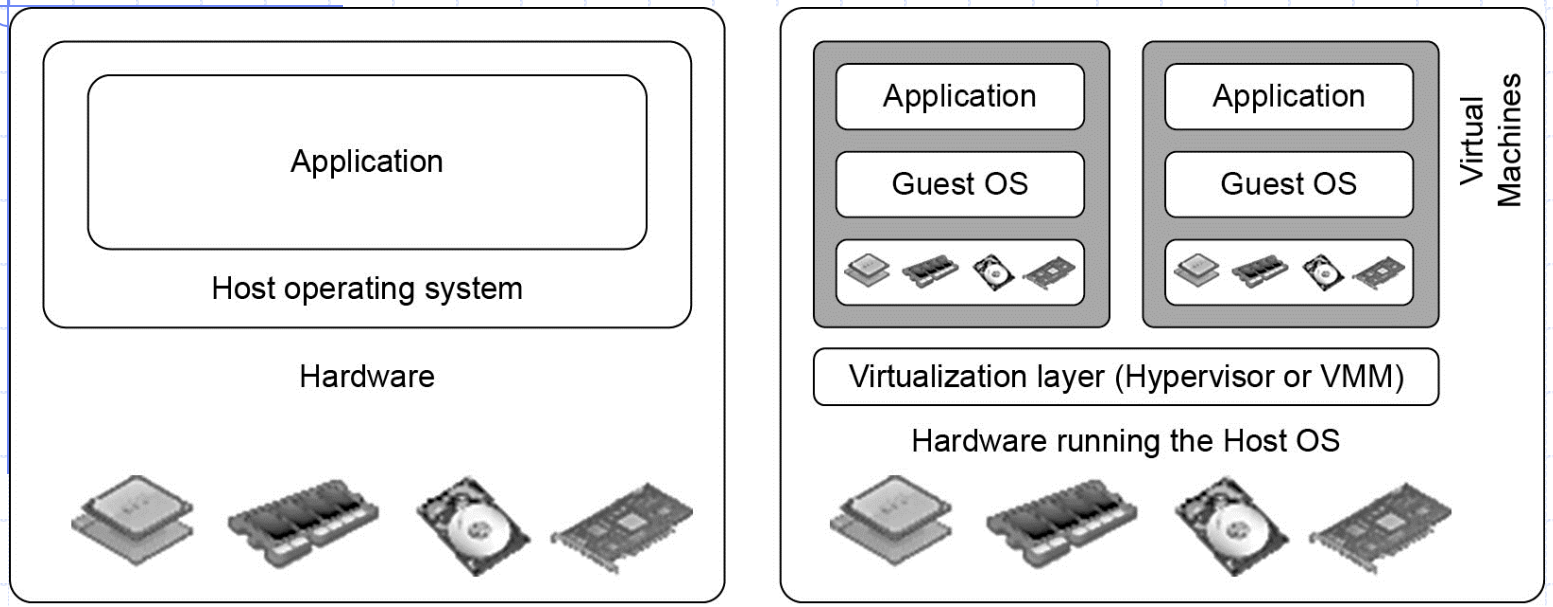
Virtuelne mašine i virtuelizacija klastera i centara za podatke

- VM, VMM
- Nivoi i vrste virtuelizacije
- Virtuelni klasteri
- Privatni oblaci

Virtuelizacija Centara za podatke (eng. data center)

- Zahtevi virtuelizacije u centrima za podatke:
  - Konsolidacija servera
  - Rezervisanje i oslobađanje virtuelnog skladišta
  - Operativni sistemi oblaka za virtuelne centre
  - Rukovanje poverenjem (Trust)

Razlika između tradicionalnog računara i Virtuelne Mašine (VM)



(a) Traditional computer

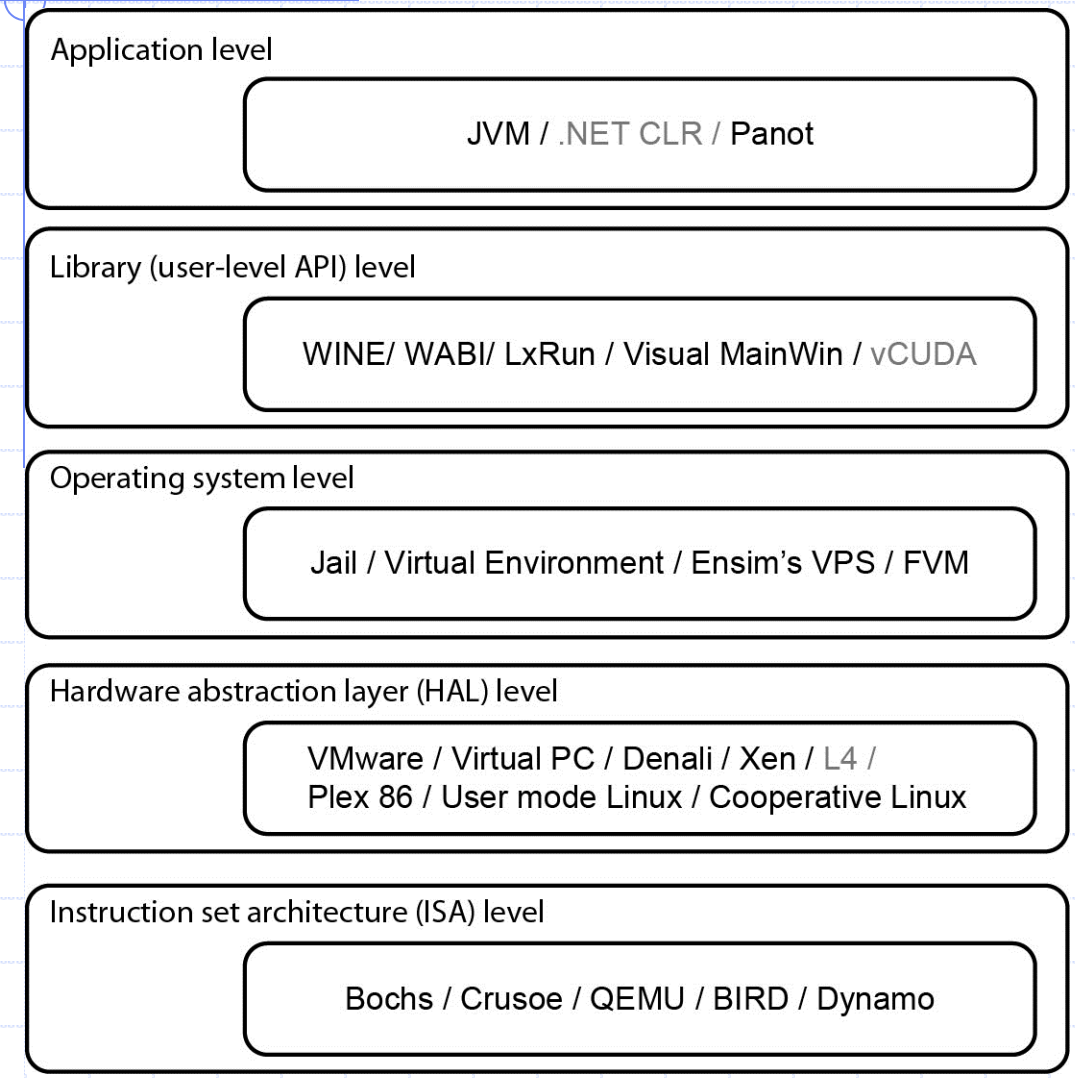
(b) After virtualization

- Osnovna razlika:
  - OS na realnom računaru radi nad realnim resursima
  - Gostujući OS na VM radi nad virtuelnim resursima

Virtuelna Mašina, Gostujući OS, i Monitor Virtuelne Mašine (VMM)

- Virtuelna mašina (VM):
  - je operativno okruženje na kom može da se izvršava Gostujući OS
- Gostujući OS:
  - je OS koji se izvršava u okruženju VM, a koji inače može da se izvršava na zasebnoj fizičkoj mašini
- VMM ili hipervizor, ili sloj za virtuelizaciju:
  - je midlver između HW i virtuelnih mašina, koje su predstavljene u sistemu

Virtuelizacija na 5 nivoa apstrakcije



- Nivo aplikacije
- Nivo biblioteke (nivo korisničkog API)
- Nivo OS-a
- Nivo apstrakcije harvdera (HAL)
- Nivo skupa instrukcija (ISA)

Virtuelizacija na nivou ISA

- Primer:
  - ISA emulacija MIPS mašinskog koda na X-86
- Prednosti:
  - Izvršenje postojećih mašinskih prog. na bilo kom domaćinu sa odgovarajućim ISA emulatorom
  - Najbolja aplikaciona fleksibilnost
- Nedostatci i ograničenja:
  - Jedna izvorna inst. može zahtevati 10-ne ili 100-ne inst. radi emulacije, što može biti sporo
  - V-ISA zahteva dodavanje posebne komponente u kompajleru koja zavisi od tipa emuliranog procesora

Virtuelizacija na nivou apstrakcije hardvera (HAL)

- Virtuelizacija se obavlja odmah iznad hardvera
  - Ona stvara okruženje virtuelnog harvera za VM-ne
- Primeri:
  - VMware, Virtual PC, Denali, Xen
- Prednosti:
  - Ima bolju performansu i dobru izolaciju aplikacija
- Nedostatci i ograničenja:
  - Vrlo skupa implementacija (složenost)

Virtuelizacija na nivou operativnog sistema (OS)

- Sloj između OS i aplikacija
  - Stvara instance OS za korisnike u centrima za podatke
- Primeri:
  - Jail, Virtual Environment, Ensim-ov VPS, FVM
- Prednosti:
  - Min cena pokretanja, troši min resursa, skalabilnost, sinhronizuje izmene stanja VM i domaćina
- Nedostatci i ograničenja:
  - Sve VM na nivou OS moraju imati istu vrstu gost. OS
  - Slaba aplikaciona fleksibilnost i izolacija

Virtuelizacija za Linux i Windows NT platforme

- Najviše virtuelizacionih sistema na nivou OS su na bazi Linux-a
  - Nivo apstrakcije jezgra Linux omogućava procesima da rade sa resursima bez poznavanja HW detalja
  - Za podršku novom HW može biti potrebno novo jezgro
  - Zato razne Linux platforme koriste zakrpe (patchs) jezgra za podršku proširene funkcionalnost
- Primeri:
  - Linux vServer, OpenVZ (Linux), FVM (WindowsNT)

Virtuelizacija na nivou biblioteke

- Stvara okruženje za izvršenje stranih programa na datoj platformi
  - Umesto da stvara VM radi izvršenja celog OS
  - To se radi presretanjem i preslikavanjem API poziva
- Primeri:
  - Wine, WAB, LxRun, VisualMainWin
- Prednosti:
  - Vrlo mali implementacioni napor
- Nedostatci i ograničenja:
  - Slaba aplikaciona fleksibilnost i izolacija

Rešenja za Virtuelizaciju na nivou midlvera/biblioteke

- WABI
  - Midlver koji konvertuje Windows pozive u Solaris
- Lxrun
  - Emulator koji izvršava Linux programe na UNIX
- WINE
  - Emulator za izvršenje Windows na Linux, Solaris, itd.
- Visual MainWin
  - Okruženje za razvoj programa u Visual Studio za Linux, Solaris i AIX
- vCUDA
  - VM za izvršenje gost. OS na GPU

vCUDA arhitektura za virtuelizaciju GPU

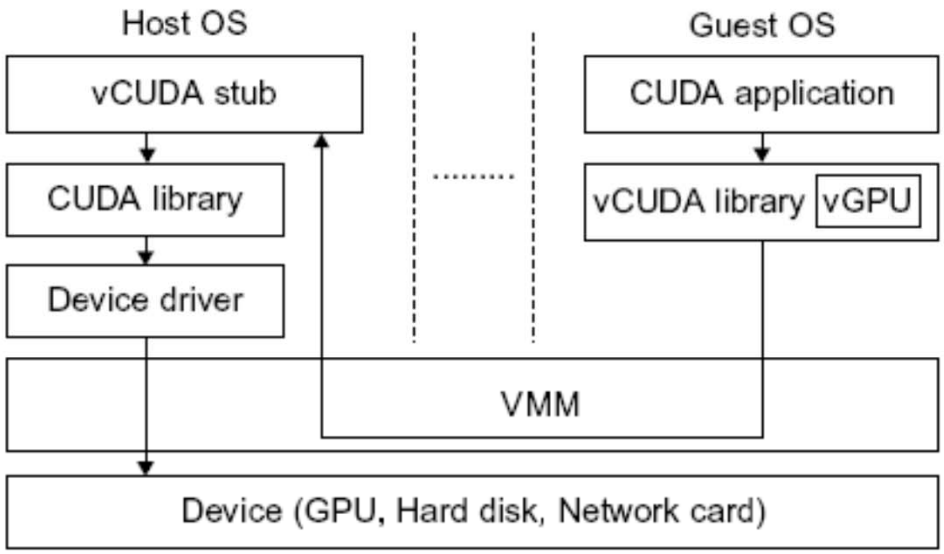


FIGURE 3.4

Basic concept of the vCUDA architecture.

(Courtesy of Lin Shi, et al. [57])

- Na VMM se izvršavaju domaćin OS i gostujući OS
- Aplikacija na gost. OS koristi virtualizovanu vCUDA biblioteku (vGPU)

Virtuelizacija na nivou korisnika – aplikacije

- Virtuelizuje neku aplikaciju kao virtuelnu mašinu
  - Može da izvršava programe pisane za određenu definiciju apstraktne mašine
- Primeri:
  - JVM , NET CLI, Panot
- Prednosti:
  - Ima najbolju izolaciju aplikacija
- Nedostatci i ograničenja
  - Niska performansa, mala aplikaciona fleksibilnost i visoka implementaciona složenost

Relativne prednosti virtuelizacije na različitim nivoima

Nivo implementacije	Veća performansa	Veća fleksibilnost	Implementaciona Složenost	Aplikaciona Izolovanost
ISA	X	XXXXX	XXX	XXX
HAL	XXXXX	XXX	XXXXX	XXXX
OS	XXXXX	XX	XXX	XX
Biblioteka	XXX	XX	XX	XX
Aplikacija	XX	XX	XXXXX	XXXXX

Hipervizor

- Virtuelizator hardvera koji omogućava da se više gostujućih OS izvršava na mašini domaćinu
  - On se još naziva i Monitor virtuene mašine (VMM)
- Tip 1: hipervizor osnovne mašine (bare metal)
  - Nalazi se iznad hardvera osnovne mašine
  - Svi gostujući OS predstavljaju sloj iznad hipervizora
- Tip 2: gostujući hipervizor (hosted hypervisor)
  - Radi iznad OS domaćina
  - Hipervizor nije u I, već je u II sloju iznad HW
  - Gostujući OS rade u sloju iznad hipervizora

Važniji dobavljači VMM i hipervizora

VMM dobavljač	Domaćin CPU	Gost. CPU	Domaćin OS	Gost. OS	VM arhitektura
VMware radna stanica	X86, X86-64	X86, X86-64	Windows, Linux	Windows, Linux, Solaris, FreeBSD, Netware, OS/2, SCO, BeOS, Darwin	Puna virtuelizacija
VMware ESX server	X86, X86-64	X86, X86-64	Nema domaćina OS	Isto ako VMware radna stanica	Para-virtuelizacija
XEN	X86, X86-64, IA-64	X86, X86-64, IA-64	NetBSD, Linux, Solaris	FreeBSD, NetBSD, Linux, Solaris, Windows XP i 2003 server	Hipervizor
KVM	X86, X86-64, IA-64, S390, PowerPC	X86, X86-64, IA-64, S390, PowerPC	Linux	Linux, Windows, FreeBSD, Solaris	Para-virtuelizacija

Arhitektura Xen (1/3)

- Koncept arhitekture Xen
  - Xen je mikro-jezgro koje realizuje sve mehanizme a politike prepušta Domenu 0
  - Xen ne uključuje rukovaoce uređaja, već obezbeđuje mehanizme za direktan pristup uređajima za gost. OS
  - Xen obezebeđuje virtuelno okruženje između HW i OS
  - Primeri Xen hipervizora: Cirtex XenServer i Oracle VM

Arhitektura Xen (2/3)

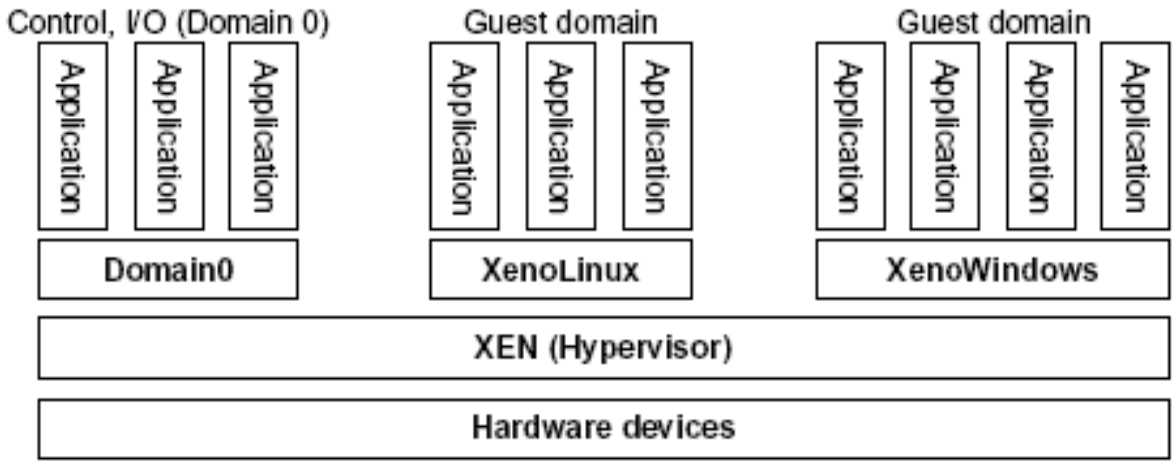


FIGURE 3.5

The Xen architecture's special domain 0 for control and I/O, and several guest domains for user applications.

- Komponente: hipervizor, gostujući OS i aplikacije
- Gostujući OS nisu ravnopravni, jedan od njih kontroliše druge, on se zove Domen 0, drugi su Domen U
- Domen 0 je privilegovan, puni se prvi bez FS, ima direktan pristup HW i rukuje uređajima – on dodeljuje uređaje gost. OS u Domenu U

Arhitektura Xen (3/3)

- Zaštita:
  - Xen je zasnovan na Linux i ima nivo zaštite C2
  - Ako se Domen 0 kompromituje, haker može da kontroliše ceo sistem
  - Domen 0 ima ulogu VMM i omogućava korisnicima rukovanje sa VM (stvaranje, kopiranje, migriranje, vraćanje unazad) kao što rukuju datotekama, dakle vrlo fleksibilno, ali sa ozbiljnim problemima zaštite

Puna i delimična virtuelizacija (1/2)

- Puna virtuelizacija:
  - Ne treba menjati gostujući OS
  - Kritične instrukcije se emuliraju u SW kroz binarno prevođenje
  - VMware Workstation koristi punu virtuelizaciju, koja putem binarnog prevođenja automatski u fazi izvršenja menja x86 SW zamenom kritičnih instrukcija
  - Nedostatak: prevođenje usporava izvršenje

Puna i delimična virtuelizacija (2/2)

- Delimična virtuelizacija (Para-virtualization):
  - Smanjuje režiju, ali cena održavanja OS je visoka
  - Poboljšanje zavisi od opterećenja
  - Mora se menjati gostujući OS, instrukcije koje se ne mogu virtuelizovati zamenjuju se sa hiper-pozivima koji direktno komuniciraju sa hipervizorom ili VMM
  - Delimičnu virtuelizaciju podržavaju Xen, Denali i VMware ESX

Izazovi virtuelizacije memorije

- Prevođenje adresa:
  - Gost. OS očekuje kontinualnu mem. od adrese 0
  - VMM mora očuvati ovu iliziju
- Održavanje kopija tabela stranica (shadowing):
  - VMM presreće operacije straničenja
  - Konstruiše kopije tabela stranica
- Režija (overheads):
  - VM režija se dodaje na vreme izvršenja
  - Kopije tabela stranica konzumiraju značajnu količinu memorije

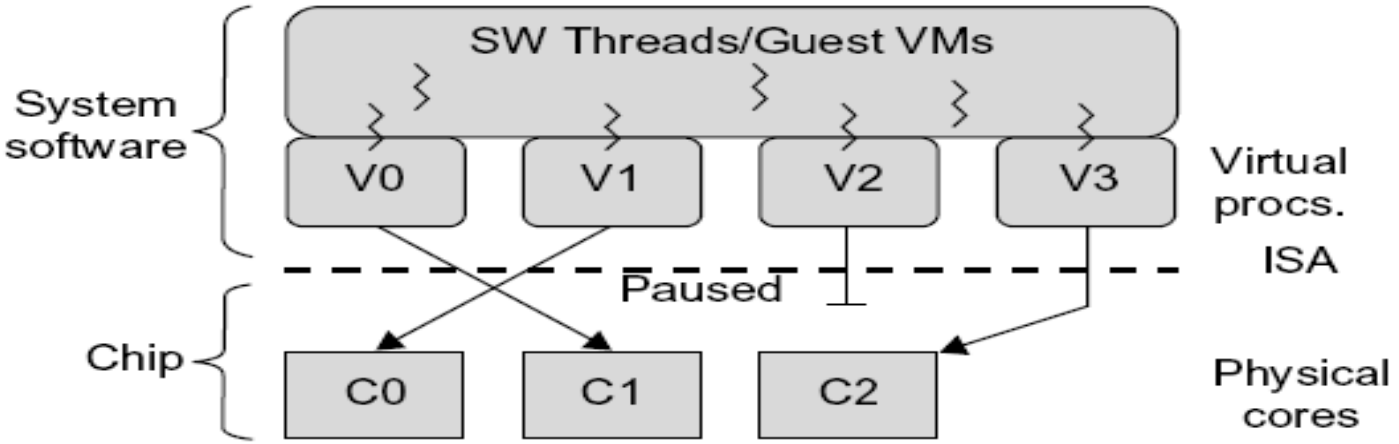
Izazovi virtuelizacije U-I uređaja

- Nivoi apstrakcije virtuelnih U-I uređaja:
  - Gostujući rukovalac U-I uređajem u gost. OS
  - Virtuelni uređaj
  - Sloj za virtuelizaciju (stvara predstavu virt. uređaja)
    - Emulira virtuelni uređaj
    - Preslikava gostujuće na realne U-I adrese
    - Multipleksira i upravlja fizičkim uređajima
    - Obezbeđuje specifične osobine U-I uređaja, npr. COW disk (COW = Copy On Write)
  - Realni uređaj
    - Može se razlikovati od virtuelnog uređaja

Zaključci o virtuelizaciji CPU, memorije i U-I:

- Virtuelizacija CPU:
  - Zahteva HW-podržane zamke za osetljive instrukcije; te zamke koristi VMM
- Virtuelizacija memorije:
  - Zahteva posebnu HW podršku (kopije tabela stranica kod VMWare ili proširene tabele stranica kod Intel) za prevođenje virt. adresa u fizičke adrese u dve faze
- Virtuelizacija U-I:
  - Najteža za realizaciju zbog složenosti U-I uslužnih rutina i emulacije između gost. OS i OS domaćina

Virtuelizacija višejezgarnog procesora: VCPU naprama CPU

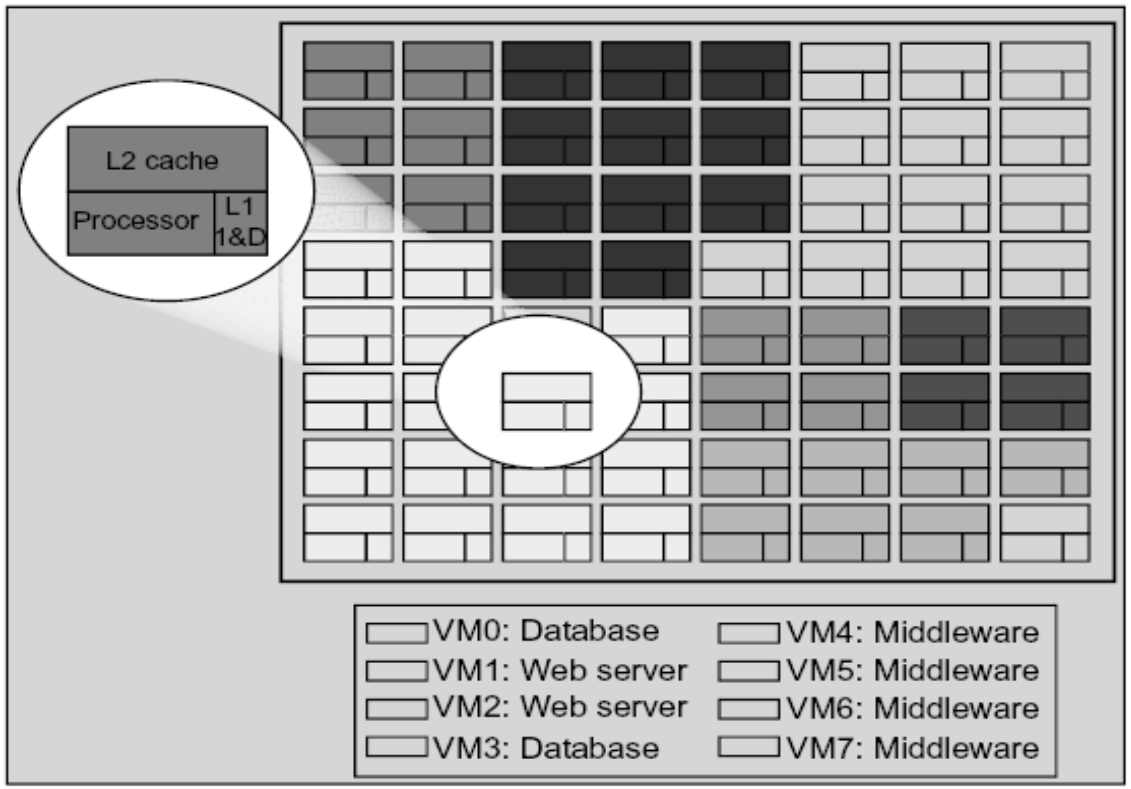


- Softveru su izložena četiri VCPU, a realno postoje samo tri jezgra
- VCPU V0, V1, i V3 su transparentno migrirali, dok je VCPU V2 transparentno suspendovan

Virtuelna naspram fizičkih jezgara

Fizička jezgra	Virtuelna jezgra
Aktuelna fizička jezgra prisutna u procesoru	Može biti više virtuelnih jezgara, vidljivih jednom OS, od stvarnog broja fizičkih jezgara
Veće opterećenje na programere da pišu aplikacije koje mogu da rade direktno na fizičkim jezgrima	Projektovanje SW postaje jednostavnije jer HW pomaže SW-u u dinamičkom iskorišćenju resursa
HW ne obezbeđuje pomoć SW-u pa je zato jednostavniji	HW obezbeđuje pomoć SW-u pa je zato složeniji
Slabo rukovanje resursima	Bolje rukovanje resursima
Mora se menjati najniži nivo sistemskog softvera	Najniži nivo sistemskog SW ne mora biti menjan

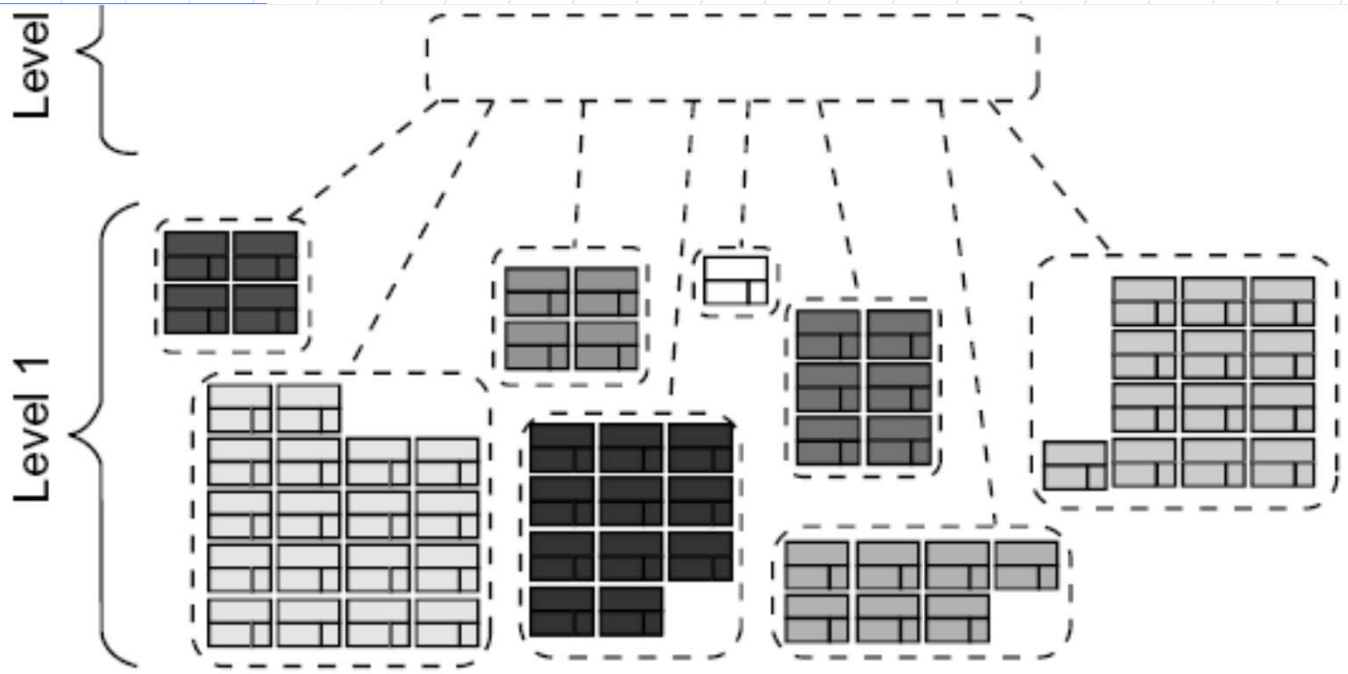
Primer preslikavanja VM na susedna jezgra



(a) Mapping of VMs into adjacent cores

- VM se lociraju na susedna jezgra
- VM posvećene raznim aplikacijama
- Npr. VM0 i VM3 – baza podtaka, VM1 i VM2 – web server VM4-VM7 - midlver

Virtuelni klasteri u sistemima sa mnogo jezgara



(b) Multiple virtual clusters assigned to various workloads

- Prostor VM-ova je moguće podeliti
- Tako nastaje virtuelna hijerarhija (sa 2 nivoa na slici iznad)
- Onda je moguće više klastera dodeliti u zavisnosti od opterećenja

Osobine virtuelnih klastera (1/2)

- Čvorovi VC mogu biti bilo fizičke ili VM
  - Više VM sa različitim OS mogu biti raspoređene na istom fizičkom čvoru
- VM radi sa gost. OS
  - Koji se često razlikuje od OS domaćina
- VM konsoliduje više funkcionalnosti na istom serveru
  - To značajno povećava iskorišćenje servera i povećava aplikacionu fleksibilnost

Osobine virtuelnih klastera (2/2)

- VM mogu biti kolonizirane (replicirane)
  - Na više servera, u cilju distribuiranog paralelizma, otpornosti na greške, i oporavka od katastrofa
- Veličina VC može da se menja dinamički
  - Kao što varira veličina prekrivačke mreže u P2P mreži
- Otkazi čvorova mogu onemogućiti VM instalirane na čvorovima u otkazu
  - Ali otkazi VM ne mogu srušiti sistem domaćina

Virtuelni klasteri naspram Fizičkih klastera

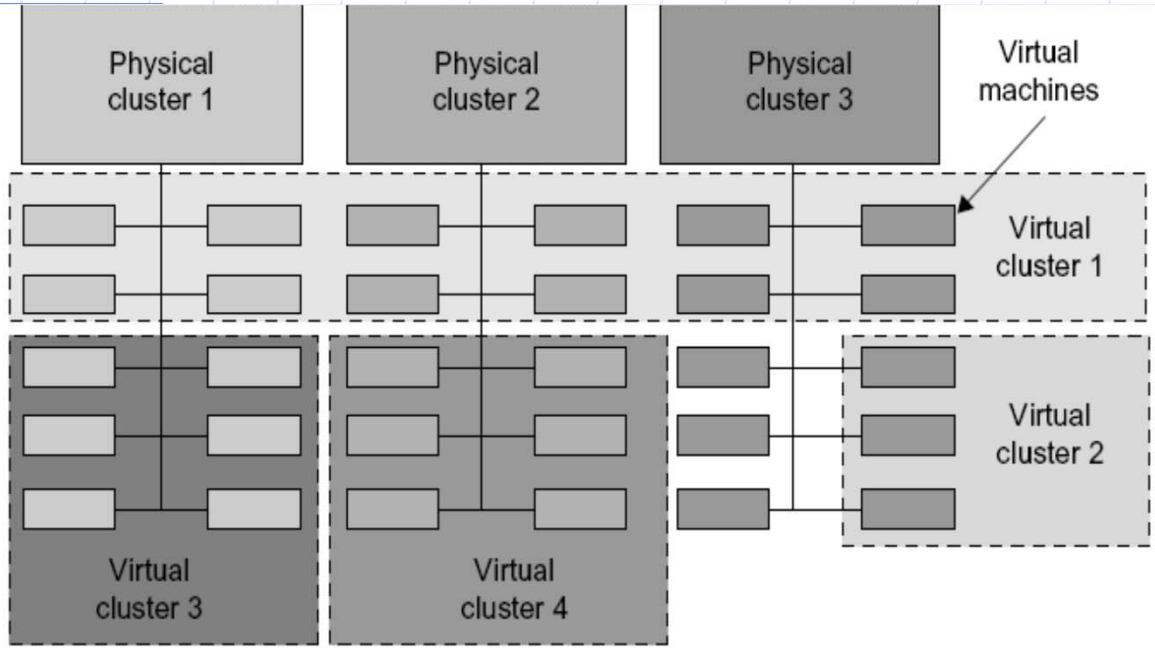


FIGURE 3.18

A cloud platform with 4 virtual clusters over 3 physical clusters shaded differently.

- Jedan VC se može izvršavati na jednom ili više fizičkih klastera (PC)
- Slika prikazuje raspored 4 VC preko 3 PC
- Npr. VC1 su dodeljeni neki čvorovi sa sva tri fizička klastera
- VC2 koristi deo PC3, itd.

Virtuelni klasteri na bazi particionisanja aplikacija

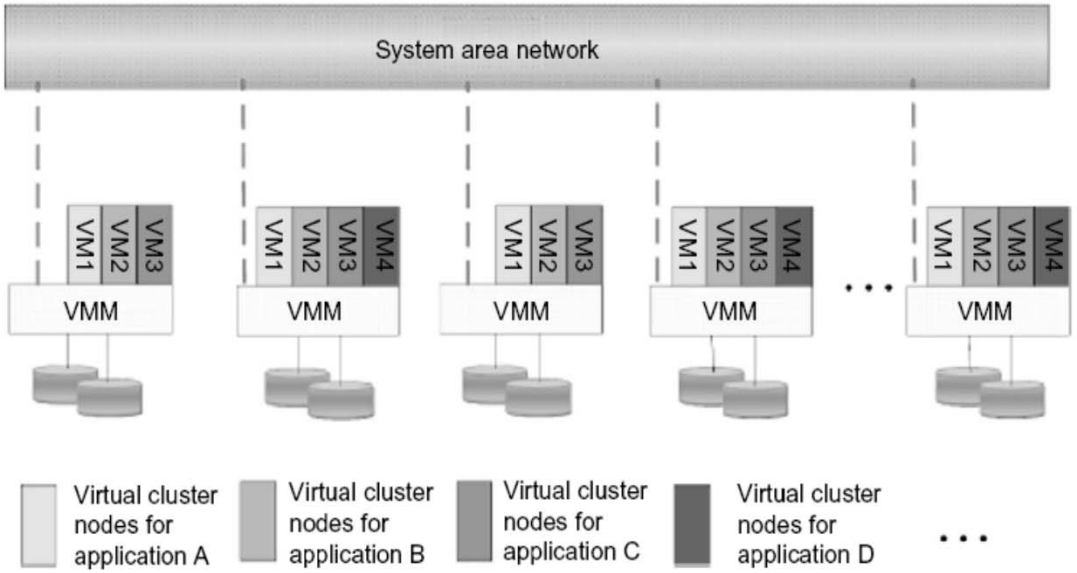


FIGURE 3.19

The concept of a virtual cluster based on application partitioning.

(*Courtesy of Kang, Chen, Tsinghua University 2008*)

- Aplikacijama se pridružuju virtuelni klasteri
- Virtuelnim klasterima se pridružuju VM na raznim fizičkim klasterima
- Slika prikazuje čvorove VC za aplikacije A, B, C i D

Živa migracija VM (1/2)

- Pet faze migracije:
  - Faza 0: „Priprema“. Aktivna VM na čvoru A. Rezervisani resursi na drugoj mašini. Blok uređaji u ogledalu.
  - Faza 1: „Rezervacija“. Inicijalizuj kontejner na ciljnom čvoru.
  - Faza 2: „Iterativno kopiranje“. Omogući kopiranje stranica (straničenje u senci) u iterativnim ciklusima.
  - Faza 3: „Stani i kopiraj“. Suspenduj VM na čvoru A. Postavi ARP za preusmeravanje saobraćaja na čvor B. Sinhronizuj preostali deo stanja na čvoru B.

Živa migracija VM (2/2)

- Pet faze migracije (nastavak...):
  - Faza 4: „Kraj ažuriranja“ (commit). Oslobodi resurse za VM na čvoru A.
  - Faza 5: „Aktiviranje“. Pokreni VM na čvoru B. Poveži VM sa lokalnim uređajima. Nastavi normalan rad VM.

Ilustracija migracije VM

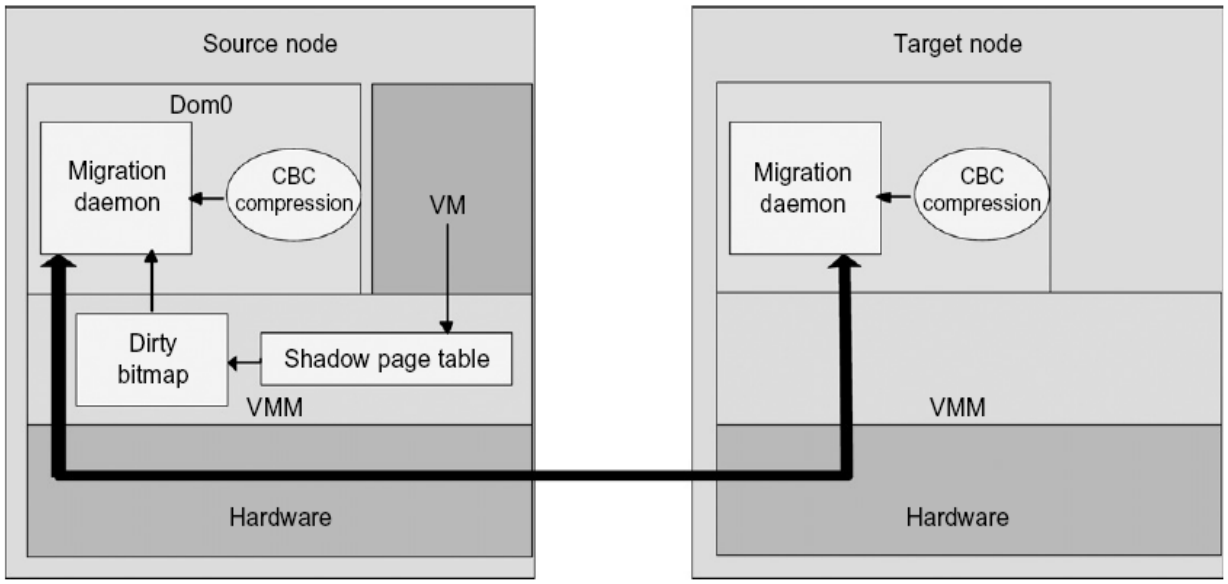


FIGURE 3.22

Live migration of VM from the Dom0 domain to a Xen-enabled target host.

- Živa migracija VM iz Domena 0 na ciljnog domaćina baziranog na Xen

Primer uticaja migracije na propusnost Web servera

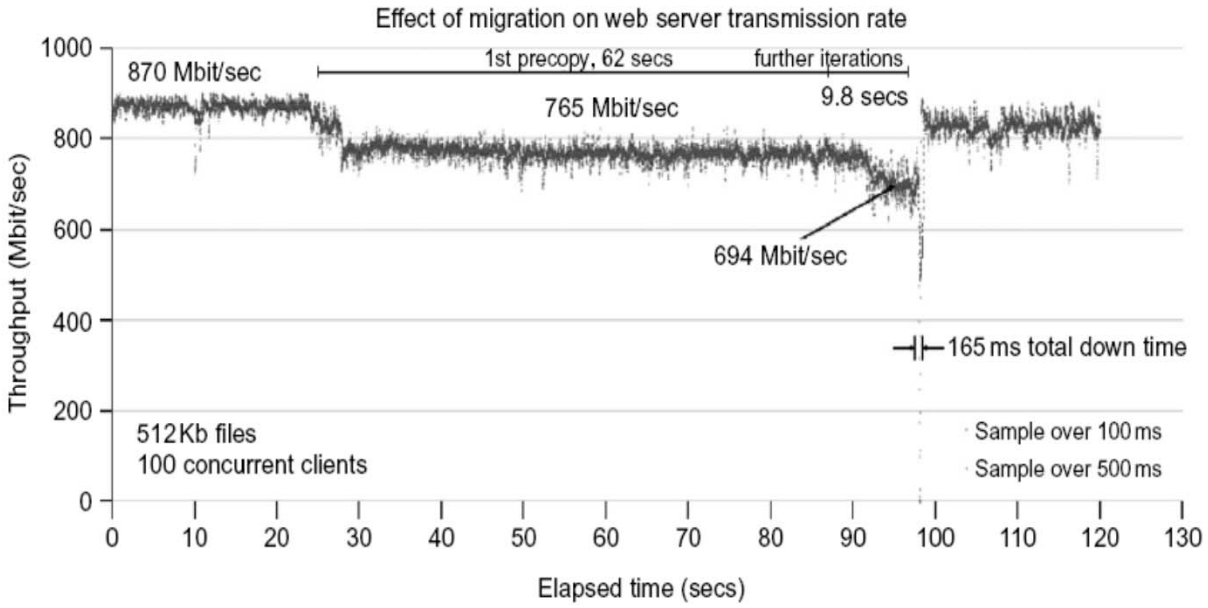


FIGURE 3.21

Effect on data transmission rate of a VM migrated from one failing web server to another.

(Courtesy of C. Clark, et al. [14])

- Na apcisi je vreme, a na ordinati propusnost u Mb/s
- Propusnost sa 870 Mb/s pada na 765 Mb/s u Fazi 2
- U Fazi 3, koja traje 165 ms, pada na 694 Mb/s

Projekti virtuelnih klastera

Ime projekta	Ciljevi	Rezultati
Cluster-on-demand (COD) na Univerzitetu Duke	Dinamička dodela resursa Sistem za rukovanje VC-om	Deljenje bazena VM od više VC korišćenjem Sun GridEngine
Cellular Disco na Univerzitetu Stanford	Postaviti VC na SMP (shared-memory multiprocessor)	Više VM na više procesora sa VMM pod nazivom Cellular Disco
VIOLIN na Univrezitetu Purdue	Klasterizacija sa više VM radi provere prednosti dinamičke adaptacije	Smanjeno vreme izvršenja aplikacija na klasteru VIOLIN
GRAAL projekat u INRIA, Francuska	Performansa paralelnih algoritama na VC na bazi Xen	75% max performanse sa 30% labavošću resursa na VM klasterima

Klaster na zahtev: Projekat COD na Univerzitetu Duke

- COD je sistem za rukovanje VC
  - Dinamička dodela servera za potrebe više VC
- Koristi se Sun GridEngine raspoređivač
  - Potvrda da je dinamička dodela servera pogodna apstrakcija za napredno rukovanje resursima VC
- COD podržava:
  - rezervaciju resursa, adaptivnu dodelu resursa, oslobađanje resursa, i dinamičko instanciranje grid usluga
- COD konfiguraciona baza podataka:
  - Obezbeđuje politike za resurse i definicione templete kao odgovore na zahteve korisnika

COD particionisanje PC u više VC

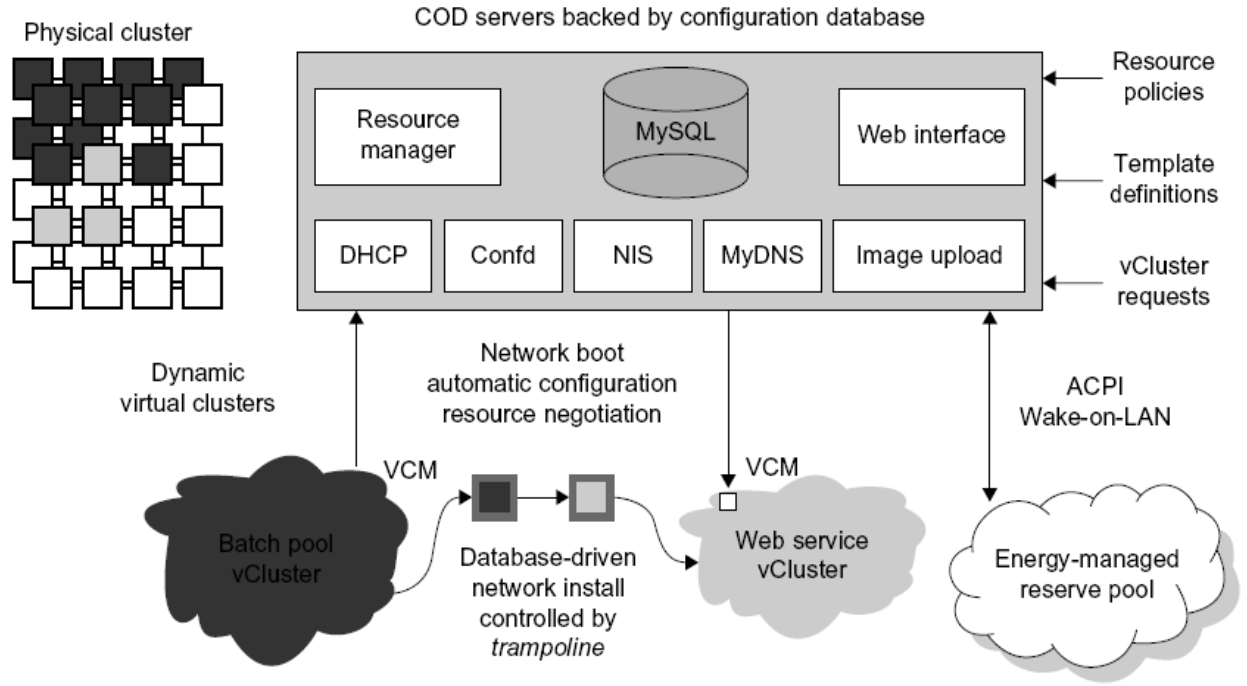


FIGURE 3.23

COD partitioning a physical cluster into multiple virtual clusters.

(*Courtesy of Jeff Chase, et al, HPDC-2003 [12]*)

- Dinamički VC sa mrežnom instalacijom, automatskom konfiguracijom i pregovaranjem o resursima
- Ugrađene politike za resurse i definicije templeta
- Dostup kroz Web servis vCluster

Primer varijacije veličine VC u COD

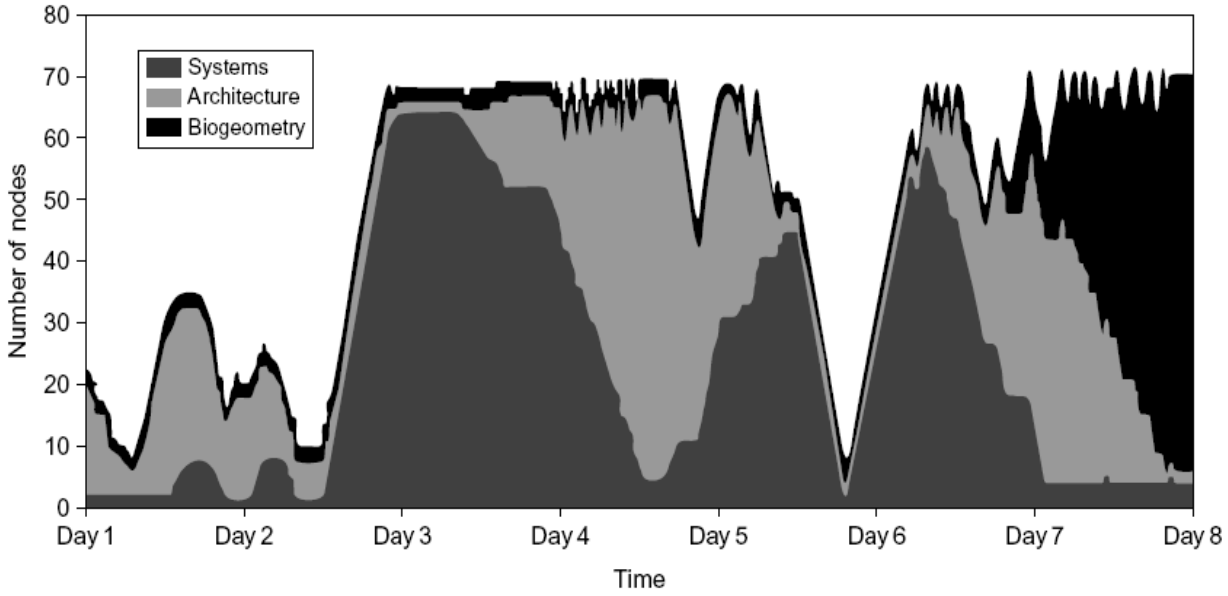


FIGURE 3.24

Cluster size variations in COD over eight days at Duke University.

(*Courtesy of J. Chase, et al. [12]*)

- Prikazana je varijacija veličine 3 VC u toku 8 dana
- Npr veličina VC Systems varira od 0 do 65 čvorova
- Najveće fluktuacije veličine su kod VC Biogeometry

Projekat VIOLIN na Univerzitetu Purdue

- VIOLIN koristi živu migraciju VM za rekonfiguri-sanje okruženja sa VC
  - Više poslova u više domena VC – bolje iskorišćenje resursa
- Domen = izolovano virtuelno okruženje nad HW
- Primer: 5 virt. okruženja, VIOLIN 1-5, koja dele 2 fizička klastera (domena)
- Rezultat: adaptacija virt okruženja može znatno poboljšati iskorišćenje resursa po cenu manju od 1% povećanja ukupnog vremena izvršenja

Ilustracija adaptacije u VIOLIN

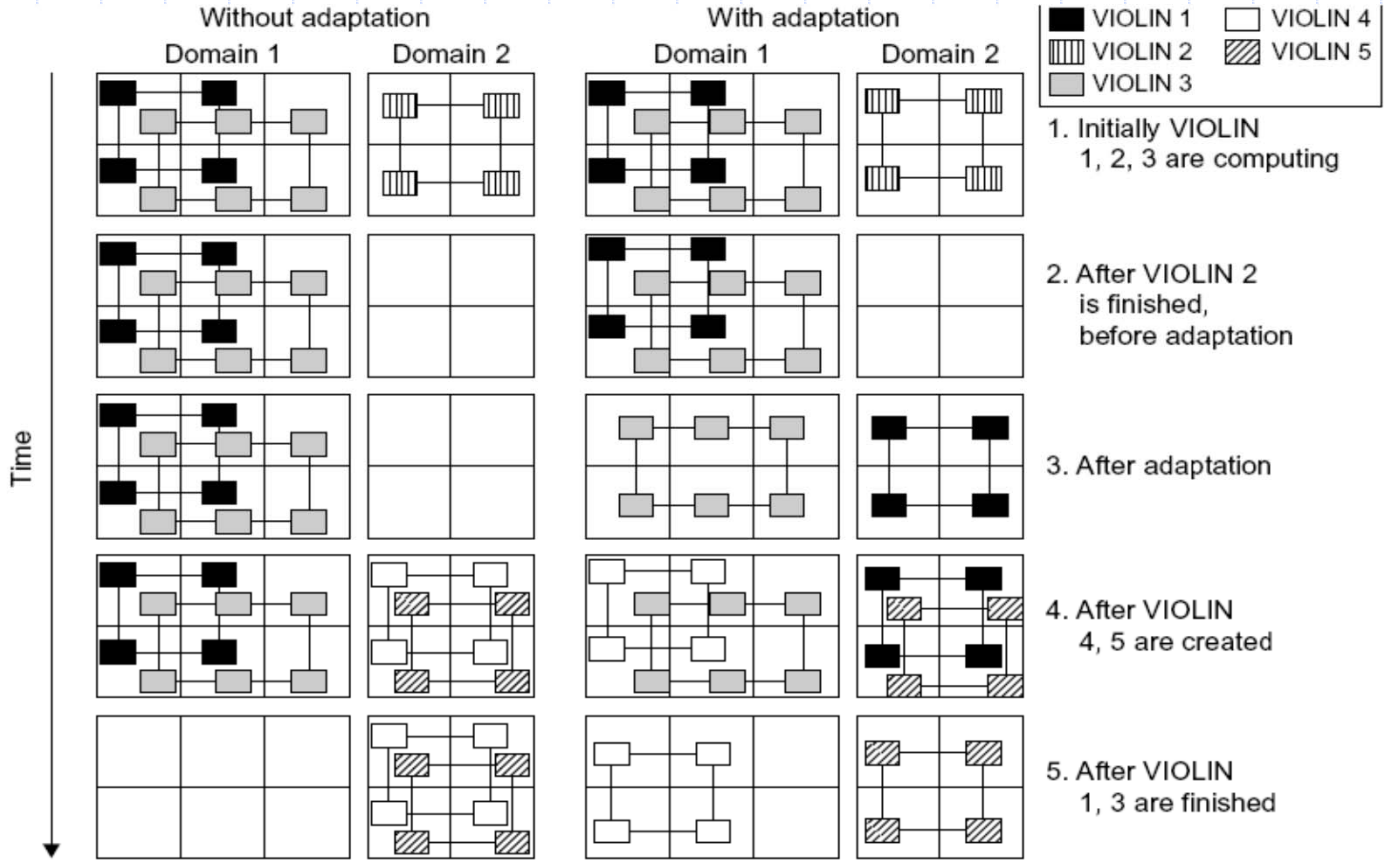


FIGURE 3.25

VIOLIN adaptation scenario of five virtual environments sharing two hosted clusters; Note that there are more idle squares (blank nodes) before and after the adaptation.

(*Courtesy of P. Ruth, et al. [24,51]*)

- Levo je prikazan sistem bez adaptiranja, a desno sa adaptiranjem
- Desni sistem uspešno koristi oba domena

Pregled rukovaoca i OS za pravljenje privatnih oblaka

Rukovaoc, OS	Koncept virtuelizuje	API klijenta, jezik	Korišćeni hipervizor	Javna sprega oblaka	Posebne osobine
Nimbus, Linux, Apache v2	VM stvaranje i VC	EC2 WS, WSRF, CLI	Xen, KVM	EC2	Virtuelne mreže
Eucalyptus, Linux, BSD	Virtuelno umrežavanje	EC2 WS, CLI	Xen, KVM	EC2	Virtuelne mreže
OpenNebula, Linux, Apache v2	Rukovanje VM, domaćin, virt. mreža i alati za raspoređivanje	XML-RPC, CLI, Java	Xen, KVM	EC2, Elastičan domaćin	Virtuelne mreže, dinamičko obezbeđivanje
vSphere 4, Linux, Windows	Virtuelizacija OS za centre za podatke	CLI, GUI, Portal, WS	VMware ESX, ESXi	VMware vCloud	Zaštita pod., vStorage, VMFS, DRM

Eucalyptus: Otvoreni OS za rukovanje privatnim oblacima

- Otvoren sistem za podršku IaaS oblaka:
  - Podržava virtuelno umrežavanje i rukovanje VM; ne podržava virt. skladište
- Privatni oblak sa kojim korisnici interaguju kroz intranet i Internet
  - Podržava kom. i sa drugim privat. oblacima
- Nedostaje zaštita i druge funkcije za opštenamenski grid i oblak

Komponente Eucalyptus-a

- Rukovalac instancama:
  - Upravlja instancama na domaćinu na kom radi
- Rukovalac grupama:
  - Raspoređuje VM na određenom ruk. instancama i rukuje mrežom virtuelnih instanci
- Rukovalac oblakom:
  - Ulazna tačka za korisnike i admin.
  - Nadzire ruk. čvorovima, i realizuje raspoređivanje kroz zahteve ruk. grupama

Hijerarhija upravljanja u Eucalyptus

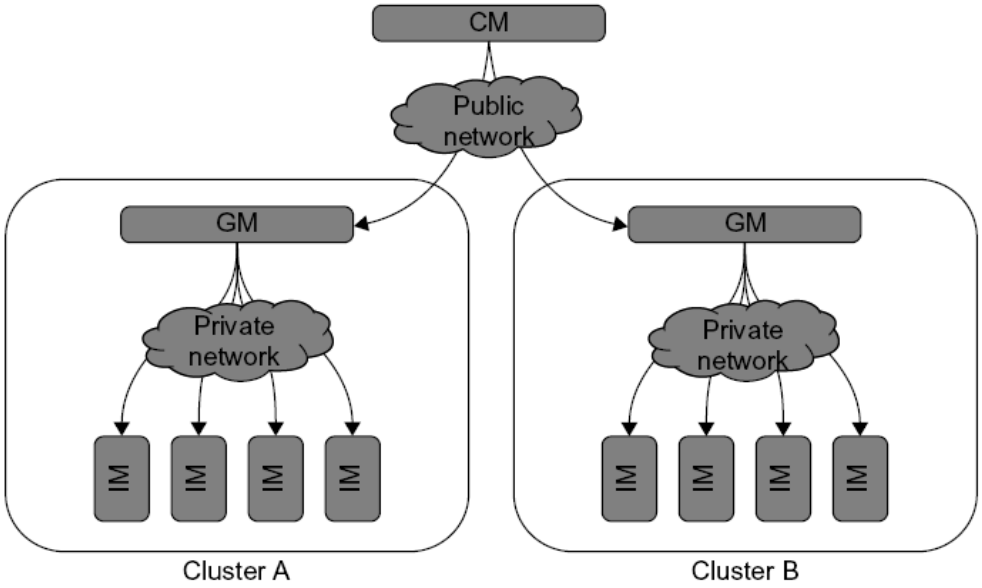


FIGURE 3.27

Eucalyptus for building private clouds by establishing virtual networks over the VMs linking through Ethernet and the Internet.

(Courtesy of D. Nurmi, et al. [45])

- Tri nivoa hijerarhije: CM, GM i IM
- Klasteri na različitim lokalitetima povezanim preko Interneta
- U primeru iznad, klaster A i B su na bazi privatnih mreža

Arhitektura vSphere/4

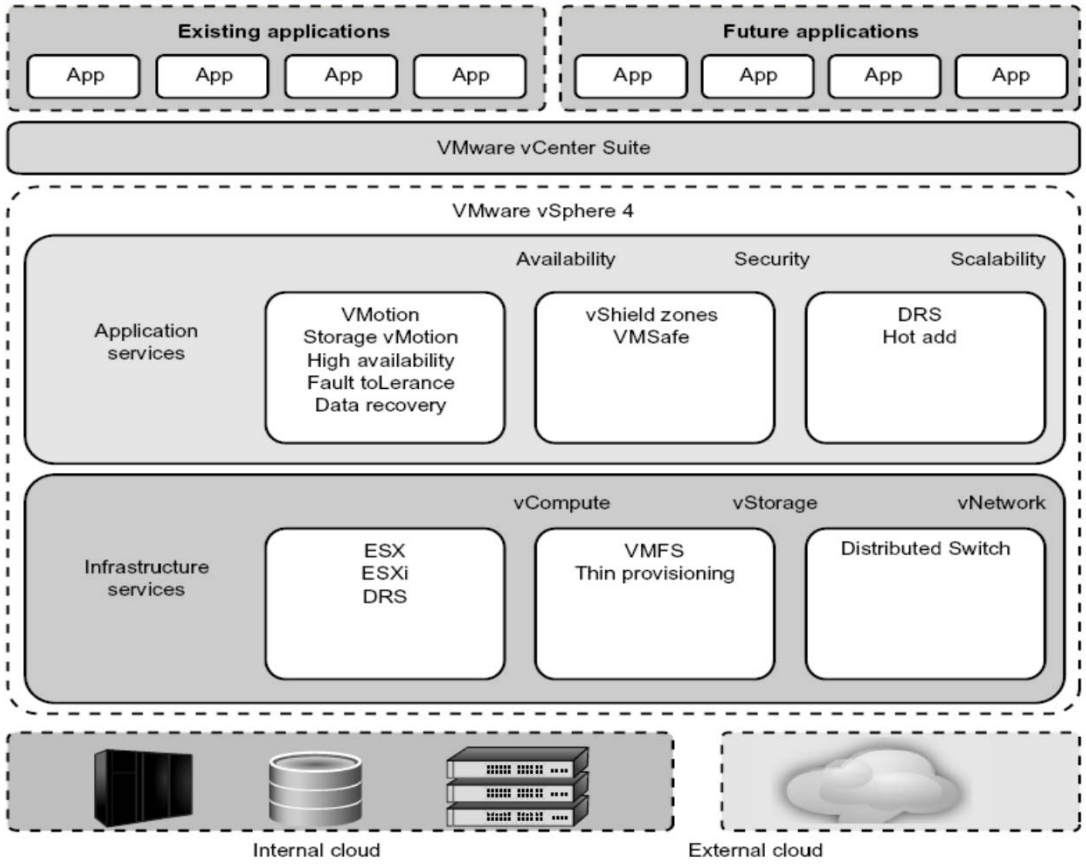


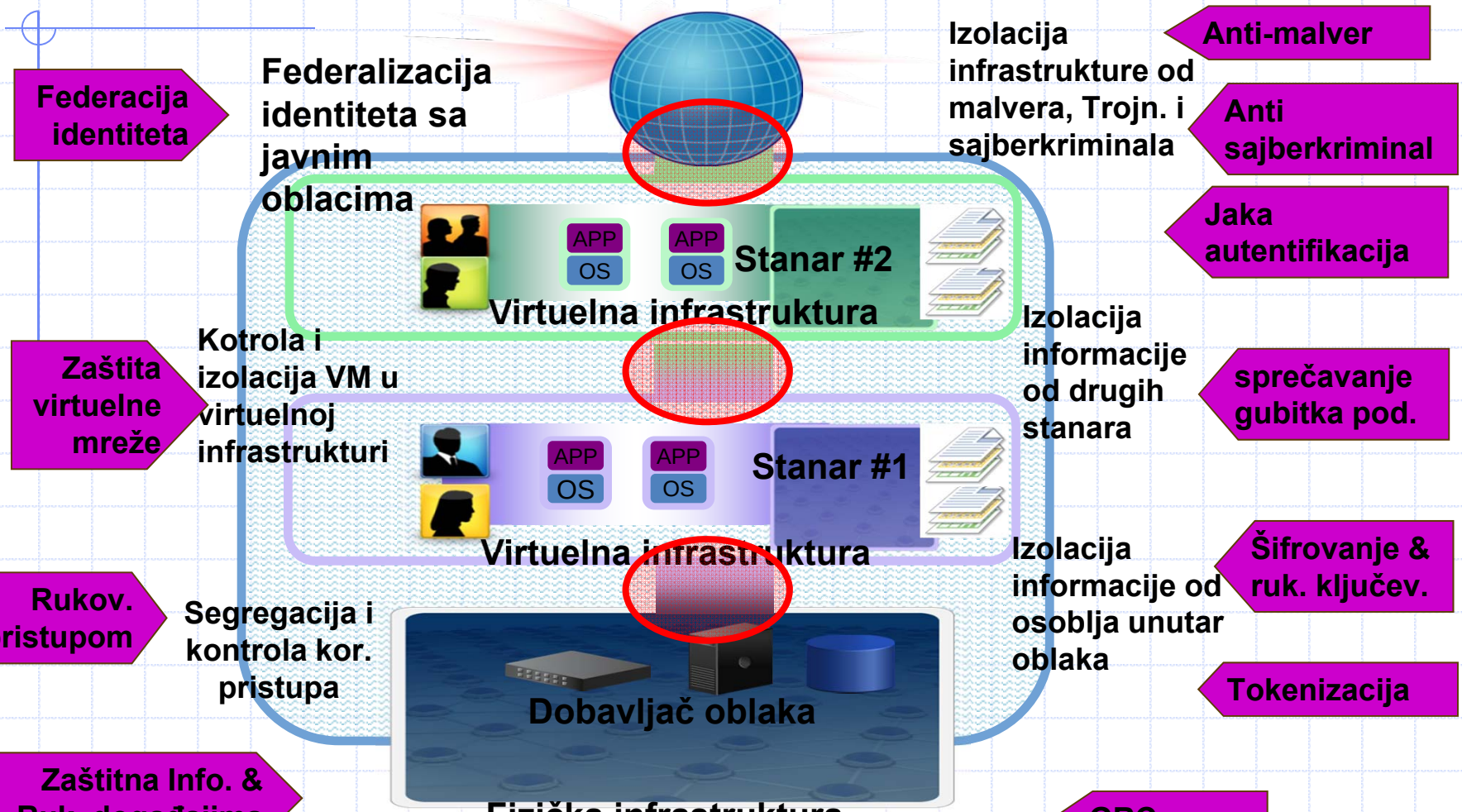
FIGURE 3.28

vSphere/4, a cloud operating system that manages compute, storage, and network resources over virtualized data centers.

(Courtesy of VMware, April 2010 [72])

- OS za oblak
- Infrastrukturni servisi: vCompute, vStorage, vNetwork
- Aplikacioni servisi: Raspoloživost, Zaštita, Skalabilnost

Zone poverenja za izolaciju VM



(Courtesy of L. Nick, EMC 2008)