

**VISOKA ŠKOLA STRUKOVNIH STUDIJA ZA INFORMACIONE TEHNOLOGIJE**



VISOKA ŠKOLA STRUKOVNIH STUDIJA ZA IT

Informacione i internet tehnologije

Seminarski rad

**Virtuelizacija**

Predmetni nastavnik:  
prof. dr Slavko Pokorni, dipl. inž. el.

Student:  
Šandor Lajko  
Datum predaje  
24.01.2010

## REZIME

Virtualizacija je koncept koji postoji već dugo. U ovom radu napravljen je kratak osvrt na istorijat virtualizacionih tehnika, kao i pregled trenutne situacije u ovoj oblasti informacionih tehnologija.

U uvodu je razjašnjen pojam virtualizacije i nekih termina koji se spominju u radu.

Poglavlje *Istorijat virtualizacije* kratkim pregledom razvoja tehnologija vezanih za ovu oblast pravi se uvod u sledeću temu, *Specifičnosti virtualizacije na x86 platformi*. U poglavlju *Virtualizacija u desktop okruženju* opisano je nekoliko primena virtualizacije na personalnim računarima, u testiranju i razvoju softvera, i kao pomoći u nastavi. Poglavlje *Virtualizacija servera* prikazuje kako se pomoći virtualizacionih tehnika rešavaju problemi niskorišćenosti hardvera, kao i slučaj servera čija je namena hostovanje virtualnih desktop sistema.

U dodatku *Ilustracije* prikazano je nekoliko interesantnih situacija primene Virtualnih mašina.

**Ključne reči:** virtualizacija, virtualna mašina, konsolidacija servera

## SADRŽAJ

<b>Rezime.....</b>	2
<b>Sadržaj .....</b>	3
<b>Uvod .....</b>	4
<b>Virtualizacija.....</b>	5
<i>Istorijat virtualizacije .....</i>	5
<i>Kategorizacija virtualizacije .....</i>	6
<i>Specifičnosti virtualizacije na x86 platformi .....</i>	7
<i>Virtualizacija u desktop okruženju .....</i>	8
Testiranje operativnog sistema.....	8
Razdvajanje razvojnih platformi .....	10
Simuliranje mreže i mrežnih konfiguracija na jednom računaru .....	11
<i>Virtualizacija servera .....</i>	12
Konsolidacija servera .....	12
Remote desktop virtualne mašine.....	13
<b>Dodatak: Ilustracije.....</b>	14
<b>Zaključak .....</b>	17
<b>Literatura.....</b>	18
<b>Indeks slika .....</b>	18

## UVOD

U najširem smislu, virtuelizacija je koncept kojim se označavaju tehnike i metodi za apstrakciju računarskih resursa.

Virtuelizacija je metodologija razdvajanja resursa računara u više zasebnih radnih okruženja, primenom tehnologija kao što su hardversko ili softversko particonisanje, time-sharing, delimična ili potpuna mašinska simulacija, emulacija i mnoge druge.

Mada je definisan kao takav, pojam virtuelizacije nije ograničen samo na particonisanje, razdvajanje nečega na više manjih celina. Virtuelizacija obuhvata i proces apstrakcije koji je logički suprotan: spajanje više fizički razdvojenih celina u jednu. Na primer, kada se nekoliko hard diskova predstavlja kao jedna logička celina, ili kada je nekoliko računara umreženo da bi se koristili kao jedan veliki računar (Grid computing, Parallel Virtual Machine).

Virtuelizacija je danas već dokazana softverska koja ima veliki uticaj na IT infrastrukturu i na način na koji se rašunari upotrebljavaju. Današnji moćni računarski sistemi koji su zasnovani na x86 arhitekturi projektovani su za izvršavanje jednog operativnog sistema i malog broja aplikacija na njemu. Ovo dovodi do veoma slabog iskorišćenja većine računara. Virtuelizacija omogućava pokretanje većeg broja virtualnih mašina na jednoj fizičkoj mašini, tako da one dele raspoložive resurse na nekoliko razdvojenih okruženja. Virtuelne mašine mogu istovremeno pokrenuti različite operativne sisteme i različite aplikacije na istom fizičkom računaru.

Izrazom *Virtualna mašina* (VM) označava se softverska implementacija računara, koja izvršava programe ni isti način kao i prava mašina.

U ovom radu predstaviću osnovne tehnike i metode virtuelizacije, kao i najčešće praktične primene.

## VIRTUALIZACIJA

### Istorijat virtuelizacije

Osnovni elementi virtuelizacije prvi put se pojavljuju se pojavljuju početkom šezdesetih godina prošlog veka.

U junu 1959. godine u okviru Međunarodne konferencije o obradi podataka (*International Conference on Information Processing*) u UNESCO-u, Kristofer Strečli (*Christopher Strachey*) objavio je rad pod naslovom „Time Sharing u velikim i brzim računarima“ (*Time Sharing in Large Fast Computers*). U ovom radu, Strečli se bavi upotrebom multi-programiranja u cilju izbegavanja čekanja procesora na periferije.

Multiprogramiranje u cilju *spooling*-a upotrebljeno je u projektu Atlas početkom 1960-ih godina. Računar Atlas je prvi koji je koristio *demand paging* i *supervisor calls*. Supervizorski pozivi čine osnovne grane supervizor programa, koji se aktiviraju pozivom interapt rutina, ili pozivom samih supervizorskih instrukcija (*S.E.R. – Supervisor Extracode Routines*). Praktično, Atlasov supervizor se pokretao unutar jedne virtuelne mašine, a korisnički programi unutar druge.

Sredinom 1960-ih, IBM-ov Watson Research Center razvija projekat M44/44x, u cilju procene sistemskih time sharing koncepata. Ova arhitektura je zasnovana na virtuelnim mašinama. Glavni računar je bio IBM 7044 (M44), a svaki virtuelni računar eksperimentalan prikaz osnovnog računara (44x). Adresni prostor svakog 44x bio je rezidentan u memorijskoj hijerarhiji M44, koja je koristila virtuelnu memoriju i multi-programiranje.



Slika 1. IBM System 370, računar na kojem je radio VM/370

IBM je nastavio da bude veoma važan pokretač na ovom polju. Razvijeno je nekoliko nekoliko sistema sa virtuelnim mašinama na bazi IBM-ovih računara: *CP-40* (koji je razvijen za prerađenu verziju računara *IBM 360/40*), *CP-67* (za *IBM 360/67*), slavni *VM/370*, i mnogi drugi. Po pravilu, IBM virtuelne mašine su bile identične kopije hardvera na koje su izvršavane. Na računaru se izvršavala komponenta nazvana *Virtual machine monitor* (*VMM*), koja je zatim omogućavala kreiranje više virtuelnih mašina, a svaka instanca je zatim pokretala svoj operativni sistem. IBM-ova VM rešenja su i danas veoma cenjena i pouzdana.

Koncept virtualizacije je praktično napušten tokom poslednje dve decenije prošlog veka kao rezultat klijent-server aplikacija i jeftinih servera na bazi x86 platforme. Široko prihvatanje *Windowsa* i pojava *Linuxa* kao serverskog operativnog sistema postavili su x86 servere na mesto industrijskog standarda. Dominacija x86 platforme na serverskim i klijentskim sistemima dovela je do nastanka nove IT infrastrukture sa specifičnim operativnim problemima. U ove probleme spadaju niski stepen iskorišćenja infrastrukture, porast troškova održavanja infrastrukture, porast troškova upravljanja, nedovoljna zaštita od otkaza i visoki troškovi održavanja i administriranja klijentskih računara. Virtualizacija na x86 platformi nastala je iz potrebe za rešavanjem upravo ovih problema.

### Kategorizacija virtualizacije

Postoji veliki broj različitih tehnika i tehnologija koje se koriste prilikom realizacije virtualizacije. Zbog toga postoji i mnogo različitih implementacija softvera za virtualizaciju.

Dve primene viruelizacije sa kojima se najčešće srećemo su *platformska virtualizacija* i *aplikativna virtualizacija*.

**Platformska virtualizacija** se bavi razdvajanjem operativnog sistema i hardverskih resursa koje operativni sistem koristi. Po stepenu apstrakcije resursa, softver za platformsku virtualizaciju možemo podeliti u sledeće grupe:

- **Potpuna virtualizacija** (*Full virtualization*) – osetljive instrukcije za CPU se zamenjuju binarnim prevodom u bezbedne instrukcije, ili se presreću od strane hardvera. Kod potpune virtualizacije bilo koji softver može da se izvršava u virtualnoj mašini. U ovu grupu spadaju IBM-ov CP/CMS, Sun-ov VirtualBox, ili VMware Workstation.
- **Hardverski potpomognuta virtualizacija** (*Hardware-assisted virtualization*) – CPU je zadužen za hvatanje osetljivih instrukcija. Može se primeniti samo ako CPU podržava ovakve operacije. Moguće je pokretanje nemodifikovanih operativnih sistema kao guest OS. Ovu tehniku koriste VMware Workstation, Xen, KVM.
- **Delimična virtualizacija** (*Partial virtualisation*) se koristi za pojedinačne namene, a ne ceo operativni sistem. U ovo grupu spadaju, na primer, programi za virtuelne optičke drajvove – OS na kome je ovakav softver instaliran praktično ne može da napravi razliku između fizički prisutnog optičkog medija i virtualnog.
- **Paravirtualizacija** je tehnika kojom se virtualnoj mašini prikazuje softverski interfejs koji je sličan, ali ne i identičan kao hardver koji ga pokreće. Ovo znači da guest OS mora biti prilagođen, modifikovan. Ovaj način virtualizacije koristile su starije verzije Xen softvera, pre verzije 3.0. Paravirtualizacija omogućava mnogo bolje performanse host i guest operativnih sistema na platformama koje su tradicionalno nezgodne za druge tipove virtualizacije (x86 platforma, na primer).
- **Virtualizacija na nivou operativnog sistema** (*Operating system-level virtualisation*) je metod kada sam operativni sistem ima ugrađene mogućnosti za virtualizaciju. Tu spada emulacija DOS-a pod novijim Windows operativnim sistemima, ili koncept „jail“ kod BSD operativnog sistema.

**Aplikativna virtualizacija** se bavi pokretanjem individualnih aplikacija u softverskom/hardverskom okruženju za koje originalno nisu bile predviđene. U ovu kategoriju spadaju:

- **Portabilne aplikacije** (*Portable applications*) su računarski programi koji se pokreću sa prenosnih memorijskih medija kao što su Flash diskovi ili optički mediji. Ovo su modifikovane aplikacije koje ne zahtevaju proceduru instalacije u OS na kojem se pokreću.
- **Međuplatformska virtualizacija** (*Cross-platform virtualisation*) omogućava da se softver kompajlirani za specifičan procesor i operativni sistem pokreću u nemodifikovanom obliku na drugom procesoru ili OS-u.
- **Virtual Appliance** je image fajl koji sadrži virtualnu mašinu za određenu virtualizacionu platformu.
- **Emulacija i simulacija** predstavljaju potpunu softversku implementaciju nekog procesora ili računarskog sistema.

## Specifičnosti virtuelizacije na x86 platformi

Ogromna većina današnjih računarskih arhitektura ima dvojnu hardversku organizaciju – postoji privilegovani i ne-privilegovani mod. U privilegovanim modu softver može da koristi

sve hardverske instrukcije, dok u neprivilegovanom ne može. Operativni sistem obezbeđuje rezidentni program koji se naziva kernel. Korisnički (*user-space*) programi izvešavaju samo neprivilegovane instrukcije. Kada korisničkom programu zatreba privilegovana instrukcija, on šalje sistemski poziv kernelu, kako bi imao pristup pristup privilegovanoj funkciji – na primer U/I operacijama. Mada ovakav pristup radi vrlo dobro u mnogim primenama, on ima nekoliko fundamentalnih nedostataka:

- Samo jedan program može da ima pristup mašinskom interfejsu. Samim tim, može da se pokrene samo jedan kernel. Ništa drugo, bio to drugi kernel (istog ili različitog operativnog sistema), ili aplikacija koja mora direktno da komunicira sa hardverom (na primer, testiranje, debugging ili dijagnostika niskog nivoa), ne može da se izvršava paralelno sa aktivnim kernelom.
- Ne mogu da se izvršavaju nikakve aktivnosti koje mogu da poremete sistem koji je u radu (upgrade, migracija, debugging sistema, itd)
- Nekom softveru ne može jednostavno da se pruži iluzija hardverske konfiguracije koja nije fizički prisutna (više procesora, različite memoriske konfiguracije, itd.)

Za razliku od mainframe računara, x86 arhitektura nije od početka dizajnirana da bi podržavala potpunu virtuelizaciju. Postoji najmanje sedamnaest instrukcija u x86 procesorima koje ih čine veoma teškim za „virtuelizovanje“. Privilegovane instrukcije IA-32 procesora izazivaju General Protection Exception kada se izvrše u neprivilegovanom modu. Upravljanje TLB-om (Translation Lookaside Buffer) na IA-32 procesorima je hardverski implementirano. Jedan ulaz u TLB-u nemože da se obeleži, na primer, identifikatorom adresnog prostora (ASID – Adress Space IDentifier), što bi VMM-u olakšalo upravljanje adresnim prostorima virtualnih mašina i njihovih kernela. Druge arkitekture kao što su SPARC ili Alpha koriste softversko upravljanje TLB-om, koje se mnogo lakše virtualizuje.

Većina savremenih sistema za virtuelizaciju na x86 računarima rešava ovaj problem na taj način, što implementiraju mehanizam za „presretanje“ ovakvih instrukcija kod njihovog nastanka, i zamenjuju ih u nizom bezbednih instrukcija. Instrukcije koje nisu problematične izvršavaju se bez intervencije. Na ovaj način se postiže brzina izvršavanja koja je približna performansama host hardvera, a održava se potpuna softverska kompatibilnost.

## **Virtuelizacija u desktop okruženju**

U zadnjih nekoliko godina virtuelizacija doživljava renesansu u sferi PC računara. Pojava softverskih rešenja koja omogućavaju pravljenje i pokretanje virtuelnih mašina na desktop računarima je veoma opravdana i može da bude vrlo korisna.

Prateći opšti trend razvoja hardverske komponente računara, došlo je do preslikavanja problema sa serverske scene na polje personalnih računara. Veoma je teško naći aplikaciju koja uspeva da optimalno iskoristi ogromnu sirovu snagu sasvim prosečnog kućnog ili kancelarijskog računara. To polazi za rukom jedino industriji za elektronsku zabavu. Doduše, ona danas i jeste glavni pogonski agregat razvoja računara. Kao rezultat toga, imamo kućne i kancelarijske računare koji su većinom vrlo neiskorišćeni.

Ima mnogo situacija kada je korišćenje desktop virtuelizacije zgodno. Navešću nekoliko primera iz ličnog iskustva.

### **Testiranje operativnog sistema**

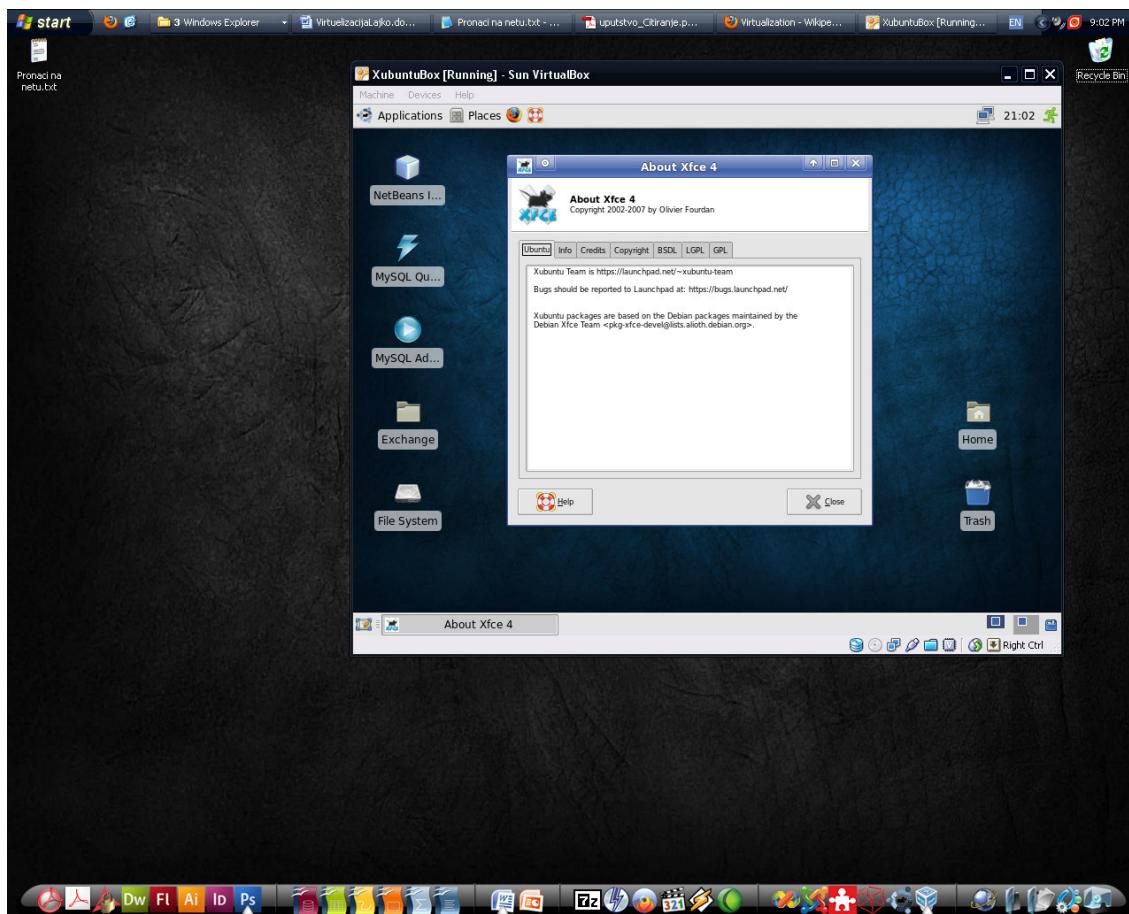
Operativni sistem Linux postoji već duže vreme. To je operativni sistem otvorenog izvornog koda, i potpuno je besplatan. U poslednjih nekoliko godina doživeo je pravi procvat. U

mnogim svojim karakteristikama sustigao je komercialne desktop operativne sisteme, a u mnogima ih i pretekao. Mada nije previše popularan u ulozi personalnog računara, smatrao sam da je došlo vreme da ga i lično isprobam.

Međutim, svako ko je više puta reinstalirao Windows na svom računaru zna koliko je to mukotrpan posao. Sama procedura instalacije je manji deo problema, najveći problem za mene je kako da sačuvam lične podatke, podešavanja programa i dokumente koje treba da prenesem na novinstalirani OS. Ovakvi rizici se naravno drastično uvećavaju kada treba instalirati OS sa kojim nisam ranije imao nikakvog iskustva.

Tu na scenu stupa virtualna mašina. Na njoj mogu da isprobam celu proceduru instalacije, pogledam šta i kako funkcioniše, i sve to bez rizika da će osetiti host sistem ili podatke na njemu. Na slici 2. prikazan je operativni sistem Xubuntu Linux operativni sistem u virtuelnoj mašini na desktopu mog Windows XP operativnog sistema.

Posle isprobavanja, instalirao sam Ubuntu Linux paralelno sa Windows-om, i koristim ih podjednako. Testirao sam još neke operativne sisteme, na primer FreeBSD, openSolaris i ReactOS.



Slika 2. Xubuntu Linux guest na Windows XP host operativnom sistemu

### Razdvajanje razvojnih platformi

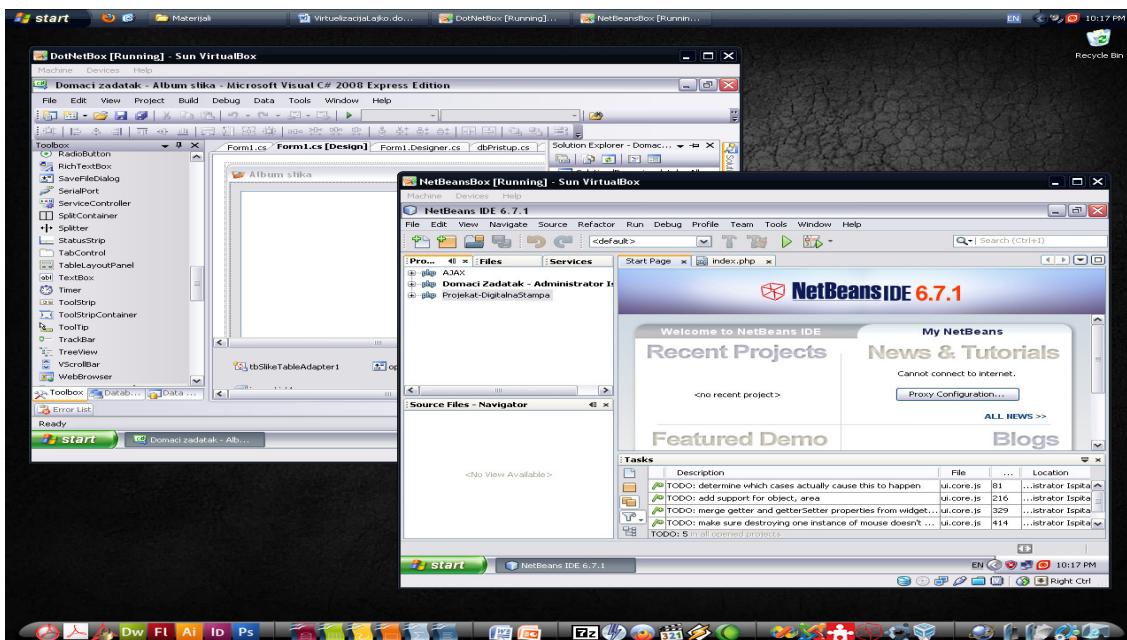
Upisao sam 2008. godine jednogodišnju školu iz web programiranja na IT Akademiji. Zaista sam uživao u učenju novih programerskih veština i tehnika, međutim, vrlo brzo sam naišao na jedan ozbiljan problem.

Pošto smo paralelno radili kurseve programiranja na nekoliko različitih platformi, bilo je potrebno da ih sve instaliram i na kućnom računaru. Problem u tome je što bi u tom slučaju trebao da na istom računaru instaliram Visual Studio za programiranje u C# programskom jeziku, Eclipse razvojno okruženje za Java programiranje, i bateriju softvera za PHP programiranje.

Pri tome, Visual Studio radi sa IIS web serverom i MS SQL Serverom baza podataka, a za PHP je potreban Apache web server i MySQL server baza podataka. Kako sam zaključio da nikako ne može biti dobro da sve to zajedno instaliram na računar, rešenje koje sam upotrebio je bilo postavljanje po jedne virtuelne mašine za svako razvojno okruženje.

Posle malo eksperimentisanja, završio sam sa tri virtualne mašine. Na jednoj su instalirani Visual Studio 2008 Express Edition, IIS web server i Microsoft SQL Server 2005 Express Edition za C# i ASP.net programiranje. Na drugoj su WAMP (Window-Apache-MySQL-PHP) server i Netbeans IDE za PHP programiranje. Na trećoj virtualnoj mašini je instaliran Xubuntu operativni sistem, Eclipse razvojno okruženje i MySQL baza podataka za programiranje u Java programskom jeziku. Za ovu poslednju nije bilo neophodno da radi pod linuxom, ali mi je jako interesantno da vidim kako isti Java program radi i na njemu, i na Windowsu.

Najkorisnija stvar kod ovakvog rešenja je to što su ova razvojna okruženja učaurena u sopstvenu virtuelnu mašinu. Serveri koje zahtevaju pokreću se samo kada za njih postoji potreba, i ne dolaze u međusobni konflikt. Druga korisna stvar je da kad dođe do situacije da se host operativni sistem mora reinstalirati, ne moraju se reinstalirati i razvojna okruženja. Ona su ponovo dostupna u nepromjenjenom obliku i na svežem host OS-u.



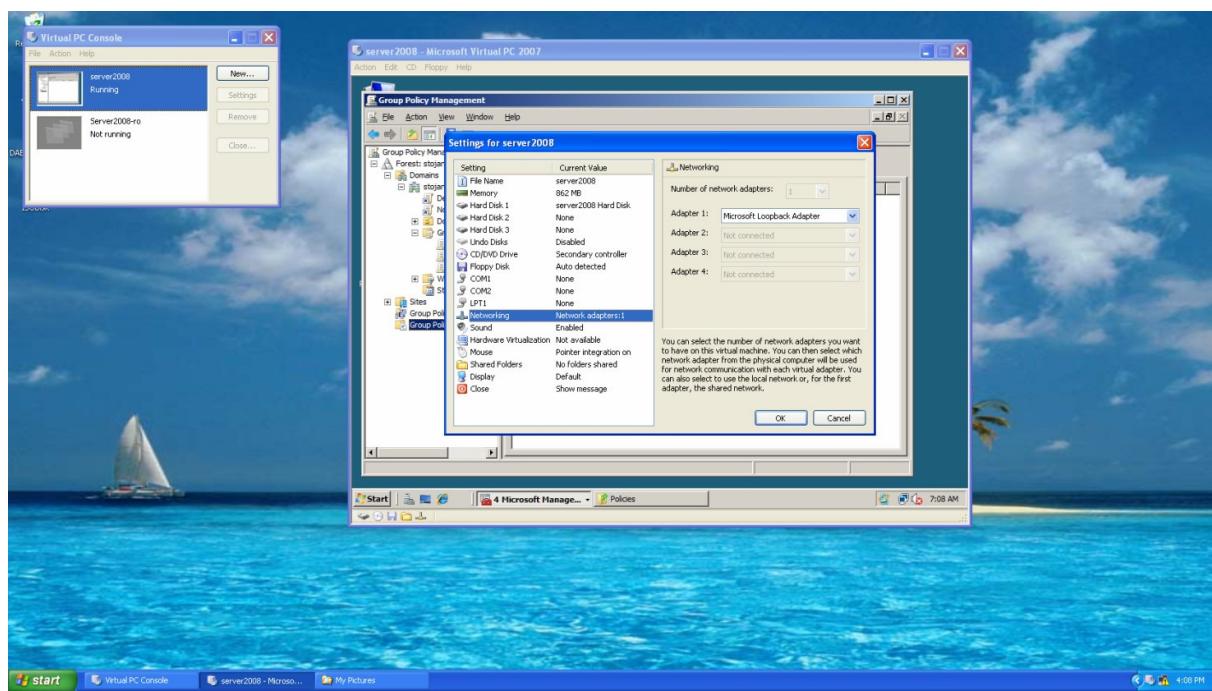
Slika 3. Visual Studio IDE i NetBeans IDE učaureni u svoje virtuelne mašine

### **Simuliranje mreže i mrežnih konfiguracija na jednom računaru**

Još jedna korisna primena desktop virtuelizacije je simulacija mreže na jednom računaru. Upravo na taj način se izvodi nastava na kursevima administracije mreža i servera na IT Akademiji.

Svaka virtuelna mašina ima sopstveni virtuelni harver za priključivanje na mrežu. Ovo se može iskoristiti na tako, što se na računaru pokrene više virtuelnih mašina. Obično je dovoljno da to bude jedan „server“ i jedan „klijentski“ računar, mada se može pokrenuti i više od toga. Pošto svaka virtuelna mašina ima sopstvenu mrežnu karticu, one mogu da se umreže međusobno, kao i sa računaram domaćinom.

Na ovaj način polaznici su u mogućnosti da isprobavaju razna rešenja za umrežavanje i konfiguraciju parametara servera i klijentskih računara, a da pri tom nema rizika od oštećenja fizičkih komponenti, kao ni mukotrpog razvlačenja kablova (i kablovi su „virtuelni“).



**Slika 4. Windows Server 2008 pokrenut u virtuelnoj mašini na Windows XP operativnom sistemu**

## Virtualizacija servera

### Konsolidacija servera

Upravo je ovaj koncept pokrenuo razvoj virtualizacije sa „mrtve tačke“ do koje se došlo tokom osamdesetih i devedesetih godina prošlog veka.

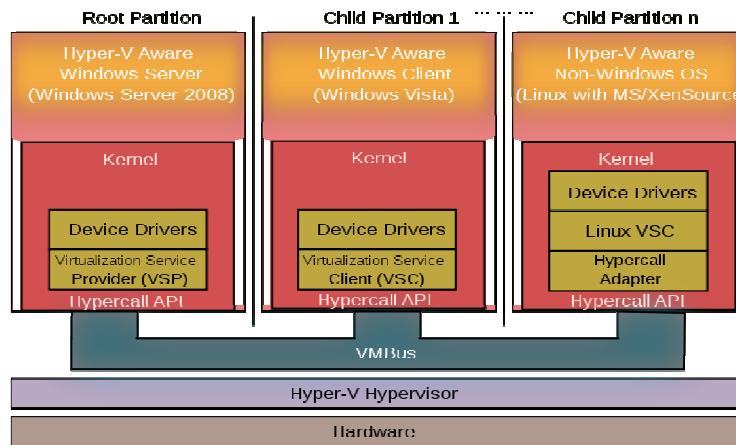
Hardver baziran na x86 platformi postao je dovoljno jak da prevaziđe svoju osnovnu namenu. PC računari, koji su prvenstveno dizajnirani za radne stанице i kancelarijske poslove, postali su sposobni da preuzmu uloge servera. Jeftini operativni sistemi opšte namene kao što je Windows, a kasnije i besplatni Linux, veoma brzo su potisnuli skupe i nemenski dizajnirane mainframe računare. Popularizacijom Interneta ova tendencija je postala još izraženija. Internet hosting kompanije su mogle jednostavno da proširuju svoje kapacitete dodavanjem nekoliko novih i jeftinih web servera.

Međutim, daljnji razvoj x86 hardvera pomerio je ovaj koncept u drugu krajnost: u situaciji kada postoji nekoliko servera koji izvršavaju specifične zadatke (na primer print server, web server, server baze podataka, itd.), ovi serveri pojedinačno postaju vrlo slabo iskorišćeni. Prosečna iskorišćenost današnjih servera iznosi 5-15% njihovog punog kapaciteta. Ovo povlači sa sobom rast troškova održavanja sistema koji je višestruko veći od porasta njegovih performansi. U troškove održavanja spadaju troškovi napajanja, hlađenja, mrežne infrastrukture, administracije, pa i troškovi nekretnina, u kojima su ovakvi sistemi smešteni.

Virtualizacija se nameće kao prirodno rešenje ovog problema. Na jednom fizičkom računaru moguće je pokrenuti nekoliko virtualnih, koji se konfigurišu kao različiti i potpuno nezavisni serveri. Upotreboom modernih rešenja za virtualizaciju, na ovaj način je moguće podići iskorišćenost hardvera sa 10-15% na čitavih 80%.

Na tržištu postoji nekoliko rešenja koja omogućavaju konsolidaciju servera putem virtualizacije. Najpoznatija su VMware-ovi *ESX* i *vSphere*, Microsoftov Windows Server 2008 R2 *Hyper-V*, kao i open source rešenja *Xen* i *Kernel-based Virtual Machine (KVM)*. Sva ova rešenja su u osnovi prilično ujednačena, i pružaju istu funkcionalnost čak i ako su bazirana na potpuno drugaćoj tehnologiji. Zato softverske kompanije koje se bave tehnologijama za virtualizaciju, daljni razvoj sve više usmeravaju ka razvoju alata za administraciju i menadžment ovakvih sistema.

Uz pomoć ovakvih alata, praktično možemo potpuno odvojiti logičku infrastrukturu računarskog sistema od njegove hardverske implementacije. Na jednom računaru može biti pokrenuto nekoliko virtualnih servera, dok dok u isto vreme i u istom sistemu nekoliko kompjutera može biti „objedinjeno“ u jedan virtualni superkompjuter. Ovakva infrastruktura najčešće se opisuje terminom *Virtualna infrastruktura*.



Slika 5. Microsoft-ova Viridian arhitektura sa Hyper-V hipervizorom

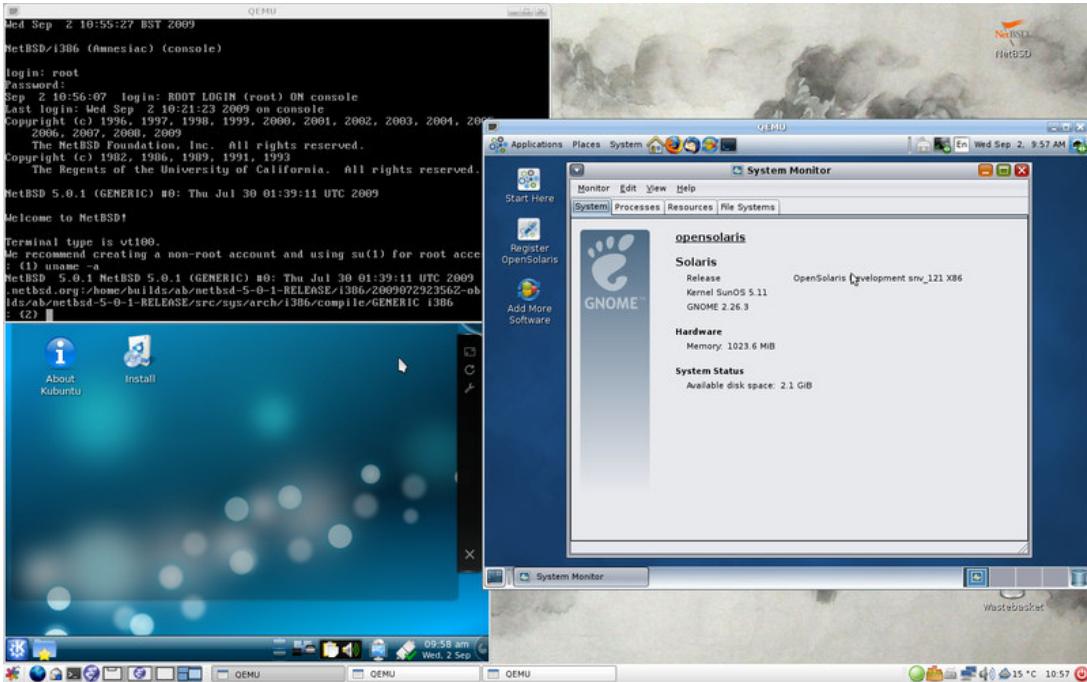
### **Remote desktop virtualne mašine**

Jedan od interesantnih načina korišćenja tehnika virtualizacije pronašao sam čitajući dokumentaciju za Sun-ov VirtualBox. Radi se o serveru, čija namena je hostovanje virtualnih desktop-a, kojima se pristupa preko RDP protokola. Tipično, Remote Desktop Protocol se koristi za daljinski pristup računaru. Slika i zvuk se pomoću ovog protokola šalju od udaljene mašine prema klijentu, a unos preko tastature i miša u obrnutom pravcu.

Scenario za ovakvu upotrebu počeo bi instaliranjem severa. Na njemu bi mogao biti bilo koji operativni sistem, ali recimo da je to Linux, pošto on može da radi bez pokretanja grafičkog korisničkog interfejsa, čime se uštede značajni računarski resursi. Zatim se pokrene VMM, u ovom slučaju VBoxHeadless server. Ova verzija VirtualBox-a ne koristi nikakav grafički interfejs, kompletan grafički prikaz sa pokrenutih virtualnih mašina emituje isključivo preko ugrađenog VRDP servera.

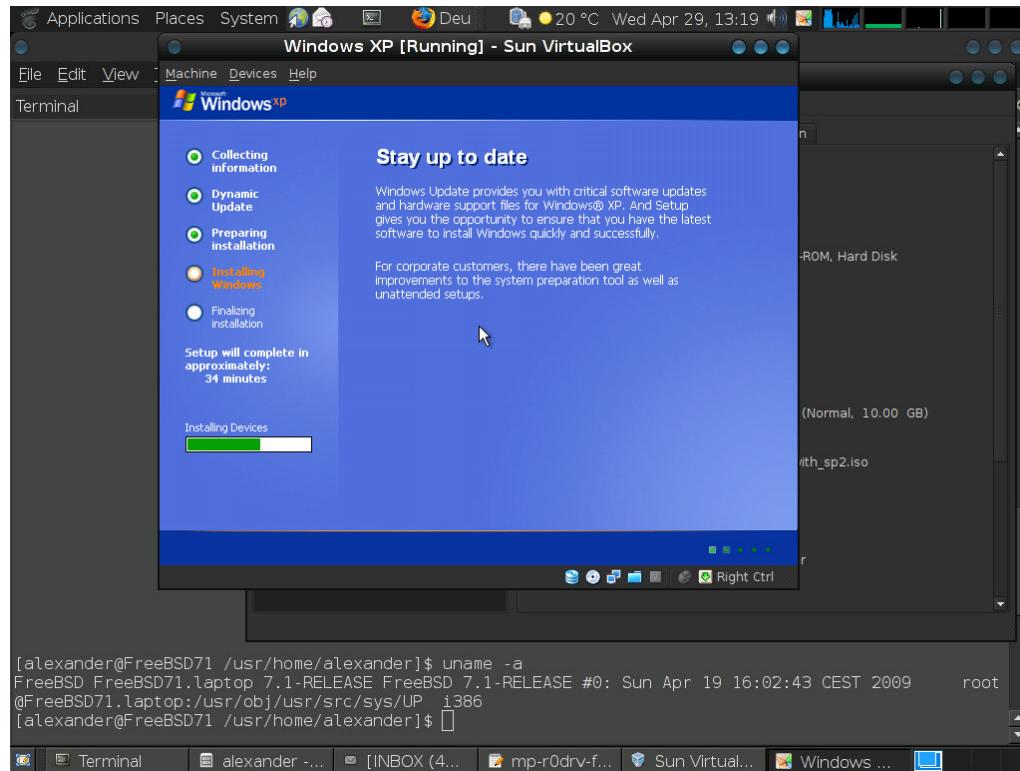
Virtualnim mašinama koje se hostuju na ovakav način, može se pristupiti preko lokalne LAN mreže ili preko Interneta. Pri tome, jedino što je na klijentskom računaru porebno je standardni klijentski softver za RDP protokol, koji postoji za sve poznate operativne sisteme.

Mogućnosti upotrebe ovakvog sistema su veoma raznolike, počev od rada na „kompanijskom računaru“ od kuće ili sa odmora na Zlatiboru, pa do izgradnje infrastrukture sa veoma jeftinim (ili namenskim) klijentskim računarima, koji se praktično koriste kao terminali prema virtualnim mašinama.

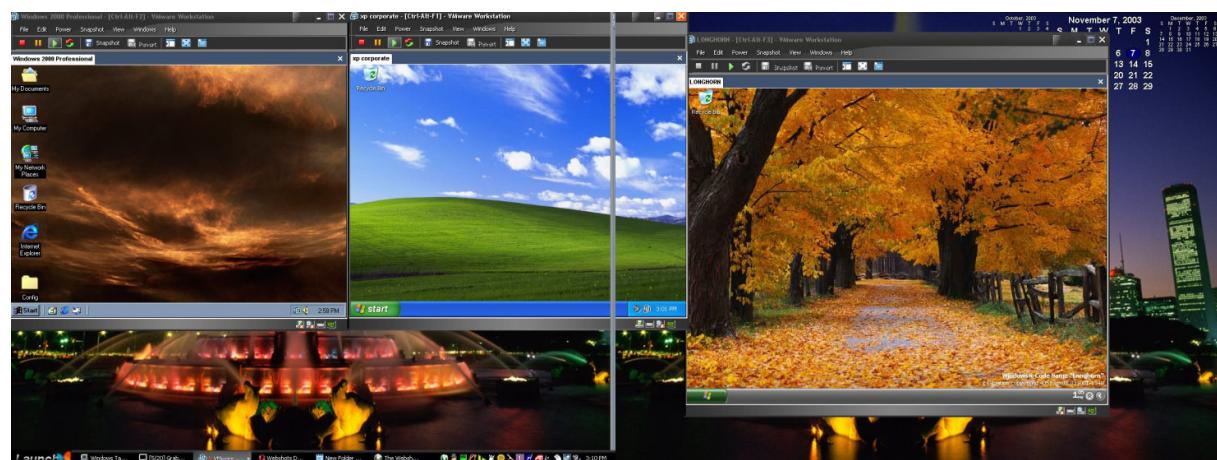


Slika 6. Kernel-based Virtual Machine sa nekoliko različitih operativnih sistema u VM

## DODATAK: Ilustracije



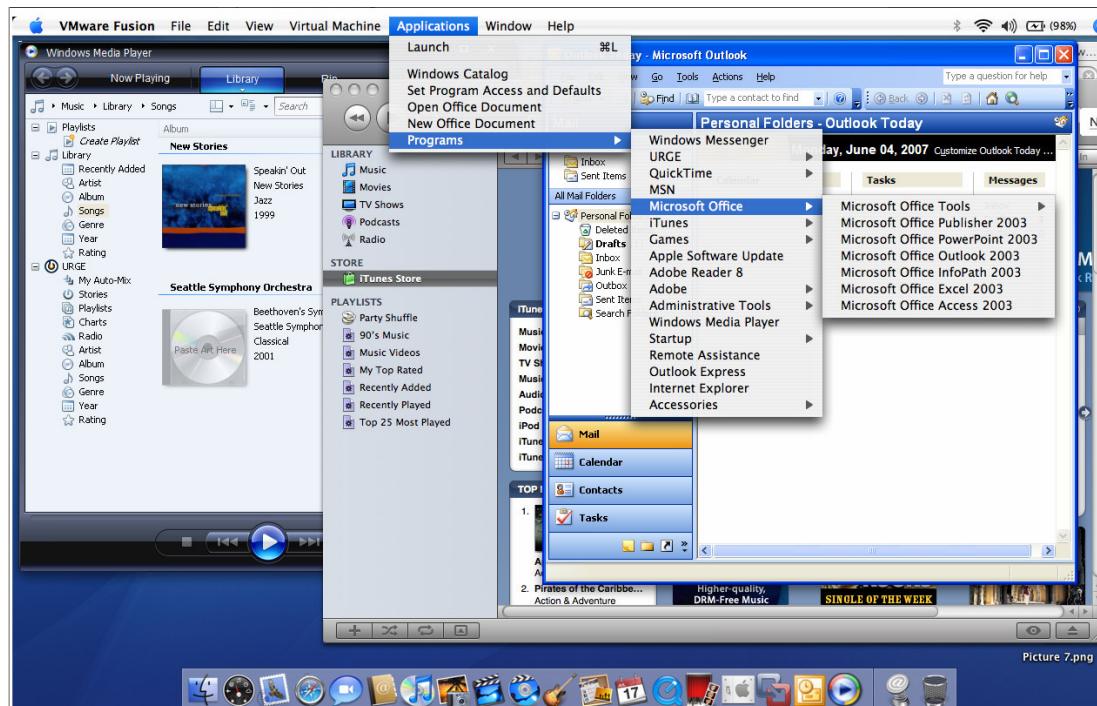
Slika 7. Windows XP u virtualnoj mašini na FreeBSD host operativnom sistemu (*Sun VirtualBox*)



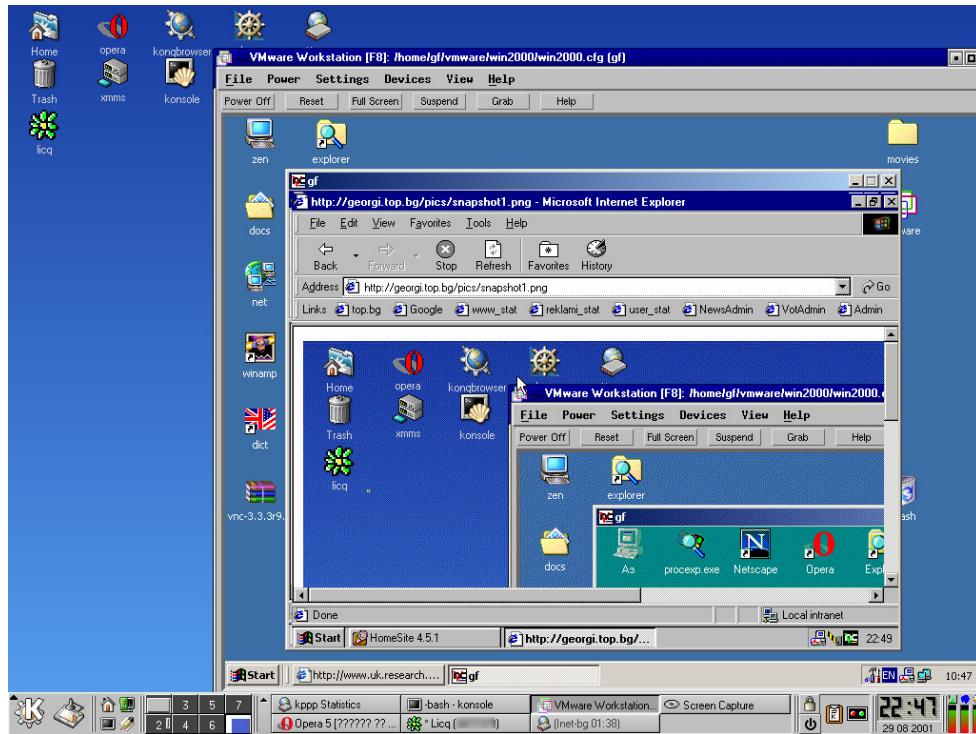
Slika 8. Nekoliko različitih Windows virtualnih mašina na Windows Vista hostu (*VMware Workstation*)



Slika 9. Windows programi na desktopu MacOSX (Sun VirtualBox seamles mod)



Slika 10. Windows programi na desktopu MacOSX (VMware Fusion)



Slika 11. Windows 2000 virtualna mašina na Linux hostu (VMware Workstation)



Slika 12. Windows 3.1 virtualna mašina na Windows Vista hostu (VMware Workstation)

## ZAKLJUČAK

Svetsko računarsko tržište je nepredvidivo koliko je i veliko. Razvoj računarskih tehnologija ima nezgodnu osobinu da odluta u prvcima koji su pre samo nekoliko godina izgledali potpuno neverovatno.

Dominacija x86 platforme je globalna. Poslednji bastioni drugih hardverskih sistema, poput Motorolinog PowePC-a ili Sunovog SPARC-a, popustili su pred naletom nekada skromne male sive kutije, koja se zvala PC. Apple MacOS i Sun Solaris operativni sistemi razvijaju se prvenstveno za rad na x86 hardveru.

Ironija cele situacije je ta, da upravo sada softver postaje sve manje zavistan od hardverske podloge na kojoj se izvršava. Moderni aplikativni softveri uglavnom se razvijaju i pokreću unutar izolovanih okruženja kao što su Java Virtualna Mašina ili Common Language Runtime. Za većinu tradicionalno desktop orientisanih aplikacija postoji sasvim pristojna online alternativa, koja je često čak i zgodnija za upotrebu.

Ne možemo pouzdano predpostaviti koja je sledeća revolucija koja će pogoditi IT industriju. Ono što jeste prepoznatljiva tendencija, je razdvajanje logičke i hardverske komponente računara. Sa jedne strane, imamo mogućnost pokretanja potpuno nezavisnih virtualnih mašina na jednom fizičkom računaru. Sa druge, ogroman broj malih računara povezuje se u celine, koje se ponašaju i koriste kao jedan virtualni superkomputer. Definitivno rešenje je verovatno negde u sredini. A virtualizacija je upravo manifestacija ovakve filozofije.

U ovom radu samo sam zagrebaoo po površini oblasti. Bilo bi još mnogo toga da se kaže i pokaže. Odustao sam od svoje prvobitne namere da detaljnije opišem i uporedim nekoliko poznatijih softverskih rešenja za virtualizaciju. U prvom redu zato, što sva ta rešenja u osnovi nude istu funkcionalnost. Čak do te mere, da se u većini slučajeva Virtualna mašina kreirana u programu jednog proizvođača može pokrenuti u VMM-u drugog. Razlika između njih se praktično svodi na implementirane alate za menadžment. A tu važi staro pravilo: „koliko para toliko muzike“. Komercijalni alati su lakši za upotrebu, kompleksniji i kompletnejši. Besplatna rešenja ih ipak slede u stopu: uz malo više uloženog truda mogu da urade isti posao.

## LITERATURA

- [1] "An Introduction to Virtualization", Amit Singh, Januar 2004,  
<http://www.kernelthread.com/publications/virtualization/> (pristupano 5.12.2009.)
- [2] "Sun VirtualBox User Manual", Sun Microsystems, 2009,  
<http://www.virtualbox.org/manual/UserManual.html> (pristupano 5.12.2009.)
- [3] "VMWare Datacenter Solutins", VMWare, 2009,  
<http://www.vmware.com/solutions/datacenter/> (pristupano 5.12.2009.)
- [4] "VMWare Virtualization Basics", VMWare, 2009,  
<http://www.vmware.com/virtualization/> (pristupano 5.12.2009.)
- [5] "VMWare Virtualization Management", VMWare, 2009,  
<http://www.vmware.com/solutions/virtualization-management/> (pristupano 5.12.2009.)
- [6] "Sun VirtualBox User Manual", Microsoft, 2009,  
<http://www.microsoft.com/virtualization/en/us/products-management.aspx> (pristupano 5.12.2009.)
- [7] "Virtualization", Wikipedia, the free encyclopedia, 2009,  
<http://en.wikipedia.org/wiki/Virtualization> (pristupano 19.12.2009.)

## INDEKS SLIKA

<i>Slika 1.</i> IBM System 370, računar na kojem je radio VM/370 .....	6
<i>Slika 2.</i> Xubuntu Linux guest na Windows XP host operativnom sistemu .....	9
<i>Slika 3.</i> Visual Studio IDE i NetBeans IDE učaureni u svoje virtuelne mašine .....	10
<i>Slika 4.</i> Windows Server 2008 pokrenut u virtuelnoj mašini na Windows XP operativnom sistemu .....	11
<i>Slika 5.</i> Microsoft-ova Viridian arhitektura sa Hyper-V hipervizorom .....	12
<i>Slika 6.</i> Kernel-based Virtual Machine sa nekoliko različitih operativnih sistema u VM .....	13
<i>Slika 7.</i> Windows XP u virtualnoj mašini na FreeBSD host operativnom sistemu ( <i>Sun VirtualBox</i> ) .....	14
<i>Slika 8.</i> Nekoliko različitih Windows virtualnih mašina na Windows Vista hostu ( <i>VMware Workstation</i> ) .....	14
<i>Slika 9.</i> Windows programi na desktopu MacOSX ( <i>Sun VirtualBox seamless mod</i> ) .....	15
<i>Slika 10.</i> Windows programi na desktopu MacOSX ( <i>VMware Fusion</i> ) .....	15
<i>Slika 11.</i> Windows 2000 virtualna mašina na Linux hostu ( <i>VMware Workstation</i> ) .....	16
<i>Slika 12.</i> Windows 3.1 virtualna mašina na Windows Vista hostu ( <i>VMware Workstation</i> ) .....	16