VMware ESXi, White Paper, URL:
<http://www.vmware.com/resources/techresources/1009>

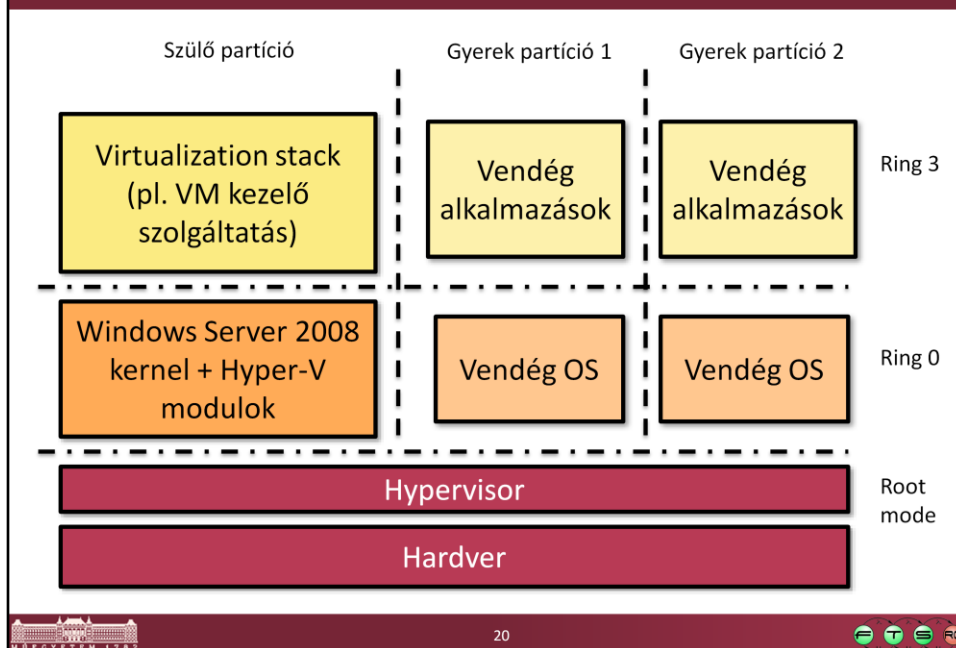
Tartalom

- Követelmények
- Szerver virtualizációs architektúrák
- Háttértárak virtualizációja
- VMware ESX és ESXi szerver architektúrája
- **Microsoft Hyper-V architektúra**
- Erőforrás gazdálkodás

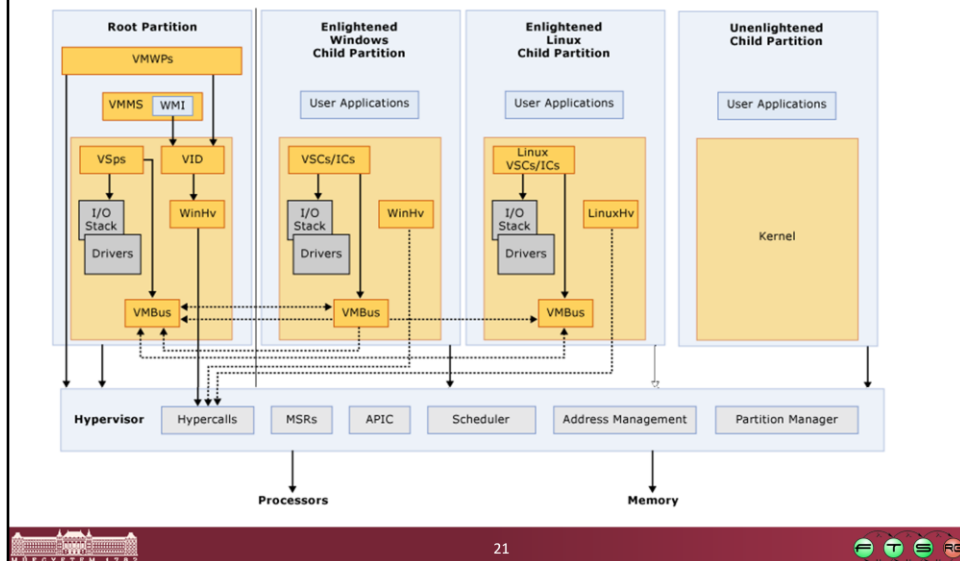
Microsoft Hyper-V

- Microsoft bare-metal virtualizációs megoldása
- Jelenleg: 2. verzió (R2)
- Két változat:
 - Windows Serverben a Hyper-V szerep
 - MS Hyper-V Server (különálló, ingyenes, csak Hyper-V)
- HW igény:
 - CPU: 64 bites, HW-es virtualizációs támogatás
- R2-be már bekerült a funkciók nagy része

Hyper-V architektúra I



Hyper-V architektúra II



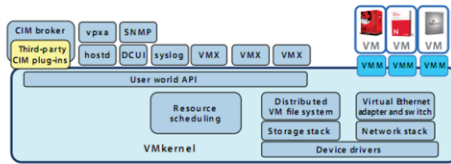
Forrás: Appendix B: Hyper-V Architecture and Feature Overview
[http://msdn.microsoft.com/en-us/library/dd722833\(BTS.10\).aspx](http://msdn.microsoft.com/en-us/library/dd722833(BTS.10).aspx)

„Acronyms and terms used in the diagram above are described below:

- **APIC** – Advanced Programmable Interrupt Controller. A device which allows priority levels to be assigned to its interrupt outputs.
- **Child Partition** – Partition that hosts a guest operating system. All access to physical memory and devices by a child partition is provided via the Virtual Machine Bus (VMBus) or the hypervisor.
- **Hypercall** – Interface for communication with the hypervisor. The hypercall interface accommodates access to the optimizations provided by the hypervisor.
- **Hypervisor** – A layer of software that sits between the hardware and one or more operating systems. Its primary job is to provide isolated execution environments called partitions. The hypervisor controls and arbitrates access to the underlying hardware.
- **IC** – Integration component. Component that allows child partitions to communicate with other partitions and the hypervisor.
- **I/O stack** – Input/output stack.
- **MSR** – Memory Service Routine.
- **Root Partition** – Manages machine-level functions such as device drivers, power management, and device hot addition/removal. The root (or parent) partition is the only partition that has direct access to physical memory and devices.
- **VID** – Virtualization Infrastructure Driver. Provides partition management services, virtual processor management services, and memory management services for partitions.
- **VMBus** – Channel-based communication mechanism used for inter-partition communication and device enumeration on systems with multiple active virtualized partitions. The VMBus is installed with Hyper-V Integration Services.
- **VMMS** – Virtual Machine Management Service. Responsible for managing the state of all virtual machines in child partitions.
- **VMWP** – Virtual Machine Worker Process. A user mode component of the virtualization stack. The worker process provides virtual machine management services from the Windows Server 2008 instance in the parent partition to the guest operating systems in the child partitions. The Virtual Machine Management Service spawns a separate worker process for each running virtual machine.
- **VSC** – Virtualization Service Client. A synthetic device instance that resides in a child partition. VSCs utilize hardware resources that are provided by Virtualization Service Providers (VSPs) in the parent partition. They communicate with the corresponding VSPs in the parent partition over the VMBus to satisfy a child partitions device I/O requests.
- **VSP** – Virtualization Service Provider. Resides in the root partition and provide synthetic device support to child partitions over the Virtual Machine Bus (VMBus).
- **WinHv** – Windows Hypervisor Interface Library. WinHv is essentially a bridge between a partitioned operating system's drivers and the hypervisor which allows drivers to call the hypervisor using standard Windows calling conventions.
- **WMI** – The Virtual Machine Management Service exposes a set of Windows Management Instrumentation (WMI)-based APIs for managing and controlling virtual machines.”

Bare metal megoldások architektúrái

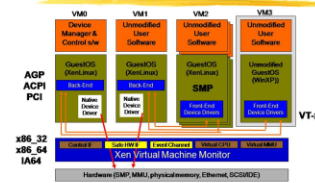
ESXi



- I/O eszközöket is a hypervisor kezeli
- Meghajtókat a VMware szállítja
- Extra kis méret: ESXi (32 MB)

Xen / Hyper-V

Xen 3.0 Architecture



- I/O eszközök kezelése a szülő partícióban
- Meghajtókat a HW gyártók szállítják



Tartalom

- Követelmények
- Szerver virtualizációs architektúrák
- Háttértárak virtualizációja
- VMware ESX és ESXi szerver architektúrája
- Microsoft Hyper-V architektúra
- **Erőforrás gazdálkodás**

Erőforrás gazdálkodás

- A virtuális gépek gyakran közös erőforráson osztoznak
- Jellemző példák:
 - Gyakran: minden virtuális gépnek virtuális CPU, összesen több vCPU, mint ahány fizikai CPU mag van
 - Ritkábban: összesen több kiosztott virtuális memória, mint amennyi fizikailag a hostban van
 - Háttértár helyfoglalás: ma még jellemzően statikusan kiosztott
 - Háttértár I/O műveletek: itt jellegzetesen osztozás van!
 - Hálózati áteresztőképesség: itt is osztozás van



Erőforrás gazdálkodás

- Versengés az erőforrásokért:
 - Kis terheléseknél ritka, hogy egyszerre több guest ugyanazt az erőforrást terhelné...
 - De szerverkörnyezetben gyakran előfordul, hogy valamelyik erőforrás szűk keresztmetszet lesz
 - A megfelelő ütemező elosztja a hozzáférést, de nem mindig megfelelően
- Cél:
 - A megosztott erőforrásokból való részesedést virtuális gépekre lebontva szabályozni tudjuk
 - Kemény korlátozások
 - „Lágy” korlátok, prioritások

Erőforrás gazdálkodás

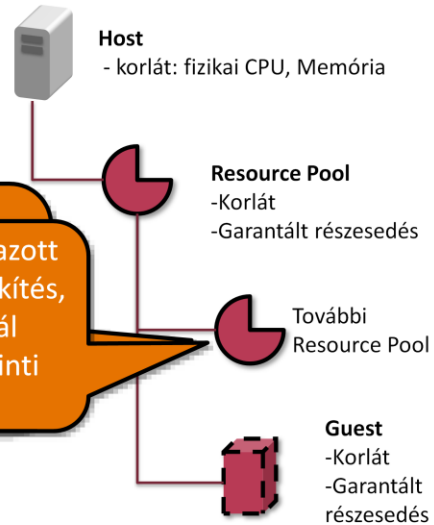
- VMware ESX/ESXi esetén 3 beállítási lehetőség:
 - **Resource Limit** – kemény felső korlát az erőforrás igénybevételére
 - Akkor is érvényes, ha egyébként van szabad erőforrás
 - **Resource Reservation** – garantált rendelkezésre álló erőforrás mennyiség
 - Nem feltétlenül használja ki, csak verseny esetén érvényesül, egyébként a keretet más használhatja
 - **Resource Shares** – prioritás
 - Verseny esetén az alapértelmezett „igazságos” elosztás módosítható ezzel

Erőforrás gazdálkodás

■ Hierarchikus erőforráskezelés

- Nemcsak virtuális gépek szintjén lehet korlátozni
- Pool-okba szervezhetők a VM-ek
- Használati esetek
 - Egy felhasználó gépére egy köz...
 - Egy feladatot e csoportjára ko...
 - Kritikus/nem kritikus alkalmazások csoportosítása

Egymásba ágyazott korlátoknál szűkítés, konfliktusnál prioritás szerinti feloldás



Erőforrás gazdálkodás

■ Speciális trükkök

- Memória működés közbeni hozzáadása/elvétele
 - Normális esetben memória beállítás működő VM mellett nem módosítható
 - Előre allokalált nagy memória + korlátozás
 - A változtatások guest újraindítás nélkül is érvényre jutnak (memóriánál néhány percet igénybe vehet...)
 - Fontos a memory ballooning driver a guestben
- Lassú hálózati kapcsolat szimulációja
 - Áteresztőképességet jól szimulálja
 - Válaszidők módosítására nincs lehetőség
- „Maradék” CPU idő kihasználása
 - Hosszú futású (batch) feladatot végző VM alacsony prioritással
 - Nem zavarja a többi gépet... (de a gyakorlatban sajnos igen...)



Erőforrás gazdálkodás

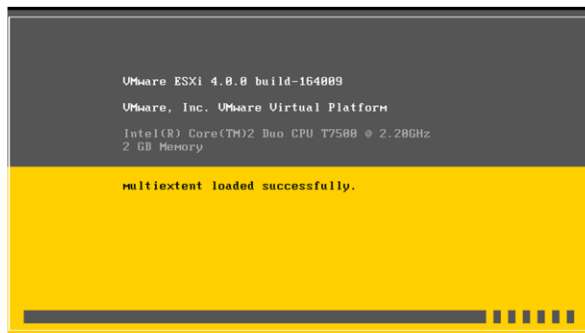
- CPU-k számának változtatására ez nem praktikus
 - Vendég OS ütemező nem tud a korlátozásról, az „ellopott” időt nem tartja számon (elvileg paravirtualizált esetben akár tudhatná, de gyakorlatban nem igazán működik...)
 - A durvább felbontású időosztás miatt a guest korlátozott esetben is a teljes CPU sebességet „érezni” fogja, de csak rövid időkre

Összefoglalás

- Követelmények
 - Sok guest, jó skálázhatóság, távoli elérés
- Szerver virtualizációs architektúrák
 - Dominánsan bare-metal
- Háttértárak virtualizációja
 - iSCSI SAN
- VMware ESX és ESXi szerver architektúrája
 - Mikrokernel, eltérés a Service Console megvalósításában
- Hyper-V architektúra

DEMO Gyakorlat: ESXi használata

- Indítás: alap konfiguráció (IP, jelszó) beállítása
- Csatlakozás VI Clienttel
- iSCSI storage beállítása datastore-nak



DEMO Hyper-V

- Hyper-V Server felülete (Server Core)
- Távoli menedzsment:
 - Server Manager, Hyper-V Manager
- Eszközkezelő: szintetikus meghajtók
- Virtuális gép létrehozása, indítása

További információ

- Microsoft Hyper-V hypercall leírás (WDK)
<http://msdn.microsoft.com/en-us/library/aa973529.aspx>
- Hyper-V Architecture and Feature Overview
[http://msdn.microsoft.com/en-us/library/dd722833\(BTS.10\).aspx](http://msdn.microsoft.com/en-us/library/dd722833(BTS.10).aspx)
- XenServer: Why?
<http://it20.info/blogs/main/archive/2009/10/29/1422.aspx>