

# Administracija računarskih mreža

Virtuelizacija

# Materijal

- sedmo poglavlje (Network Functions Virtualization: Concepts and Architecture) knjige “Foundations of Modern Networking” (Stallings, 2015)
- web izvori navedeni u prezentaciji

# Sadržaj

- Virtuelizacija
- Virtuelne mašine
- Kontejneri
- Virtuelizacija mrežnih funkcija

# Virtuelizacija

- Stvaranje privida postojanja nečeg
- Virtuelna realnost
- Virtuelna memorija
- Virtuelni LAN
- Virtuelne privatne mreže
- Virtuelne mašine
- NFV (*Network Functions Virtualization*)

# Ideja virtuelizacije

- Pretvaranje fizičkih resursa u logičke (virtuelne)
- Apstrakcija fizičkog hardvera kroz softverske interfejsse
- Korisnici, aplikacije i upravljački softver ne zanimaju se fizičkim detaljima

# Virtuelizacija u praksi

- Prostora za čuvanje podataka (*storage*)
  - Razdvajanje logičkog od fizičkog prostora
- Hardvera
  - Virtuelne mašine
- Mreže
  - kombinovanje dostupnih fizičkih mrežnih resursa u virtualne koji se dodjeljuju po potrebi

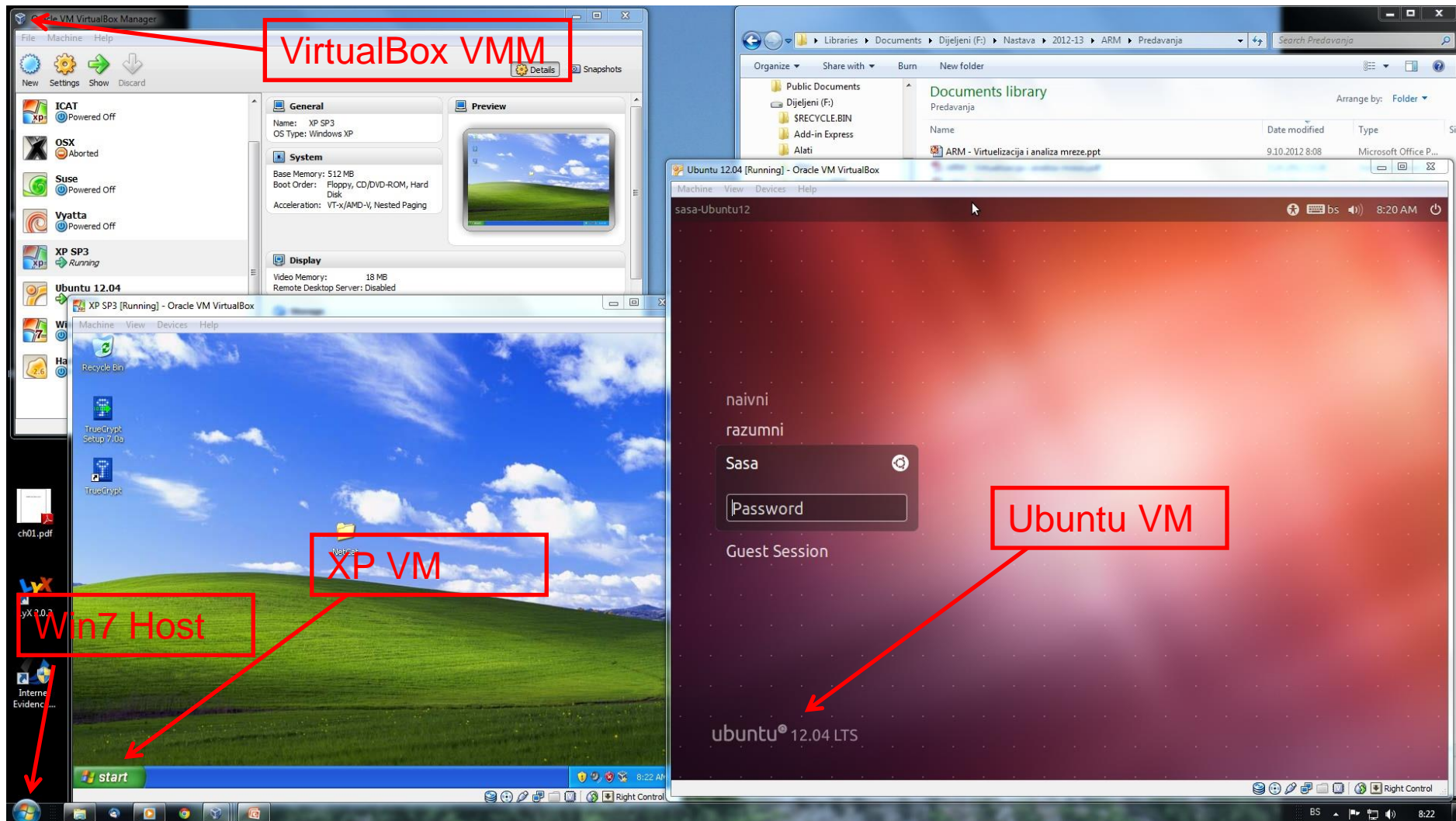
# Virtuelizacija hardvera

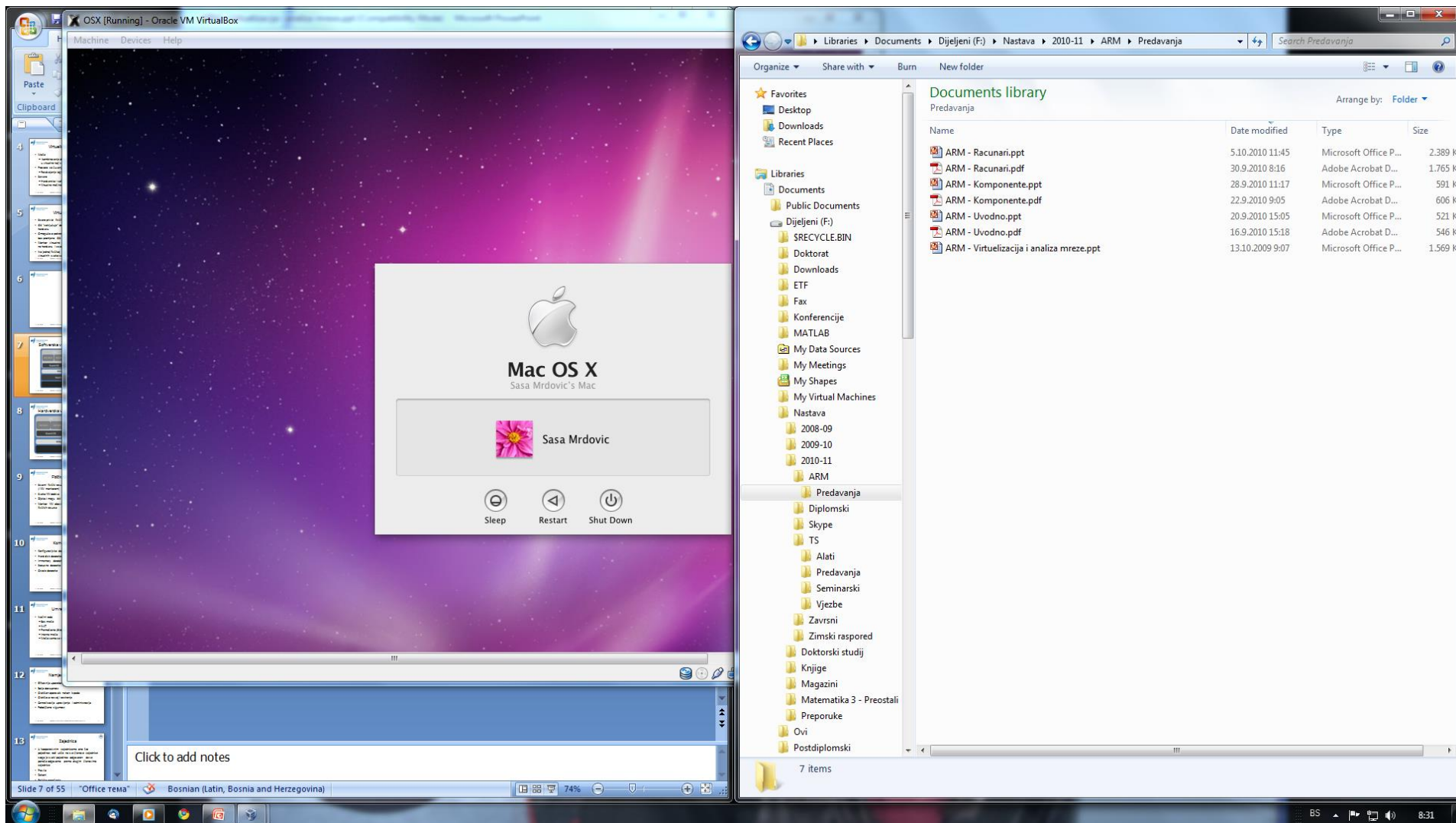
- Aplikacije pisane za OS
- Na računaru jedan OS se izvršava u jednom trenutku
- Virtuelizacijske tehnologije
  - na jednom računaru istovremeno izvršavanje više OS (više virtuelnih mašina - VM)
  - počeo IBM 1970.-ih
  - aktuelizirano 2000.-ih
    - hardver nadrastao softver – postao neiskorišten

# Virtuelna mašina

- Stvara privid fizičke mašine (hardvera)
- OS “zaključuje” da se izvršava direktno na hardveru
- Omogućava pokretanje OS i procesa u njemu bez promjene OS ili programa
- Monitor virtuelne mašine izvršava se direktno na hardveru i stvara ovaj privid
- Na jednoj fizičkoj mašini može biti više virtuelnih svaka sa svojim OS

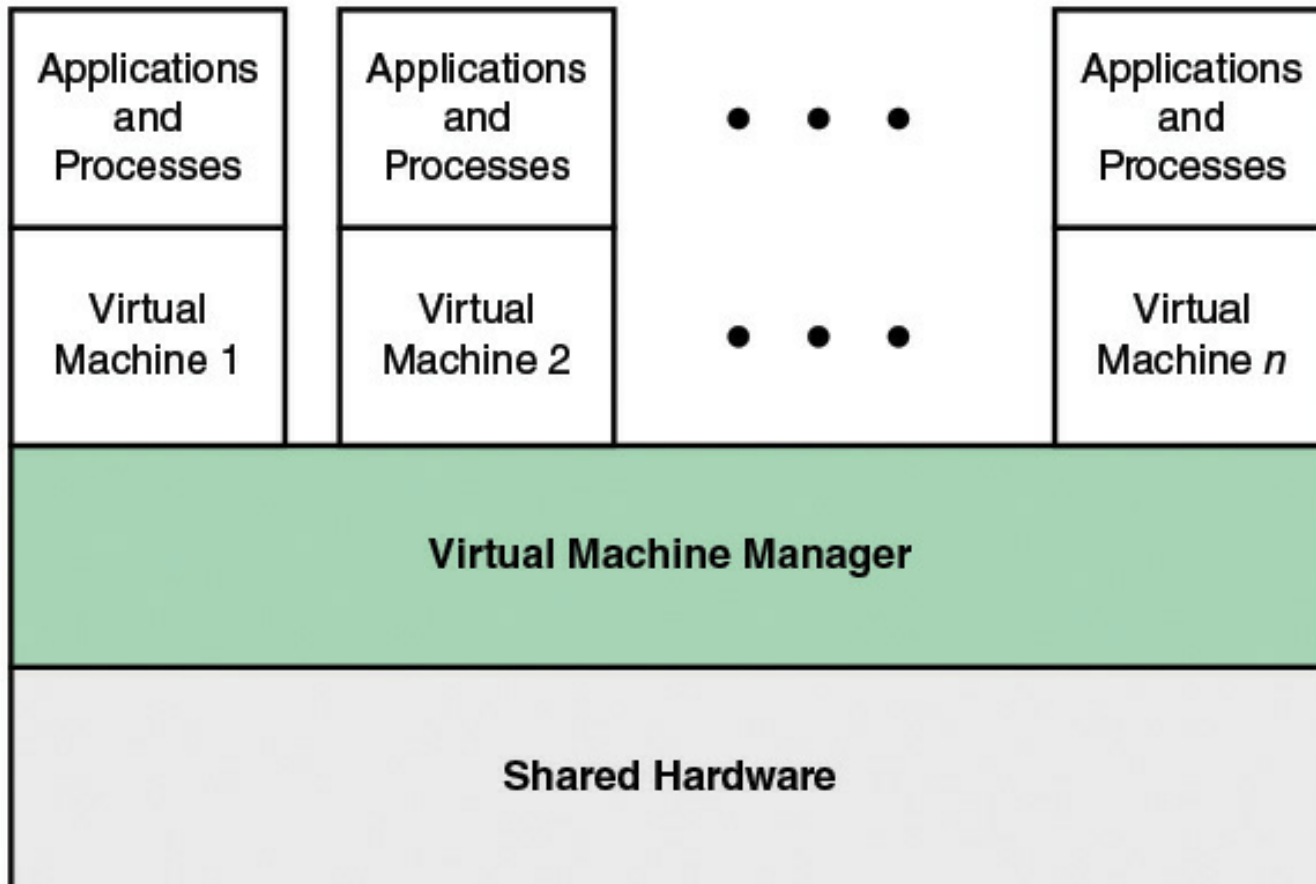






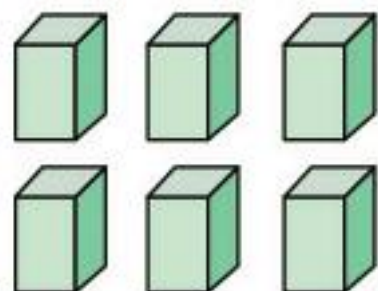
# Koncept

- Virtual machine monitor/manager (VMM) ili *hypervisor*

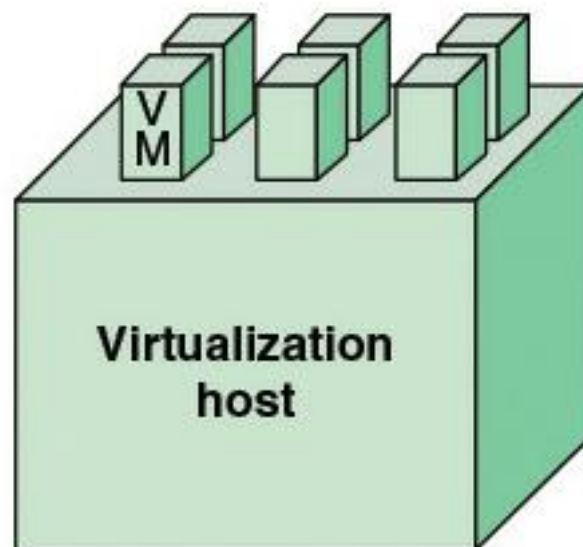


# Konsolidacijski odnos (*ratio*)

- Broj VM koje računar podržava
  - Inicijalno 4:1 do 12:1



**Physical servers**



**6:1 consolidation ratio**

# Korist od virtuelizacije

- Potreban manji broj računara (u data centru)
- Manji troškovi (konsolidacija)
  - manja potrošnja energije
  - manje hlađenja potrebno
  - manje kablova
  - manje mrežnih *switch*-eva
  - manje fizičkog prostora
- Danas je više virtuelnih nego fizičkih servera

# Korist od virtuelizacije (2)

- Hardver postaje apstraktan resurs
  - OS i aplikacije dobijaju hardvera koliko im treba
  - nije im bitno kako (od kog fizičkog hardvera)
  - bolja iskorištenost hardvera
  - laka i brza migracija VM na drugi hardver
    - povećane/smanjene potrebe
    - raspoređivanje opterećenja
    - otkaz hardvera



# Arhitekture

- VMM posrednik (*proxy*) između VM i hardvera
- VM softver koji imitira karakteristike fizičkog računara
  - procesor(i)
  - RAM
  - prostor za pohranu
  - mrežne veze
- VM se pokreće kao fizički server sa OS i aplikacijama

# Fizički resursi i VM

- Stvarni fizički resursi se raspoređuju među VM (i VM monitorom)
- Svaka VM dobiva dio resursa
- Dijelovi mogu biti različiti
- Monitor VM obavlja mapiranje virtuelnih i fizičkih resursa
- Određena degradacija performansi zbog posredovanja VMM



# VM je skup datoteka

- Konfiguracijska datoteka
  - opis virtuelnog računara (CPU, RAM, HD, NIC, I/O, ...)
- Hard disk datoteka(e)
  - virtuelni hard disk
- In-memory datoteka
- Statusna datoteka
- Ostale datoteke

# Pravljenje kopije VM

- Kopiranje VM datoteka
- Time se kopira sve:
  - podaci
  - OS
  - aplikacije
  - hardverska konfiguracija
- Mnogo jednostavnije nego za fizički server

# Umnožavanje VM

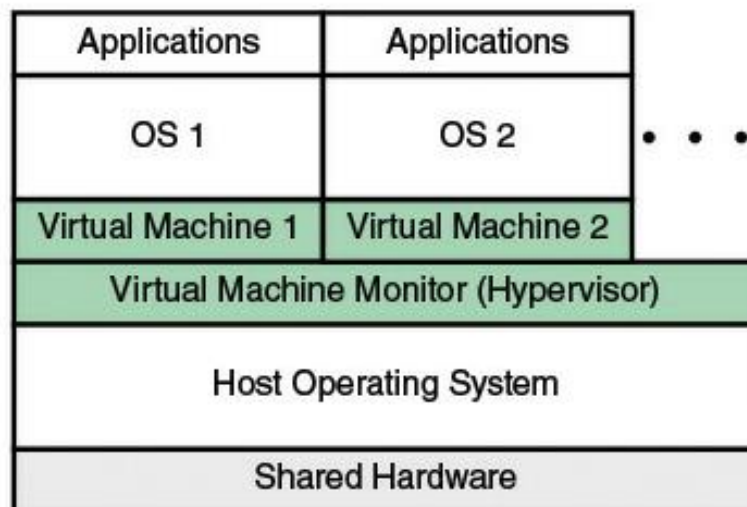
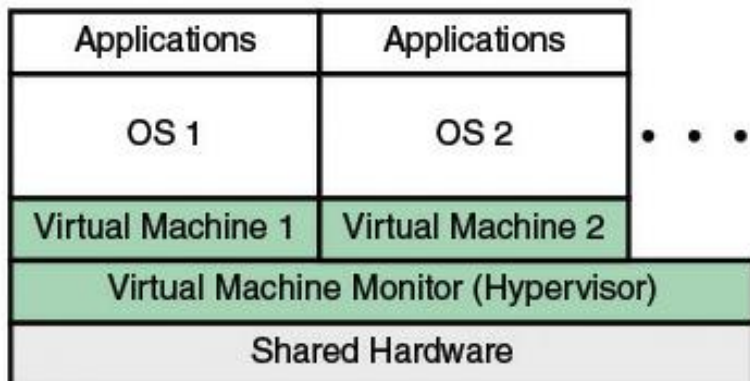
- Kopiranje VM datoteka
  - uz promjenu imena servera i IP adrese (bar)
- Mnogo brže od fizičkih servera
- Upotreba predložaka (urneka, *template*)
  - standardizovana HW i SW konfiguracija
  - unijeti samo jedinstvene identifikatore
  - softver pravi VM na osnovu ovoga

# VM za povećanje pouzdanosti

- Brzo prebacivanje VM sa jednog fizičkog servera na drugi
- Rezervni VM na drugom fizičkom serveru
  - izvršava se paralelno sa primarnim
    - trenutno prebacivanje u slučaju ispada
- Premještanje VM bez gašenja
  - Live Migration, vMotion
  - Ako nema dovoljno HW resursa prebaci na drugi HW

# Tipovi hipervizora (VMM)

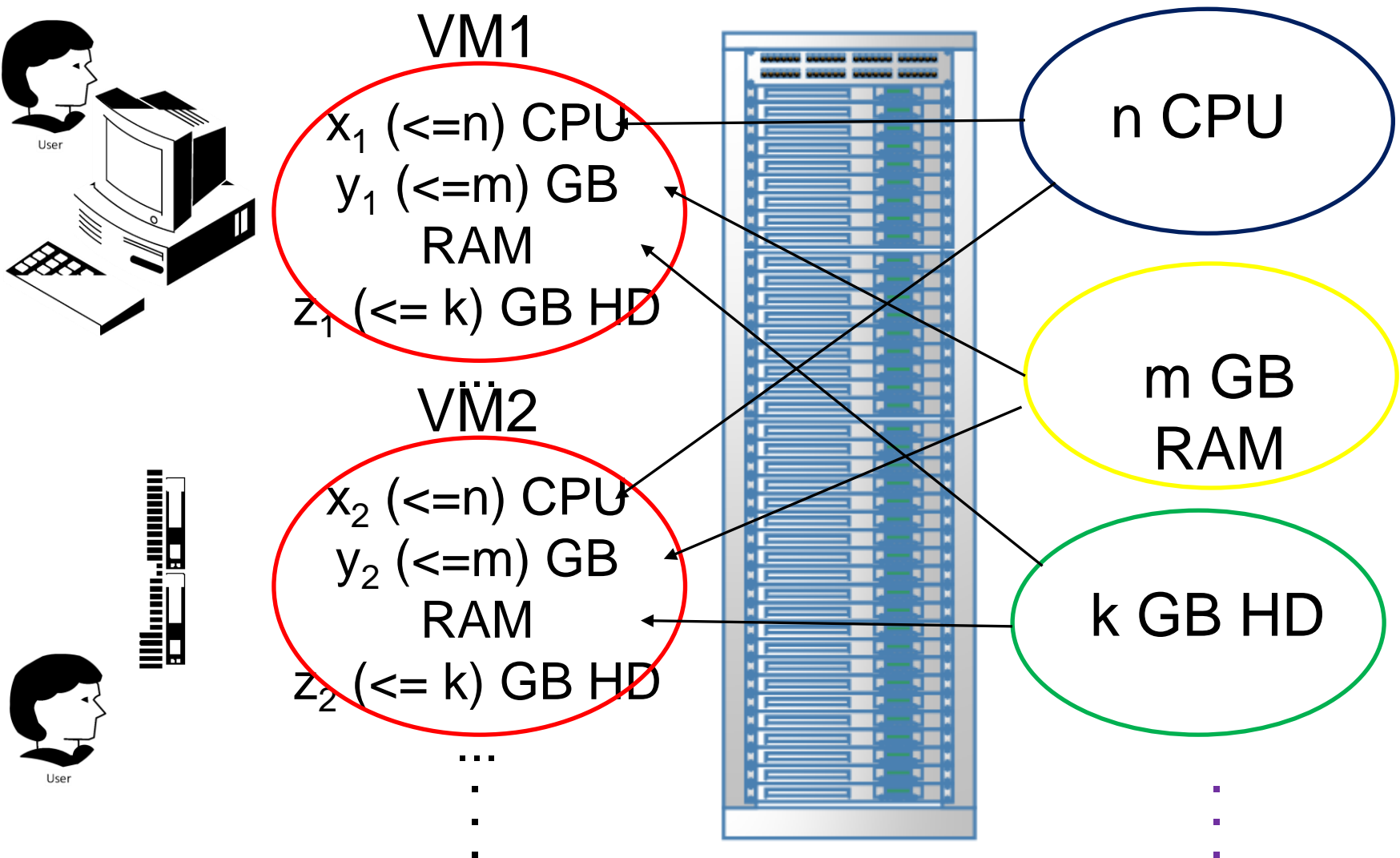
- Ima li OS između VMM i HW
  - Tip 1: Nema
    - VMware ESXi, Microsoft Hyper-V, open source Xen
  - Tip2 : Ima
    - VMware Workstation i Oracle VM Virtual Box

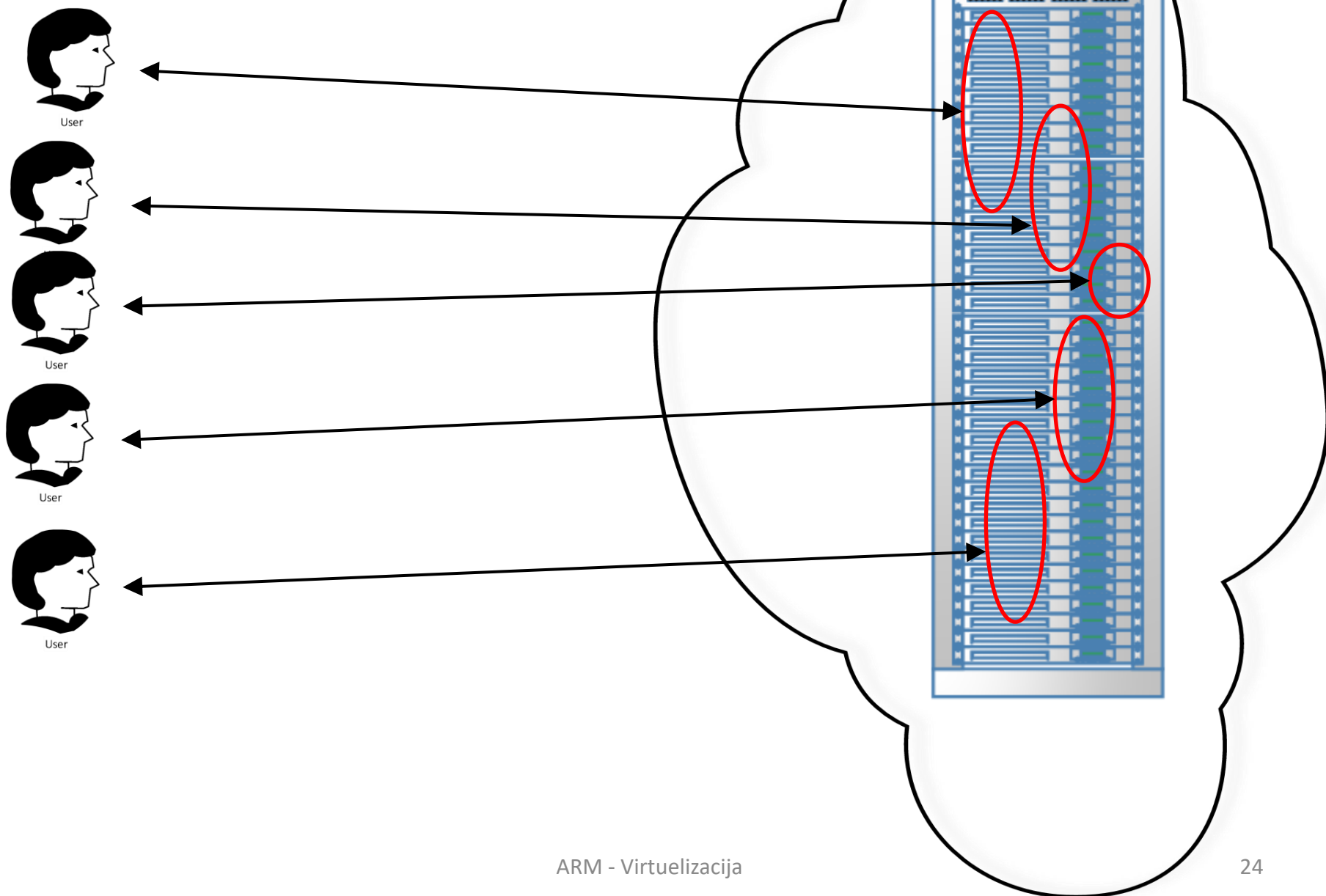


# Umrežavanje i VM

- Načini rada
  - Bez mreže
  - NAT
  - Premošteno (*Bridged*)
  - Interna mreža
  - Mreža samo sa monitorom VM (Host-only)

# Virtualne mašine







# Kontejneri



- 6, 09 (12,18) m x 2,44 m x 2,59 (2,90) m
- malo više od 25 tona tereta

# Kontejneri za prevoz (1956.)

- Prije
  - *Break bulk shipping*
  - Utovar
    - prosječno 200.000 pojedinačnih stvari po brodu
    - brzina 1.3 t/h (oko sedmice po brodu)
    - trošak oko 6 \$ (60 - 75% troškova prevoza)
    - veliki broj radnika
- Poslije
  - Container shipping
  - Utovar
    - oko 10.000 kontejnera po brodu
    - brzina 10.000 t/h
    - trošak oko 0,15 \$
    - velika nezaposlenost u lukama
      - radnika koji su utovarali

# Kontejneri za softver (2013.)

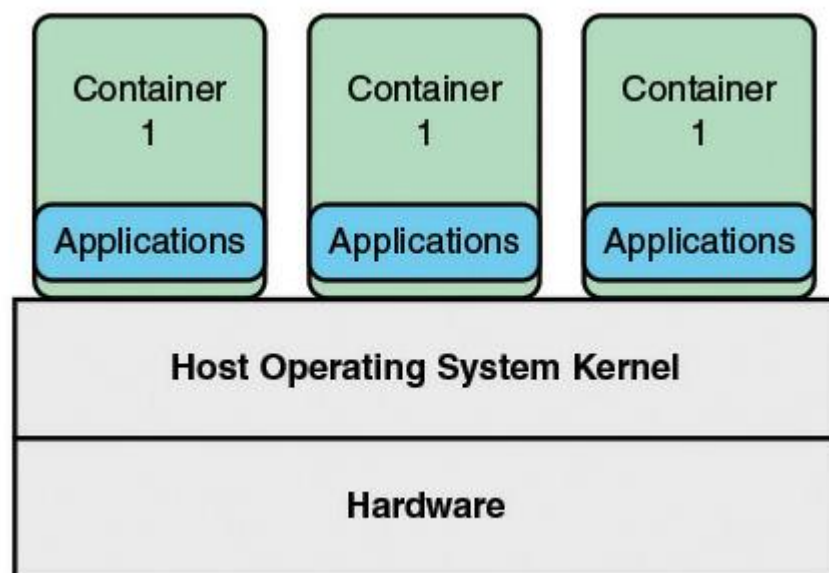
- Prije
  - spor i komplikovan „deployment“
  - Razvojno okruženje
    - programski jezik
    - biblioteke
    - OS
  - Produkciono okruženje
    - programski jezik ???
    - biblioteke ???
    - OS ???
- Poslije
  - brz i jednostavan „deployment“
  - Razvojno okruženje
    - kontejner
  - Produkciono okruženje
    - kontejner (isti)

# Historija

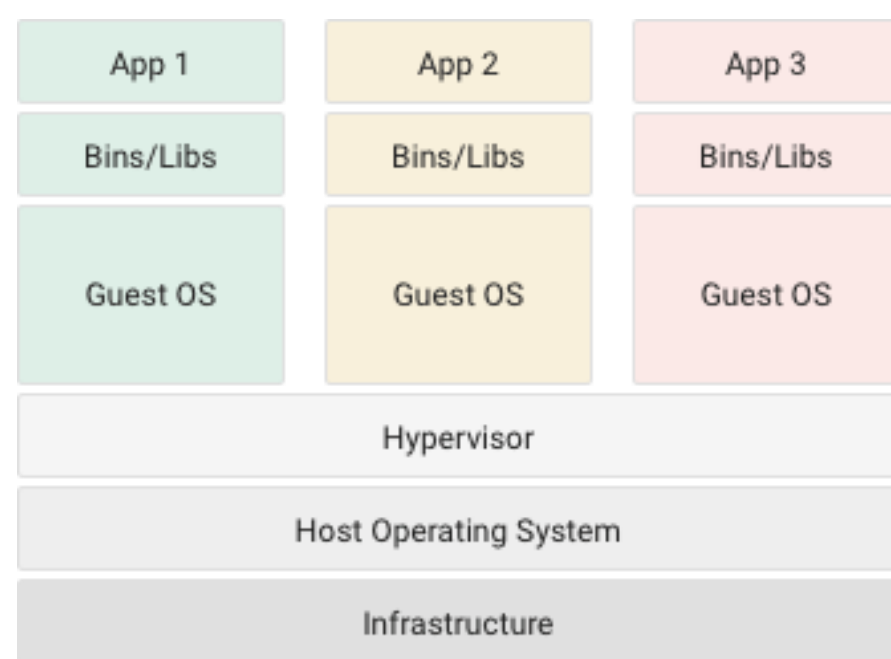
- chroot (Unix) (1979.)
- jail (FreeBSD) (1990., 2000.)
- Zones (Solaris) (2004.)
- cgroup (2006.)
- LXC (Linux Containers) (2008.)
  - „dyno“ (Heroku) (2007.)
- Docker (2013.)

# Virtuelizacija kontejnerima

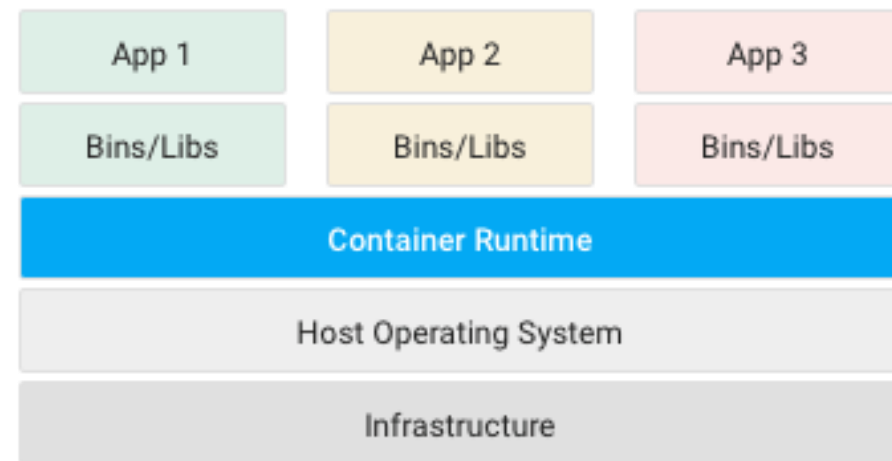
- Softver, virtuelizacijski kontejner, izvršava se na OS
- Stvara (samo) izvršno okruženje za aplikacije
  - a ne kompletan prividni računar ili OS
  - sve aplikacije u kontejnerima na računaru dijele OS (kernel)
  - nema potrebe za poseban OS za svaku aplikaciju



# VMs vs Containers



**Virtual Machines**



**Containers**

# Konzistentno okruženje

- Predvidivo
- Izolovano od drugih aplikacija
- Može uključivati zavisnosti (*dependencies*)
  - specifične verzije jezika i biblioteka
- Pruža konzistenciju nezavisno od toga gdje se koristi (pokreće, *deploy*)
- Veća produktivnost - manje promjena okruženja
  - razvojno, testno, produkciono

# Pokretanje svugdje

- Kontejneri se mogu pokrenuti, gotovo, bilo gdje
- Na svakom OS (Linux, Windows, Mac)
- Na VM ili fizičkom računaru (*bare metal*)
- Kod razvojnog inženjera (*developer*), u lokalnom podatkovnom centru ili u *cloud*
- *Docker image format*
  - popularan, gotovo standard



# Manje resursa od VM

- Koriste kernel OS na kom su pokrenute
  - ne treba im cijeli OS (u *image*)
- Manja *image* datoteka
- Brže pokretanje
- Lakše premještanje
- Lakše umnožavanje
  - bolje skaliraju
- Pogodno za *cloud*

# Podrška savremenom razvoju

- Modularnost
  - male veličina
  - portabilnost
- Podrška za modularne aplikacije
  - umjesto monolitnih
- Mikro servisne arhitekture
- Serverless
- DevOps

# Bolje korištenje resursa

- Manje resursa nego VM
- Korištenje (pokretanje, umnožavanje) komponenti (kontejnera) po potrebi
  - granularnije (preciznije)
- Umjesto skaliranje cijele (monolitne) aplikacije
  - zbog opterećenja jedne komponente

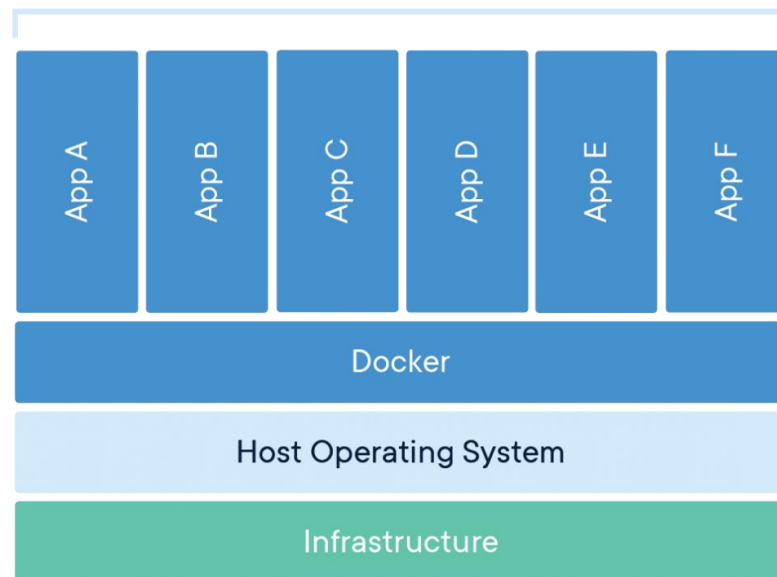
# Aplikacija -> kontejner

- *Containerization*
- Pakovanje koda i ostalog potrebnog (*dependencies*)
  - biblioteke
  - konfiguracije
  - postavke
- u jednu datoteku
- koja se izvršava na (*container*) *runtime engine*

# Upotreba

- Mikroservisi
- DevOps
- Hibrid, *multi-cloud*
- Modernizacije i migracija aplikacija

# Docker



- *De facto* standard
- *Docker engine*
- *Docker image*
- Veliki broj aplikacija u obliku pripremljenih *Docker image*
- Jednostavan proces pretvaranja aplikacije u *Docker image*
- Široka podrška
  - OS i *cloud* platforme

# Upravljanje kontejnerima

- Razbijanje (monolitne) aplikacije na mikro servise izvedene kroz kontejnere
- Raste broj kontejnera kojim treba upravljati – pogotovo u većim sistemima (hiljade)
- Postaje kompleksno

# Orkestracija

- Pokretanje (*Provisioning*)
- Redundantnost
- Nadzor rada (*Health monitoring*)
- Alokacija resursa
- Skaliranje i raspoređivanje opterećenja
- Premještanje između fizičkih računara (*hosts*)



# Platforme za orkestraciju

- Apache Mesos
- Nomad
- Docker Swarm
- Kubernetes

# Kubernetes

- Razvio Google
- Danas open source (Cloud Native Computing Foundation)

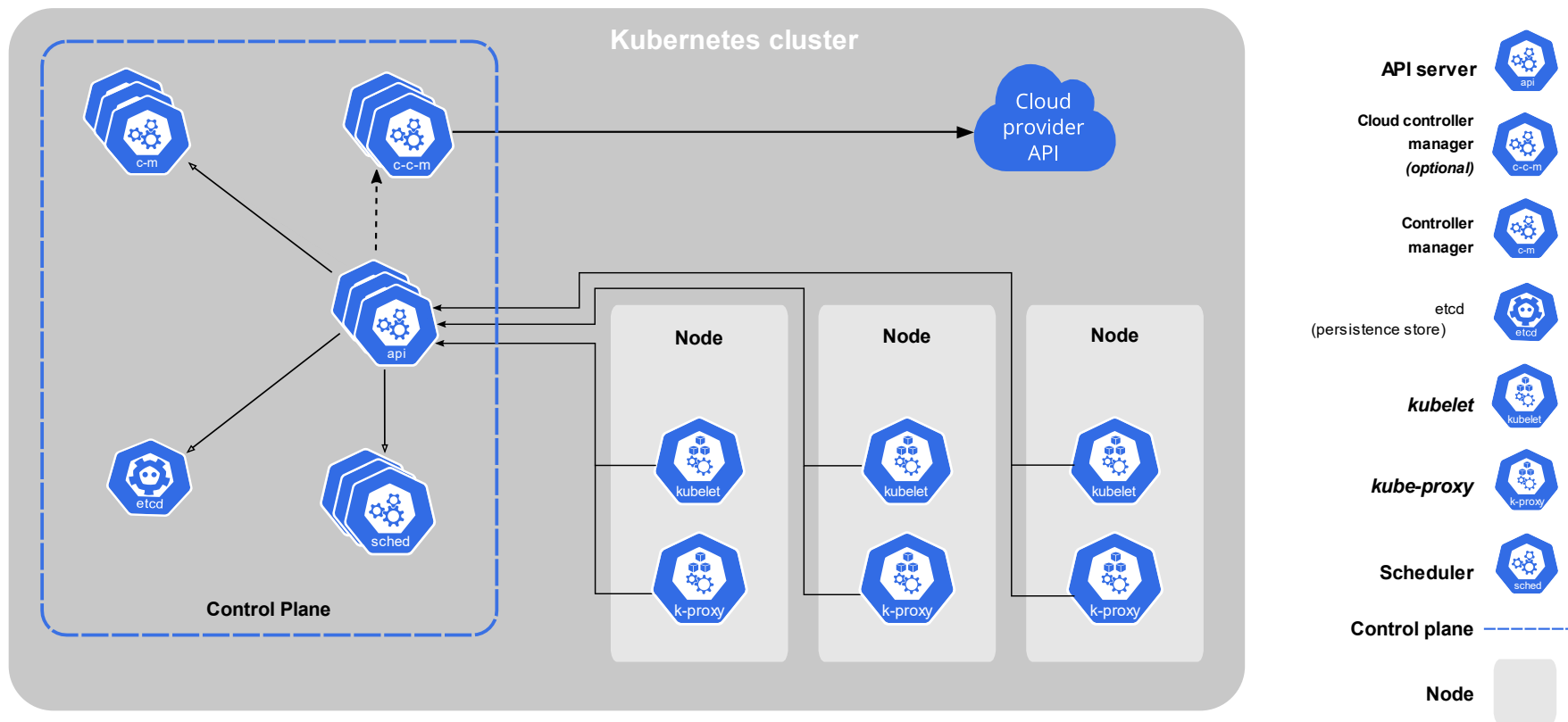
# Kubernetes usluge

- Pristup kontejnerima i balansiranje opterećenja
- Orkestracija prostora za pohranu
- Automatska aktivacija (*rollout*) i povrat (*rollback*)
- Automatsko raspoređivanje kontejnera
  - na *nod*-ove prema potrebnim CPU i RAM
- Samo-oporavak
- Upravljanje konfiguracijama i sigurnosnim postavkama

# Kubernetes - komponente

- Klaster - jedinica kojom upravlja Kubernetes
- Sastoji se od
  - *Nod*-ovi
    - čvorovi (VM, fizički računari) na kojim se izvršavaju kontejneri
    - minimalno jedan
  - *Pod*-ovi
    - grupa kontejnera na *nod*-ovima
    - *logical host* (*storage*, IP adresa)
  - Kontrolna ravan

# Kubernetes - komponente



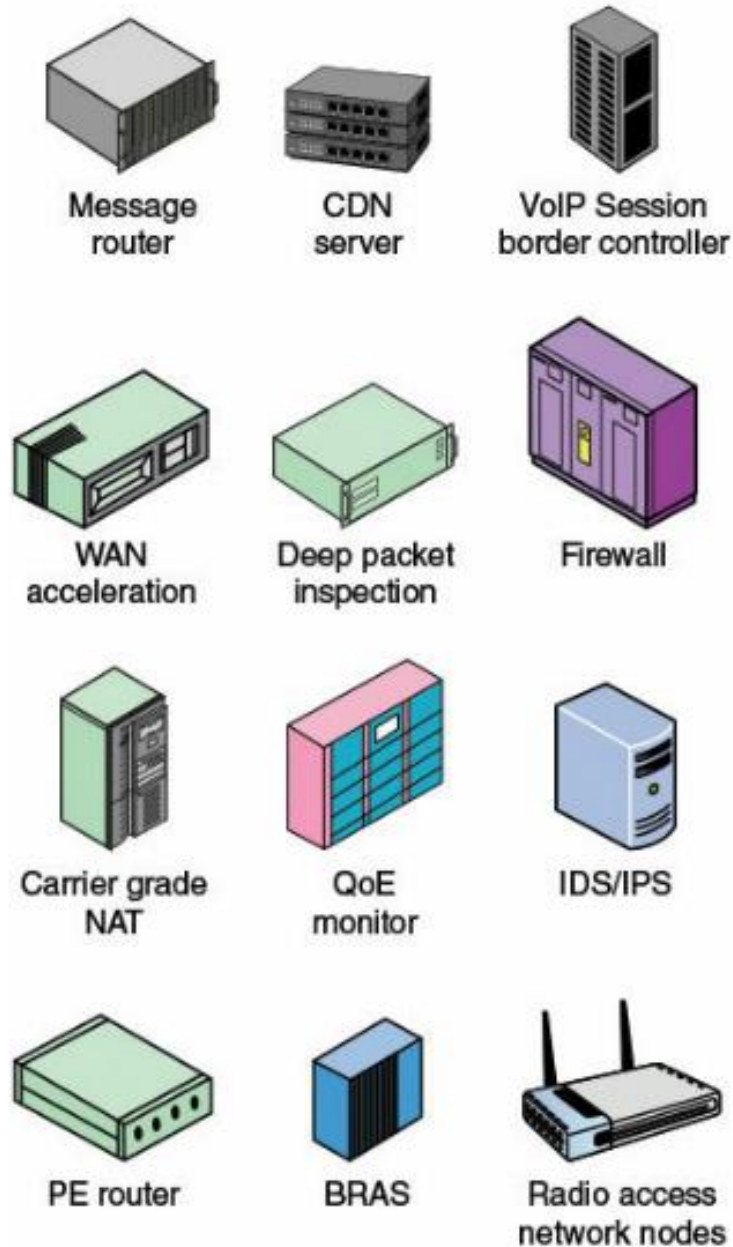
# Početne reference

- <https://cloud.google.com/containers>
- <https://www.ibm.com/cloud/learn/containers>
- <https://www.docker.com/resources/what-container>
- <https://kubernetes.io/docs/tutorials/kubernetes-basics/>

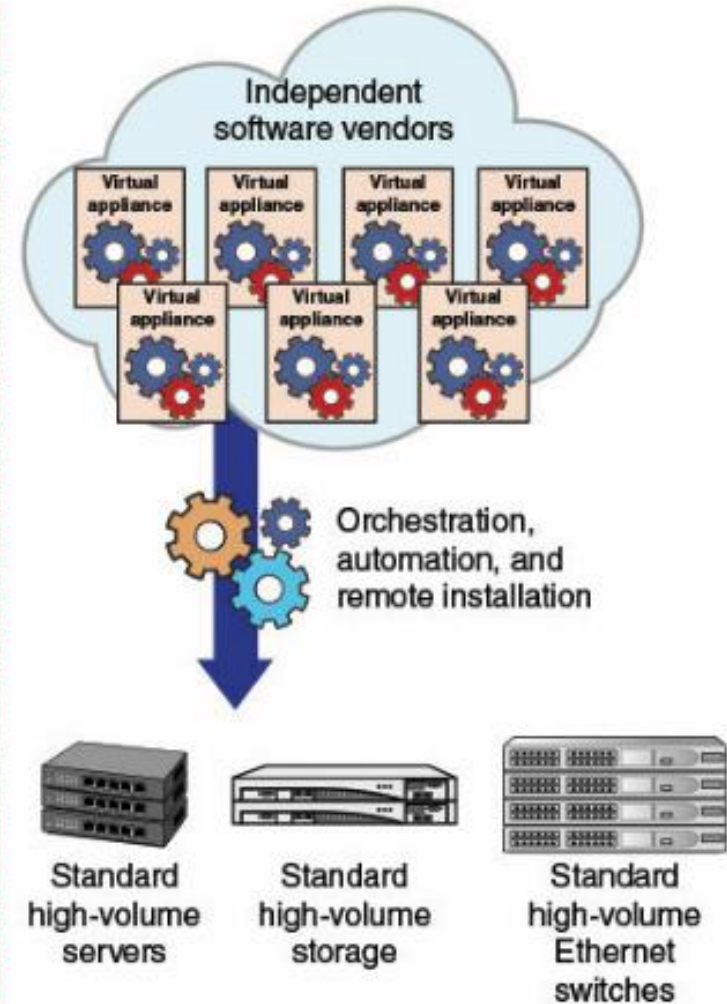
# NFV

- Izvedba mrežnih funkcija u softveru u VM
  - Mrežnih uređaja (switch, ruter, ...)
  - Računarsko-mrežnih uređaja (firewall, IDS,...)
  - *Network-attached storage*: File i DB serveri

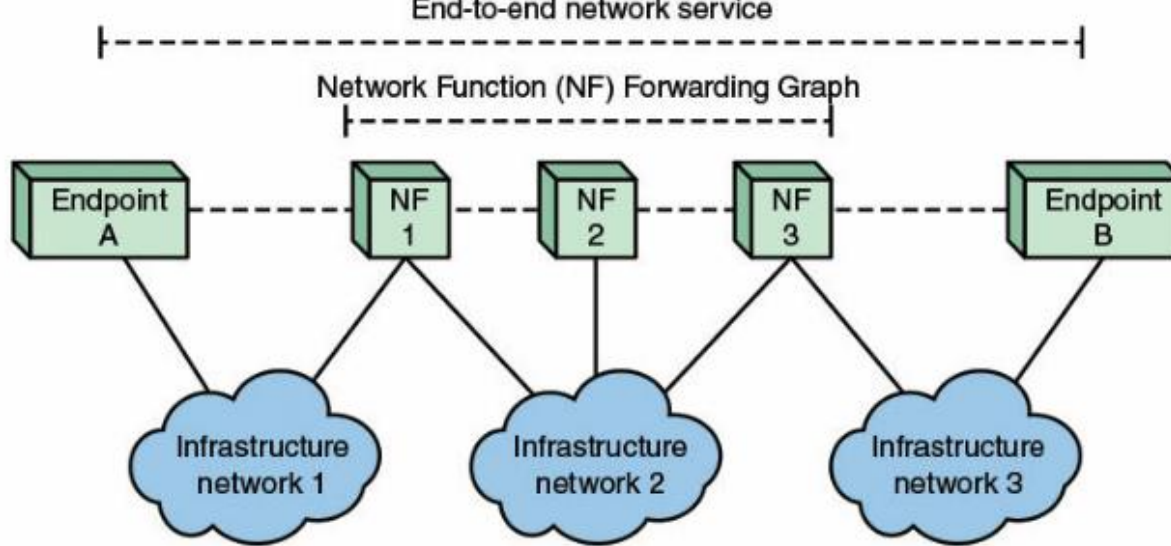
## Traditional Network Application Deployment



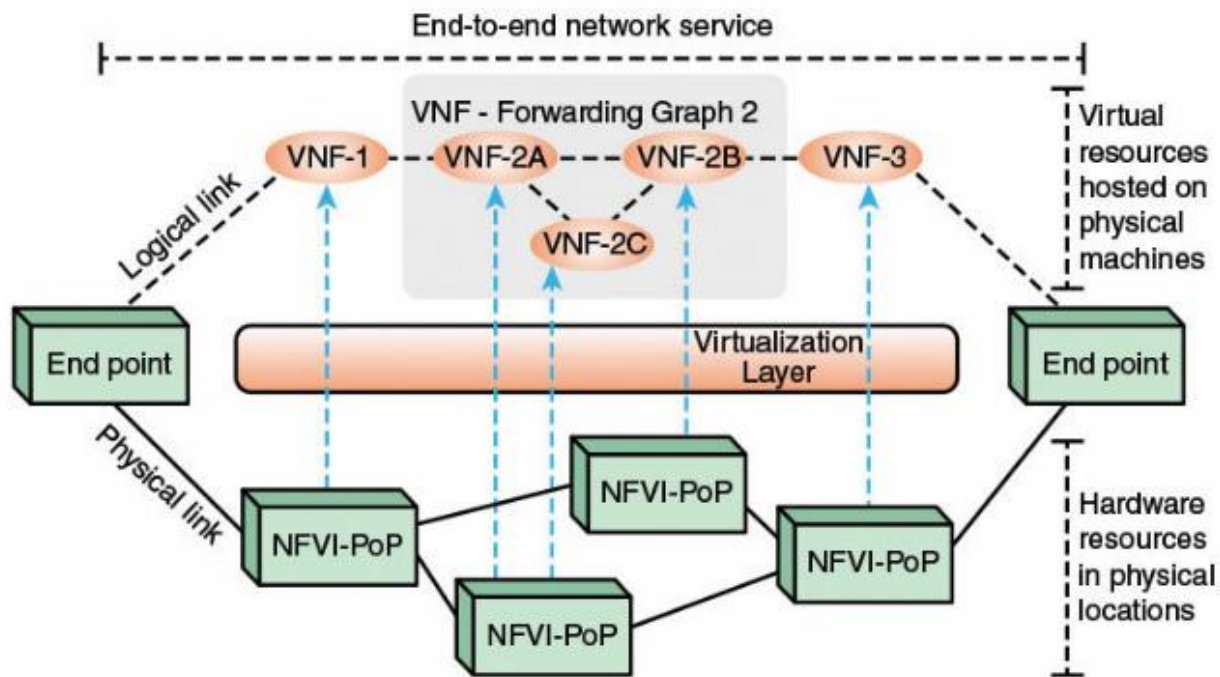
## NFV Network Appliance Deployment







(a) Graph representation of an end-to-end network service



(b) Example of an end-to-end network service with VNFs and nested forwarding graphs