Computação na Nuvem: Virtualização e Migração de VM

André Meireles

Estêvão Monteiro

Monique Soares

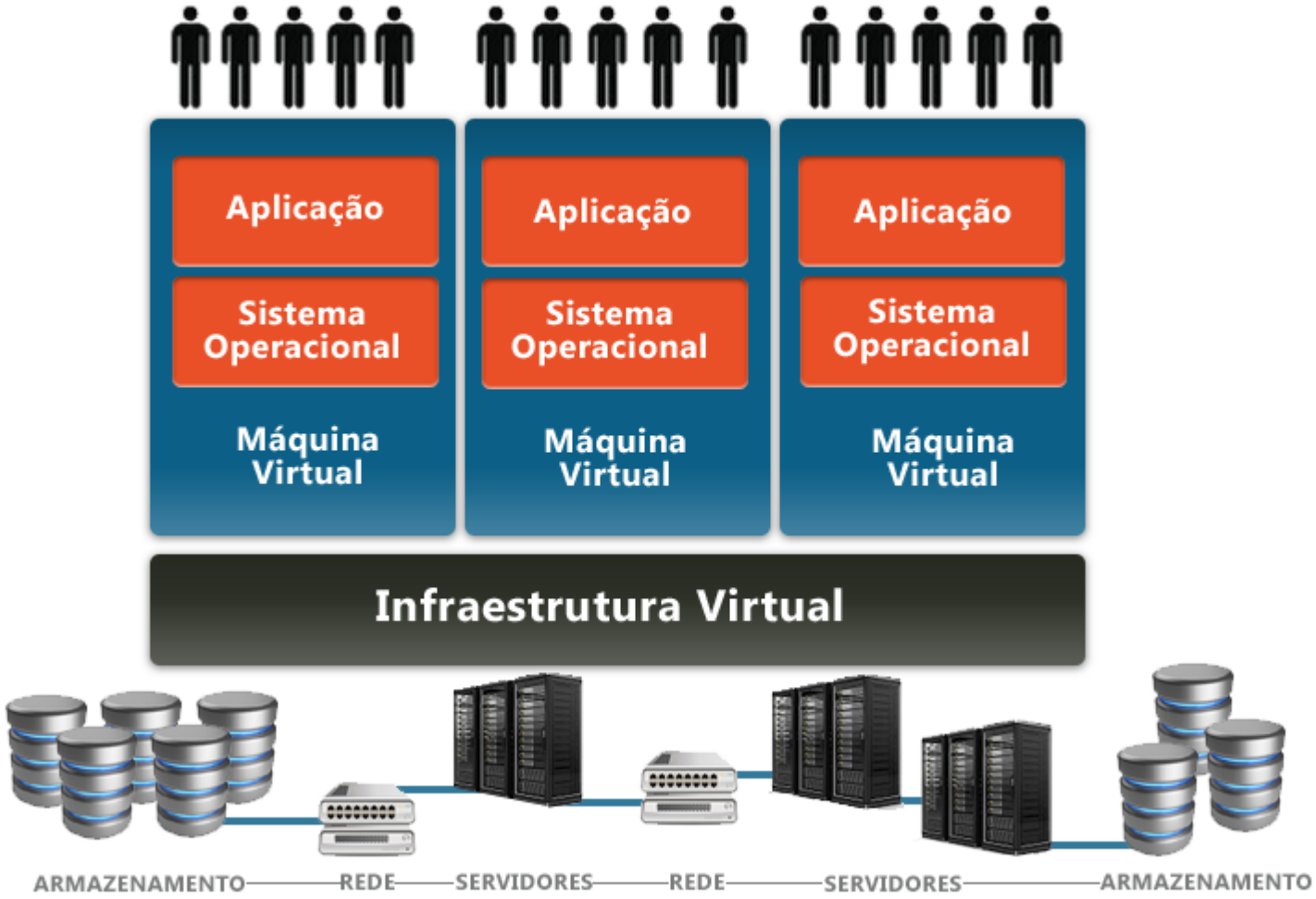
© TemplatesWise.com

Agenda

- Overview
- Histórico
- Abordagens
- Desafios em x86
- Snapshots

Virtualização

- Virtualização de Hardware/Plataforma/Sevidor:
  - Máquina virtual
    - Imagem do hardware com seu próprio SO
  - Host X Guest
  - Múltiplas MVs sobre uma máquina física
  - Multiplexação de recursos físicos
  - Qualquer combinação de SOs



http://blog.opus-software.com.br/?p=1065

Virtualização

- Vantagens:
  - Segurança;
  - Separação limpa das tarefas;
  - Alocação de usuários;
  - Cobrança aos usuários;
  - Otimização do aproveitamento dos recursos físicos*;
  - Consolidação de servidores;
  - Abstração da localização física: migração;
  - Recuperação de desastres.

*Bem feito, compensa o overhead!

Virtualização

- Casos de uso
  - Uma VM por serviço, que pode ser facilmente migrada entre servidores
  - Cloud computing:
    - IaaS, PaaS, SaaS...
  - Aplicações distribuídas que criam e destroem VMs
    - Jogos online multi-jogador, aplicações multimídia distribuídas.
  - Múltiplos SOs diferentes num mesmo desktop
    - Ensino e desenvolvimento de software.

Virtualização

- Casos de uso
  - Uma VM por serviço, que pode ser facilmente migrada entre servidores
  - Cloud computing:
    - IaaS, PaaS, SaaS...
  - Aplicações distribuídas que criam e destroem VMs
    - Jogos online multi-jogador, aplicações multimídia distribuídas.
  - Múltiplos SOs diferentes num mesmo desktop
    - Ensino e desenvolvimento de software.

Virtualização

- Casos de uso
  - Uma VM por serviço, que pode ser facilmente migrada entre servidores
  - Cloud computing:
    - IaaS, PaaS, SaaS...
  - Aplicações distribuídas que criam e destroem VMs
    - Jogos online multi-jogador, aplicações multimídia distribuídas.
  - Múltiplos SOs diferentes num mesmo desktop
    - Ensino e desenvolvimento de software.

Virtualização

- Casos de uso
  - Uma VM por serviço, que pode ser facilmente migrada entre servidores
  - Cloud computing:
    - IaaS, PaaS, SaaS...
  - Aplicações distribuídas que criam e destroem VMs
    - Jogos online multi-jogador, aplicações multimídia distribuídas.
  - Múltiplos SOs diferentes num mesmo desktop
    - Ensino e desenvolvimento de software.

Virtualização

- Requisitos essenciais para um VMM [Popek & Goldberg 1974]
  - Fidelidade
  - Desempenho
  - Segurança

Virtualização

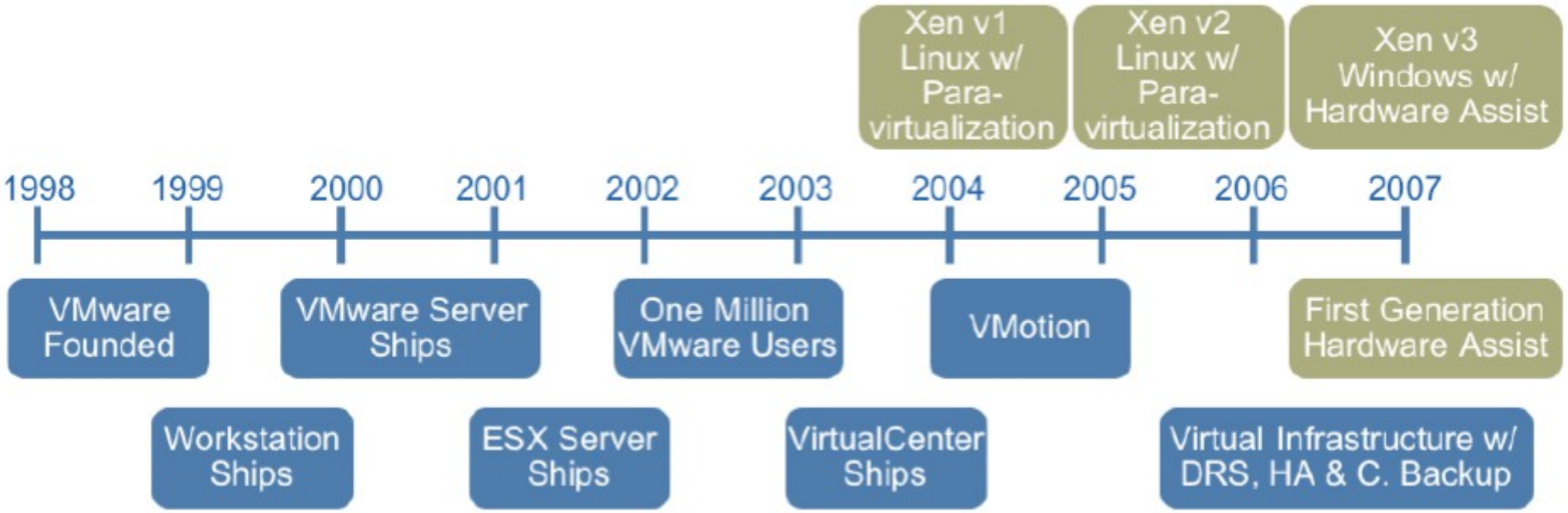
- Precursor: IBM M44/44X (1964)
  - O objetivo do projeto M44/44X era avaliar os conceitos de sistema de compartilhamento de tempo, então emergente;
  - A arquitetura foi baseada em máquinas virtuais
    - Principal máquina foi uma IBM 7044 (M44) e cada máquina virtual foi uma imagem experimental da máquina principal (44X).

Virtualização

- Connectix Virtual PC – 1997
  - Emulação de SO!
  - Host: Mac OS 8.5+ com CPU G3
  - Guests: Windows 95, OS/2, Red Hat Linux
  - Comprado pela Microsoft em 2003
  - Apenas em 2007 implementou virtualização de hardware

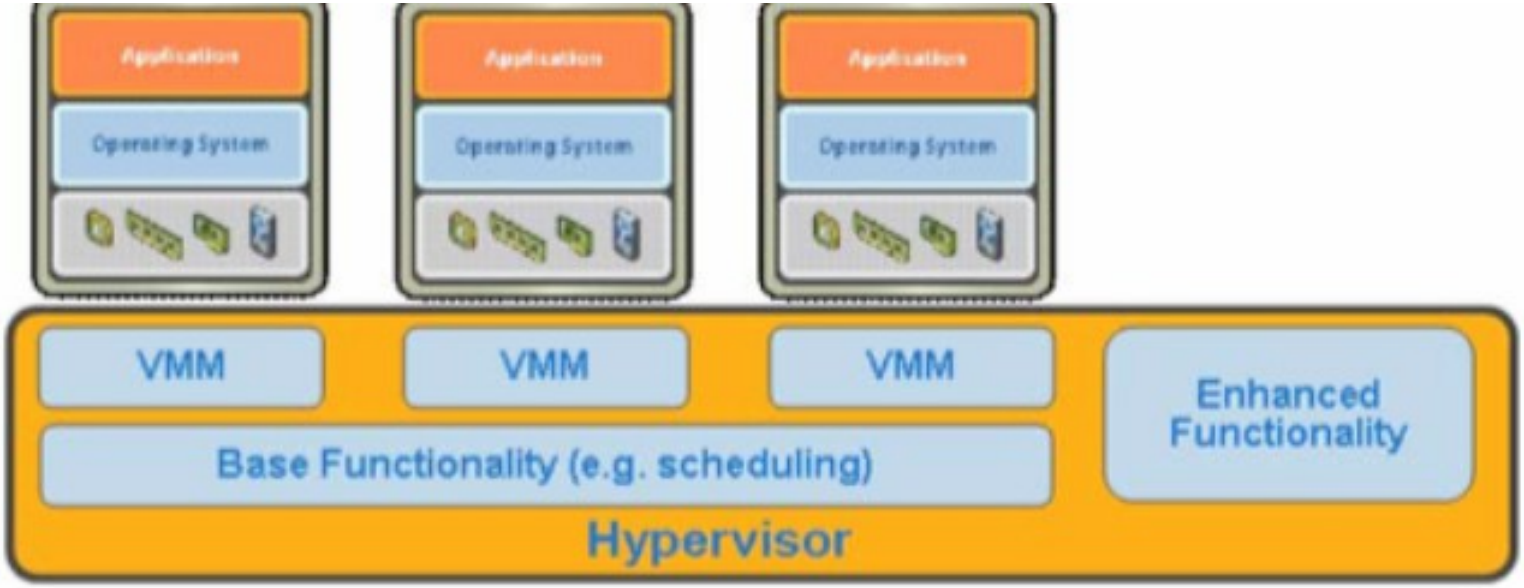
Virtualização

- VMware - 1998 (publicado em 1999)
  - Pioneiro na virtualização de sistemas x86-32/IA-32/i386
  - Tradução binária
  - Execução direta
  - SOs virtualizados rodam plenamente isolados


Xen v1 Linux w/ Para-virtualization
Xen v2 Linux w/ Para-virtualization
Xen v3 Windows w/ Hardware Assist
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
VMware Founded
VMware Server Ships
One Million VMware Users
VMotion
First Generation Hardware Assist
Workstation Ships
ESX Server Ships
VirtualCenter Ships
Virtual Infrastructure w/ DRS, HA & C. Backup

Virtualização

- Hosted x Native
- Hosted
  - Hypervisor é uma aplicação sobre um SO
  - Conveniente para usuários
- Native (bare-metal)
  - Hypervisor roda diretamente sobre o hardware
  - Ideal para servidores e centros de dados


Application
Operating System
Application
Operating System
Application
Operating System
VMM
VMM
VMM
Enhanced Functionality
Base Functionality (e.g. scheduling)
Hypervisor

Virtualização

- Paravirtualização
  - Usa interface modificada para melhor desempenho
  - Denali (2001)
  - Xen (2003)
  - VMWare (2005-2009)

Virtualização

- Virtualização Total
  - Monitor de VMs / Hipervisor
  - Desempenho era deficiente em arquiteturas x86, entre outras
  - Deficiência mitigada por Intel VT-x (2005) e AMD-V (2006)

Virtualização

- Virtualização Total
  - Microsoft Virtual PC (1997/2006/2009)
  - VMWare (1999)
  - Microsoft Hyper-V (2003/2008)
  - Xen (2003) também!
  - Parallels (2005)
  - Linux KVM (2007)
  - Virtualbox (2007)

Virtualização

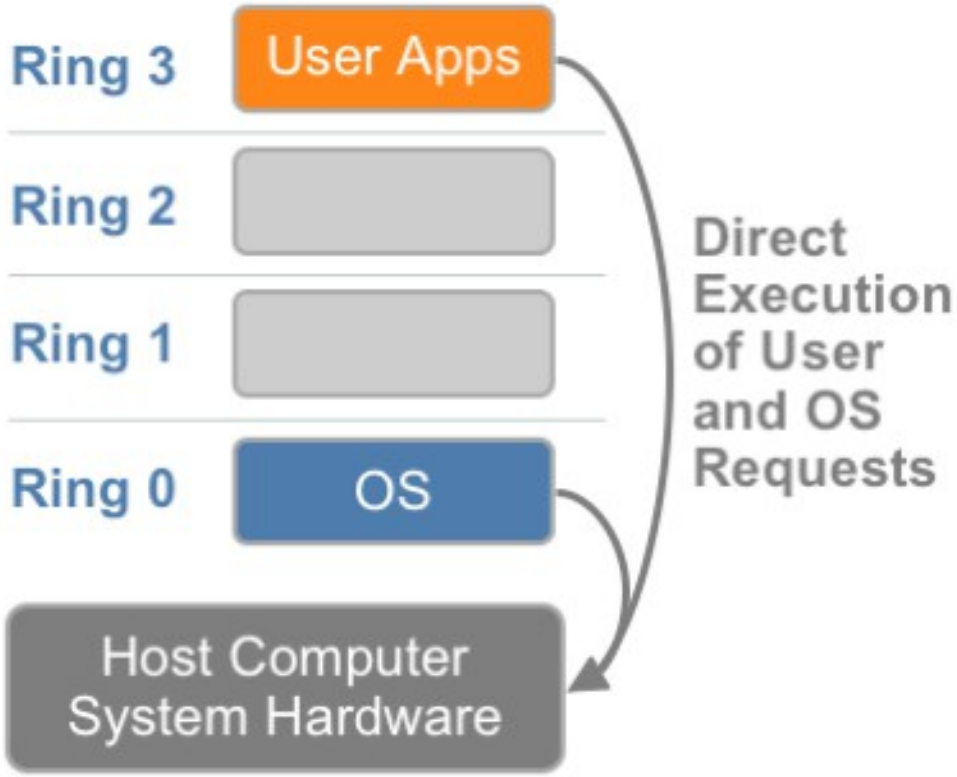
- Host costuma impor certas restrições de acesso aos recursos físicos
  - Oculta periféricos específicos;
  - Limita capacidades de dispositivos.

Virtualização

- Comparação com Microkernel
  - MK emula SO sobre seus módulos;
  - VMs fazem acesso direto a hardware;
  - Na VM, aplicações não precisam ser re-compiladas.

Virtualização

- Desafios da virtualização de hardware x86
  - SO precisa executar instruções no anel 0
  - Aplicações rodam no anel 3
  - Semântica de instruções fora do anel 0 pode ser diferente


Ring 3
User Apps
Ring 2
Ring 1
Ring 0
OS
Host Computer
System Hardware
Direct
Execution
of User
and OS
Requests

Virtualização

- Kernel mode vs. User mode
  - Kernel mode - instruções sensíveis:
    - Entrada e saída;
    - Configuram MMU.
- Instruções privilegiadas provocam "trap" quando em modo usuário.

Virtualização

- Virtualização requer que instruções sensíveis sejam instruções privilegiadas (provoquem trap) [Popek & Golberg 1974].
- Em x86:
  - Instruções sensíveis em modo usuário são ignoradas;
  - O programa pode detectar o modo em que está sendo executado:
    - SO pode tomar decisões indevidas quando souber que está em modo usuário.

Virtualização

- Intel VT-x (2005) e AMD-V (2006) permitem rodar máquinas virtuais em containers capazes de provocar traps, permitindo emulação das instruções privilegiadas
  - 9 novas instruções
  - Geram traps demais, que arruinam cache de CPU, TLB* e tabelas de predição de ramificações.

* Tradutor entre endereçamento de memória lógico e físico.

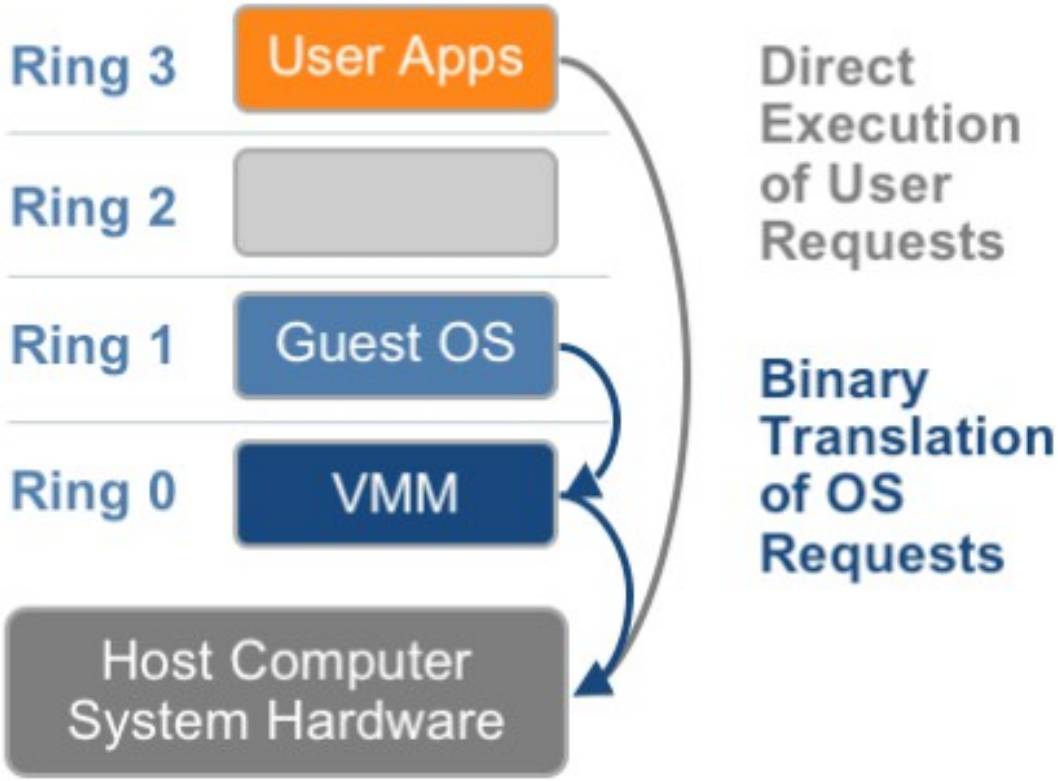
Virtualização

- Tradução binária

  Ao executar código, o hipervisor:

  1) identifica os blocos básicos*;
  2) substitui instruções sensíveis por hiperchamadas**;
  3) substitui também a última instrução (a de fluxo de controle);
  4) grava a instrução modificada (emulada) em cache;
  5) executa a instrução modificada;
  6) repete até concluir o programa.

* Sequências de instruções que terminam em jump, call, trap etc. (fluxo de controle).
** Se não houver, executa nativamente. Se já estiver em cache, executa a partir dela. É feita apenas para o sistema operacional: programas de usuário são ignorados.

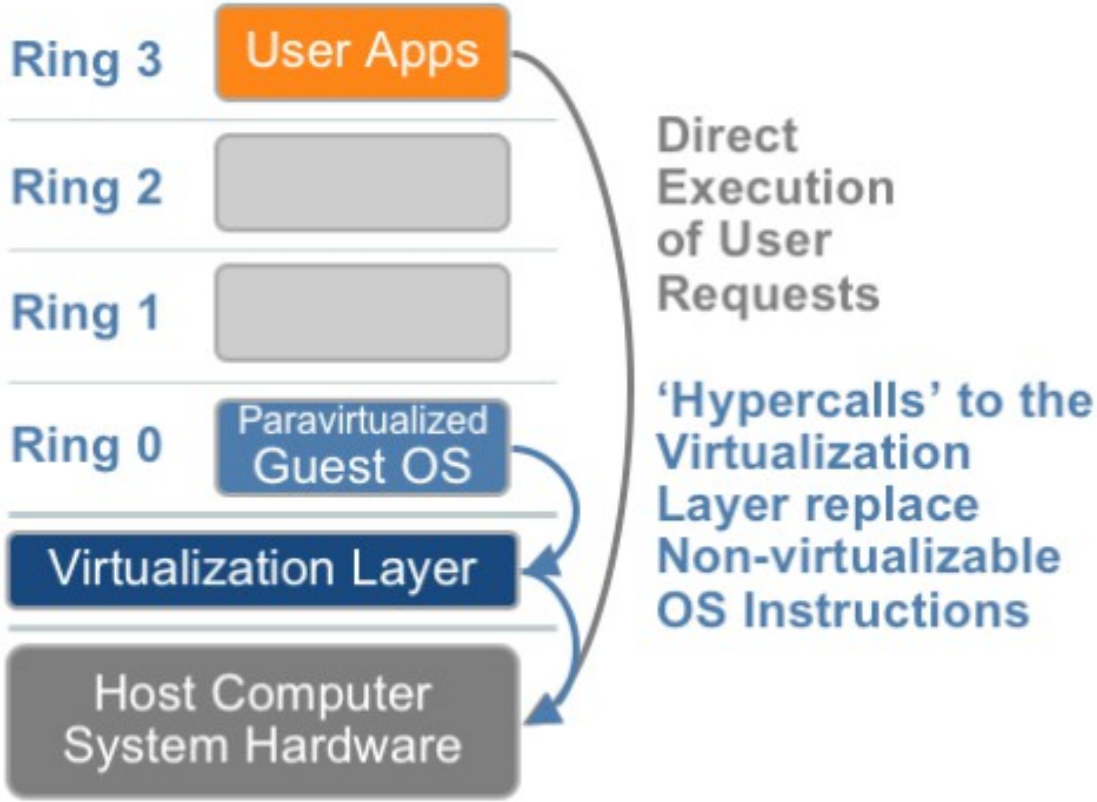

Ring 3
User Apps
Ring 2
Ring 1
Guest OS
Ring 0
VMM
Host Computer System Hardware
Direct Execution of User Requests
Binary Translation of OS Requests

Virtualização

- Paravirtualização
  - SO guest é modificado para fazer hypercalls
  - Hypercall: chamada ao hypervisor pelo kernel guest
  - Hypervisor provê interfaces para hypercalls
    - Gerenciamento de memória
    - Tratamento de interrupções
    - Relógio

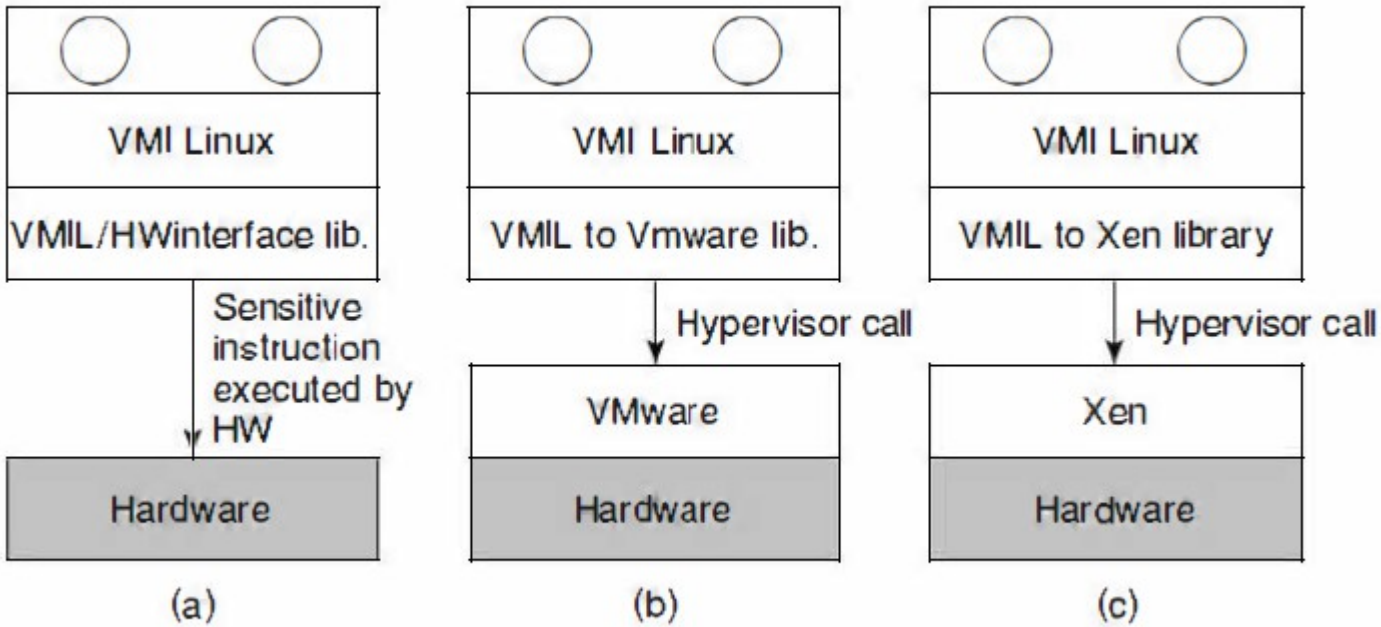
Virtualização

- Paravirtualização
  - Reduz overhead
  - Mais simples de implementar do que tradução binária
  - Baixa compatibilidade e portabilidade
  - Exemplo: Xen (2003)


Ring 3
User Apps
Ring 2
Ring 1
Ring 0
Paravirtualized
Guest OS
Virtualization Layer
Host Computer
System Hardware
Direct
Execution
of User
Requests
'Hypercalls' to the
Virtualization
Layer replace
Non-virtualizable
OS Instructions

Virtualização

- Paravirtualização
  - Hipervisor funciona como microkernel;
  - Uma API de paravirtualização padrão universal a tornaria transparente;
  - Virtual Machine Interface - VMI [Amsden et al. 2006, VMware];
  - Virtual Machine Disk Format – VMDK;
  - Open Virtualization Format - OVF: distribuição de pacotes de VMs;
  - Paravirt-ops: componente do kernel Linux, suporta VMI, Lguest e Xen.


VMI Linux
VMIL/HWinterface lib.
Sensitive instruction executed by HW
Hardware
(a)
VMI Linux
VMIL to Vmware lib.
Hypervisor call
VMware
Hardware
(b)
VMI Linux
VMIL to Xen library
Hypervisor call
Xen
Hardware
(c)

Virtualização

- Otimização de virtualização total:
  - São comuns adendos mínimos ao SO, não-intrusivos, que não afetam o kernel;
  - Adendos usam APIs do SO para implementar backdoors ao hypervisor;
  - São chamados serviços e drivers "paravirtualizados";
  - Partilham estruturas de dados comuns com o hypervisor;
  - Melhor vazão de processamento (throughput);
  - Menor carga à CPU;
  - Ex.: VMware Tools.

Virtualização

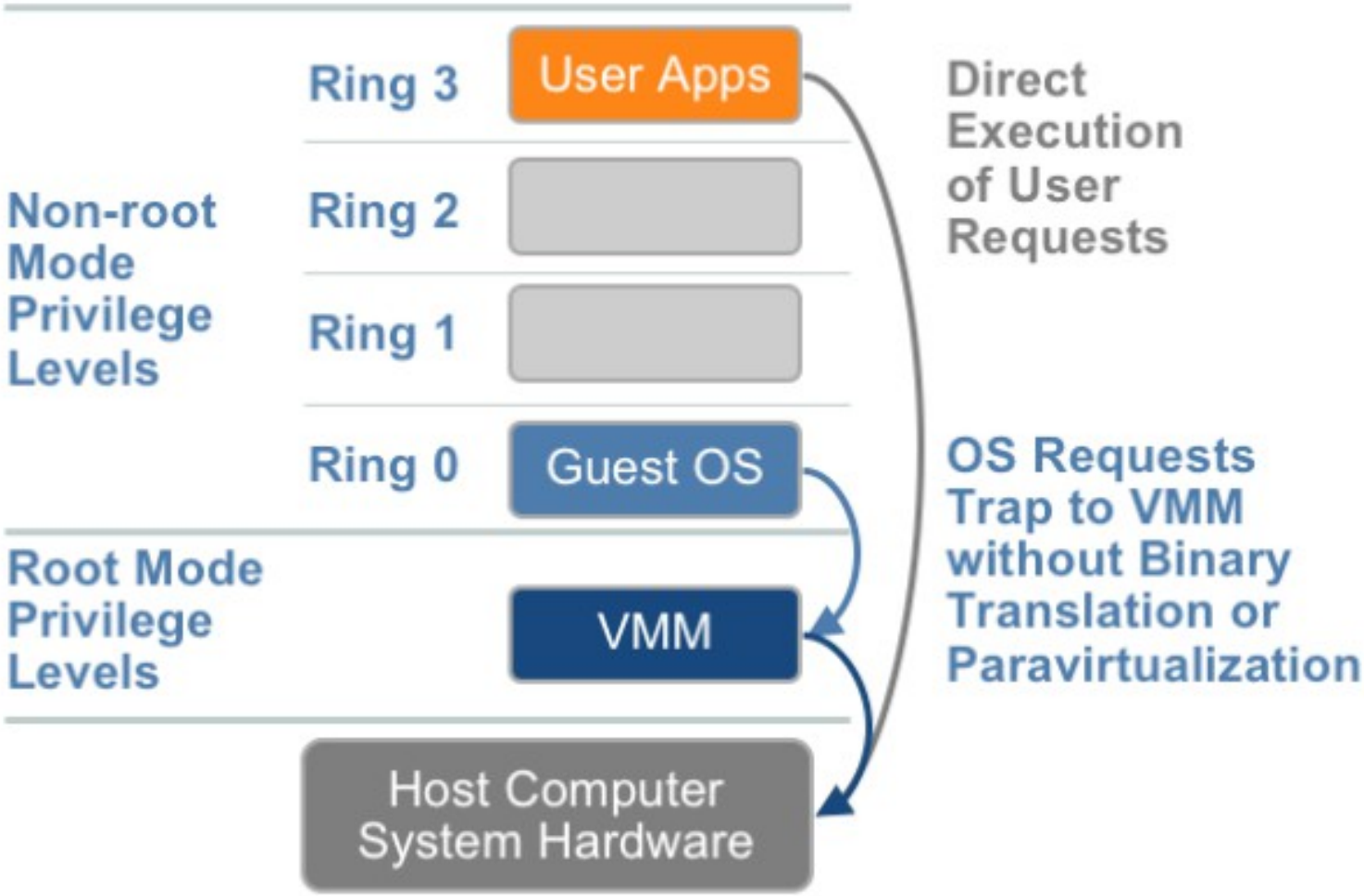
- Serviços "paravirtualizados":
  - Comunicação de mensagens entre hóspede e hospedeiro;
  - Sincronização do relógio
  - Ajuste imediato da resolução do hóspede de acordo com o tamanho da janela;
  - Cursor transparente entre hóspede e hospedeiro (captura desnecessária);
  - Copiar e colar texto e arquivos entre hóspede e hospedeiro;
  - Scripts para automatizar o hóspede.

Virtualização

- Drivers "paravirtualizados":
  - SVGA em vez de VGA: 32 bits, resoluções altas, aceleração de vídeo (Windows Aero, jogos, DXVA etc.);
  - SCSI;
  - VMXNet NIC;
  - Mouse;
  - Áudio;
  - Compartilhamento de pastas;
  - Drag'n'drop;
  - ThinPrint.

Virtualização

- Virtualização assistida por hardware
  - Intel VT-x (2005) e AMD-V (2006)
  - Root mode: VMM roda abaixo do anel 0
  - Trap-and-emulate
  - Virtual Machine Control Structures/Blocks: estado da VM
  - Primeira geração perde para tradução binária [Adams & Agesen 2006]
  - Desempenho pode ser melhorado com drivers paravirtualizados


Ring 3
User Apps
Ring 2
Ring 1
Ring 0
Guest OS
Non-root Mode Privilege Levels
Root Mode Privilege Levels
VMM
Host Computer System Hardware
Direct Execution of User Requests
OS Requests Trap to VMM without Binary Translation or Paravirtualization

Virtualização

- Hardware Assist - 1ª Geração
  - Modelo de programação rígida;
  - Grande overhead nas transições de hypervisor para guest.
- Hardware Assist - 2ª Geração
  - Redução do overhead nas operações de memória.
- Hardware Assist - 3ª Geração: ?

Virtualização

- Snapshots
  - Aproveita discos virtuais com camada de tradução;
  - Captura o estado do disco virtual;
  - Pode capturar o estado de memória da VM;
  - Idealmente acumulativo (versões diferenciais);
  - Permite migração/teleportação e failover;
  - Snapshots periódicos promovem rapidez nessas operações;
  - Mesclagens periódicas otimizam a manutenção.

Referências

- G. Coulouris, J. Dollimore and T. Kinberg, Distributed Systems: Concepts and Design, Edition 3, Addison-Wesley 2001.
- http://www.docstoc.com/docs/83094767/Virtualization
- http://blog.opus-software.com.br/?p=1065
- POPEK, G. J., AND GOLDBERG, R. P. Formal requirements for virtualizable third generation architectures.
- http://www.vmware.com/files/pdf/VMware_paravirtualization.pdf