

ELEKTROTEHNIČKI FAKULTET UNIVERZITETA U BEOGRADU



DATA CENTRI

Diplomski rad

Kandidat:

Miloš Radosavljević 2009/213

Mentor:

doc. dr Zoran Čiča

Beograd, Jul 2015.

SADRŽAJ

SADRŽAJ	2
1. UVOD.....	3
2. GENERALNE INFORMACIJE O DATA CENTRIMA.....	4
2.1. ISTORIJAT	4
2.2. NAMENA.....	6
2.3. PODELA DC.....	7
2.3.1. Aplikacije	7
2.3.2. Infrastruktura i njeni zahtevi	9
3. OPREMA.....	12
3.1. MEHANIČKA OPREMA.....	12
3.2. ELEKTRIČNA OPREMA	13
3.3. IT OPREMA	16
3.3.1. Serveri.....	17
3.3.2. Mrežna oprema	19
3.3.3. Oprema za skladištenje	21
4. TOPOLOGIJE	24
4.1. KLASIFIKACIJA TOPOLOGIJA	24
4.2. FIKSNE TOPOLOGIJE BAZIRANE NA STABLU	26
4.2.1. Topologija osnovnog stabla.....	28
4.2.2. Clos mreže	28
4.2.3. Fat tree topologija	29
4.2.4. VL2.....	31
4.3. FIKSNE REKURZIVNE TOPOLOGIJE	33
4.3.1. Dcell.....	33
4.3.2. BCube	36
4.3.3. MDCube.....	37
4.3.4. FiConn	38
4.4. FLEKSIBILNA TOPOLOGIJA	40
4.4.1. Hibridna topologija	40
4.4.2. c-Through topologija.....	40
4.4.3. Helios.....	41
4.4.4. OSA- Optical Switching Architecture	42
4.5. OSTALE TOPOLOGIJE	44
4.5.1. Dragonfly.....	44
4.5.2. Scafida	45
5. PROTOKOLI.....	47
6. PROBLEMATIKA I TRENDOVI U DATA CENTRIMA	59
6.1. IZAZOVI ENERGETSKE EFIKASNOSTI.....	60
6.1.1. PUE (Power Usage Effectiveness).....	60
6.1.2. Iskorišćenost servera	61
6.1.3. Podela troškova	62
6.2. PREPORUČENE AKCIJE I TRENDOVI.....	62
7. ZAKLJUČAK.....	65
LITERATURA.....	66

1. UVOD

Data centri se mogu definisati kao mesta gde su smešteni računarski sistemi, sistemi skladištenja podataka i telekomunikacione opreme. Iako mnogi nisu svesni, data centri su u sve većoj meri počeli da postaju nezaobilazni deo svakodnevnog života. Od izbora adekvatnog data centra uveliko zavisi bezbedno, pouzdano i nesmetano poslovanje kompanija.



Slika 1.1: Savremeni data centar [1]

Cilj ovog rada biće da se čitaocu:

- približi značaj savremenih data centara,
- objasni njihov princip rada,
- predstave podele koje su vršene na osnovu potreba, primene, infrastrukture i arhitekture,
- objasne zahtevi koje moraju da ispunjavaju,
- objasne pristupi za njihovu implementaciju.

Rad će biti koncipiran na sledeći način: drugo poglavlje opisuje generalne informacije o data centrima kao što su kratak istorijat, namena, aplikacije, generalni zahtevi u pogledu infrastrukture. Treće poglavlje se bavi podelom opreme u data centrima, zahtevima koje ona mora da ispunjava i virtuelizaciji i njoj primeni u data centrima. Četvrto poglavlje opisuje topologije koje se koriste u data centrima, njihovu implementaciju kao i njihove prednosti i mane. U petom poglavlju se obraća pažnja na protokole koji se koriste u data centrima. Šesto poglavlje je osvrt na problematiku sa kojom se susreću data centri, preporuke koje treba primeniti za njihovo rešavanje i aktuelni trendovi u data centrima. U poslednjem poglavlju, sedmom, se rezimiraju rezultati ovog rada.

2. GENERALNE INFORMACIJE O DATA CENTRIMA

2.1. Istorijat

Početak data centara (*eng. data center*) može se pratiti još od nastanka prvih kompjuterskih sistema koji su bili veoma kompleksni zbog velikog prostora koji su zahtevali i kontrolisanog okruženja koje je bilo neophodno za njihovo funkcionisanje. Oko 1960-te vojska je razvila ogromnu mašinu nazvanu ENIAC (*eng. Electronic Numerator, Integrator, Analyzer and Computer*). Težina ENIAC je bila 30 tona i zauzimala prostor od 168 m². Zahtevala je konstantno prisustvo 6 tehničara da bi mašina obrađivala oko 5000 operacija u sekundi [2]. Prvi kompjuteri bili su kompleksni za upotrebu. Kompjuteri su bili smešteni u specijalne sobe odakle bi ih tehničari održavali. Upravo te specijalno dizajnirane sobe predstavljaju korene data centara. Pored prostora koji su zauzimali, neophodan je bio i sistem za hlađenje zbog njihovog brzog pregrevavanja. Zbog velikog broja komponenti koji su sačinjavali ove prve kompjutere, potreban je bio i ogroman broj kablova, koji bi povezali sve delove. To je prouzrokovalo velike probleme. Problem je prevaziđen upotrebom ormarića (*eng. racks*) za stavljanje opreme, nosačima kablova i podignutim podovima (*eng. raised floors*). Ako se sve sagleda, vidi se da su rani kompjuteri bili veoma skupi. Korišćeni su u vojne i civilne svrhe, pa se zbog toga velika pažnja posvećivala bezbednosti, koja je bila prioritet, tako da je uspostavljena kontrola pristupa tim sobama [2][4].

1960

Počevši od 1960. godine, kompjuteri su prešli sa uređaja na bazi vakuumske cevi, na uređaje koji su koristili poluprovodničku tehnologiju kao što su tranzistori koji su duže trajali, bili manji, efikasniji, pouzdaniji i mnogo jeftiniji. Ranih 1960-ih godina kompjuteri su koštali oko 5 miliona dolara i mogli su da se iznajme za 17 hiljada dolara mesečno. Sredinom ove dekade, računari su počeli da se koriste u komercijalne svrhe i delilo ih je nekoliko korisnika. Udruživanje American Airlines i IBM dovelo je do razvoja programa nazvanog *Sabre System*, koji je instaliran na 2 IBM 7090 kompjutera smeštena u specijalno dizajniranu sobu u Njujorku. Ovaj sistem je bio u mogućnosti da prosleđuje 84000 poziva dnevno [3]. Memorija kompjutera je počela polako da se odvaja od uređaja sa magnetnim jezgrom i da se postavlja kao statička i dinamička memorija poluprovodnika, što je u velikoj meri smanjilo troškove, veličinu i potrošnju [3].



Slika 2.1.1: ENIAC (Electronic Numerator, Integrator, Analyzer and Computer) [2]

1970

Intel je 1971. godine proizveo prvi mikroprocesor: 4004. Data centri u Americi počeli su da od 1973. godine beleže formalne planove za oporavak od katastrofe (*eng. disaster recovery*). Ako bi se desila katastrofa, ona ne bi bitno uticala na poslovne operacije. Tadašnji kompjuteri nisu obavljali kompleksne funkcije i one su uglavnom bile samo administrativne prirode [3].

Kompanija SunGard je 1978. godine razvila prvi komercijalni posao za oporavak od katastrofe i iznajmila je prostor za svoja postrojenja od skoro 3000 m² u Njujorku gde se i danas nalaze [3].

Godine 1973. Minikompjuter Xerox Alto bio je orijentir za razvoj personalnih računara zbog svog grafičkog korisničkog interfejsa, za tadašnje vreme visoke rezolucije ekrana, velike interne i eksterne memorije za skladištenje, miša i specijalnog softvera.

Godine 1977. pojavljuje se ARC-net, prva svetska komercijalna lokalna mreža, instalirana u banci Chase Manhattan Bank, u Njujorku, kao beta sajt. To je bio najjednostavniji i najjeftiniji tip lokalne mreže, koji je koristio *token-ring* arhitekturu, podržavajući brzine prenosa podataka od 2,5 Mb/s i povezujući do 255 računara. Oni su zahtevali specijalno hlađenje i pri kraju 1970-ih, ovi kompjuteri se sele u kancelarije, čime se gasi prvi pojam data centara [3].

1980

Tokom ovih godina dolazi do evolucije u industriji mikrokompjutera zahvaljujući nastanku IBM personalnog kompjutera (PC, *eng. Personal Computer*). Kompjuteri su počeli da se instaliraju svuda, a da se nije brinulo o okruženju i specifičnim uslovima potrebnim za njihov rad. Tada dolazi do problema gubljenja podataka jer je bilo veoma teško da se organizuju sve te informacije. Kao pokušaj rešenja ovog problema došlo je do formiranja informacionih timova, koji bi sastavljali i održavali ove mikrokompjutere, ali to iz više aspekata nije bilo dobro niti krajnje rešenje. Počevši od 1985. IBM je obezbedio više od 30 miliona dolara u proizvodima i podršci, koje je u toku od 5 godina uložio u *Super Computer* ustanovu (postrojenje) izgrađenu na Kornel univerzitetu u Njujorku. Godine 1988. IBM izbacuje aplikacioni sistem /400(AS/400) i veoma brzo postaje jedna od vodećih kompjuterskih kompanija. Kako su informacione tehnologije (IT) počele da napreduju i da se razvijaju, kompanije su postale svesne potrebe za kontrolom IT resursa [3][4].

1990

Kompjuterski sistemi se u ovom periodu baziraju na modelu klijent-server čime postižu prilagodljivost i dostupnost podataka ka krajnjim korisnicima. Ovaj model u idealnom slučaju dostavlja podatke krajnjem korisniku u onom obliku koji njemu najviše odgovara. Mikrokompjuteri, danas „serveri”, polako su počeli da zauzimaju prostor u starim kompjuterskim sobama unutar kompanija i termin data centar je ponovo „rođen” [3][4].

Zahvaljujući pojeftinjenju opreme i pojavi novih standarda za strukturno kabliranje omogućeno je formiranje data centara. Time je postignut hijerarhijski dizajn za njihovo smeštanje. Ekspanzija data centara se desila tokom perioda .com domena. Kompanijama je bio potreban brz pristup Internetu kao i njihovo stalno prisustvo na njemu, zbog različitih vrsta poslovnih operacija. Oprema potrebna za ove operacije nije bila dostupna svima pre svega gledano sa ekonomskog aspekta manjih kompanija. Zato se rađa ideja da se grade ogromni objekti koji bi pružali usluge sa nizom rešenja za raspoređivanje i rad sistema. Sa povećanjem potražnje, rasli su i zahtevi koje su data centri morali da ispunjavaju, tako da su otkrivane nove tehnologije i procedure da bi se ispunili ovi uslovi [3][4].

2000

Data centri se konstantno razvijaju i šire kako raste potreba za njima. Uz razvoj, stvaraju se i novi problemi i prepreke koje je potrebno prevazići, kao što je povećanje potrošnje energije. Od 2007. godine prosečni data centri trošili su električnu energiju ekvivalentno 25000 domova. U SAD data centri učestvuju u potrošnji sa 1,5% od ukupne potrošnje energije na godišnjem nivou i njihova potrošnja konstantno je u porastu. Ali i pored tolikih troškova, rast broja data centara je neprekidan što se može najbolje videti iz primera da je broj državnih data centara 1999. godine bio 432, a preko 1100 danas, dok se prosečno broj servera na godišnjem nivou poveća za 5,75 miliona [2]. Današnji najveći problem u modernim data centrima je energetska efikasnost i potrošnja koja konstantno raste, pa se stalno teži ka smanjenju potrošnje. Za smanjenje energetske potrošnje Facebook je lansirao *OpenCompute Project*, obezbeđujući specifikacije za njihov Prinevill, Oregon data centar koji koristi 38% manje energije nego njihove ostale ustanove i košta 24% manje [3].

2.2. Namena

Teško je naći preciznu definiciju kojom bi se opisali data centri. Data centri predstavljaju fizičku ili virtuelnu infrastrukturu koju koriste kompanije za smeštanje računara, servera i mrežnih sistema i komponenata koje su neophodne za funkcionisanje informacione tehnologije (IT) kompanije, a to uglavnom uključuje skladištenje, preradu i prosleđivanje velikih količina podataka na zahtev klijenta u klijent/server arhitekturi. U data centrima se takođe nalaze sistemi za napajanje, sistem za bekap napajanje (baterije, agregati), protivpožarni sistemi, sistemi za održavanje radne okoline, bezbednosni sistemi. Data centar treba da omogući kompanijama smeštaj i funkcionisanje njihove IT infrastrukture po prihvatljivim cenama i na prihvatljiv način. Oni se još nazivaju i „farme servera”.

Savremeni data centri predstavljaju jezgro današnjeg poslovanja velikih kompanija. Pojavom virtuelizacije i njenoj primeni u data centrima, ostvaruje se evidentan napredak i samih data centara, što je povuklo i ogromne promene u savremenom poslovanju. Virtuelizacija dobija na ogromnom značaju i ima najveću primenu upravo u data centrima. Funkcionisanje kompanija i njihovo poslovanje zavisi u ogromnoj meri od pouzdanosti i raspoloživosti njihovih informacionih sistema. Dostupnost i protok podataka neophodnih za funkcionisanje firme su od vitalne važnosti i ti podaci se moraju prosleđivati i čuvati bez gubitaka. Iz ovoga se vidi da je svaki element u prenosu, skladištenju i procesiranju podataka od kritične važnosti. Data centri su možda jedna od najbitnijih karika u obradi podataka, jer informacije i podaci moraju biti dostupni 24/7, a to će biti moguće jedino ako se obezbedi bezbedna i

pouzdana IT infrastruktura. U slučaju havarije nekog dela informacionog sistema, funkcionisanje kompanije može doživeti potpuni kolaps. Od krucijalne važnosti je uspostaviti pouzdanu i bezbednu IT infrastrukturu i smanjiti šansu na minimum da dođe do prekida njenog rada. Treba postaviti visoke standarde koji će obezbeđivati garancije za stabilan rad i funkcionisanje koji idu zajedno sa odgovarajućim mikroklimatskim uslovima i redundantnošću kablovskog povezivanja i napajanja za informacionu opremu i sisteme za skladištenje podataka. Da bi se sve to postiglo, potreban je ogroman početni kapital da bi se obezbedilo kvalifikovano osoblje, oprema i adekvatan prostor u kome će ona biti smeštena, a ono što mnogi i ne shvataju kada žele da izgrade data centar su kasniji ogromni troškovi koji idu na održavanje, unapređivanje sistema i proširivanje data centra kako rastu potrebe kompanije. Iz svega ovoga zaključujemo da data centri igraju centralnu ulogu u današnjem poslovnom svetu. Standardi za data centre su definisani međunarodnim aktom TIA-942 [1][4].

2.3. Podela DC

Data centri se na osnovu potreba mogu podeliti na:

- privatne (*eng. private*),
- javne (*eng. public*),
- virtuelno privatne (*eng. virtually private*) [1].

Privatni data centri

Privatni data centri pripadaju jednom preduzeću ili kompaniji i mogu da podrže više korisnika unutar kompanije. Oni su povezani preko privatne ili virtuelno privatne mreže unutar kompanije. Usluge unutar privatnog data centra mogu da ostanu privatne za tu kompaniju ili mogu delimično da imaju pristup Internetu preko sigurnih kanala. Održavanje opreme i tehnička podrška u korporativnim data centrima je u nadležnosti sopstvenog IT sektora ili je ugovorena spolja sa nekim od poslovnih partnera.

Javni data centri

Javni data centri povezani su preko Interneta i često služe za višekorisničke usluge kao i za poslovanje javnih servisa. Ovi Internet data centri mogu biti u vlasništvu telekomunikacionog operatora ili Internet servis provajdera (ISP), koji takođe brinu o održavanju data centra. Telekomunikaciona mreža služi za povezivanje kompanija sa Internet data centrima, gde pristupaju svojim servisima. U te servise spadaju pristup Internetu, iznajmljivanje kapaciteta, skladištenje podataka, hostovanje veb sajtova, hostovanje aplikacija, skladištenje podataka, usluge za oporavak od katastrofe itd.

Virtuelno privatni data centri

Virtuelno privatni data centri su izgrađeni na zajedničkoj data centar infrastrukturi, često obezbeđenoj od data centar provajdera ili Cloud servis provajdera (CSP) u formi infrastrukture kao usluge. Osnovna mreža za virtuelno privatne data centre pruža korisniku privatnost i bezbednost.

2.3.1. Aplikacije

Jedna od glavnih uloga data centara je da obezbede rad aplikacija i operativnih baza podataka koje su neizostavne za uspešan rad kompanija. Te aplikacije su uglavnom aplikacije IT sistema koje služe za upravljanje poslovnim operacijama i podacima te organizacije. One mogu biti napravljene od

strane kompanija u čijem su vlasništvu ili kupljene od neke softverske kompanije. Ove aplikacije najčešće mogu biti: ERP (*eng. Enterprise Resource Planning*) i CRM (*eng. Customer Relation Management*) [1][4].

Aplikacije u većini slučajeva rade na nekoliko hostova, od kojih svaki opslužuje po jedan segment aplikacije. Zajedničke komponente aplikacija su npr. *middleware*, baze podataka, fajl serveri, aplikacioni serveri i mnoge druge. Data centri se često koriste za bekap na udaljenoj lokaciji, pri čemu kompanije mogu da se pretplate za bekap koji im pruža data centar. Ovakav vid zaštite i čuvanja podataka na nekoj drugoj lokaciji za cilj ima dodatnu zaštitu i sigurnost. To znači da se podaci iz jednog data centra mogu paralelno staviti i čuvati u drugom data centru koji može biti na velikoj udaljenosti od prvobitnog. Možda i najbitnija prednost data centra je oporavak od katastrofe - mehanizam koji obnavlja infrastrukturu ili podatke od krucijalnog značaja za rad kompanije nakon većih kvarova ili nepogoda. Oporavak od katastrofe je pouzdaniji i jednostavniji u okviru data centara. Glavni server ima svoju alternativnu opciju, jer se podaci paralelno upisuju i čuvaju na dva različita mesta i ako dođe do pada primarnog servera, rezervni će zauzeti njegovo mesto. Sve ovo je veoma bitno za rad aplikacija [1][4].

Što se tiče ERP i CRM aplikacija koje su najčešće u data centrima:

ERP je planiranje resursa u korporacijama i može se definisati kao spoj prakse upravljanja poslovanjem i savremene tehnologije. Informacione tehnologije (IT) integrišu najvažnije poslovne procese u organizaciji da bi se osavremenili i postigli definisani poslovni ciljevi. ERP je sinteza tri bitna faktora: upravljanja poslovanjem, informacionih tehnologija i specifičnih poslovnih ciljeva [4][5].

Može se definisati još i kao softverska arhitektura koja se bavi usmeravanjem podataka i raspodelom raštrkanih poslovnih podataka ka svim jedinicama firme kojima su ti podaci potrebni. ERP omogućava izvršnom menadžmentu kompanije celokupan pregled funkcionisanja svih poslovnih aktivnosti, što može dovesti do poboljšanja poslovnih odluka u daljem procesu. Jezgro ERP čini dobro uređeno, centralizovano skladište podataka koje prikuplja informacije od nepotpune aplikacije i snabdeva je novim, tako da te aplikacije funkcionišu na univerzalnoj kompjuterskoj platformi [4]. U velikim poslovnim organizacijama podaci se prikupljaju na različitim serverima u sklopu više funkcionalnih jedinica koje mogu biti geografski razdvojene. Današnji smisao ERP sistema je da može u celini da pokrije širok spektar funkcija preduzeća i integriše ih u jedinstveno skladište podataka. Funkcije kao što su menadžment ljudskih resursa, upravljanje lancem snabdevanja, finansije, upravljanje odnosima sa klijentima, upravljanje proizvodnjom i zalihama i logistika ranije su bile usamljene softverske aplikacije, koje su uglavnom bile snabdevene sopstvenim aplikacijama, bazom podataka i mrežom. Danas, svi ovi sektori mogu da rade zajedno „pod istim kišobranom” – ERP sistemom. Softverski ERP sistem mora obezbediti preduzeću mnoge funkcionalnosti sa karakteristikama kao što su fleksibilnost, modularnost i jasnost, širina, najjeftiniji poslovni procesi i globalni fokus [5]. Kao primere najčešćih ERP aplikacija navode se: *Exact*, *Portera*, *QAD* i *SSA* [4][5].

CRM je poslovna strategija za uspostavljanje kontakata i upravljanje odnosima sa mušterijama kako sadašnjim tako i budućim. CRM ne predstavlja samo poslovnu strategiju već i filozofiju i kulturu koja je bazirana na odnosu prema klijentima, da bi se ostvario uspešan marketing, prodaja i usluživanje. CRM aplikacije omogućuju uspešan menadžment u oblasti odnosa sa kupcima, pod uslovom da kompanija ima dobro vođstvo i jasno definisanu strategiju. Primeri CRM aplikacija su: *Sage Base*, *Sales Force* i *InfusionSoft*. Struktura data centara obezbeđuje efikasan i pouzdan rad pomenutih sistema i aplikacija. Tehnike virtuelizacije u data centru (virtuelizacija servera) omogućavaju sve to [6][7].

2.3.2. *Infrastruktura i njeni zahtevi*

Po pitanju infrastrukture gradivni elementi data centara se mogu podeliti u četiri grupe:

- *building shell*,
- IT infrastruktura,
- električna infrastruktura,
- mehanička infrastruktura [10].

Svaka infrastruktura se sastoji iz komponenata i zahteva koje komponente moraju da ispune u pogledu infrastrukture npr. potrošnja.

- *Building shell* predstavlja prostor veličine jedne sobe, jednog ili više spratova, a može biti i čitava zgrada u kojoj je smeštena sva potrebna oprema za funkcionisanje data centara.
- IT infrastruktura se može podeliti na više delova i u nju spadaju serveri, oprema za skladištenje, mrežna oprema. Pomoću ove opreme se vrše sve funkcije zbog kojih se koriste data centri. Ona zahteva konstantan dotok električne energije kao i dobre sisteme hlađenja da ne bi došlo do prestanka njenog rada i havarije u sistemu [5].
- Električna oprema podrazumeva napajanja i linkove kojima se obezbeđuje dotok energije za rad data centara. Ova infrastruktura je bazirana na redundantnosti.
- Mehanička infrastruktura u stvari predstavlja opremu u kojoj se smešta sva ostala oprema npr. rek (*eng. rack*) i fizički elementi. Oprema se uglavnom funkcionalno grupiše i smešta po odgovarajućim zonama.

Osim mrežne i računarske infrastrukture, ostale bitne komponente u data centrima su:

- sistemi za nadzor hardvera i softvera - NOC (*eng. Network Operation Center*),
- sistem za hlađenje i odvođenje toplote,
- uzemljenje,
- fizičko obezbeđenje i protivpožarna zaštita [1].

Data centri poseduju najsavremeniju opremu, sofisticirane sisteme sigurnosti, što znači i bolju bezbednost i pouzdanost IT sistema i infrastrukture.

Sušтина data centara je upravljanje podacima. U današnjoj eri sve moćnijih računara, mobilnih uređaja, mreža velikih propusnih moći i Interneta u globalu, količina podataka sve više raste. To stvara sve veće potrebe za njihovim skladištenjem na kraći ili duži rok. Kompanije su prinuđene da proširuju svoje data centre. Međutim, proširenje podrazumeva dodatni prostor i dodatnu opremu, tako da infrastruktura data centara predstavlja sve veći udar na budžet kompanije. Upravo iz tih razloga, mnoge kompanije se odlučuju za iznajmljivanje računarske infrastrukture, a to dovodi do novog načina poslovanja u svetu, koji se popularno naziva *Cloud*. Ovo je novi trend u data centrima u koje se polaže velika nada. Iznajmljivanjem kapaciteta u data centrima štedi se na ulaganjima u sopstvenu infrastrukturu. U zavisnosti od potreba, iznajmljeni kapaciteti se mogu smanjiti ili povećati u bilo kom trenutku.

U pogledu pouzdanosti i raspoloživosti (vremena ispravnog rada), uspostavljeni su međunarodni standardi za data centre, po kojima se razlikuju četiri ranga data centara (Tier 1-4) [4]. Veći rang znači i pouzdaniji, bolji i skuplji data centar. Tier 4 je najrobustniji i najmanje osetljiv na smetnje, jer ima potpuno redundantnu opremu i sisteme, a pristup se kontroliše biometrijskim metodama. Najjednostavniji data centar Tier 1 (serverska soba, *eng. server room*) obično poseduju mala preduzeća ili veće trgovinske radnje [4].

i) Tier 1

Nema redundantnih komponenti (jedinstven server, jedan mrežni aplink, jedan vod za napajanje, jedan klima uređaj, može ali ne mora da ima UPS, nema dupli pod). Garantovana raspoloživost je 99,671% (dozvoljen prekid u trajanju od 28,8 sati godišnje) [4]. U slučaju preventivnog održavanja sistema neophodan je potpuni prekid rada [4][8].

ii) Tier 2

Sastoji se od računarskih komponentata koje rade na principu (N+1) - (imaju nezavisnu aktivnu ili pasivnu bekap komponentu koje ne učestvuju aktivno tokom normalnih operacija sistema), UPS ili agregata, ima jedan neredundantan vod za napajanje i klimatizaciju kao i dupli pod. Garantovana raspoloživost je 99,741% (dozvoljen prekid u trajanju od 22 sata godišnje). Preventivno održavanje napajanja ili druge infrastrukture zahteva prekid rada sistema [4][8].

iii) Tier 3

Pored redundanse i karakteristika koje ima Tier 2, ima i redundantne (duplirane) vodove za napajanje i mrežne aplinkove (zaštitu putanja). U principu, samo jedan napojni vod istovremeno napaja opremu, drugi je u *stand-by* modu. Ima dovoljne kapacitete na međuvezama koji dozvoljavaju i omogućavaju preuzimanje saobraćaja i podataka u slučaju neophodne intervencije i održavanja sistema. Garantovana raspoloživost je 99,982% (dozvoljen prekid u trajanju od 1,6 sati godišnje). Preventivno održavanje ne zahteva prekid rada hardverske infrastrukture i sistema, ali postoje neplanirani događaji i slučajevi koji mogu izazvati potpuni prekid rada sistema [4][8].

iv) Tier 4

Potpuna tolerancija na smetnje i greške svih komponentata sistema. Postoji puna redundantnost i fizička odvojenost svih komponentata (napajanje, rashladni sistemi, serveri, hard diskovi, svi mrežni linkovi i veze). Napajanje svih komponentata je udvojeno, ima nezavisne vodove i dva aktivna odvojena puta. Isto važi i za rashladne sisteme. Garantovana raspoloživost je 99,995% (dozvoljen prekid u trajanju od 0,4 sata, tj. 24 minuta godišnje) [4][8].

Jedan od osnovnih zadataka sa kojim se danas susreću stručnjaci je unapređenje energetske efikasnosti data centra. Cilj je povećati stepen obrade podataka po jednom m² i povećati produktivnost po kW. Implementacija standarda za nadzor i kontrolu doprinosi i pomaže boljem radu i organizaciji data centara. Sistemi ventilacije i hlađenja u data centrima moraju da zadovolje specijalne zahteve koji su neophodni za pouzdan rad:

- temperatura 24°C +2/-4 °C,
- monitoring pomoću senzora,
- relativna vlažnost vazduha 35-55%,
- kapacitet hlađenja do 1kW/m²,
- ventilacija i spoljašnji vazduh se dovode posebnom instalacijom,

- nezapaljivi ventilacioni i rashladni kanali,
- vazduh u prostoriji se menja 1,5 puta na sat,
- dozvoljena prašina do nivoa 5 mikrona [4].

Rezime onoga što data centar mora da obezbedi je:

- dostupnost i neprekidan rad 24/7/365 u toku godine,
- pouzdanost u slučaju pada sistema kao i neprekidno praćenje,
- upravljanje napajanjem i mrežnom komunikacijom,
- redundantnost i različiti putevi komunikacije,
- bezbednost mreže, kontrola pristupa i video nadzor,
- kontrola životne sredine,
- suzbijanje požara kao i njihova detekcija [1][4].

3. OPREMA

U prethodnom poglavlju je definisana infrastruktura data centara. Takođe je navedena i podela opreme na:

- mehaničku,
- električnu,
- IT opremu [10].

3.1. Mehanička oprema

Mehanička oprema data centara se odnosi na rekove i šasije. Data centri su uglavnom organizovani u redove rekova, pri čemu oni služe za skladištenje komponenata kao što su serveri, svičevi itd. Standardna veličina reka je 78 inča visine, 23-25 inča dužine i 26-30 inča širine. Uvedena je standardna veličina za rekove koja se naziva *rack unit* ili samo U i iznosi 44,5 mm. Veći deo servera koji su sa jednim ili dva ulaza mogu stati u veličinu 1U, ali veći serveri kao što je na primer multiprocesor sa 4 ulaza zahteva veličine 2U ili veće. Standardni rekovi mogu primiti 42 komada opreme koji zahtevaju veličinu od 1U da bi bili u potpunosti ispunjeni. Složenost rekova može da varira od najjednostavnijih, koji ne predstavljaju ništa više od metalnih kućišta, pa sve do rekova koji imaju dodatne funkcije kao što su distribucija energije, ugrađen KVM (*eng. keyboard-video-mouse*) svič, sopstveno hlađenje itd. Za veću kompaktnost serveri mogu biti smešteni u samostalne šasije koja je i sama po sebi rek. Ona ima sopstveno napajanje, bežičnu konekciju i menadžment infrastrukture. Obezbeđuje standardnu veličinu slotova gde se mogu ubacivati *modular assets* poznati još i kao blejd. Ona je visine 13 inča i 6 šasija se može staviti u rek, pri čemu ona može primiti 16 1U servera tako da se teoretski ukupno u rek može staviti 96 *modular assets*. Povećanje gustine servera koristeći blejd formu dovodi do povećanja potrošnje energije, na primer u tradicionalnom data centru trošeno je 7 kW električne energije po reku dok rek ispunjen blejd serverima troši 21 kW. Ovo očigledno dovodi do problema pregrevavanja koji sistem za hlađenje delimično može da reši. Procena je da troškovi izgradnje i opremanja data centra površine oko 1000 m², zavisno od primenjenog stepena redundantnosti, iznose od 10 do 35 miliona dolara. Mehanička infrastruktura sadrži ceo mehanički sistem koji joj omogućava da podrži rad data centara. Ove strukture mogu podržati i modularne data centre kao i tradicionalne data centre. Poenta cele mehaničke infrastrukture u data centrima je da se u njoj smešta i održava oprema, pre svega IT oprema koja je neophodna za rad data centara, kao i da se stvore uslovi za njen nesmetan rad [9][10].

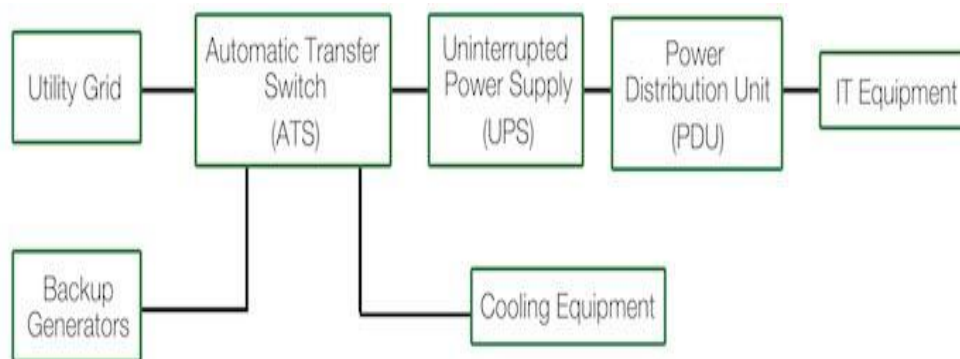
3.2. Električna oprema

Električna oprema kod data centara je bazirana na redundantnosti i njena glavna uloga je konstantno obezbeđivanje i dovođenje električne energije, koja je potrebna za rad IT opreme kao i za rad samog data centra. Ova oprema se uglavnom odnosi na napajanje i nju u većini data centara čine: bekap generator, predajnici snage, automatizovani svičevi za transfer ATS (*eng. Automatic transfer switches*), UPS (*eng. Uninterrupted Power Supplies*), jedinica za distribuciju snage (PDU) i akumulatori [10].

Najjednostavniji sistem za distribuciju energije čine: UPS, PDU, *Remote Distribution Cabinet* (isturen kabinet za distribuciju energije) i energetska traka (*eng. power strip*).

Opis najbitnijih delova električne opreme je dat u nastavku:

- *Utility Grid* (Komunalna Mreža) - je komercijalni distributivni elektroenergetski sistem koji distribuira energiju do data centra koji predstavlja potrošača. On je ultimativni izvor energije za data centre, a nekada ih može biti i dva na koje su povezani objekti da u slučaju pada jednog drugi nastavi sa snabdevanjem [10].



Slika 3.2.1: Dijagram el. mreže [10]

- Bekap generatori - predstavljaju generatore na bazi dizela koji služe u svrhu testiranja ili u situacijama kada dođe do havarije npr. nestanka struje ili kvara na nekom delu opreme, tada se on uključuje i snabdeva energijom data centar kako ne bi došlo do prestanka njegovog rada. On može snabdevati data centar energijom satima pa čak i danima dok se kvar ne otkloni. Za velike data centre oni se obično raspoređuju u setovima [10].



Slika 3.2.2: Pomoćni generatori [10]

- **ATS** (*eng. Automatic transfer switches*) - ATS svičevi automatski prebacuju pristup sa jednog izvora napajanja na drugi, tako da se u slučaju kvara izvora napajanja koje se koristilo, prebaci automatski na rezervni (alternativni) izvor energije.
- **UPS** (*eng. Uninterrupted Power Supplies*) - uređaj koji omogućava opremi da radi još neko vreme dok se sistem ne prebaci na pomoćno napajanje usled prekida rada primarnog izvora energije. Uglavnom se sastoji od baterije u kojoj se skladišti energija taman toliko da se premosti vremenski jaz do paljenja rezervnih izvora napajanja [10].



Slika 3.2.3: UPS [10]

- PDU (eng. *Power Distribution Unit*) - zove se i MDU (eng. *Main Distribution Unit*) i predstavlja svojevrsnu razvodnu kutiju [10].



3.2.4: Power Distribution Unit [10]

- Oprema za hlađenje - njena svrha je da ukloni toplotu koju stvara IT oprema. Kao i sva oprema pri radu tako i serveri, komunikaciona oprema i oprema za skladištenje se zagrevaju pri konstantnom korišćenju i ako se zagrevanje ne neutrališe doći će do kvara na njoj i prestanka rada data centra. Infrastruktura za hlađenje se sastoji od vazdušnih jedinica za hlađenje *Computer Room Air Conditioners* (CRAC) ili *Computer Room Air Handler* (CRAH).
 - CRAC je uređaj koji prati i održava temperaturu, protok vazduha i vlažnost u data centru. Ove jedinice zamenjuju klima uređaje koji su korišćeni za hlađenje u prošlosti [11].*
 - CRAH je uređaj koji se koristi u data centrima da bi se smanjilo pregrejavanje opreme. Za razliku od CRAC koji koristi mehaničko hlađenje da bi ohladio vazduh u data centru, CRAH koristi ventilatore, rashladne kalemove i sisteme pumpi za ukljanjanje toplote [11].*



Slika 3.2.5: Liebert CRAC Unit [10]

Takođe u opremu za hlađenje spadaju i *Chiller plants* (postrojenja za hlađenje) koji se često koriste kao izvor hlađenja u većim data centrima, jer efikasnije otklanjaju toplotu nego DX (eng. *Direct expansion*) jedinica. One stvaraju hladnu vodu koju šalju jedinici za hlađenje unutar data centra.



Slika 3.2.6: Chiller-plant [10]

Ovaj sistem sadrži i sledeće komponente:

- hladnjak,
- pumpe,
- cevovod sa hladnom vodom,
- kondenzatore ili kule za hlađenje,
- jedinice za kontrolu klime.

Pošto se potrošnja u data centrima ubrzano povećava, naročito sve većom upotrebom blejd servera u data centrima, za ovakvo opterećenje potreban je izvor velike energije. Pri pretvaranju i prenosu električne energije, dolazi do velikih gubitaka koji mogu dostići na samom kraju i do 50%. Stepem efikasnosti korišćenja električne energije značajno utiče na konačne troškove u održavanju data centara. Najveći problem za električnu opremu predstavlja vlažnost tako da moraju biti ispunjeni određeni uslovi za njeno smeštanje. Dobra lokacija mora biti suva i dobro provetrena prostorija u klimatizovanoj zgradi. Vlažnost treba biti manja od 80%, a temperatura u rasponu od 0-40 stepeni celzijusa. Treba izbegavati drastične promene temperature i vlažnosti i po svaku cenu treba uslove uskladiti tako da se izbegne kondenzacija. To se može izvršiti podizanjem temperature unutar opreme [4][9].

3.3. IT oprema

Kompanije predstavljaju složenu organizaciju ljudi i tehnologija. Arhitektura preduzeća obezbeđuje suštinsku poslovnu logiku za poslovne procese kao i za IT infrastrukturu i funkcije koje podržavaju operativni model samog poslovanja. Većina preduzeća je postala gotovo u potpunosti zavisna od IT usluga da bi isporučile robu i usluge klijentima. Osnovne komponente IT infrastrukture su:

- konfiguracijski artefakt (*eng. Configuration Artifact*) - predstavlja podatke i kod koji se koristi za upravljanje konfiguracijom infrastrukture softvera i usluga (npr. *Puppet* moduli),
- infrastrukturni uređaj (*eng. Infrastructure Device*) - hardver koji može biti dodeljen softveru ostvarenom pomoću konfiguracijskog artefakta (npr. *Rack-mounted blade server*),
- čvor (*eng. Node*) - računarski resursi unutar uređaja u kojima se nalaze infrastrukturni resursi, čija je konfiguracija definisana pomoću konfiguracijskog artefakta (npr. *Amazon EC2 instance*),

- softver (*eng. Software*) - skup aplikacija koje usmeravaju čvor da vrši neku funkciju (npr. *MySQL software*),
- infrastrukturna usluga (*eng. Infrastructure Service*) - funkcionalnost sa dobro definisanim interfejsom smeštenim na jednom ili više čvorova (npr. *Apache Tomcat service*),
- mreža (*eng. Network*) - fizička infrastruktura koju koriste dva ili više uređaja radi komunikacije (npr. *Network Interface Card*),
- put kroz mrežu (*eng. Network Path*) - logička veza na mreži koju koriste dva ili više čvorova da bi komunicirali (npr. mrežni put između dva uređaja).

Na osnovu uvida u IT infrastrukturu, IT oprema koja se koristi u data centrima se deli na:

- servere,
- mrežnu opremu,
- opremu za skladištenje [10].

3.3.1. Serveri

Serveri koji se koriste u data centrima dele se na tri tipa: *tower*, *rack-mounted* i blejd servere. Serverska infrastruktura može biti zasnovana na Linux ili Windows platformi. *Rack-mounted* serveri i blejd serveri se najčešće koriste u data centrima, tako da će oni biti detaljno objašnjeni i biće navedena poređenja između njih, a takođe će biti date i osnovne informacije o virtuelizaciji na serverima.

Tower serveri

Predstavljaju kompjutere čija je uloga da služi kao server koji su ugrađeni u uspravnim kućištima. Ovo kućište, koje se još naziva i kula (*eng. tower*), slično je po obliku kao i personalni kompjuter u stilu *tower*. Ovaj server je u suprotnosti sa rek ili blejd serverima. Njegove prednosti su: lakše hlađenje, jer nije velika gustina njegovih komponenata kao i skalabilnost, jer neograničen broj servera može biti dodat na postojeću mrežu. Ipak, njegove mane su ono što ga čini nepogodnim za data centre. Skup ovakvih servera je glomazniji i teži od blejd ili rek servera. Kabliranje može biti problem ako je veći broj ovih servera na jednom mestu i svaki toranj zahteva namenski ventilator, pa skup ovakvih servera može stvarati veliku buku [12].

Rack-mounted serveri

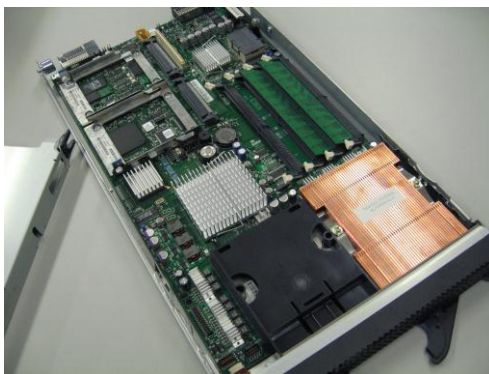
Takođe se nazivaju i rek serveri. To su kompjuteri koji služe kao server i dizajnirani su da mogu da se instaliraju u okvire koji se nazivaju rekovi. Stalac za montažu sadrži više slotova koji se nazivaju *bays* i svaki je dizajniran da održi hardversku jedinicu bezbednom. Rek serveri imaju nisko kućište za razliku od *tower* servera. Jedan rek može da sadrži više servera naslaganih jedan iznad drugog, čime se konsoliduju mrežni resursi i smanjuje zauzimanje prostora u data centru. Konfiguracija rek servera pojednostavljuje kablovsko povezivanje između mrežnih komponenata. U reku koji je ispunjen serverima, potrebno je specifično hlađenje da bi se izbeglo pregrejavanje koje nastaje zbog velikog broja komponenti na jednom mestu [13].



Slika 3.3.1: Rack-mounted server [13]

Blade serveri

Blejd serveri su tanke, modularne elektronske ploče koje sadrže jedan ili više mikroprocesora kao i memoriju, koja je namenjena za jednu aplikaciju i oni se lako mogu ubaciti u rek prostor sa sličnim serverima. Blejd serveri su serveri na kartici koji sadrže procesor, memoriju, ugrađene kontrolore mreže i opcioni *Fiber Channel host bus* adapter (HBA) i druge ulazno/izlazne portove. Omogućavaju veću procesorsku snagu u rek prostoru, pojednostavljuju kabliranje i smanjuju potrošnju energije. Njihovom primenom se 85% smanjuje kabliranje tokom njihovog rada u odnosu na standardne 1U ili *tower* servere. Samim tim IT osoblje može da utroši manje vremena na održavanje infrastrukture i više vremena da posveti obezbeđivanju veće dostupnosti. Svaki blejd server dolazi u paketu sa jednim ili dva lokalna ATA (*eng. Avanced Technology Attachment*) ili SCSI (*eng. Small Computer System Interface*) diska. Za dodatno skladištenje oni mogu da se povežu sa bazenom za skladištenje (*eng. storage pool*) pomoću NAS (*eng. network attached storage*), *Fiber Channel* ili *iSCSI storage area network* (SAN). Prednost blejd servera nije samo u skladištenju nekoliko servera u jednu šasiju, već i u konsolidaciji resursa (kao što su mrežna oprema i oprema za skladištenje) u manje arhitekture kojima se upravlja preko jednog interfejsa. Blejd serveri se još i nazivaju serveri velike gustine, imaju centralizovanu konzolu za upravljanje i koriste se za grupisanje servera koji su posvećeni nekoj funkciji [14].



Slika 3.3.2: Blade server [14]

Uvek se postavlja pitanje koje servere treba izabrati za rad data centara. Neke kompanije se radije odlučuju za *rack-mounted* servere zbog niže cene implementacije i zbog toga što nema potrebe za modifikacijama prema izvorima energije data centara. Drugi smatraju da je centralizovano upravljanje koje pružaju blejd serveri pomoću konzole veoma velika prednost ovih servera. Prednost je i to što obezbeđuju potrebnu snagu, kablovsku i infrastrukturnu efikasnost koja je neophodna u sve prenatrpanijim današnjim data centrima. Iz ovih razloga se vodi debata koji izbor napraviti po pitanju

arhitekture u virtuelizovanom okruženju. Posmatrano sa stanovišta rek servera oni manje koštaju od blejd servera, ne zahtevaju dodatno napajanje na višoj amperaži, koje je teško održavati, a i lakše se prelazi na virtuelizaciju. Ako posmatramo sa strane blejd servera, centralizovana konzola za upravljanje omogućava administratorima da vide fizičko okruženje servera uključujući imena, IP adrese, konfiguraciju hardvera i radnog statusa. Administratori kontrolišu snagu servera, vide konzolu servera, odgovaraju na upozorenja i prosleđuju resurse. Sve ovo smanjuje broj fizičkih mesta koje administrator mora da poseti, postiže se veća efikasnost i smanjuje broj kablova koji se koriste čime se pravi više mesta u reku [13][14].

Rezime je da je najbolje kombinovati obe tehnologije, jer se tako postiže bolje upravljanje i sve to uz povećanje vidljivosti i dostupnosti podataka i aplikacija.

i) Virtuelizacija

Virtuelizacija kao tehnologija je postala trend u svetu bez koga se ne može zamisliti rad data centara modernih kompanija. Iako je kao tehnologija virtuelizacija dosta dugo prisutna, tek razvojem podrške za virtuelizaciju na x86 platformi postala je široko zastupljena kako u velikim preduzećima tako i kod manjih korisnika. Nasuprot tradicionalnom konceptu „jedna aplikacija – jedan fizički server”, virtuelizacija servera predstavlja model gde se između hardvera i operativnog sistema uvodi sloj apstrakcije tzv. hipervizor. Nakon toga je na jednom fizičkom serveru moguće hostovati više virtuelnih mašina i servisa na njima. Na ovaj način se drastično smanjuju troškovi data centra kroz efikasnije iskorišćenje hardvera i smanjenje operativnih troškova (troškovi za napajanje i hlađenje, zakup prostora) i troškova održavanja. Sa druge strane, dostupnost servisa više ne zavisi od hardvera [15].

Globalne procene su da je prosečna iskorišćenost serverske snage 10-15%. Njihovom virtuelizacijom se postiže maksimalno iskorišćenje resursa, a data centar se pretvara iz statičkog u dinamički centar za isporuku IT servisa.

Virtuelizacija servera daje najbolje efekte ukoliko se kombinuje sa deljenim sistemom za masovno skladištenje podataka - centralizovanim skladištem podataka (*eng. storage*). Pouzdanost i raspoloživost sistema se na taj način podižu na najviši nivo.

Najvažniji proizvođači hipervizora su svakako Microsoft, Vmware, Oracle i Citrix, a Datatek ima višegodišnje iskustvo sa sva tri rešenja kroz mnoge realizovane projekte. Najveći proizvođači serverskih sistema su IBM i HP [1].

3.3.2. Mrežna oprema

Mrežna oprema je oprema koja olakšava povezivanje uređaja i deljenje podataka, kako unutar data centra tako i van njega. Njene funkcije su raznovrsne: povezivanje (*eng. interfacing*) uređaja na komunikacioni sistem, sinhronizacija između predajnika i prijemnika, razmena podataka prema odgovarajućem protokolu, usaglašavanje brzine slanja i prijema podataka, adresiranje i usmeravanje, formatiranje podataka, zaštita tih podataka itd. Sve ove funkcije mrežnog sistema su okarakterisane uslovima apstraktnih slojeva. Apstraktni sloj se koristi za skrivanje detalja implementacije, posebno skupa funkcionalnosti. Veći deo mrežne opreme i softvera u data centrima koristi OSI/IEC-7498-1 standard *Open System Interconnection* (OSI) model sa 7 slojeva za protokole kompjuterske mreže. U ovom modelu slične funkcije komunikacije su grupisane u logičke slojeve. Sloj sa manjom numeracijom služi sloju iznad sebe.

Fizički sloj

Fizički sloj (*eng. Physical Layer*) je prvi i najniži od sedam slojeva OSI modela računarske mreže. Prevodi komunikacijske zahteve od sloja veze u specifične operacije u tehničkim uređajima, tj. prevodi bitove okvira u odgovarajući kodiran električni, optički ili radio signal, a potom vrši predaju, kontrolu prenosa i prijem. Definišu se naponski nivoi, broj pinova na konektorima (odnosno parica u kablovima) ili debljina oplata koaksijalnog kabla. Mrežne kartice (integrisane na matičnoj ploči ili samo utaknute u sabirnicu na matičnoj ploči), habovi i ripiteri su primeri uređaja na fizičkom sloju OSI modela [16].

Sloj veze

Brine se za razmenu podataka u vidu okvira između mrežnih uređaja i za detekciju/korekciju mogućih grešaka u fizičkom sloju. Uređaji komuniciraju pomoću adresa (MAC adrese kod *ethernet* mrežnih uređaja) i komunikacija na ovom nivou je moguća samo unutar lokalnih mreža. Svičevi (*eng. switches*) su uređaji koji rade na sloju podataka jer oni čuvaju u memoriji MAC adrese svih mrežnih uređaja čije okvire su prosleđivali. Kada do svičeva dođe okvir oni, na osnovu određene MAC adrese i sadržaja tabele usmeravanja, odlučuju da li se okvir prosleđuje ili odbacuje, i u slučaju da se okvir prosleđuje na koji port/portove se prosleđuje [16].

Mrežni sloj

Mrežni sloj je OSI sloj čija se uloga ogleda u usmeravanju podataka na njihovo krajnje odredište i adresiranju koje omogućava jedinstvenu adresu svakom korisniku u globalnoj mreži. IP je protokol mrežnog sloja, koji predstavlja okosnicu Interneta. Ruteri su primer uređaja koji rade na mrežnom sloju [16].

Transportni sloj

Ovaj sloj vodi računa o paketima koji putuju između dva računara. Primeri protokola na transportnom sloju su TCP i UDP. Ako se neki paket „izgubi“ na putu, TCP će tražiti da se ponovo pošalje, pa je zbog toga pogodan za razmenu podataka, gde je pouzdanost prenosa od esencijalnog značaja. S druge strane, UDP nema kontrolu da li se poneki paket zagubio, pa je zgodan za multimedijalne aplikacije, gde nije toliko bitno da li se zagubi poneki paket, nego je bitna brzina komunikacije [16].

Sloj sesije

Bavi se uspostavljanjem veze između krajnjih korisnika i sinhronizacijom iste. Najlakše ga je objasniti kod videa preko Interneta, gde nije prihvatljivo imati ton bez slike ili sliku bez tona ili oboje, ali bez sinhronizacije. Za to se brine ovaj sloj [16].

Sloj prezentacije

Podaci koji se koriste na raznim računarima se kodiraju na različite načine (*eng. littleendian, eng. bigendian*). Tekstualne datoteke na Mac, Unix i Windows operativnim sistemima na različite načine označavaju prelazak u novi red. Sve takve konverzije se izvode (ukoliko su implementirane) na prezentacionom sloju [16].

Sloj aplikacije

Na ovom sloju programer koristi API, kojima ostvaruje mrežnu komunikaciju s određenom svrhom, a da pritom ne mora voditi računa o nižim slojevima, za koje se brine operativni sistem. Primeri protokola na ovom sloju su HTTP, FTP, telnet, SMTP, NNTP i mnogi drugi [16].

Mrežna opremu koja se najčešće koristi:

- Ripiter (*eng. Repeater*) je elektronski uređaj koji prima i reemituje signal. Ripiteri se koriste za pokrivanje većih udaljenosti.
- Čvorište (*eng. Hub*) je centralno mesto povezivanja koje najčešće služi da međusobno poveže delove LAN mreže. Ima više portova i kada paket stigne na jedan port kopira se ka ostalim portovima da svi delovi LAN-a mogu da vide sve pakete.
- Most (*eng. Bridge*) je uređaj koji povezuje LAN sa drugom LAN mrežom koja koristi isti protokol.
- Svič (*eng. Switch*) je uređaj koji vrši komutiranje okvira i praktično predstavlja most sa više portova. Kada se primi okvir, on pogleda MAC tabelu za MAC adresu iz primljenog okvira, da se odredi preko kog porta je dostupno konačno odredište okvira. Taj okvir se šalje na taj port samo ako nije u pitanju isti port sa kog je primljen okvir, u tom slučaju se okvir odbacuje. Ako nema zapisa u MAC tabeli svič se ponaša kao hab.
- Ruter (*eng. Router*) je mrežni uređaj koji vrši komutiranje paketa (na mrežnom sloju).
- Gejtvej (*eng. Gateway*) je tačka u mreži koja se ponaša kao ulaz za drugu mrežu. Tipično se koristi za ostvarivanje komunikacije između mreža baziranih na različitoj tehnologiji.
- Mrežna kartica (*eng. Network card*) je deo koji se brine za komunikaciju računara preko računarske mreže. Moderne matične ploče obično na sebi imaju integrisan mrežni čip i priključak, ali takođe postoje i mrežne kartice koje se ubacuju u PCI ili drugi tip slot. Danas se manje vidaju odvojene mrežne kartice, obično se uzima dodatna kartica (uz integrisanu) zbog mogućnosti priključivanja više mrežnih uređaja (npr. ADSL modem (eternet) i mrežni hab), iako neke matične ploče dolaze i sa dva mrežna čipa, odnosno mrežna priključka.

3.3.3. Oprema za skladištenje

Skladištenje u data centrima se može ostvariti na više načina. Često se skladišta sa visokim performansama smeštaju u takozvane *storage towers* koje omogućavaju transparentan daljinski pristup skladištu bez obzira na tip i broj fizičkih uređaja koji se koristi. Skladištenje može takođe biti obezbeđeno i u manjim *storage bricks* koje su smeštene u rekove ili slotove šasija ili su direktno povezane sa serverom. Data centar tipično zahteva četiri tipa mrežnog pristupa i potencijalno može koristiti četiri tipa fizičke mreže. Mreža klijent-server omogućava spoljašnji pristup data centru i obavezno koristi tehnologije kao što su žičani *Ethernet* ili bežični LAN. Server-na-server mreže mogu da koriste *Ethernet*, *InfiniBand* (IB) ili druge tehnologije. Pristup skladištu tradicionalno obezbeđuje *FiberChannel*, ali se takođe može koristiti i *Ethernet* ili *InfiniBand*.

InfiniBand predstavlja standard za komunikaciju koji podržava visoke protoke i malo kašnjenje, pri čemu se može koristiti i za komunikaciju jedinica unutar računarskog sistema i za komunikaciju između računara. On nudi protok od 2.5Gb/s pa naviše. Podržava rad i sa bakarnim provodnicima (do 30m) i sa optičkim kablovima (do 10km).

Skladištenje je veoma važna i složena komponenta u savremenim data centrima. Zahtevi koji moraju da se ispunjavaju pri gradnji skladišta u data centrima su:

Pouzdanost

Podaci koji se jednom sačuvaju neće nestati ili se promeniti. Ovo se odnosi na RAID zaštitu i druge vrste kopija podataka koje se često čuvaju na daljinu tj. da neće doći do njihove promene ili gubitka usled npr. kvara na glavnom disku [17].

Dostupnost

Meri se u procentima. Uglavnom aplikacijama u realnom vremenu je od male koristi sigurnost podataka ako ne mogu zbog protokola ili nekog sistemskog pada da pristupe istim podacima. Na godišnjem nivou u nekim data centrima kao što je napomenuto dostupnost dostiže 99,999% [17].

Zadržavanje i usklađenost

Podrazumeva da mnoge kompanije imaju uslove za zadržavanje podataka u skladu sa raznim pravnim direktivama kao što su Sarbanes/Oxley, HIPAA itd. Bilo kakvo zadržavanje podataka podrazumeva namenski sistem za upravljanje dokumentima i skladište koje njemu odgovara. Ovo može ograničiti izbor za sistemsku memoriju za tu komponentu [17].

Oporavak od katastrofe (*eng. disaster recovery*)

Po pitanju skladištenja suština je da se podaci čuvaju na drugom mestu koji su sinhronizovani sa podacima u glavnom skladištu. Moraju se ispuniti uslovi da u takvoj situaciji skladište na udaljenoj lokaciji pruži podacima uslove za nesmetan nastavak rada i to je poenta bezbednog poslovanja [17].

Postoje brojne tehnologije za skladištenje:

- *Storage Area Networks (SANs)*

Razvila se od direktno priključenog diska. Otvoreni kompjuterski sistemi kompanija koriste protokol SCSI da komuniciraju sa svojim hard diskovima. Rani SAN sistemi su objedinjavali nekoliko diskova čineći ih kao jedan i davali administratorima sposobnost da rezbare particije nazvane *Logical Units (LUN)*. Današnji sistemi koriste nekoliko tipova virtuelizacije kako bi napredne tehnike upravljanjima podataka mogle da kopiraju podatke na udaljenim lokacijama. SAN infrastruktura je dalje nastavila da se razvija pojavom *FibreChannel*. Omogućena je zamena i postavljanje podataka na udaljenim lokacijama i da se velikim nizovima diskova opslužuje stotine hiljada hostova. Primer moderne verzije SAN tehnologije je iSCSI koji sažima SCSI preko TCP/IP mreže na ethernet. SAN tehnologije ostaju dominantne na tržištu opreme za skladištenje.

- *Network Attached Storage (NAS)*

Deli dosta istih ciljeva kao i SAN kao što je na primer konsolidacija skladišta na centralnu lokaciju. Služe za skladištenje nestrukturiranih podataka. SNIA (*eng. Storage Networking Industry Association*) definiše ove podatke kao podatke koji su bazirani kao fajl i mogu se posmatrati kao podaci generisani od strane aplikacije na nivou korisnika. Novell i NetApp su prve kompanije na tržištu koje su krenule sa ovom tehnologijom skladištenja dok tu još učestvuju i EMC, IBM i HP. Razlika između NAS i SAN je u lokaciji sistema datoteka. U SAN sistemima host je odgovoran za ugradnju i održavanje bilo kakvih fajl sistema. NAS sistemi koriste protokole za deljenje datoteka kao što su NFS ili CIFS da stvore, pročitaju, napišu ili obrišu fajlove. Fajl sistem se nalazi na serveru za skladištenje, a ne na hostu. NAS još nudi i blok skladištenje koje se postavlja na vrh fajl sistema. SAN i NAS instalacije zahtevaju značajnu mrežnu infrastrukturu i dovode do 30% potrošnje električne energije koja ide na sisteme za skladištenje.

Ako posmatramo hardverski deo opreme za skladištenje, on se sastoji iz dva fundamentalna dela: kontrolera i police za diskove. Energija koju koristi kontroler uglavnom odlazi na veliku keš memoriju koja se sastoji od DRAM (*eng. Dynamic Random Access Memory*). Kontrolori se često kombinuju u HA (*eng. high availability*) parove koji mogu biti aktivni-aktivni ili aktivni-pasivni. Ono što mora da se zna pri dizajniranju data centra je koliki će biti broj kontrolera i koliku će snagu koristiti svaki od njih u normalnim i ekstremnim uslovima. Police za diskove gledano sa energetskeg stanovišta se sastoje iz diskova i ventilatora. Radi uštede energije ventilatori moraju imati promenljivu brzinu i za veliki niz od preko 1000 diskova biće potrebno više od 100 ventilatora da ih hladi. Diskovi mogu biti različitih snaga, kapaciteta i performansi.

NAS su osnova sistema za skladištenje i dele se na:

- SSD (*eng. Solid State Drives*)

SSD je uređaj za skladištenje podataka, koji za to koristi integrisana kola. Namena mu je trajno skladištenje podataka, kao i hard disk. Iako se ponekad naziva „SSD diskom“, on ne sadrži mehaničke delove. SSD tehnologija za komunikaciju s ostatkom računara koristi isto elektroničko kolo kao i hard diskovi, što olakšava ugradnju i zamenu sporijih mehaničkih diskova. Zbog veće brzine pisanja i čitanja, konstruisano je novo ulazno/izlazno kolo SATA *Express* kako bi ostatak računarskog sistema pratio i maksimalno iskoristio novu tehnologiju. Glavni delovi SSD su memorijski kontroler i fleš memorija, tako da od njih najviše zavise performanse uređaja. Od 2010. većina SSD koristi NI baziranu fleš memoriju, koja čuva podatke bez napajanja. Za aplikacije koje zahtevaju brz pristup, ali ne nužno trajnost podataka nakon prestanka napajanja, SSD mogu biti napravljeni na memoriji sa slučajnim pristupom (RAM). Takvi uređaji mogu koristiti različite izvore napajanja, kao što su baterije, da održe podatke nakon prestanka napajanja. Nude najnižu cenu po I/O zahvaljujući niskom kašnjenju koje prave. Ali su takođe najskuplji po uskladištenom terabajtu čak 10 puta više od ostalih diskova.

- Diskovi visokih performansi FC (*eng. Fiber Channel*)

Oslonac za skladištenje u kompanijama u poslednjih 20 godina. Oni rade 2-3 puta brže nego sporiji SATA hard diskovi ali imaju između 1/3 i 1/2 kapaciteta SATA diskova. Od 2008 su uglavnom zamenjeni od strane SAS (*eng. Serial Attached SCSI*) diskova. SAS ima nešto bolje performanse od FC jer se uklapa sa *Fibre Channel* protokolom. Zbog ovoga nije pogodan za veće razdaljine pa se koristi za mrežno *backend* skladištenje podataka između kontrolera i njihovih diskova. Diskovi sa velikim performansama koriste do 6 puta više energije nego SATA, ali zato nude veću pouzdanost i znatno bolji učinak zbog njihove povećane rotacione brzine.

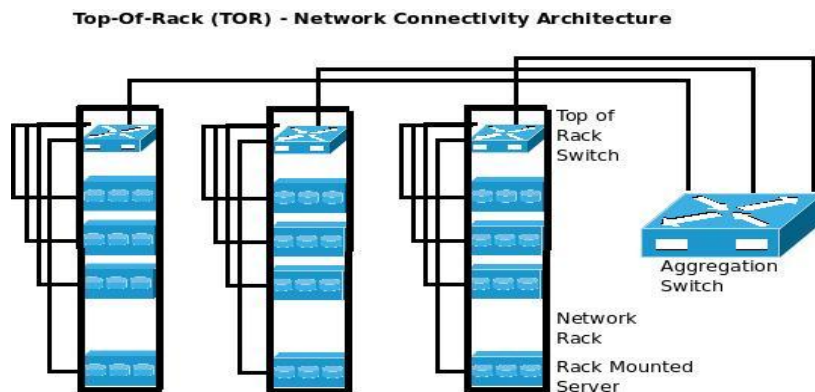
- Serijski ATA (*eng. Serial Advanced Technology Attachment*)

Ovi diskovi su prvobitno usmereni na potrošačko tržište. Ali onda su se preselili u kompanije kada im je povećana pouzdanost i životni vek i oni nude najbolju energetske efikasnost po terabajtu. Zamenio je stariji paralelni ATA (PATA) standard. Nudi više prednosti kao što su smanjenje veličine i dužine kablova i njihovu cenu, brže prosleđivanje podataka i efikasniji prenos preko I/O protokola. Njegove komponente komuniciraju pomoću kablova velike brzine preko dva para provodnika (SATA interfejs).

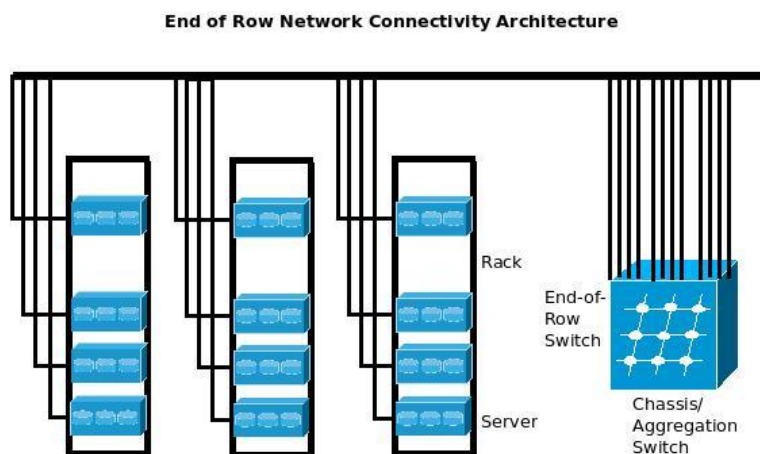
4. TOPOLOGIJE

4.1. Klasifikacija topologija

Mrežne topologije u data centrima koje povezuju servere imaju značajan uticaj na agilnost i mogućnost rekonfiguracije infrastrukture data centara, kao odgovor na rastuće zahteve aplikacija i usluga. Koriste se uglavnom u klasičnoj topologiji data centara ToR (*eng. Top of Rack* - dizajn u kome se serveri povezuju sa jednim ili dva ethernet sviča, koji se nalaze unutar reka; stvarna pozicija sviča ne mora da bude na vrhu reka), EoR (*eng. End of Row* - tradicionalan pristup kada se na veliki svič povezuju serveri iz jednog ili više rekova) i obični svičevi.



Slika 4.1.1: ToR svič [18]



Slika 4.1.2: EoR svič [18]

Arhitektura mreže mora biti skalabilna, isplativa, laka za upravljanje, otporna na kvarove (*eng. fault tolerant*). Da bi se ovo postiglo, arhitektura data centara mora ispuniti određene uslove [19]:

- bilo koja VM (*eng. Virtual Machine*) može da se prebacuje na bilo koju mašinu bez potrebe za menjanjem njene IP adrese,
- ne postoji potreba da administrator rekonfiguriše bilo koji svič pre postavljanja,
- bilo koji krajnji host treba da ima mogućnost da efektivno komunicira sa bilo kojim drugim hostom preko bilo kog slobodnog puta,
- nema pojave petlje - kruženja podataka u mreži,
- mora se ostvariti efikasna i brza detekcija kvarova.

Tipična konfiguracija mreže u data centru je karakteristična po tome da se krajnji host povezuje na ToR svičeve koristeći 1GigE link. ToR svičevi obično imaju 48 1GigE portova povezanih sa krajnjim hostovima i do četiri 10GigE apinkova. Dizajn mrežne topologije data centra treba da obezbedi bogatu povezanost između ToR svičeva tako da bi se ispunili gore navedeni uslovi.

Ethernet se najčešće koristi u data centrima i čvorovi se mogu konfigurisati da rade u *Ethernet-switched* modu ili *IP-routed* modu. Može se koristiti i *Myrinet*, koji predstavlja veoma brz LAN sistem, dizajniran za povezivanje više mašina, u cilju stvaranja kompjuterskih klastera. Adresiranje može biti ravno tj. interfejsu se dodeljuje bilo koja adresa bez obzira na položaj u topologiji i to je *Ethernet*. Može biti hijerarhijsko, tada dodeljivanje adrese interfejsu zavisi od položaja u topologiji i to je IP adresiranje. *IP-routed* mreže su skalabilnije od *Ethernet-switched* mreža zbog manjih tabela za prosleđivanje. Ovo su najčešće korišćene mreže u data centrima. Pitanje razvrstavanja topologija u data centrima je veoma kompleksno i postoji više tipova takve podele. Hijerarhijske i ravne topologije, *switch-only* i *server-only* topologije, fiksne i fleksibilne topologije su neke od podela. U ovom radu podela je predstavljena fiksnim i fleksibilnim arhitekturama, gde su topologije razvrstane na osnovu toga da li je mrežna topologija ostala fiksna od trenutka postavljanja [19].

Fiksne topologije dele se na:

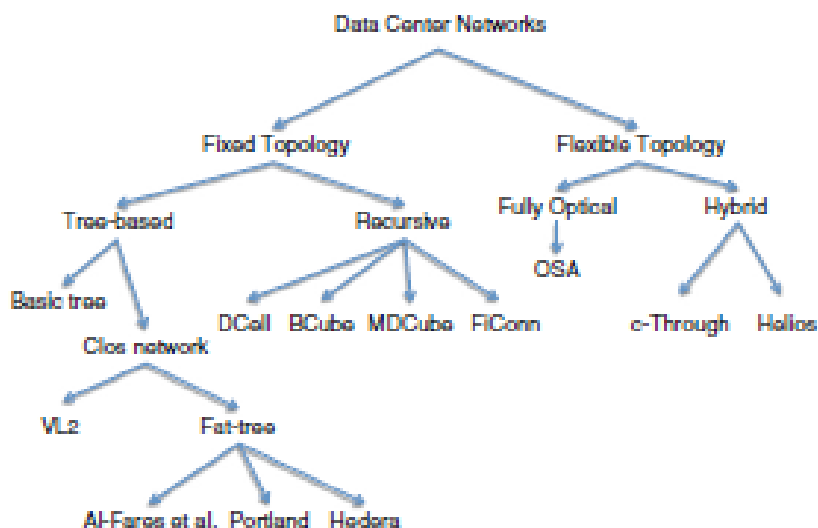
- topologije bazirane na stablu (*eng. tree based*),
- rekurzivne topologije.

Fleksibilne topologije dele se na:

- hibridne topologije,
- potpuno optičke topologije.

Dalje grananje topologija baziranih na stablu se može definisati na osnovno stablo (*eng. basic tree*) i Clos mrežu (*eng. Clos network*) koja se opet deli na VL2 i *Fat-tree* topologiju. U rekurzivne topologije spadaju DCell, BCube, MDCube, FiConn i potpuno optička topologija OSA. Hibridne su c-Through i Helios. Topologije koje još treba pomenuti su: Scafida i Dragonfly. Oznake koje se koriste u kasnijem objašnjavanju topologija:

- n - broj portova na sviču,
- k - broj portova na serveru,
- N - ukupan broj servera u mreži data centra.



Slika 4.1.3: Podela topologija [19]

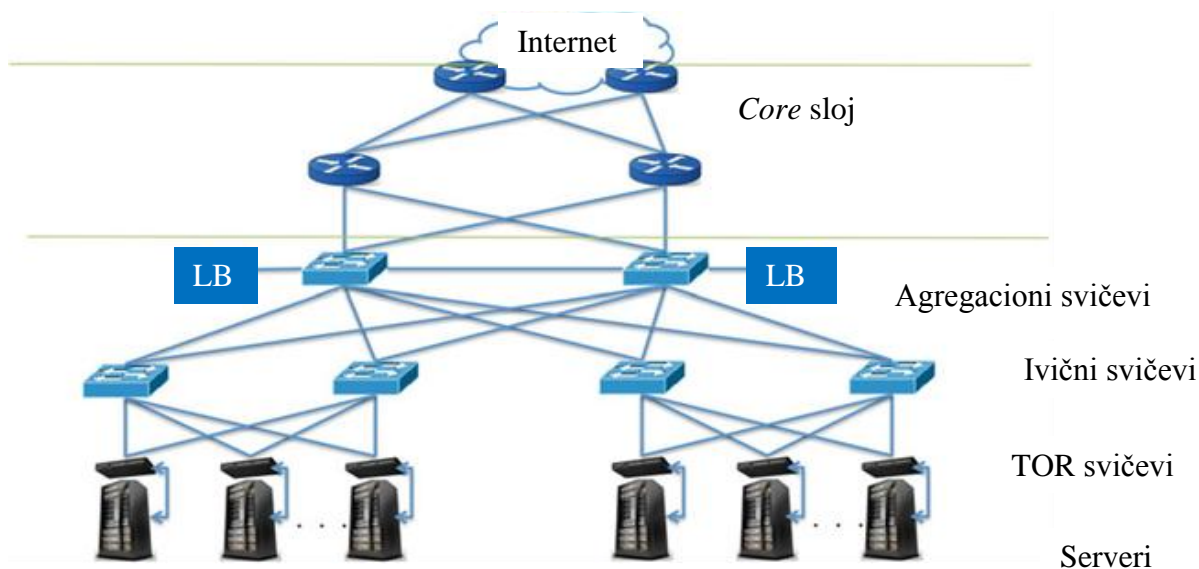
4.2. Fiksne topologije bazirane na stablu

Ove topologije su široko rasprostranjene u data centrima. Cisco kao jedna od vodećih kompanija na polju data centara je predložila topologiju 1+1 redundantnosti koja je *multi-tier* topologija stabla. Kod nje su ToR svičevi povezani na ivične (*eng. edge*) svičeve, a ovi opet na agregacione svičeve. Ivični i ToR svičevi predstavljaju svičeve pristupnog sloja. Agregacioni svičevi su povezani na *core* sloj. Ova mreža je hijerarhijska i redundantna, zbog povezanosti na višestruke svičeve iz susjednog sloja. Svičevi na višem sloju obrađuju više saobraćaja što zahteva veću robusnost tih svičeva i ove topologije su uglavnom skalabilne [19]. Po pitanju zahteva, fiksne topologije na bazi stabla se ponašaju na sledeći način [20]:

- skalabilnost – moderni data centri moraju biti prilagodljivi da prime ogroman broj servera i omoguće proširenje mreže. Dodavanje servera je relativno jednostavno i sa tog stanovišta ove topologije su skalabilne. Problem predstavlja potencijalno zagušenje jezgra mreže, pošto veliki deo saobraćaja prolazi kroz jezgro mreže.
- mapiranje aplikacija - u topologijama na bazi stabla, svaka aplikacija se preslikava na jedan VLAN, konstruisanim da bi se podržao server koji je zadužen za rad aplikacije. U optimalnim data centrima, svaki server može da se dodeli bilo kom servisu prema potrebnim zahtevima.
- konekcija *server-to-server* i protok kroz uska grla - međuserverski saobraćaj je ograničen prekomernim dodeljivanjem servera (*eng. oversubscription*), tako da kod topologija na bazi stabla dolazi do uskog grla na agregacionim i *core* svičevima.
- podela resursa i agilnost - hijerarhijska struktura topologije i prekomerno dodeljivanje servera utiče na zagušenost hijerarhijski viših slojeva (agregacioni i *core* slojevi) čime se slabe performanse celog data centra. Poenta podele resursa i agilnosti je da se aplikacije koje zahtevaju više servera dodele klasterima koji su međusobno bliski u topologiji. Ako aplikacija zahteva dodatne servere, može se desiti da slobodni serveri nisu bliski sa stanovišta topologije pa može doći do zagušenja viših slojeva topologije. Ovo ograničava agilnost u dodeli servera tj. rast aplikacije, pa se pokušava unapred proceniti koliko će servera biti potrebno za rad aplikacije u najgorem slučaju i na taj način alocirati u početku

sve servere što može dovesti opet do slabijeg iskorišćenja servera i problema prekomerne dodele servera.

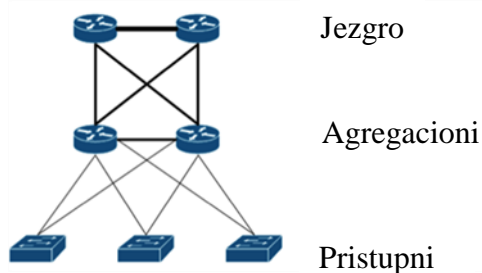
- pouzdanost i iskorišćenost - tipične topologije na bazi stabla imaju slabu pouzdanost i iskorišćenost. Višestruki putevi nisu efektivno iskorišćeni u ovim arhitekturama, pošto se koristi samo jedan put od strane STP (*eng. Spanning Tree Protocol*) unutar domena drugog sloja, iako postoji više puteva između svičeva. Ove topologije nude najviše dva potpuno odvojena puta. U agregacionom i *core* sloju hijerarhije stabla, maksimalan model elastičnosti je 1:1 što vodi ka tome da je maksimalna iskorišćenost kapaciteta svakog sviča i linka najviše 50%. Tako da klasičan model na bazi stabla, koji se koristi godinama, postaje nepodesan. Cilj data centra je da raspodeli resurse u pravo vreme tamo gde su potrebni, da bi se poboljšala pouzdanost servisa i iskorišćenost servera.
- balansiranje saobraćaja (*eng. load balancing*) - linkovi u *core* sloju su mnogo više iskorišćeni nego oni u pristupnom sloju tj. preko njih ide mnogo veća količina saobraćaja, dok je cilj da se saobraćaj ravnomerno raspoređuje kroz mrežne puteve u celoj mreži data centra. U ovim topologijama balansiranje saobraćaja ne može mnogo da pomogne zbog postojanja relativno malog broja alternativnih puteva.
- robusnost - kvarovi su česti u data centrima, najčešće su to kvarovi na serveru, reku, linku ili sviču. U data centrima na bazi stabla, ako bi došlo do kvara sviča u *core* sloju, došlo bi do prestanka rada data centra na nekoliko sati za milione korisnika. Dodavanje redundantnih svičeva bi moglo da smanji ovaj problem, ali ga ne rešava zbog slabe konekcije ovih arhitektura.
- potrošnja energije - pri gradnji ovih data centara potrošnja energije nije uzeta u obzir. Prekomerno dodeljivanje servera za rad aplikacija, dovodi do toga da veći deo vremena većina servera ne radi ništa, ali troši energiju čak i u stanju mirovanja. Samim tim dovodi do većeg utroška energije potrebne za servere i sisteme hlađenja, čime se i povećava cena održavanja data centra na bazi stabla.
- izolacija performansi - više različitih servisa se smešta u data centre. U tipičnim data centrima na bazi stabla, protok saobraćaja može da bude ometen od drugog protoka neke različite usluge ako koriste istu mrežu u podstablu.
- kapacitet mreže - kapacitet je ograničen zbog potencijalne zagušenosti viših slojeva, a koji može da se desi ako se aplikaciji dodele serveri koji nisu međusobno bliski u topologiji pa većina saobraćaja ide preko viših slojeva, pa ako se to dešava za sve aplikacije onda će viši slojevi biti zagušeni saobraćajem jer balansiranje saobraćaja ne može mnogo da pomogne usled postojanja relativno malog broja alternativnih putanja.



Slika 4.2.1: Cisco DCN topologija [19]

4.2.1. Topologija osnovnog stabla

Ovo je tradicionalna topologija za data centre koja ima dva ili tri sloja. Sastoji se od dva ili tri sloja svičeva/rutera sa serverima kao „lišćem“. U topologijama sa tri sloja nalazi se *core* sloj (sloj jezgra) u vrhu stabla koji obezbeđuje usluge rutiranja drugim delovima data centra, kao i za usluge van njega (npr. Internet). Agregacioni sloj u sredini obezbeđuje povezanost za svičeve ivičnog sloja i redove rekova data centara. Krajnji svičevi pristupnog sloja povezuju host sa mrežom. U topologiji sa dva nivoa ne postoji srednji ili agregacioni sloj. Ne postoji povezanost između svičeva na istom nivou ili svičeva koji se nalaze u nesusednim nivoima što se najbolje vidi sa slike 4.2.2. U ovoj topologiji svičevi na višem nivou prosleđuju veću količinu saobraćaja između većeg broja servera i samim tim su potrebni svičevi boljeg kvaliteta i performansi čime se odmah povećava i cena. U ovim topologijama može doći do zagušenja saobraćaja u agregacionom ili *core* sloju. Skalabilna je tj. mreža se proporcionalno može širiti dodavanjem novih svičeva u njene slojeve [19].

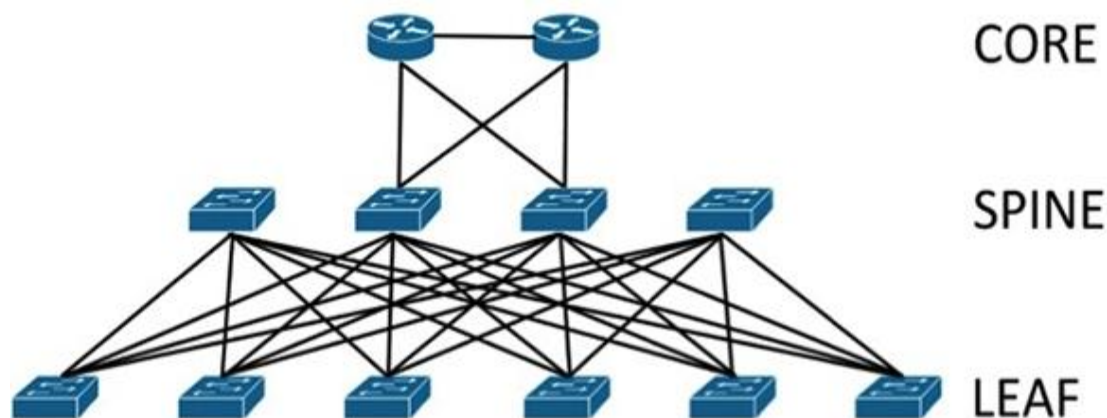


Slika 4.2.2: Osnovna topologija stabla sa tri nivoa [19]

4.2.2. Clos mreže

Osnovna *Clos* mreža se sastoji od tri sloja svičeva koji se nalaze na ulazu, sredini i izlazu mreže i ona u modernim data centrima predstavlja alternativu topologijama na bazi stabla. Ona se još naziva i *leaf-spine* topologija i može se posmatrati kao topologija sa tri sloja. Zbog drugog naziva ove topologije, za nju će se koristiti nešto drugačiji nazivi za svičeve, dok se analogija može videti na

slikama 4.2.2 i 4.2.3. U data centrima Clos mreža se uobičajeno sastoji iz tri vrste svičeva: *leaf* (ToR) svičevi, koji su direktno povezani sa serverima, agregacionih koji su povezani sa *leaf* svičevima i *core* svičevi, koji su povezani sa *spine* (agregacionim) svičevima. Ova topologija je slična topologijama stabla sa tri sloja samo ima više svičeva u *spine* sloju. U ovoj arhitekturi serije *leaf* svičeva čine pristupni sloj. Mreža obezbeđuje da su svičevi pristupnog sloja veoma blizu jedan drugom čime se kašnjenje i verovatnoća zagušenja svodi na minimum. Svi linkovi prenose saobraćaj tj. nisu blokirani, jer je STP (eng. *Spanning Tree Protocol*) zamenjen drugim protokolima. OSPF (eng. *Open Shortest Path First*) protokol se često koristi da izračuna puteve između *leaf* i *spine* svičeva. Radi efikasno u virtuelnim okruženjima što mu daje prednost pri virtuelizaciji. Virtuelizacija se uglavno bazira na *East-West* saobraćaju (komunikacija između servera).



Slika 4.2.3: Clos topologija [8]

4.2.3. Fat tree topologija

Topologija mreže u data centrima, koja se bazira na *fat tree* topologiji koristi standardne ethernet svičeve. Pristup stvaranja topologije je sledeći: svič sa n portova u pristupnom sloju je povezan sa $n/2$ servera. Preostalih $n/2$ portova je povezano sa $n/2$ svičeva u agregacionom sloju. $n/2$ agregacionih svičeva, $n/2$ svičeva na pristupnom sloju i serveri, koji su povezani na pristupne svičeve, formiraju osnovnu ćeliju *fat tree* topologije. Ova ćelija se još zove i *pod*. U *core* sloju postoje $(n/2)^2$ svičeva sa n portova, pri čemu je svaki od njih povezan sa svakom od n ćelija. Za razliku od osnovnog stabla sva tri sloja koriste istu vrstu svičeva i svičevi visokih performansi nisu potrebni u agregacionom i *core* sloju. Maksimalan broj servera u *fat tree* topologiji sa svičevima sa n portova je $(n)^3/4$. Ovakva topologija je specijalan slučaj *folded-Clos* mreže zato je i stavljena pod *Clos* mrežu u klasifikaciji.

Prednosti ove topologije su:

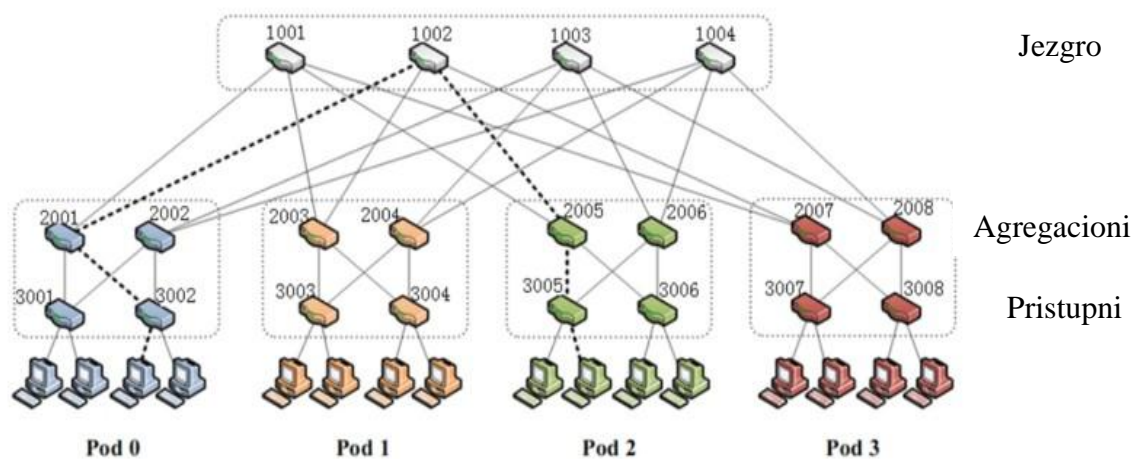
- ima isti protok u svim delovima mreže,
- svaki sloj ima istu količinu protoka,
- jeftina je, jer svuda koristi iste komponente,
- svaki port podržava istu brzinu kao i krajnji host,
- ako se paketi uniformno šalju slobodnim putevima, teoretski svi uređaji mogu da rade istom brzinom,
- dobra skalabilnost jer može da opsluži $(n)^3/4$ servera.

Mane ove topologije su:

- kompatibilna unazad sa IP/Ethernet-om, što može biti i mana i prednost, jer će algoritmi za rutiranje izabrati najkraći put što brzo može dovesti do zagušenja,
- kompleksno je povezivanje zbog nedostatka portova visoke brzine,
- ne koristi sve slobodne puteve podjednako,
- problem sa prekomernim kabliranjem zbog velikog broja svičeva.

Ono što je takođe značajno po pitanju *fat tree* topologije je da je glavni cilj rutiranja da raspodeli saobraćaj ravnomerno između *core* svičeva. Navešćemo tri predloga za rešavanje navedenog problema:

- Al-Fares je predložio tabelu sa dva nivoa za rutiranje po pitanju adresiranja da bi rešio ovaj problem. Za razliku od uobičajene tabele za rutiranje u kojoj jedan zapis povezuje nekoliko IP adresa sa jednim portom na sviču, u dvonivovskoj tabeli svaki zapis može imati pokazivač ka maloj sekundarnoj tabeli. Zapisi u sekundarnoj tabeli su u formi (*suffix, port*), dok zapisi u primarnoj tabeli imaju formu (*prefix, port*). Ne moraju svi primarni zapisi da imaju pokazivač ka sekundarnom zapisu, mada ka sekundarnom zapisu može pokazivati više primarnih zapisa. Pri rutiranju paketa, krajnja IP adresa će prvo biti proverena da odgovara najdužem prefiksu u primarnoj tabeli. Nakon toga, IP adresa se ponovo proverava da odgovara najdužem sufiksu i krajnji port će biti određen. Po pitanju odgovora na kvar Al-Fares predlaže *failure broadcast* protokol za detektovanje kvarova. Ovaj protokol rešava samo proste kvarove linka. Za *core* svičeve, tabele rutiranja su jednostavnije. Kako jedan *core* svič ima samo jedan link ka agregacionom sviču u svakoj ćeliji, a svaki agregacioni svič ima samo jedan link ka svakom pristupnom sviču u ćeliji, kada paket stigne u *core* svič, podsetnik njegove putanje je jedinstven. Rasprostranjenost (difuzija) saobraćaja se dešava samo na prvoj polovini puta, pre nego što on stigne do *core* sviča i to je prema Al-Faresu rešenje ovog problema [19][20].
- PortLand koristi 48-bitnu hijerarhijsku Pseudo MAC (PMAC) šemu za adresiranje. Kod *fat tree* topologija svi serveri unutar iste ćelije imaju isti prefiks u PMAC adresama. Jedinstven PMAC se dodeljuje svakoj fizičkoj ili virtuelnoj mašini, šifrujući njihovu lokaciju. Treba i dalje da bude podržan ARP (*eng. Address Resolution Protocol*). PortLand koristi LDP (*eng. Location Discovery Protocol*) da otkrije lokaciju svičeva i da poboljša prosleđivanje i rutiranje paketa. U PortLand svičevi će prosleđivati pakete tek pošto se ustanovi lokacija. LDM (*eng. Location Discovery Message*) se koristi za ispitivanje pada linka. Ako ćelija ne primi LDM određeni vremenski period, razumeće da je link pao i poslaće tu informaciju menadžeru linkova. On koristi matricu greške za čuvanje statusa svakog linka u celoj mreži. Kada primi LDM obaveštenje, menadžer linkova ažurira matricu greške (*eng. fault matrix*) i obaveštava sve pogođene ćelije o padu ovog linka. Ćelije koriste novu informaciju da ažuriraju svoje tabele usmeravanja. Pri oporavku procedura je slična, šaljući poruku oporavka umesto poruke greške [19][20].
- Hedera se oslanja na PortLand implementaciju, ali ima više uslova pri rutiranju paketa od PortLanda i veću osetljivost na grešku. Koristi centralizovano rešenje radi upravljanja. Svičevi analiziraju tokove i šalju podatke o njima centralnom uređaju koji na osnovu tih podataka proračunava optimalne putanje za tokove i šalje te informacije svičevima koji na osnovu njih vrše prosleđivanje saobraćaja. Takođe po pitanju pada linka koristi isti princip kao i PortLand [19].

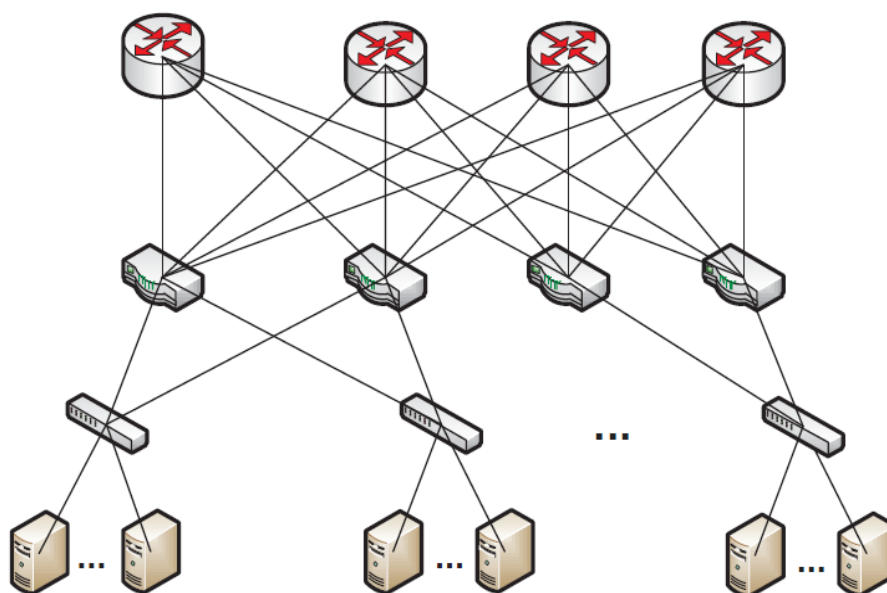


Slika 4.2.4: Fat tree topologija [19]

4.2.4. VL2

VL2 se zasniva na *folded-Clos* mrežnoj topologiji. Ova topologija stvara iluziju bilo kom skupu servera u data centru da su priključeni u fizički odvojene ethernet svičeve sa neophodnim brojem portova. Napravljena je da bi ispunila tri cilja:

- *Uniform High Capacity*
Znači da je maksimalna brzina *server-to-server* saobraćaja ograničena slobodnim kapacitetom mrežnih kartica servera. Potrebno je takođe da se serverima mogu zadavati usluge bez razmatranja koja je topologija u pitanju.
- *Performance Isolation*
Saobraćaj jednog servisa ne treba da bude pogođen saobraćajem drugog servisa, treba da se ponašaju kao da je svaki servis povezan na različit, fizički odvojen svič.
- *Layer 2 semantics*
Svaki server bi trebalo da se ponaša, bez obzira u kojoj mreži radi, kao da je u LAN-u. VM bi trebalo da slobodno migriraju sa istom adresom na bilo koji server.



Slika 4.2.5: Primer VL2 topologije [20]

Ideja pri dizajniranju ove topologije je da se koriste jeftini svičevi, uređene u *Clos* topologiju što obezbeđuje raznolikost puteva između servera. Ovaj dizajn zamenjuje skupe svičeve, koji se koriste u osnovnim topologijama stabla, širokim slojem jeftinih svičeva. Ovi svičevi se mogu dodavati na mrežu, da bi se postigao veći kapacitet i robusnost. VL2 topologija ne zahteva nikakve izmene u samim aplikacijama. Nude manji propusni opseg od optičkih ili hibridnih topologija, ali pružaju jednostavnost i visoke performanse. Ubacuje 2.5 *shim* sloj u grupu serverskih protokola tj. koristi MPLS (nezavisan i skalabilan protokol prenosa u kome se paketima dodeljuju labele i na osnovu njih se prosleđuju paketi, bez potrebe gledanja u sadržaj paketa), čime se dodaju nove mogućnosti nedostupne u mrežnim adapterima/drajverima. Nema potrebe za novim softverom ili svičevima. Adresiranje je bazirano na IP i koristi iste ethernet i IP protokole. Svičevi koji sačinjavaju mrežu, ponašaju se kao ruteri trećeg sloja sa tabelama za rutiranje, izračunatim od strane OSPF. Ovim se omogućava korišćenje više puteva za prenos paketa. Ipak, IP adrese koje koriste servisi u data centrima, ne treba da budu vezane za neki poseban svič ili će agilnost servera između servisa biti izgubljena. VL2 dodeljuje adrese serverima, samo po imenima bez topološkog značaja. Kada server pošalje paket, *shim* sloj na serveru poziva sistem direktorijum da sazna stvarnu lokaciju odredišta i tamo tuneluje originalni paket. *Shim* sloj pomaže da se reši problem po pitanju skalabilnosti, koje unosi ARP u mreži sa dva nivoa. Tunelovanje pomaže pri implementaciji VLB (eng. *Valiant Load Balancing* - šema za rutiranje, koja je decentralizovana tj. u kojoj svaki čvor donosi lokalne odluke: šema je prilagodljiva, što je njena najveća prednost i podržava sve matrice saobraćaja, a takođe se koristi i za građenje mreže svičeva velike skalabilnosti i garantovanih performansi, bez potrebe za centralizovanim upravljanjem). Za razliku od data centara na bazi stabla pomenutih gore u ovoj topologiji postoji bogata povezanost između agregacionih i *core* svičeva. Veliki broj puteva između svaka dva agregaciona sviča daje joj prednost pri padu nekog od linkova. Jedino što zahteva je dodatni sloj između IP i MAC u protokolskom steku krajnjih uređaja. Koristi VLB za slanje saobraćaja kroz sve slobodne linkove bez potrebe za centralizovanom kordinacijom. Svaki server nezavisno bira put kroz mrežu za svaki tok koji šalje drugim serverima. Pomoću VL2 se postiže visok stepen izolacije performansi i veliki kapacitet. Ipak broj servera koje zahteva za svoj rad dovodi do većih troškova, teško se implementira u data centre sa desetina hiljada servera, svi svičevi i serveri rade konstantno, pa

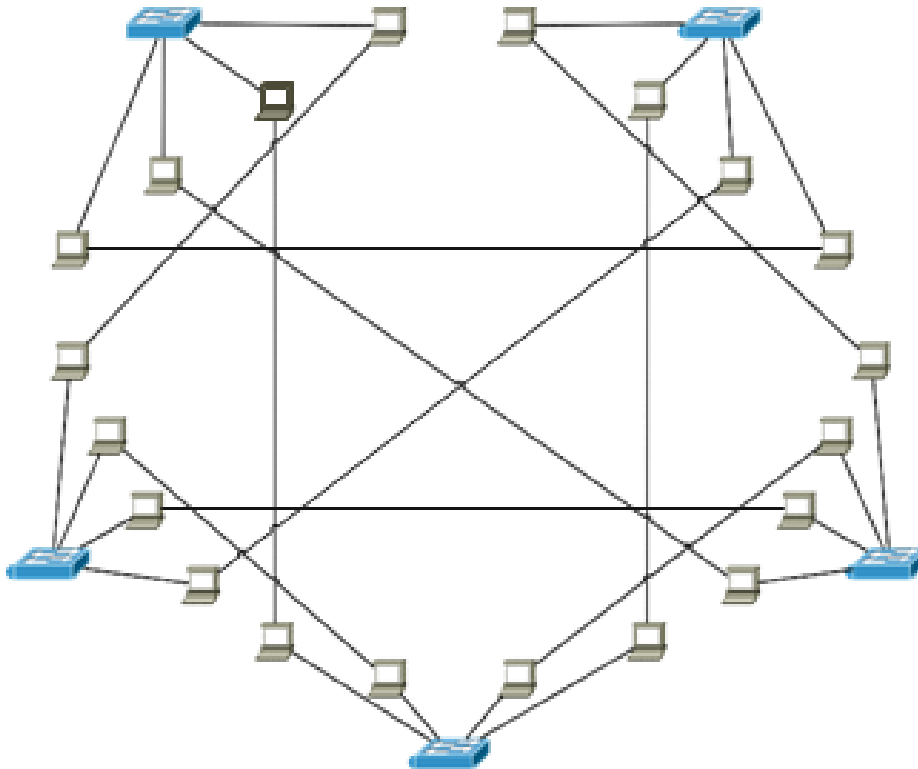
nije energetski efikasna i nema procenu performansi u realnom vremenu što predstavlja mane ove topologije [19][20].

4.3. Fiksne rekurzivne topologije

Ove topologije i dalje imaju slojeve kao i u topologijama na bazi stabla. Razlika je u tome što rekurzivne topologije koriste strukture na nižem sloju kao ćelije, da bi izgradili strukture višeg sloja. Serveri u ovim topologijama mogu biti povezani na svičeve i servere sa različitih slojeva topologije. Imaju više mrežnih portova na serverima, čime se dosta razlikuju od *tree based* topologija. Nazivaju se još i modularni data centri, jer oni ne koriste samo standardne rekeve za skladištenje servera, već prenosne kontejnere u kojima se smešta nekoliko hiljada servera na rekovima koji se nalaze unutar ovih kontejnera. Modularni data centri su veoma mobilni data centri. Pojednostavljen je pristup energiji, tako što se direktno povezuju na izvor energije bez posrednika. Manje je vreme postavljanja ovih data centara i smanjeni su troškovi proizvodnje i infrastrukture potrebne za hlađenja [19].

4.3.1. Dcell

Topologija data centara, DCell ima četiri komponente: DCell skalabilnu mrežnu infrastrukturu, efikasan algoritam rutiranja za tu strukturu, robusno rutiranje koje se bavi raznim tipovima kvarova i šemu za proširivanje koja omogućava postepenu ekspanziju ove topologije. Osnovni element DCell arhitekture je $DCell_0$ i sastoji se od n servera i jednog sviča sa n portova. Svaki server u $DCell_0$ je povezan sa svičom u toj ćeliji i svi DCell na istom sloju su povezani međusobno. Ovo je rekurzivna mreža velikog kapaciteta, napravljena od mini svičeva koji povezuju servere. Serveri su povezani na DCell različitih nivoa preko više linkova. DCell na višem nivou se gradi rekurzivno formiranjem potpuno povezanog grafa sa mnogo DCell manjeg sloja. Saobraćaj se raspoređuje ravnomerno između servera i linkova na serveru. Nema jedinstvene tačke pada kod DCell i ona je otporna na različite tipove kvarova na linku, serveru ili reku. Robusna je zahvaljujući bogatoj fizičkoj povezanosti i primenom *fault-tolerant routing* protokola. Linkovi na višem nivou neće predstavljati mesto zagušenja, kao što se dešavalo kod topologija na bazi stabla i obezbeđuju dva puta veću propusnu moć od ovih topologija. Ova topologija se takođe može postepeno proširivati i predstavlja održivu strukturu za povezivanje u data centrima. DCell sa stepenom 4 može da podrži nekoliko miliona servera bez upotrebe skupih svičeva jezgra ili rutera [19][20][21].



Slika 4.3.1: DCell₁ od DCell₀ [19]

i) *Konstruisanje DCell*

Neka je DCell_k k-nivovska DCell. Prvi korak je konstruisati DCell₁ od nekoliko DCell₀. Svaka DCell₁ ima n+1 DCell₀ i svaki server svake DCell₀ u DCell₁ je povezan sa serverom u drugom DCell₀. Rezultat je da su DCell₀ povezane jedna sa drugom, sa tačno jednim linkom između svake DCell₀. Slična procedura se koristi kada se konstruiše DCell_k od DCell_{k-1}. U DCell_k svaki server će na kraju imati k+1 linkova, prvi link je povezan sa svičom kada se pravi DCell₀, a link na k nivou će biti povezan sa serverom u istoj DCell_k, ali različitom DCell_{k-1}. Broj nivoa DCell je ograničen brojem mrežnih portova na serverima koji se nalaze u njoj [19][21].

ii) *Rutiranje u DCell*

Po pitanju rutiranja data centri, koji koriste DCell, ne mogu da koriste globalnu šemu za rutiranje, jer je njihov cilj da povežu milione servera. Takođe, ne mogu da koriste ni hijerarhijsku OSPF, jer se stvara usko grlo. Da bi se izbegli padovi na hardveru, softveru kao i energetske problemi koji su česti u data centrima predloženo je korišćenje specijalno dizajniranog protokola za data centre na bazi DCell, DCell *Fault-tolerant Routing protokol* (DFR) i *DCellRouting*. *DCellRouting* je osnovni protokol rutiranja u DCell i koristi podeli i vladaj pristup, pri čemu ovo nije rutiranje najkraćom putanjom ali jeste rutiranje jednom putanjom (*eng. single-path*). Ako su izvor i odredište u istoj DCell_k a različitim DCell_(k-1) koristi se sledeći princip. Algoritam će prvo naći link koji povezuje ove dve DCell_(k-1) i rutiranje će biti odvojeno preko dva podputa. Ovaj proces se ponavlja sve dok svi podputevi ne postanu direktni linkovi. DFR omogućava robusno rutiranje u DCell. Definiše tri tipa otkaza: servera, reka i linka. Koristi tri tehnike za njihovo rešavanje: lokalno rerutiranje, lokalno stanje veze i *jump up* [19][20].

- lokalno rerutiranje predstavlja pristup pri padu linka koji koristi specifično interfejs prosleđivanje i po padu linka potiskuje *link state advertisement* i izaziva lokalno rerutiranje koristeći *backwarding* tabele. Sa ovim pristupom kada je potisnuto ne više od jednog obaveštenja pada linka, paketu je garantovano prosleđivanje slobodnim putem ako on postoji.
- lokalno stanje mreže je tehnika koja koristi *local-link* rutiranje za rutiranje u DCell. U DCell svaki čvor koristi *DCellBroadcast* da emituje status svih njegovih linkova periodično ili kada se ustanovi kvar. Kada čvor primi paket on koristi *DCellRouting* da izračuna putanju ka čvoru u istoj ćeliji. Po dobijanju prvog linka kojim može to da uradi, koristi lokalno stanje mreže da vidi kako da dođe do drugog čvora. Kada identifikuje da je link pao, on traži proksi (pomoćnika) pomoću lokalnog rerutiranja kome će poslati paket, a koji će ga posle proslediti na određište preko drugog linka.
- *jump up* je tehnika koju je najbolje opisati na sledeći način. Ako cela jedna DCell padne tj. otkazu joj svi linkovi, onda bi se paket beskonačno rerutirao oko nje. Tada čvor koji šalje paket bira proksi, DCell ćeliju višeg nivoa, i na taj način zaobilazi onu koja je otkazala.

iii) *DCellBroadcast*

DCellBroadcast je robusna brodkasting šema za DCell. U *DCellBroadcast* kada server inicijalizuje brodkast, poslaće paket svim ostalim $k+1$ komšijama. Kada prima brodkast paket, server proverava da li je već primio ovakav paket. Ako nije, šalje taj paket svim svojim komšijama. Ovakav način slanja duplira brodkast pakete, ali je robusan jer može da isporučuje pakete svakom serveru dok god je mreža povezana. Serveri se ne ponašaju samo kao krajnji hostovi, već kao i relejni čvorovi jedan prema drugom.

Prednosti ove topologije:

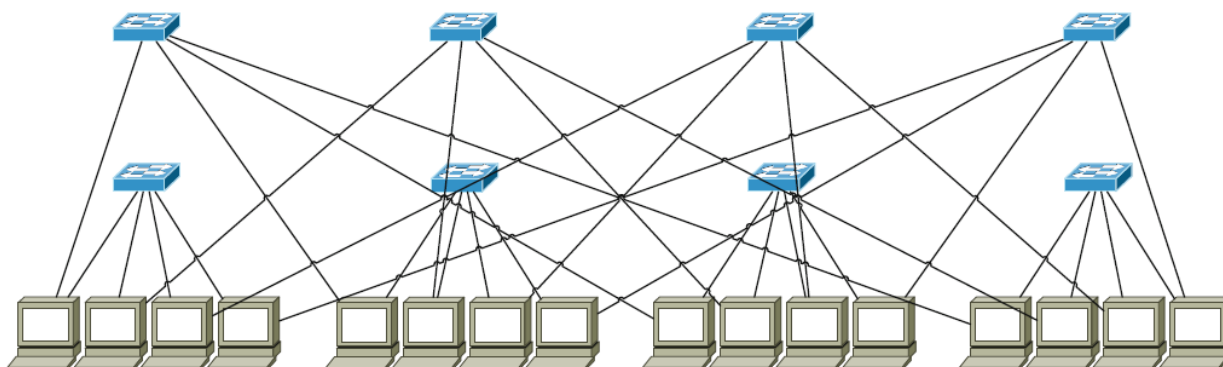
- velika širina bisekcije (širina bisekcije predstavlja minimalni broj linkova koje je potrebno odstraniti iz topologije da bi se mreža podelila na dve međusobno nepovezane mreže jednake veličine),
- mala je verovatnoća da će doći do zagušenja,
- robusnost,
- ne zahteva skupe mrežne komponente,
- podržava skalabilan i efikasan algoritam za rutiranje,
- visok kapacitet mreže,
- skalabilna zbog svoje arhitekture koja može vrlo lako da se proširuje.

Mane ove topologije ostaju:

- *cross section bandwidth* nije jednak (protok između dve podmreže jednake veličine pri čemu ove dve podmreže zajedno čine čitavu originalnu mrežu),
- kašnjenje u mreži,
- opterećenje nije ravnomerno izbalansirano između linkova u *all-to-all* komunikaciji.

4.3.2. BCube

BCube je specijalno dizajnirana topologija da bi se opremili modularni data centri smešteni u prenosne kontejnere visine do 40 stopa. Dok je vreme razmeštanja znatno smanjeno kod ovog tipa data centra, mana ovog okruženja je da je teško vršiti servis i zamenu delova kada se jednom postavi. Najosnovniji element BCube je $BCube_0$. Kao i kod $DCell_0$, n servera je povezano sa jednim svičem sa n portova. Glavna razlika između ove dve topologije je način proširivanja. BCube koristi više svičeva kada pravi više nivoa, dok DCell koristi svičeve sa n portova nultog nivoa. DCell ima veći kapacitet mreže nego BCube i kako DCell podržava *server-to-server* konekciju, tako BCube podržava *server-to-switches*. Srž BCube arhitekture predstavlja njena mrežna struktura orijentisana na server, gde se serveri sa višestrukim portovima povezuju sa višestrukim nivoima COTS (eng. *Commercial Off The Shelf*) mini svičevima. Kao i kod DCell, serveri se ne ponašaju samo kao krajnji hostovi, već kao i relejni čvorovi jedan prema drugom. Podržava širok opseg aplikacija tako što ubrzava *one-to-one*, *one-to-several* i *one-to-all* tipove saobraćaja i pruža visok kapacitet mreže za *all-to-all* saobraćaj. Nema velike degradacije performansi što je veoma bitno, jer kada se postavi kontejner teško je zamenjivati komponente data centra. Može se integrisati sa TCP/IP protokolima. BCube je robusna topologija i ravnomerno raspoređuje opterećenje na serverima i linkovima. Može da opslužuje nekoliko hiljada servera tj. data centre srednje veličine [19][22].



Slika 4.3.2: $BCube_1$ od $BCube_0$ [19]

i) Konstruisanje BCube

Postoje dve vrste uređaja u BCube: serveri sa više portova i svičevi koji povezuju konstantan broj servera. Ova topologija je rekurzivno definisana struktura. Pri konstrukciji $BCube_1$, n dodatnih svičeva se koristi i povezuju sa samo jednim serverom u svakoj $BCube_0$. Zato se $BCube_1$ sastoji od n $BCube_0$ i $2n$ svičeva, računajući i one u $BCube_0$. $BCube_k$ je konstruisana iz $n * BCube_{(k-1)}$ sa n^k dodatnih svičeva sa n portova. Ovi dodatni svičevi su povezani sa tačno jednim serverom u svakoj $BCube_{(k-1)}$. BCube sa k nivoa, na nivou i sadrži n^i svičeva, pri čemu je $i=1,2,..k$. Kao i kod DCell broj nivoa zavisi od broja portova na serverima [22].

ii) Rutiranje u BCube

- *Single-Path* rutiranje u BCube

Struktura BCube omogućava jednostavan način da se implementira *single-path* rutiranje u BCube. Ovaj tip protokola razvrstava i određuje odredišta BCube adresa, određuje identifikaciju paketa i dužinu prenosa paketa. Svaki paket uključuje BCube zaglavlje (gde se čuva indeks hopa svakog BCube paketa) između ethernet i IP zaglavlja. Serveri povezani sa istim svičem razlikuju se samo u jednoj cifri u njihovoj adresi. Kada se prosleđuju paketi od izvora ka odredištu, srednji čvorovi će

odrediti cifre u njihovoj adresi po kojima se razlikuju i menjati po jednu cifru u svakom koraku. Ovo znači da se paketi prosleđuju sa jednog servera na drugi preko sviča koji ih povezuje. Paket može da putuje preko najviše $k+1$ svičeva i k servera do svog odredišta [22].

- *Source* rutiranje u BCube

BCube Source Routing (BSR) je *source* protokol rutiranja koji je dizajniran specijalno za BCube arhitekturu. Ovaj protokol troši bez potrebe veći deo mrežnog kapaciteta BCube. U ovom protokolu mrežni saobraćaj je podeljen na tokove [22].

Prednosti ove tehnologije su što:

- podržava sve tipove saobraćaja,
- ima veću redundantnost zbog više portova na njihovim serverima,
- mala je verovatnoća zagušenja.

Mane ove topologije su:

- teža zamena komponentata,
- koristi veliki broj svičeva što dovodi do povećanja cene.

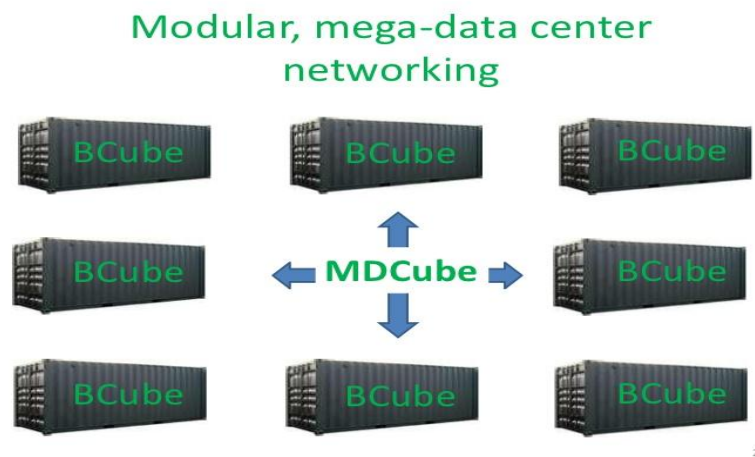
4.3.3. *MDCube*

BCube je predložen da bi se izgradili modularni data centri velikih kapaciteta, korišćenjem servera sa više portova i mini svičeva. Ali glavna prepreka kod BCube je da podrži milione servera dodavanjem portova na servere i dodavanjem novih mini svičeva u topologiju. MDCube je rekurzivna topologija izgrađena na osnovu BCube topologije, tako što su iskorišćeni optički kablovi za povezivanje više BCube kontejnera korišćenjem portova svičeva velike brzine (10Gb/s). Na osnovu toga postoje dva tipa linkova u MDCube: jedan je link koji povezuje servere i svičeve, a drugi je *inter-container* link velike brzine između svičeva, koji se nalaze u različitim kontejnerima. Portovi velikih brzina u BCube formiraju potpuni graf MDCube. Da bi bila skalabilna sa povećanim brojem kontejnera, topologija grafa se može proširiti na kocku uvođenjem dimenzija u MDCube. Ona obezbeđuje veliki kapacitet mreže i održivo kabliranje. MDCube je serverski dizajn topologije tako da se logika ostavlja serverima. Problem je nedovoljna iskorišćenost propusnog opsega, u slučaju kada se susretne sa *bursty* saobraćajem, jer bira najkraći put na nivou kontejnera. Mala je verovatnoća da dođe do zagušenja. Rutiranje ka spoljašnjim mrežama je obezbeđeno rezervnim svičevima i serverima u BCube, koji se koriste za spoljašnju konekciju. U odnosu na BCube i fat tree broj kablova za unutrašnju konekciju se drastično smanjio, zbog korišćenja virtuelnih portova svičeva (*eng. switch virtual interface* (SVI) - port na sviču koji se koristi za prosleđivanje saobraćaja između različitih VLAN bez potrebe za korišćenjem rutera. Kao i kod BCube ili Helios topologija, za MDCube je potrebno kraće vreme postavljanja i manji troškovi hlađenja. Ona zahteva velike protoke unutar kontejnera. Cena opreme za povezivanje komponentata, pre svega imajući u vidu optičke kablove za povezivanje kontejnera, takođe predstavlja problem [20][23][24].

i) Rutiranje u MDCube

- *MDCubeRouting* je dizajniran da ispravi zapise o ID-u kontejnera jedan po jedan da bi stigao do odredišnog kontejnera i redosled korekcija je kontrolisan permutacijom. Rutiranje unutar kontejnera je određeno BCube protokolima [24].

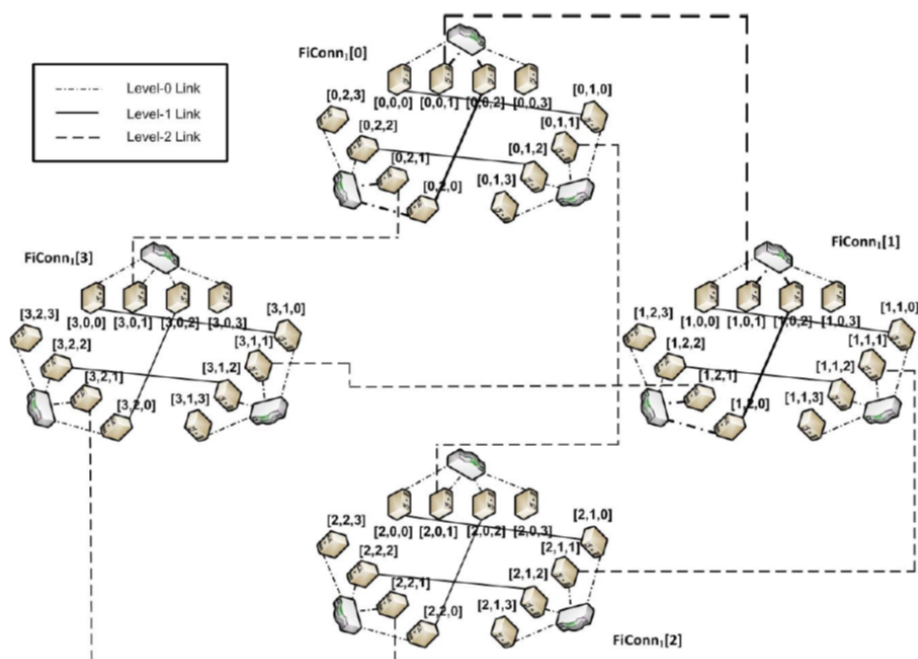
- *MDCubeDetourRouting* balansira opterećenje između kontejnera, tako što inicira rutiranje prema slučajnom skoku na nivou kontejnera, a onda prati *MDCubeRouting* od ovog slučajno izabranog čvora do odredišta da bi se izbeglo nepotrebno preklapanje u tom kontejneru [24].
- Robusno rutiranje u MDCube je dizajnirano oslanjajući se na strukturu MDCube: izvorni čvor balansira opterećenje na mogućim putevima na nivou kontejnera, opterećenje se javlja zbog slabo kontrolisanog *source routing* protokola; rutiranje unutar kontejnera vrše i dalje BCube protokoli. Za pad linka ili sviča unutar kontejnera generisaće se poruka o grešci putanje i poslati izvornom čvoru da pokrene preusmeravanje na nivou kontejnera birajući druge paralelne puteve [24].



Slika 4.3.3: MDCube [23]

4.3.4. FiConn

FiConn je skalabilno rešenje topologije koja radi sa serverima koji sadrže samo dva porta i jeftinim svičevima. On definiše rekursivnu strukturu mreže u više nivoa. FiConn višeg nivoa je konstruisan od FiConn nižeg nivoa. Kada se konstruiše FiConn višeg nivoa, FiConn nižeg nivoa koriste polovinu svojih raspoloživih bekap portova za povezivanje i formiraju mrežu. Broj servera u FiConn raste kao dupla eksponencijalna funkcija (matematička definicija, $f(x) = a^{b^x}$, gde su a i b promenljive), kako rastu nivou FiConn strukture (ako se koriste svičevi sa 48 portova, FiConn drugog nivoa će moći da podrži 361200 servera tj. broj servera u FiConn_k , N_k , je veći ili jednak od $2^{k+2} * (n^2/16)^k$, gde je n broj servera u FiConn_0 i $n > 4$) [25]. Iako se koriste bekap portovi svakog servera, pouzdanost servera nije kompromitovana. On nam nudi veoma skalabilnu strukturu sa *off-the-shelf* serverima sa dva mrežna porta i jeftinim svičevima. U FiConn je mala verovatnoća zagušenja i ima dobru iskorišćenost kapaciteta linkova. Proces povezivanja mrežnih uređaja u ovoj topologiji nije složen, ipak ovo ne prolazi bez dodatnih troškova, kabliranje je skuplje u ovoj topologiji nego onim na bazi stabla. Serveri koriste više CPU resursa pri prosleđivanju paketa, kao i kod svih rekursivnih topologija navedenih gore. Napravljen je da podrži velike data centre [19][20][25].



Slika 4.3.3: FiConn topologija [25]

i) *Konstruisanje FiConn*

Kada se konstruiše FiConn višeg nivoa, FiConn strukture nižeg nivoa koriste pola svojih slobodnih bekap portova za povezivanje i stvaranje mreže preko $FiConn_{(k-1)}$ modula. $FiConn_0$ je osnovna jedinica za konstrukciju ove topologije, koja se sastoji od n servera i sviča sa n portova koji povezuje servere. Svaki server u FiConn ima jedan port povezan sa svičom u $FiConn_0$ i ovaj port zovemo port nultog nivoa. Ovaj port se može posmatrati kao originalan operacioni port na serverima. Ako bekap port servera nije povezan sa drugim serverom, zovemo ga slobodan bekap port. Ako postoji ukupno b servera sa slobodnim bekap portovima u $FiConn_{(k-1)}$, broj $FiConn_{(k-1)}$ u $FiConn_k$ će biti $(b/2 + 1)$. U svakoj $FiConn_{(k-1)}$, $b/2$ servera od ukupnog broja servera b sa slobodnim bekap portovima je izabrano da poveže preostalih $b/2$ servera $FiConn_{(k-1)}$ koristeći bekap portove, svaki za jednu $FiConn_{(k-1)}$. Konstruišemo FiConn topologiju k -tog nivoa, koristeći bekap portove $FiConn_{(k-1)}$ servera [19][25].

Postoje tri glavne razlike između FiConn i *fat tree* topologije. Prva je da se za razliku od *fat tree* topologije ovde povezivanje ostavlja serverima, a ne svičevima. Kod *fat tree* postoje tri sloja svičeva a kod FiConn samo jedan najniži i broj iskorišćenih svičeva je mnogo manji u FiConn (ako je N broj servera, a n broj portova koji se koriste, tada je broj iskorišćenih svičeva kod *fat tree* $(5*N)/n$, a kod FiConn N/n). Nema ograničenja u broju servera kao kod *fat tree*. Mali broj svičeva i linkova, doprinosi nižoj ceni i lakšem povezivanju. Ravnomerno koristi različite nivoe linkova i na osnovu saobraćaja pokušava da bolje iskoristi kapacitet linkova. FiConn deli iste principe dizajna kao i DCell, da bi serverima prepustio povezivanje. Razlika se ogleda u nekoliko aspekata. Broj mrežnih portova čvora servera je u DCell $k+1$, a u FiConn uvek 2. FiConn koristi postojeći bekap port na svakom serveru za povezivanje i nema potrebe za dodatnim troškovima na hardveru servera. Mnogo manje košta povezivanje u FiConn, jer svaki server ima samo dva porta. Rutiranje koristi ravnomerno linkove na različitim slojevima, što DCell ne može. Za razliku od BCube, izgrađen je za ogromne data centre od više desetina hiljada servera. BCube ima veći broj portova na serveru, ali FiConn ima prednost u lakšem povezivanju uređaja [19][20][25].

4.4. Fleksibilna topologija

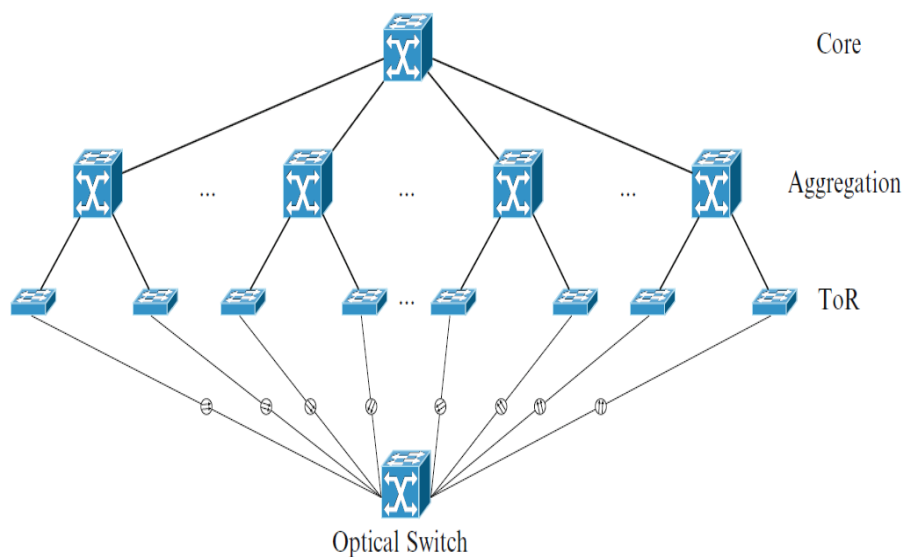
Ove topologije su bazirane na optičkim svičevima za izgradnju mreža data centara. Nude veliki protok i veliku fleksibilnost, jer topologija može biti rekonfigurisana tokom rada. Ova osobina je važna, ako se uzme u obzir neuravnoteženi i promenljivi obrasci saobraćaja u data centrima. Delimo ih na hibridne i potpuno optičke topologije.

4.4.1. Hibridna topologija

Ove topologije povećavaju ethernet sloj za prosleđivanje paketa, optičkim kolima i svičevima obezbeđuju veliki opseg mreže za aplikacije koje to zahtevaju. Primeri ovih topologija su c-Through i Helios, koje se mogu konstruisati trenutno dostupnim uređajima da bi podržale široke zahteve aplikacija i uslova saobraćaja [19].

4.4.2. c-Through topologija

c-Through je hibridna tehnologija, naziva se još i HyPaC (*eng. Hybrid Packet and Circuit*) mreža i sastoji se iz dva dela: *tree based* električne mreže, koja povezuje svaki par ToR svičeva i optičke mreže koja pruža veliki opseg za povezivanje parova rekova kada je to potrebno. Pošto je optička oprema skupa i propusni opseg njihovih linkova veliki, nije isplativo da se uspostavi optički link između svakog para rekova. Umesto toga c-Through povezuje svaki rek sa tačno jednim rekom u potrebnom trenutku. Optički linkovi velikog kapaciteta povezuju rekove na prolaznoj bazi, zavisno od zahteva saobraćaja. Odgovornost za procenu zahteva saobraćaja i njegovog demultipleksiranja u c-Through vrše krajnji hostovi. U cilju postizanja najboljeg iskorišćenja optičkog kola, c-Through koristi servere da baferuju saobraćaj, kao i da prikupe dovoljne količine saobraćaja radi brzog prenosa. Baferovanje saobraćaja se vrši povećanjem granica soket bafera, ali da ne dođe do blokiranja ili dodatnog kašnjenja. Procenu saobraćaja rekova i rekonfiguracija mreže se kod ove topologije postiže korišćenjem sistema kontrole ravni. Da bi se konfigurisao optički deo c-Through mreže, saobraćaj između rekova mora da zna prioritete. c-Through procenjuje zahteve za *rack-to-rack* saobraćaj, tako što posmatra okupiranost nivoa soket bafera. Kako rek može da bude optički povezan samo sa jednim rekom istovremeno, veza treba da se uspostavi između rekova, koji razmenjuju najveću količinu saobraćaja i to se radi preko servera, jer svaki server u c-Through ima servis za upravljanje podacima (*eng. data plane*) kojim se raspodeljuje saobraćaj u električnoj i optičkoj mreži. Ovaj dizajn mreže koristi krajnje hostove da nadgledaju saobraćaj i koristi parcijalan pristup da podeli optičku i električnu mrežu. c-Through i Helios pokušavaju da što veću količinu saobraćaja pošalju optičkim putem. Ima centralizovanu kontrolu koja dinamički konfigurise optičku mrežu i svičeve. Da bi rešio problem dodele saobraćaja mrežama, c-Through koristi demultipleksiranje [19][26].



Slika 4.4.1: c-Through mreža [19]

i) Demultipleksiranje saobraćaja u c-Through

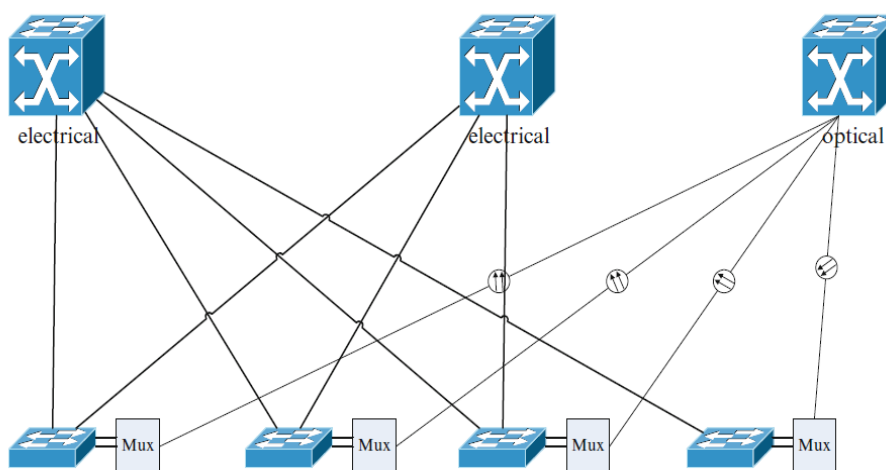
Hibridne mreže nailaze na problem kako da dodele saobraćaj na dva različita tipa mreže, električnu i optičku. Kod c-Through ovaj problem je rešen korišćenjem VLAN-ova. ToR svičevi se dodeljuju dvema različitim VLAN koje su logički odvojene. Ovo razdvaja optičku od električne mreže. Pošto se optički VLAN može menjati prema promenljivim zahtevima saobraćaja, *spanning tree* protokol se ne uzima u obzir zbog dugog vremena koje je potrebno za njegovo prilagođavanje. Servis za upravljanje podacima se nalazi na svakom serveru i on služi da bi dodelio saobraćaj optičkoj i električnoj mreži i odredio prema prioritetu prenosa između kojih linkova će se uspostaviti optička veza. Put kojim kreće paket je određen njegovim odredištem. U praksi, da bi se efikasnije iskoristila optička mreža, optički povezana odredišta će imati veći prioritet u prenosu od električnih odredišta.

Mane ove mreže su svakako nedovoljna iskorišćenost optičkih linkova i problem dodele saobraćaja različitim tipovima mreže čije je rešenje gore navedeno ali ne rešava u potpunosti problem, cena koja je neophodna za njihov rad kao i vreme potrebno za prebacivanje optičkog linka na drugi rek, koje može dostići i do 80 ms. Sa druge strane pružaju veliki protok i fleksibilnost zbog topologije koja se može rekonfigurisati [19].

4.4.3. Helios

Helios je još jedan tip hibridne mreže koja se sastoji od električnih i optičkih svičeva velikog protoka (mogu biti i MEMS (*eng. MicroElectroMechanical Systems* - mali uređaji koji mogu da se koriste za pravljenje optičkih svičeva i koji reflektuju svetlosne zrake na željeni izlazni port) optički svičevi). Helios je *multi-routed tree* topologija sa dva sloja sastavljena od *core* i *pod* (ToR) svičeva. *Core* svičevi mogu biti ili optički ili električni da bi se iskoristile prednosti obe tehnike. Helios je sličan c-Through topologiji, ali koristi WDM (*eng. Wavelength Division Multiplexing* - multipleksiranje po talasnim dužinama). Na svakom sviču ćelije, aplinkovi su opremljeni optičkim predajnikom. Polovina aplinkova je povezana sa električnim svičevima, dok je druga polovina povezana sa optičkim svičevima preko optičkog multipleksera. Za razliku od c-Through, Helios koristi električnu mrežu za prosleđivanje paketa *bursty* saobraćaja. Helios koristi algoritam Hedere da ustanovi zahteve saobraćaja između rekova. Cilj ovakve mreže je da zadovolji što veću količinu saobraćaja. U Heliosu se najveći problem definiše kao *maximum-weight matching* (optimizovani problem u Heliosu kako da se

povećaju zahtevi saobraćaja, koje topologija može da ispuni) i rešava se koristeći *Edmond's Algorithm* kao što radi i c-Through. Ideja je da se nađe podgraf koji ima maksimalan zahtev saobraćaja, pri čemu je svaki čvor podgrafa povezan sa maksimalno jednim susednim čvorom i oduzima se od početnog grafa mreže kako bi se dobio rezultujući graf, tako da on definiše koji su čvorovi povezani. Ima iste mane i prednosti kao i prethodna hibridna topologija. Skalabilna je mreža, ali sa velikim troškovima koji je prate. Glavna razlika između Heliosa i c-Through je što Helios koristi svičeve za procenu i demultipleksiranje saobraćaja, a ne na hostovima kao c-Through. Ovakav pristup Helios topologije, pravi kontrolu saobraćaja koja je transparentna krajnjim hostovima, ali zahteva modifikovanje svičeva. Prednost c-Through je kada baferuje podatke u hostovima, može da gomila saobraćaj i da napuni efektivno optički link kada on bude bio dostupan [19][27].



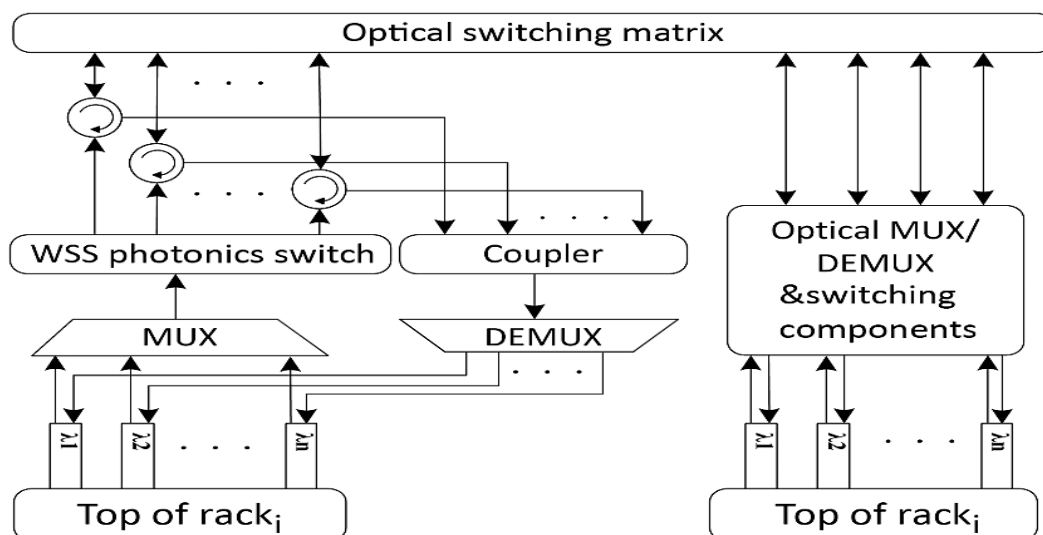
Slika 4.4.2: Helios mreža [19]

4.4.4. OSA (eng. *Optical Switching Architecture*)

Potpuno optička mreža, što znači da ne koristi električne *core* svičeve, već samo optičke svičeve da bi napravila optičko jezgro. OSA postiže fleksibilnost koristeći mogućnost konfigurisanja MEMS svičeva. Topologija se pravi tako što se svaki od N ToR svičeva poveže sa jednim portom na MEMS sviču sa N portova. Uzevši u obzir bipartitno podudaranje portova kod MEMS (eng. *bipartite port-matching* - svaki ulazni port je direktno povezan sa najviše jednim izlaznim portom, na osnovu para rekova koji treba da se povežu, pri čemu MEMS svičevi imaju $N \times N$ portova (N ulaznih koji se povezuju na N izlaznih)), ToR svič može komunicirati samo sa jednim ToR svičem u datom trenutku. Ako povežemo N/k ToR svičeva, gde je k broj portova ToR sviča, na k portova MEMS svičeva, svaki ToR će moći da komunicira sa k ToR svičeva istovremeno. Konfiguracija MEMS određuje koji ToR svičevi će biti međusobno povezani i OSA mora da obezbedi povezanost grafa ToR svičeva kada konfigurise MEMS [19][28].

Graf ToR svičeva je povezan optičkim kolima kroz MEMS, koristi se *hop-by-hop stitching* optičkog kola da se postigne široka povezanost mreže. Da bi došao do udaljenih ToR svičeva na koje nije direktno povezan, ToR svič koristi jednu od svojih k konekcija. Prvim hopom, ToR svič prima prenos preko vlakna, prebacuje ga u električni signal, čita zaglavlje paketa i šalje paket ka odredištu. Na svakom hopu, svaki paket se pretvara iz optičkog u elektronski i onda ponovo u optički (O-E-O) i menja se na ToR sviču. Čista O-E-O konverzija može se uraditi u delovima nanosekundi. Na svakom portu mora se naglasiti da bilo koji dolazni ili odlazni saobraćaj ne može biti veći od kapaciteta porta. Zato konekcije velikog protoka koriste što manji broj hopova. OSA topologija ovo omogućava. U suštini ToR svičevi obezbeđuju portove između električnih veza servera i optičkih signala koji dolaze

sa *switching core*. OSA omogućava, kako je gore navedeno, višestruke veze svakog ToR sviča sa *switching core*. Povezivanje se vrši na osnovu saobraćajnih zahteva. Pošto mreža ne obezbeđuje direktan optički link između svakog para rekovala, kontrolni sistem pravi topologiju u kojoj je ToR svič zadužen za prenos saobraćaja između druga dva ToR sviča. OSA procenjuje zahteve saobraćaja koristeći isti metod kao i Helios. Razlika između ove dve topologije je u tome što kod OSA mogu postojati višestruki linkovi ka svakom reku i problem nije više *maximum-weight matching*. OSA ima problem *multi-commodity* protoka. Kod OSA topologije je dozvoljeno da u konačnom rešenju čvorovi imaju stepen veći od jedan (dok kod Heliosa je maksimalan stepen čvorova 1) tj. maksimalan broj čvorova može biti b , sa maksimalno b susednih čvorova može biti povezan čvor u rezultujućem podgrafu koji je maksimalne težine, ali tako da on ima maksimalan zahtev saobraćaja. Zato se kod OSA u cilju jednostavnijeg rešavanja primenjuje više puta *maximum-weight matching* algoritam. Kada se nađe jedan podgraf, on se oduzme od početnog grafa, dobije se neki rezultujući graf. Na rezultujući graf se primenjuje isti algoritam i ovaj proces se ponavlja sve do trenutka, kada više ne može da se nađe novi podgraf da bi se dodao onim već ranije pronađenim [28].



Slika 4.4.1: Celokupan prikaz OSA topologije [29]

Za razliku od fiksnih topologija, OSA nije složena za povezivanje i može dodeliti dodatni propusni opseg na zahtev saobraćaja. U odnosu na hibridne tehnologije OSA može da ostvari slične performanse kao i one po manjoj ceni i potrošnji energije.

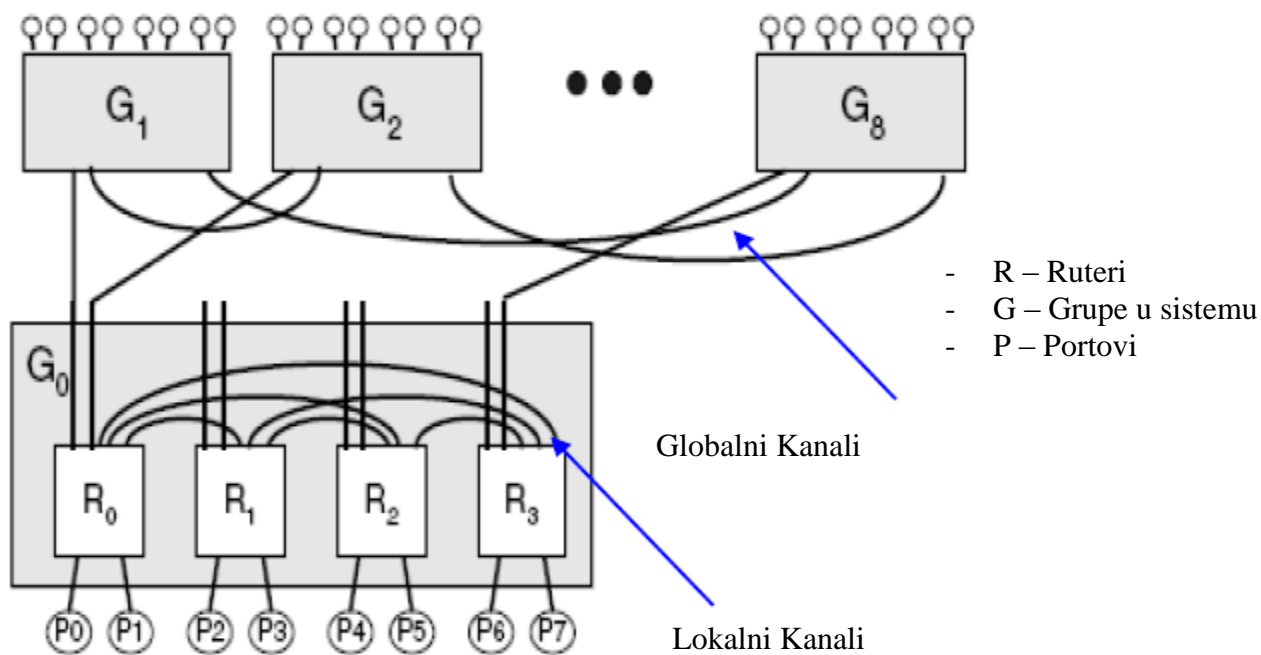
Poenta u potpunosti optičke OSA topologije je da se ona oslanja na mogućnost konfigurisanja optičkih uređaja u jedan dinamički skup *one-hop* optičkih kola. Tada ona koristi *hop-by-hop stitching* više optičkih linkova da obezbedi *all-to-all* povezanost za male protoke i *bursty* saobraćaj, a i da bi se izborila sa opterećenjima, sa kojima postojeće električne i optičke arhitekture ne mogu da se efikasno izbore preko svojih optičkih interkonekcija. OSA dinamički podešava kapacitete svojih optičkih linkova, da bi zadovoljila promenljive zahteve saobraćaja. Takođe da bi efikasno iskoristila skupe optičke portove, OSA koristi i *Circulator* (bidirekciona komponenta za simultan prenos u oba smera u odnosu na isto kolo, što potencijalno povećava duplo iskorišćenost MEMS portova) [19][20][28].

4.5. Ostale topologije

Topologije koje još treba da se pomenu vezane za data centre su Dragonfly i Scafida.

4.5.1. Dragonfly

Napredak tehnologije i povećanje propusnog opsega po pinu, dovodi do korišćenja rutera sa velikim brojem portova da bi se smanjilo kašnjenje, dijametar i troškovi povezivanja mreže. Ipak mreže ovog tipa zahtevaju duže kablove za povezivanje. Pošto kabliranje zauzima veliki deo u troškovima data centra, broj kablova treba smanjiti da bi se napravila isplativa mreža. *Dragonfly* topologija koristi skup rutera sa velikim brojem portova, kao virtuelni ruter, da bi se postigla efikasnost mreže [28]. Da bi se definisala topologija *Dragonfly* sledeće stavke su definisane:



Slika 4.5.1: Osnovna DragonFly topologija [31]

- N - broj servera,
- p - broj terminala povezanih povezanih na svaki ruter,
- a - broj rutera u svakoj grupi,
- k - stepen rutera,
- k_l - efektivan stepen grupe ili virtuelnog rutera,
- h - broj kanala u svakom ruteru kojima se povezuju sa ostalim grupama,
- g - broj grupa u sistemu.

Na donjem nivou svaki ruter je povezan sa p servera, $a-1$ lokalnih kanala je povezano sa ruterima iz iste grupe i h globalnih kanala je povezano na rutere u drugoj grupi. Stepenn svakog rutera je $k = p + a + h - 1$. Grupa se sastoji od a rutera koji su povezani preko mreže, napravljene od lokalnih kanala

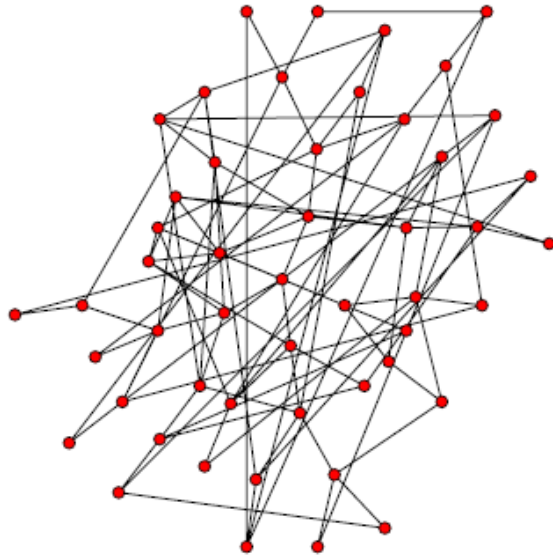
unutar grupe. Topologija ove lokalne mreže može biti bilo koji tip topologije. Svaka grupa ima $a \cdot p$ konekcija ka serverima i $a \cdot h$ konekcija ka globalnim kanalima. Svi ruteri u grupi rade kao virtuelni ruter sa stepenom $k_l = a \cdot (p + h)$. Ovakav stepen omogućava da se mreža ostvari sa malim dijametrom, što povlači da neće dolaziti do zagušenja. Pri punoj veličini ($N = a \cdot p \cdot (a \cdot h + 1)$) topologije Dragonfly, postoji samo jedna veza između svakog para grupa. Parametri a , p , h mogu imati bilo koju vrednost. Povećanjem efektivnog stepena Dragonfly topologija postaje veoma skalabilna [30].

Cilj ove topologije je da smanji dužinu linkova i mrežni dijametar. Dragonfly zahteva podršku za adaptivno rutiranje, što nije podržano ni od jednog postojećeg hardvera. Zahteva i višestruke virtuelne kanale za izbegavanje dedloka (*eng. deadlock*). Dragonfly smanjuje troškove povezivanja za 52% u odnosu na *Clos* topologije [30][31].

4.5.2. Scafida

Za razliku od ostalih navedenih arhitektura, koje su simetrične, Scafida je nesimetrična mrežna arhitektura inspirisana *scale-free* mrežom.

Može se deliti na manje celine i manje homogene delove. Imaju malu verovatnoću zagušenja i veliku robusnost, jer su izgrađene na principu *scale-free* mreže. U ovoj topologiji se veštački ograničava broj linkova koje čvor može da ima tj. maksimalan stepen čvora, da bi se ispoštovao broj portova koje mrežni ruteri i svičevi mogu da imaju. Iako se ograničava stepen čvorova u mogućnosti je da održi prednosti *scale-free* mreže. Kratke razdaljine i robusnost ostaju nepromenjene. Pored toga ograničenje stepena čvora ne utiče mnogo na propusni opseg topologije. Kako serveri u data centrima komuniciraju jedan sa drugim, prosečna dužina linkova koji ih povezuju značajno utiče na performanse data centra. Serveri mogu da utiču na proces rutiranja ako je broj portova servera veći od jedan. Bez obzira na veličinu mreže, prosečne dužine linkova rastu umereno uprkos ograničavanju. Još jedna bitna karakteristika ove topologije je robusnost. Neke aplikacije zahtevaju neprekidnu komunikaciju između servera data centra, što može dovesti do zagušenja i pada performansi topologije. Zato je ovaj tip topologije veoma kompatibilan za aplikacije. Kapacitet za sve linkove je jednak i serveri i svičevi imaju dovoljne protoke tako da ne ograničavaju performanse topologije. Iako su stepeni čvorova veštački ograničeni, ova topologija ima dobre osobine za rad data centra. Scafida je veoma skalabilna i fleksibilna. Može podržati 50000 servera. Kao dodatak, veličina data centra može se podesiti preko *fine-grained* skale. Ova arhitektura se može napraviti koristeći samo par parametara, kao što su broj servera, dijametar, dužina putanje kretanja paketa, tako da broj servera u strukturi može biti podešen samo na *course-grained* skali. Veoma je fleksibilna po pitanju broja i tipa mrežne opreme. Može da koristi različite tipove svičeva. Ideja ove topologije da neki svičevi završe kao gusto povezana hab mesta [32].



Slika 4.5.2: Scafida topologija – maksimalnog stepena 5 zasnovana na scale free mreži [32]

5. PROTOKOLI

Protokoli koji se koriste u data centrima su:

- DNS (*eng. Domain Name System*) protokol, u najširem smislu, zadužen je za prevođenje domenskih imena, koja prirodno pogoduju čoveku zbog toga što su mu bliska i zgodna za pamćenje u IP adresu, sastavljenu iz niza brojeva, koja prirodno pogoduje računaru. Ovaj protokol je ključan za funkcionisanje velikog broja Internet servisa (npr. servis elektronske pošte na domenu) dodeljujući alfanumerički naziv nekoj IP adresi pomoću kojeg je omogućeno drugim korisnicima da lociraju dotičnu mašinu koja je jedinstvena na globalnom nivou. DNS je definisan sa RFC (*eng. Request for Comments*) dokumentima 1034 i 1035, objavljenim od strane *Internet Engineering Task Force* [33][34].
- HTTP (*eng. Hypertext Transfer Protocol*) je mrežni protokol na aplikacionom sloju, čija je osnovna svrha razmena podataka. HTTP je osnova data komunikacije za WWW (*eng. World Wide Web*). Hipertekst (*eng. Hypertext*) je strukturni tekst koji koristi logičke linkove (hiperlinkove) između čvorova koji sadrže tekst. HTTP je protokol za razmenu i transfer hiperteksta. HTTP je zamenjen verzijom HTTP/1.1 (Jun 2014) koja je redefinisana serijom protokola RFC 7230, RFC 7231, RFC 7232, RFC 7233, RFC 7234, i RFC 7235 [33].
- HTTPS (*eng. HTTP Secure*) protokol bazira se na istim principima kao i HTTP, samo je uložena dodatna napor kako bi se postigla bezbednost prenetih podataka, primenom nekog od sigurnosnih protokola na aplikacionom sloju (*eng. Transport Layer Security* (TLS) ili *eng. Secure Sockets Layer* (SSL)). Osnovni zadatak HTTPS protokola je autentifikacija veb strane kojoj pristupamo, kao i obezbeđivanje privatnosti i integriteta prenesenih podataka [33].
- SMTP (*eng. Simple Mail Transfer Protocol*) predstavlja osnovni Internet standard za razmenu elektronske pošte na aplikacionom sloju. Funkcioniše po klijent-server modelu, u kojem se definišu zadaci i komunikacija između servera kao snabdevača i klijenta kao korisnika usluga. Iako serveri e-pošte i softveri za transfer elektronskih poruka između računara, kao što su MTA (*eng. Mail Transfer Agent*), koriste SMTP za slanje i primanje elektronskih poruka, na korisničkom nivou se SMTP jedino koristi za slanje poruka serverima za e-poštu. Protokol koji je danas u širokoj upotrebi je zasnovan na RFC 5321 preporuci [33].
- FTP (*eng. File Transfer Protocol*) je standardan mrežni protokol aplikacionog sloja koji se koristi za razmenu podataka preko TCP (*eng. Transmission Control Protocol*). Zasnovan je na principu komunikacije klijent-server i koristi odvojene kanale za kontrolu i podatke između klijenta i servera. Korisnici FTP mogu potvrditi svoj identitet korišćenjem lozinke i korisničkog imena (*eng. clear-text sign-in protocol*) ili se mogu konektovati anonimno ako to server dozvoljava. Kako bi osigurali prenos koji enkriptuje podatke i zaštićuje lozinku i korisničko ime, FTP često koristi kriptografski protokol SSL/TLS (FTPS) koji omogućava sigurnost komunikacije preko kompjuterske mreže. Zasnovan je na preporuci RFC 114 [35].

- Protokol za rutiranje je vid komunikacije između rutera, kojim se utvrđuju pravila za odabir putanje paketa kroz mrežu. Zaseban ruter poseduje informacije samo o onim ruterima sa kojima je direktno povezan, i pomoću ovih protokola, odnosno međusobne komunikacije, svaki ruter spoznaje topologiju čitave mreže, što je veoma bitno za proces usmeravanja, odnosno formiranje tabele usmeravanja u ruteru. Po oblasti rutiranja, protokoli za rutiranje mogu se svrstati u grupu internih i grupu eksternih protokola, a po drugom načinu klasifikacije definišu se protokoli zasnovani na stanju linka (OSPF i IS-IS) i protokoli zasnovani na vektoru rastojanja (*Routing Information Protocol*, RIPv2, IGRP) [33].
- OSPF (*eng. Open Shortest Path First*) je protokol za rutiranje IP mreža. On spada u grupu internih protokola koji koriste *link state routing* algoritam i operiše unutar jednog autonomnog sistema. Ruteri razmenjuju međusobno informacije o topologiji mreže i na osnovu toga rekonstruišu kompletnu mrežnu topologiju na osnovu koje formiraju i popunjavaju tabele usmeravanja. Tabela usmeravanja se pretražuje na osnovu određene IP adrese paketa i kao rezultat pretrage vraća izlazni port na koji treba proslediti paket. OSPF ne koristi TCP/IP transportni protokol nego enkapsulira podatke u IP datagrame što je u suprotnosti sa drugim protokolima za rutiranje, na primer RIP. Definisani su kao verzija 2 u IPv4 u RFC 2328 i kao verzija 3 u IPv6 u RFC 5340 [33][36].
- IS-IS (*eng. Intermediate System to Intermediate System*) je interni protokol za rutiranje zasnovan na stanju linka. Veoma je sličan OSPF-u. Jedna od osnovnih razlika u odnosu na OSPF je u tom što je OSPF razvijen specifično za IPv4, dok je IS-IS generalniji protokol i može da se primeni i za druge tipove adresa. IS-IS se koristi i u IPv4 i u IPv6 mrežama. IETF je predložio ovaj protokol u RFC dokumentu 1142 [37].
- RIP (*eng. Routing Information Protocol*) je protokol za rutiranje zasnovan na vektoru rastojanja koji koristi brojač skokova (hopova) za merenje dužine puta. On sprečava pojavu petlji ograničavajući broj skokova od izvora ka destinaciji (postoje situacije kada se privremene petlje javljaju i razvijene su brojne tehnike za rešavanje ovog problema – trovanje ruta, deljenje horizonta). Maksimalan broj skokova je u RIP ograničen na 15. RIP koristi UDP (*eng. User Datagram Protocol*) transportni protokol. RIP verzija 1 je bila razvijena za klasno adresiranje, a kao što je poznato zbog lošeg iskorišćenja IP adresa sa klasnog se prešlo na besklasno adresiranje. RIP verzija 2 se koristi u slučaju besklasnog adresiranja. RIPv2 je definisan u preporuci RFC 2453 [33].
- IGRP (*eng. Interior Gateway Routing Protocol*) je interni protokol za rutiranje zasnovan na vektoru rastojanja koji je razvio Cisco [38].
- UC (*eng. Unified Communications*) skup proizvoda koji pruža jedinstven korisnički interfejs u cilju što jednostavnije i efikasnije upotrebe integrisanih servisa koji se izvršavaju u realnom vremenu kao što su trenutno razmenjivanje poruka i slično, sa servisima koji ne rade u realnom vremenu, npr e-mail [39].
- VoIP (*eng. Voice over Internet Protocol*), popularno Internet telefonija, predstavlja princip prenosa telefonskog govora u realnom vremenu preko Interneta. Slično fiksnim telefonskim mrežama (*eng. Public Switch Telephone Network, PSTN*), u VoIP se telefonski poziv uspostavlja, analogni signal se digitalizuje i kodira. Razlika je u prenosu digitalne informacije, koji je u slučaju VoIP-a paketski, gde se informacija prenosi pomoću IP u malim nezavisnim paketima [40].
- RTP (*eng. Real-time Transport Protocol*) je mrežni protokol na aplikacionom sloju, namenjen prenosu audio i video sadržaja preko IP mreže poput Interneta. Radi u

kolaboraciji sa RTCP (*eng. RTP Control Protocol*), pri čemu je RTP zadužen za prenos multimedijalnih tokova (*eng. stream*), dok je RTCP zadužen za sinhronizaciju i kontrolu kvaliteta tokova. IETF ga je predložio i definisan je u RFC 1889 [41].

- ICMP (*eng. Internet Control Message Protocol*) je protokol Internet sloja koji služi za slanje kontrolnih poruka mrežnim uređajima (poput rutera). Kontrolne poruke mogu se odnositi na obaveštenja o dostupnosti određene usluge, zagušenosti mreže, kao i poruke koje sadrže različite upite (*eng. query messages*). ICMP se razlikuje od transportnih protokola kao što su TCP i UDP u tome što se u suštini ne koristi za razmenu podataka između sistema. Definisan je u dokumentu RFC 792 [42].
- *Open Flow Protocol* je komunikacioni protokol koji preko mreže pristupa tabeli usmeravanja (*eng. lookup table*) mrežnog uređaja. Ovaj protokol omogućava kontrolerima da odrede put mrežnih paketa kroz mrežne svičeve. Najmanje dva kontrolera su preporučena: primarni i sekundarni kao podrška. Ova podela kontrole prosleđivanja omogućava bolje upravljanje saobraćajem nego pri korišćenju ACL (*eng. Access Control Lists*) i protokola rutiranja. Omogućava i korišćenje različitih tipova svičeva, kojima se upravlja pomoću jednog otvorenog protokola. Smatra se da *OpenFlow* omogućava softverski definisano umrežavanje. *OpenFlow* omogućava daljinsko upravljanje tabelama za prosleđivanje na trećem sloju tako što dodaje, modifikuje i uklanja njihove zapise i pravila pretraživanja tih tabela. Na ovaj način odluke rutiranja mogu se doneti periodično ili ad hoc od strane kontrolera i prevesti na pravila i postupke sa promenljivim životnim vekom. Paketi za koje je pretraga neuspešna se prosleđuju do kontrolera. Kontroler onda modifikuje postojeća pravila tabele usmeravanja i tokova na jednom ili više svičeva ili dostavlja nova pravila, da bi sprečio prevelik protok saobraćaja između kontrolera i sviča. Može čak i da odluči da prosledi saobraćaj sam, pod uslovom da je rekao svičevima da proslede cele pakete a ne samo zaglavlja paketa. *OpenFlow* protokol se nalazi na vrhu TCP i propisuje upotrebu TLS (*eng. Transport Layer Security*). Kontroleri bi trebalo da slušaju na TCP portu 6653 zahteve ako neki svič želi da uspostavi konekciju. Ranije verzije *OpenFlow* protokola su koristile port 6633. Trenutno se koristi verzija 1.4 *OpenFlow* protokola, ali postoje i verzije 1.1, 1.2, 1.3 [43].
- ARP (*eng. Address Resolution Protocol*) je protokol koji se koristi za prevođenje adresa sa mrežnog sloja (kao što je IP adresa) u adrese na sloju linka (kao što je MAC adresa), veoma je zastupljen i kompatibilan je za rad sa velikim brojem tehnologija koje se sprovode na mrežnom sloju i sloju linka (najčešći slučaj funkcionisanja je IPv4 u kombinaciji sa IEEE 802.3 ili IEEE 802.11). Protokol funkcioniše po principu prosleđivanja poruka, u kojima je sadržan ili zahtev da se adresa prevede, ili informacija da je adresa prevedena. U okviru protokola implementirana je i tehnika obrnutog procesa, odnosno prevođenja adresa sa sloja linka na adrese mrežnog sloja, koja se zove *Reverse ARP* ili *Inverse ARP*. Postoje četiri tipa poruka, dve kojima se šalje zahtev za prevođenje (*ARP request* i *RARP request*) i dve kojima se šalje odgovor na zahtev (*ARP reply* i *RARP reply*). Veličina poruke zavisi od veličine adrese koja se prevodi, što diktira sloj sa kojeg je adresa. Tip mreže, kao i veličina i tip poruke, definisani su u zaglavlju, a u korisnom delu poruke (*eng. payload*) nalaze se četiri adrese od značaja - IP i MAC adrese primaoca i pošaljioaca [19].
- *Proxy based ARP* je tehnika koja se bazira na principima ARP protokola, sa razlikom što je u stanju da prevodi adrese koje dolaze izvan mreže. Po pristizanju podataka, ARP proksi saznaje lokaciju sa koje podaci stižu, proglašava svoju MAC adresu kao krajnju

destinaciju, a zatim pomoću posrednika (*eng. proxy*) prosleđuje podatke ka destinaciji za koju su predviđeni [19].

- HSRP (*eng. Hot Standby Router Protocol*) predstavlja mehanizam koji daje privid hostovima da koriste jedan te isti ruter kao gejtvej ka ostatku Interneta. Ovaj protokol omogućava da se izvrši automatsko prebacivanje na rezervni ruter u slučaju da primarni ruter otkáže, a da korisnici i dalje imaju privid da koriste jedan te isti ruter. Očigledno, ovaj protokol omogućava da ruteri koji mogu da igraju ulogu gejtvej rutera za neku LAN mrežu, u međusobnoj komunikaciji izaberu aktivni ruter koji će zaista i prosleđivati saobraćaj, kao i rezervni koji će preuzeti na sebe ulogu aktivnog rutera ukoliko dođe do otkaza aktivnog rutera. Termin virtuelni ruter označava grupu rutera koji međusobno komuniciraju preko HSRP protokola, jer ih korisnik u suštini vidi kao jedan isti ruter. Pomoću multikasta, HSRP šalje pozdravne poruke (*eng. hello messages*) ka svim HSRP slobodnim ruterima, između kojih se utvrdi prioritet i ruter sa najvećim prioritetom koristi se kao aktivni ruter sa predefinisanim gejtvej adresom. Kada ovaj ruter padne, po ustanovljenoj hijerarhiji prioriteta prelazi se na sledeći ruter najvišeg prioriteta, pri čemu protokol ni na koji način ne menja tabele rutiranja niti su korisnici svesni da je došlo do promene. U slučaju pada aktivnog rutera, funkciju aktivnog uzima *Standby* ruter. Definisan u RFC dokumentu 2281 [33][44].
- IGMP (*eng. Internet Group Management Protocol*) je protokol mrežnog sloja koji omogućava hostovima da se prijave na željene multikast sesije, a koje oglašavaju IGMP ruteri. Opisan je u RFC dokumentu 1112 [33].
- IP (*eng. Internet Protocol*) je protokol trećeg sloja OSI referentnog modela (sloja mreže). Sadrži informacije o adresiranju, čime se postiže da svaki mrežni uređaj (računar, server, radna stanica, interfejs rutera) koji je povezan na Internet ima jedinstvenu adresu i može se lako identifikovati u celoj Internet mreži, a isto tako sadrži kontrolne informacije koje omogućuju paketima da budu prosleđeni (rutirani) na svoja konačna odredišta. Ovaj protokol je dokumentovan u RFC 791 i predstavlja sa TCP protokolom jezgro Internet mreže, TCP/IP stek protokola (*eng. Transmission Control Protocol/Internet Protocol*). Protokol ima dve verzije: IPv4 i IPv6 [33].
- IPv4 (*eng. Internet Protocol version 4*) je najzastupljenija verzija Internet protokola i rutira najviše saobraćaja na Internetu. Radi po principu najboljeg pokušaja i ne garantuje dostavu. Opisan je u RFC 791 dokumentu. Naslednik ovog protokola je IPv6 [45].
- IPv6 (*eng. Internet Protocol version 6*) je najnovija verzija Internet protokola, koja se bavi rešavanjem problema nedostatka adresa. Pored ove prednosti, novom verzijom nudi se i niz pogodnosti poput hijerarhijskog adresiranja, koje ograničava prekomeran rast tabela usmeravanja, pojednostavljenja multikast adresiranja, sigurnosti i mobilnosti uređaja. Koristi 128-bitne adrese dok su u IPv4 adrese 32-bitne [46].
- IPSec (*eng. Internet Protocol Security*) je protokol koji obezbeđuje sigurnost komunikacije putem IP protokola. Bezbednost se zasniva na obostranoj autentifikaciji i korišćenju kriptografskih ključeva [47].
- STTP (*eng. Stateless Transport Tunneling Protocol*) protokol omogućava formiranje virtuelnih mreža preko fizičke infrastrukture što se radi kreiranjem tunela, a cilj je razdvajanje virtuelne mreže od fizičke infrastrukture sa stanovišta korisnika a sve u cilju veće fleksibilnosti, npr. lakše je seliti virtuelne mašine, moguće je definisati znatno veći broj virtuelnih mreža u odnosu na VLAN tehnologiju, adresiranje u virtuelnim mrežama

je odvojeno od onog u fizičkoj mreži (što je dobro za prvu stavku i seljenje virtuelnih mašina) [48].

- MPLS protokol (*eng. Multiprotocol Label Switching*) je mehanizam za usmeravanje podataka sa jednog mrežnog čvora na sledeći pomoću kratkih labela (*eng. labels*) umesto dugih mrežnih adresa. Kako se prosleđivanje vrši samo na osnovu sadržaja labele, izbegava se ispitivanje sadržaja samog paketa i pretraga u tabeli usmeravanja. Ruter koji obavlja usmeravanje na osnovu oznake je LSR (*eng. Label Switch Router*) ili tranzitni ruter. Kada LSR primi paket, na osnovu njegove labele određuje sledeći korak putanje i odgovarajuću novu labelu iz tabele usmeravanja. Stara labela se uklanja i menja novom pred prosleđivanje [49].
- VRRP (*eng. Virtual Router Redundancy Protocol*) je protokol koji dinamički dodeljuje odgovornost za virtuelne rutere, apstraktnu reprezentaciju više rutera npr. bekap rutera, nekim VRRP ruterima na LAN. VRRP ruter koji kontroliše adrese povezane sa virtuelnim ruterom se zove *master* i on prosleđuje pakete koji su poslani na ove IP adrese. Izborni proces omogućava dinamički bekap po pitanju odgovornosti prosleđivanja ako je *master* nedostupan. Ovo omogućava bilo kom virtuelnom ruteru na LAN da se koristi kao podrazumevani difolt gejtvej ruter krajnjih hostova. Prednost korišćenja ovog protokola je veća dostupnost, bez potrebe za promenom dinamičkog rutiranja ili *router discovery* protokola na krajnjim hostovima. VRRP pruža informacije o stanju rutera, ali ne daje informacije o obrađenim i razmenjenim putanjama od strane tog rutera. Svaki VRRP je ograničen na jednu podmrežu. Može se koristiti u eternetu, MPLS i *token ring* mrežama, IPv4 i IPv6. IETF ga je predložio u dokumentu RFC 5798 [50].
- UDP (*eng. User Datagram Protocol*) je jednostavan protokol transportnog sloja. UDP ne garantuje pouzdan prenos. Pomoću UDP, aplikacije mogu da šalju drugim hostovima na IP mreži bez prethodnog uspostavljanja putanje podataka. UDP je pogodan za situacije kada provera i korekcija grešaka nisu potrebni ili se obavljaju u samoj aplikaciji, jer ne daje garancije za isporučivanje poruka i kada ih pošalje nema informacije o njihovom statusu. Brojne Internet aplikacije koriste UDP, uključujući DNS (*eng. Domain Name System*), SNMP (*eng. Simple Network Management Protocol*), RIP (*eng. Routing Information Protocol*) i DHCP (*eng. Dynamic Host Configuration Protocol*). Audio i video snimci se generalno prenose pomoću UDP. Opisan je u RFC dokumentu 768 [51].
- UDT (*eng. UDP-based Transfer Protocol*) je protokol koji se može koristiti u BDP (*Bandwidth-Delay Product*) mrežama da bi se iskoristio bogat kapacitet mreže na efikasan i ravnopravan način. Očekuje se da se koristi za prenos podataka aplikacija preko veoma brzih mreža. Danas najčešće korišćena TCP verzija je *TCP NewReno/SACK* i ona ne radi dobro u ovakvim okruženjima zbog svog sporog oporavka i otkrivanja slobodnog propusnog opsega (*eng. bandwidth*) i ne može da isprati rast BDP mreže. UDT je protokol visokih performansi aplikacionog sloja i pre svega služi za visoke performanse za transfer velikih skupova podataka preko WAN mreža. Početne verzije ovog protokola su razvijene i testirane na veoma brzim mrežama (do 10 Gbit/s), ali kasnije verzije su prilagođene i standardnim Internet brzinama. UDT je razvijen na UDP protokolu, dodavanjem kontrole zagušenja i mehanizama kontrole pouzdanosti. Podržava pouzdan protok podataka. UDT koristi AIMD (*eng. Additive Increase Multiplicative Decrease*) kontrolni algoritam zagušenja. Sa strane razvoja, UDT je izvor ranjivosti jer se kao i mnogi drugi protokoli oslanja na postojeće sigurnosne mehanizme za TCP i UDP protokole. Opisan je u dokumentu 2026 sekcija 10 [52] [53].

- PPP (*eng. Point to Point Protocol*) je protokol koji se koristi za uspostavljanje direktne veze između dva čvora u serijskim kablovima, telefonskim linijama, mobilnim telefonima, fiber-optičkim vezama, specijalnim radio vezama, i dr. Ovaj protokol obezbeđuje autentičnost konekcije, enkripciju prenošenja (koristeći ECP) i kompresiju. PPP se često koristi kao protokol na sloju linka za povezivanje sinhronih i asinhronih kola. Sadrži tri komponente: (1) komponenta za prenos datagrama preko određenog fizičkog sloja, (2) LCP (*eng. Link Control Protocol*) za uspostavljanje, konfiguraciju i testiranje veze i (3) jedan ili više kontrolnih mrežnih protokola (NCP) za pregovaranje opcionalnih parametara konfiguracije. PPP omogućava da više protokola mrežnih slojeva rade na istoj komunikacionoj vezi. Za svaki sloj obezbeđuje se drugi NCP za pregovaranje. PPP pored autentifikacije i kompresije uključuje i mogućnosti detekcije grešaka i multilink. Definisani su u RFC 1661 [54].
- LLDP (*eng. Link Layer Discovery Protocol*) je protokol iz grupe *Internet Protocol Suite* koji mrežni uređaji koriste za oglašavanje svog identiteta, mogućnosti i suseda, uglavnom na Ethernetu. Ovaj protokol je definisan u dokumentu IEEE 802.1AB. LLDP informaciju šalju uređaji sa svakog porta u fiksnim intervalima, u formi ethernet okvira. Svaki okvir sadrži jednu LLDP jedinicu podataka (LLDPDU), koja predstavlja niz struktura tip-dužina-vrednost (*eng. type-length-value TLV*). Ethernet okviri u LLDP imaju svoju određenu MAC adresu, postavljenu na specijalnu multikast adresu koju 802.1D kompatibilni svičevi ne prosleđuju. Ostale multikast i unicast određene adrese su dozvoljene. Svaki LLDP okvir počinje sa obaveznim TLV: šasijom ID, portom ID i vremenom života (*time-to-live*). Svaki obavezni TLV prati bilo koji broj opcionalnih TLV. Okvir se završava specijalnim TLV, koji se naziva kraj LLDPDU. LLDP se koristi kao komponenta mrežnog upravljanja i aplikacija za monitoring. Informacije sakupljene ovim protokolom se skladište u bazi podataka koja čuva menadžment informacije (MIB). Odatle se mogu saznati naziv i opis sistema i porta, VLAN ime, IP adresa, mogućnosti sistema (komutacija, rutiranje i dr.), MAC/PHY informacije, MDI snaga i agregacija veza. Često se koristi poboljšana verzija LLDP, LLDP-MED (*eng. Media Endpoint Discovery*) koji pruža mogućnosti autootkrivanja LAN pravila, formiranje baza lokacije, automatsko upravljanje snagom krajnjih tačaka i praćenje i utvrđivanje karakteristika mrežnih uređaja [55].
- TCP (*eng. Transmission Control Protocol*) je jedan od osnovnih protokola unutar IP grupe protokola. TCP je protokol transportnog sloja. Korišćenjem protokola TCP, aplikacija na nekom od hostova umreženog u računsku mrežu kreira virtuelnu konekciju prema drugom hostu, pa putem te ostvarene konekcije zatim prenosi podatke. Stoga ovaj protokol spada u grupu tzv. konekciono orijentisanih protokola, za razliku od nekonekcionih npr. UDP. TCP garantuje pouzdan prenos podataka u kontrolisanom poretku od pošiljaoca prema odredištu slanja. Osim toga, TCP pruža i mogućnost višestrukih istovremenih konekcija prema jednoj aplikaciji na jednom hostu od strane više klijenata, gde su najčešći primeri za to veb. TCP podržava neke od najčešće korišćenih aplikacionih protokola na Internetu, kao što su HTTP (protokol za pregled veb stranica), SMTP (protokol za razmenu elektronske pošte), telnet, SSH (protokole za udaljeni rad na računaru), itd [56].
- STP (*eng. Spanning Tree Protocol*) je mrežni protokol koji obezbeđuje topologiju bez petlji za bilo koju povezanu ethernet LAN mrežu. Osnovna funkcija STP je da spreči pojavu petlje. Pojava petlje se javlja kada postoje dva ili više čvorova u mreži koji međusobno komuniciraju preko više od jednog puta. Problem koji tada nastaje je

beskonačno kružno slanje paketa podataka, koje se naziva *broadcast* oluja. Ona nepotrebno troši propusni opseg mreže čime se značajno smanjuju performanse mreže. Ovo je posebno značajno za ethernet LAN mreže u kojima na ovaj način može doći do zagušenja. STP dozvoljava obezbeđivanje dodatnih linkova, kako bi se obezbedili rezervni putevi ako neki aktivni link padne. Takođe, nema potrebe za ručnom aktivacijom rezervnog linka i ne postoji opasnost od stvaranja petlje.

Ovaj protokol je standardizovan kao IEEE 802.1D, ali je kasnije uključen u 802.1Q-2014 zajedno sa SPB (*eng. Shortest Path Bridging*) koji obezbeđuje jednostavnije kreiranje i konfigurisanje mreže, dok omogućava rutiranje po više putanja.

STP protokolom se u mreži povezanih uređaja drugog sloja (najčešće ethernet svičeva) stvara *spanning tree* i linkovi koji nisu deo *spanning tree* se ne koriste za prenos korisničkih okvira, ostavljajući samo jedan put između dva mrežna čvora. Protokol se zasniva na algoritmu koji je izmislila Radia Perlman.

Operacije protokola: LAN se može opisati kao graf, čiji čvorovi su mostovi i LAN segmenti, a linije grafa su linkovi. Da bi prekinuli petlju u LAN, dok održavaju pristup svim LAN segmentima, mostovi kolektivno računaju *spanning tree* koristeći STP. Koren most *spanning tree* je most (svič) sa najmanjim ID. Svaki most ima prioritet broj koji se može podešavati i MAC adresu. ID mosta se sastoji od kombinacije ova dva. Kada se porede prioriteti dva mosta, prvo se porede prioritet brojevi, a tek onda MAC adrese u slučaju da su brojevi jednaki. Most sa najmanjim ID će postati koren stabla. STP predviđa mogućnost da koren most ima više od jednog porta na istom LAN segmentu i u tom slučaju port sa najmanjim ID će postati port tog segmenta i stavlja se u režim prosleđivanja. Ostali portovi istog sviča u tom segmentu, postaviće se u blokirajuće stanje. Izračunati *spanning tree* ima svojstvo da poruke sa bilo kog uređaja koji je povezan na koren most se kreću najjeftinijim putem (što je veći propusni opseg linka, jeftiniji je put) od puteva koji povezuju koren most i taj uređaj. Ovakav pristup je garantovan pomoću dva pravila:

- ♦ najjeftiniji put svakog mosta: kada se odredi koren most, svaki preostali most određuje cenu svih puteva koji ga povezuju sa koren mostom. Bira najjeftiniji put i port na mostu koji ga povezuje sa tim putem, postaje koren port (*eng. root port RP*).
- ♦ najjeftiniji put svakog segmenta mreže: mostovi na segmentu mreže određuju koji most ima najjeftiniji put između segmenta mreže i korena. Port koji ovakav most povezuje sa segmentom mreže postaje nadređeni port (*eng. designated port DP*).

Svaki aktivni port koji nije RP ili DP je blokiran port (BP). Mostovi u STP moraju da odrede koren most i uloge portova (RP, DP, BP).

Da bi se osiguralo da svaki most ima dovoljno informacija, mostovi koriste okvire podataka BPDU (*eng. Bridge Protocol Data Unit*) da bi menjali informacije o ID mostova i cenama puteva do korena. Most šalje BPDU okvir koristeći jedinstvenu MAC adresu svog porta kao izvorišnu adresu i odredišnu adresu STP multikast adrese. Postoje dva tipa BPDU: konfiguracijski BPDU, CPBDU (koristi se za računanje *spanning tree*) i TCN (*eng. Topology Change Notification*) BPDU (objavljuje promene u topologiji mreže). BPDU se šalju na svake 2 sekunde i omogućavaju svičevima da prate promene u mreži i da prestanu ili počnu prosleđivanje na portovima prema potrebi. Kada se uređaj prvi put poveže sa portom sviča neće odmah početi sa prosleđivanjem podataka. Proći će kroz brojna stanja dok obradi BPDU i odredi topologiju mreže. Port može biti u stanju:

- ♦ blokiranja - port koji bi izazvao pojavu petlje da je u aktivnom stanju. U blokirajućem stanju port ne prima niti šalje podatke, ali može preći u stanje prosleđivanja ako padnu drugi linkovi koji se koriste i algoritam *spanning tree* odluči da taj port može preći u stanje prosleđivanja. BPDU podaci se i dalje primaju u blokirajućem stanju.
- ♦ slušanja - svič obrađuje BPDU i čeka moguće nove informacije koje bi port vratile u blokirajuće stanje. U ovom stanju se ne popunjava tabela MAC adresa i ne prosleđuju okviri.
- ♦ učenja - još uvek se ne prosleđuju okviri, ali se uče adrese izvora iz primljenog okvira i dodaju se u filtrirajuću bazu podataka. U ovom stanju popunjava se tabela MAC adresa.
- ♦ prosleđivanja - port normalno radi. Prosleđuje i prima podatke. STP i dalje prati dolazeće BPDU ako treba port da vrati blokirajuće stanje da bi sprečio petlju.
- ♦ isključeno - ne mora biti deo STP, administrator može ručno isključiti port.

Kada se host (npr. server) poveže na port sviča, port će se prebaciti u stanje prosleđivanja ali sa zakašnjenjem od oko 30 sekundi dok prođe kroz stanja slušanja i učenja. Ali ako se poveže drugi svič umesto hosta, port može ostati u blokirajućem stanju ako se ustanovi da će to dovesti do petlje u mreži. TCN BPDU se koriste da obaveste druge svičeve o promenama na portu. Svič koji nije koren može da generiše TCN obaveštenje koje propagira do korena. Po prijemu TCN, koren svič će postaviti zastavicu promene topologije (*eng. topology change flag*) u svoj normalni BPDU. Zastavica se propagira ka svim svičevima da zastare svoje ulaze u tabelama prosleđivanja. Da bi se smanjilo kašnjenje pri priključivanju hostova na svičeve uvedene su ekstenzije *spanning tree* protokola: *Rapid Spanning Tree Protocol (RSTP)*, *Per-VLAN Spanning Tree (PVST)*, *Per-VLAN Spanning Tree Plus (PVST+)*, *Multiple Spanning Tree Protocol (MSTP)* [57][58].

- *Rapid Spanning Tree Protocol (RSTP)* je ekstenzija STP protokola uvedena od strane IEEE 2001. godine kao 802.1w. RSTP obezbeđuje znatno bržu konvergenciju *spanning tree* posle promene topologije, uvođenjem novih ponašanja tokom konvergencije i novih uloga portova mostova. Protokol je dizajniran da bude kompatibilan unazad sa STP. STP je trebalo da odgovori na promenu topologije 30-50 sekundi, dok RSTP odgovara za $3 \cdot \text{Hello times}$ (po difoltu $3 \cdot 2$ sekunde) ili par milisekundi ako je došlo do pada fizičkog linka. *Hello time* je vremenski interval koji se koristi u RSTP i njegova difolt vrednost je 2 sekunde. Standard IEEE 802.1D-2004 uključuje RSTP i nepotreban originalni STP standard [57].

RSTP dodaje nove uloge portovima mosta da bi ubrzao konvergenciju pri padu linka. Broj stanja portova je smanjen sa 5 na 3 stanja. Stanje porta kod RSTP može biti:

- ♦ odbacivanja - korisnički podaci se ne šalju preko porta.
- ♦ učenja - port ne prosleđuje okvire još uvek, ali popunjava svoj tabelu MAC adresa.
- ♦ prosleđivanja - port je u potpunosti operativan.

Uloge porta mosta u RSTP:

- ♦ koren - najbolji port između mosta koji nije koren i koren mosta. Služi za prosleđivanje podataka.
- ♦ nadređen - port za prosleđivanje za svaki LAN segment.

- ♦ alternativni – ovaj port obezbeđuje alternativni put do koren mosta. Ovaj put se razlikuje od onog koji koristi koren port.
- ♦ bekap - obezbeđuje bekap put za segment koji je već povezan sa drugim portom mosta.
- ♦ isključeno - ne mora biti deo STP, administrator može ručno isključiti port [58].

Otkrivanje pada koren sviča se vrši u 3 *Hello times*. Portovi se mogu konfigurisati kao ivični portovi ako su priključeni na LAN koja nema druge mostove povezane na nju. Ivični portovi se direktno prebacuju u stanje prosleđivanja. RSTP prati na portovima prijem BPDU okvira u slučaju da je most povezan. Može automatski da detektuje ivične portove. Čim most detektuje dolazeći BPDU na ivični port, taj port prestaje da bude ivični. RSTP naziva vezu između dva ili više svičeva kao *link-type* vezu. Port koji radi u punom dupleks modu se pretpostavlja da je *point-to-point* link, dok se poludupleks portom smatra deljeni port (port koji ide na LAN segment, hab) po defaultu. Ovakvo automatsko podešavanje tipa linka može se premostiti eksplicitnom konfiguracijom. RSTP poboljšava konvergenciju na vezama point-to-point smanjujući *max-age* vreme na 3 *Hello* intervala, uklanjajući STP stanje slušanja i *handshake* dva sviča brzom tranzicijom porta u stanje prosleđivanja. RSTP se ponaša isto kao i STP za deljene portove. Cilj RSTP je da spreči pojavu petlje, tako što obezbeđuje da postoji samo jedan put između krajnjih čvorova u mreži. Gde postoje dodatni putevi, ovaj protokol te puteve stavlja u pasivno stanje ili ih blokira, ostavljajući samo jedan put aktivnim između dva čvora. Ako padne glavni put između čvorova, RSTP može aktivirati bekap put čime se održava povezanost mreže. Njegovo vreme konvergencije pri promeni topologije mreže je mnogo brže nego kod STP. Za razliku od STP, RSTP će odgovoriti na BPDU poslatu iz pravca koren mosta. RSTP most će predložiti svoju *spanning tree* informaciju svojim označenim portovima. Ako drugi RSTP most primi ovu informaciju i utvrdi da je ona superiornija koren informacija od njegove, stavlja sve svoje druge portove u stanje odbacivanja. Taj most može poslati poruku „dogovora“ prvom mostu, slažući se da je njegova *spanning tree* informacija superiornija. Prvi most po primanju „dogovora“ zna da može brzo da prebaci taj port u stanje prosleđivanja, zaobilazeći tradicionalna stanja slušanja/učenja. Ovo stvara kaskadni efekat daleko od koren mosta gde nadređeni mostovi mogu brzo da predlože promene drugim mostovima, a ovi mogu brzo da mu potvrde da li sme da uradi tranziciju ili ne. Ovo je jedan od glavnih elemenata koji omogućavaju RSTP da obezbedi brže vreme konvergencije od STP. RSTP održava bekap detalje po pitanju statusa odbacivanja portova. Ovim se izbegavaju vremenska ograničenja, ako trenutni portovi prosleđivanja padnu ili BPDU nisu primljeni na koren port u određenom vremenskom intervalu [57].

- U ethernet okruženjima gde postoji više virtuelnih LAN (VLAN), poželjno je stvoriti više *spanning trees* tako da saobraćaj iz različitih VLAN koristi različite linkove. Vlasničke verzije STP Cisco kompanije su *Per-VLAN Spanning Tree* (PVST) i *Per-VLAN Spanning Tree Plus* (PVST+) koje stvaraju odvojeni *spanning tree* za svaki VLAN. Oba protokola su osmišljena od strane Cisco i nekoliko tipova svičeva drugih proizvođača ih podržava. Ovi protokoli koriste različite multikast adrese. Neki uređaji kao što su Force10 Networks, Alcatel-Lucent, Extreme Networks, Avaya i BLADE Network Technologies podržavaju ove protokole. Extreme Networks podržavaju ove protokole sa dva ograničenja: nedostatak podrške na portovima gde su VLAN neoznačeni (*eng. untagged* – port na VLAN gde se host fizički povezuje na taj VLAN) i nedostatak podrške za VLAN sa ID 1.

PVST radi samo sa ISL (*eng. Inter-Switch Link* - vlasnički protokol Cisco koji održava VLAN informacije u ethernet okvirima, kao saobraćajne tokove između svičeva i rutera ili svičeva i svičeva) zbog ugrađenih ID STP. PVST je podrazumevani protokol za Cisco svičeve. Zbog zavisnosti od IEEE 802.1Q standarda (standard koji podržava VLAN na ethernet mreži) i ISL Cisco je definisao dodatni standard PVST+ koji je kompatibilan sa 802.1Q enkapsulacijom. Ovo je postao standardni protokol za Cisco svičeve kada je Cisco prekinuo i uklonio ISL podršku sa svojih svičeva. Ovaj protokol može da tuneluje kroz MSTP (*eng. Multiple Spanning Tree Protocol*) region [57][58].

- MSTP je prvobitno definisan u IEEE 802.1s, a kasnije spojen u IEEE 802.1Q-2005 standardu. Definiše ekstenziju RSTP za unapređenje iskorišćenja VLAN. MSTP konfiguriše odvojene *spanning tree* za svaku VLAN grupu i blokira sve osim jednog alternativnog puta u svakom *spanning tree*. Ako postoji samo jedan VLAN u mreži tradicionalni STP će raditi na odgovarajući način. Ako postoji više od jednog VLAN u mreži, logička mreža konfigurisana tradicionalnim STP će raditi, ali je moguće bolje iskoristiti slobodne alternativne puteve, korišćenjem alternativnog *spanning tree* za različite VLAN ili grupe VLAN. MSTP omogućava formiranje MST regiona koji mogu da pokreću više MSTP instanci, MSTPI (individualnih *spanning trees* u MSTP). Više regiona i drugi STP mostovi su povezani preko jednog zajedničkog *spanning tree*. Za razliku od vlasničkih per-VLAN *spanning tree* implementacija, MSTP uključuje sve svoje informacije o *spanning tree* u jedan BPDU format. Ovim se smanjuje potreban broj BPDU na LAN da bi komunicirale informacijama o *spanning tree* svake VLAN, ali takođe omogućava kompatibilnost unazad sa RSTP. MSTP ovo radi šifrujući dodatne informacije posle standardnog RSTP BPDU kao i MSTI poruka. Svaka od ovih MSTI konfiguracijskih poruka prenosi *spanning tree* informaciju za svaku instancu (individualni *spanning tree*). Svakoj instanci može biti dodeljen broj konfigurisanih VLAN i okvira dodeljenih ovim VLAN. Oni rade u ovoj *spanning tree* instanci kad god su unutar MST regiona. Da bi se izbeglo prenošenje njihovih celih VLAN u *spanning tree* mapiranju u svakoj BPDU, mostovi šifruju kratak prikaz MD5 njihovih VLAN u tabelu instanci u MSTP BPDU. Ovaj kratak prikaz posle koriste drugi MSTP mostovi sa drugim administrativnim vrednostima da bi odredili da li je njihov susedni most u istom MST regionu kao i oni. MSTP je u potpunosti kompatibilan sa RSTP mostovima, u smislu da se MSTP BPDU može tumačiti od strane RSTP mosta kao RSTP BPDU. Ovo omogućava kompatibilnost sa RSTP mostovima bez potrebe za konfiguracijskim promenama, kao i da bilo koji RSTP most izvan MSTP regiona vidi taj region kao jedan RSTP most, bez obzira na broj MSTP mostova unutar tog regiona. Da bi olakšao pogled da je jedan MST region jedan RSTP most, MSTP koristi promenljivu nazvanu preostali hopovi (*eng. hops*) koja je brojač vremena života, umesto *message age* tajmera koji se koristi RSTP. *Message age* (poruka starosti) se povećava samo jednom kada informacija *spanning tree* uđe u MST region i zato će RSTP mostovi videti region kao samo jedan "hop" u *spanning tree*. Portovi na ivici jednog MST regiona su povezani sa jednim RSTP ili STP mostom ili krajnjom tačkom i nazivaju se granični portovi. Kao u RSTP, ovi portovi se mogu konfigurisati kao ivični portovi da bi olakšali brze promene u stanje prosleđivanja kada su povezani na krajnje tačke [57][58].
- SPB (*eng. Shortest Path Bridging*) je odobren od strane IEEE kao IEEE 802.1aq standard 2012. godine. SPB omogućava svim linkovima da budu aktivni kroz više puteva jednakih cena. SPB obezbeđuje mnogo veće topologije drugog sloja, bržu konvergenciju i poboljšava korišćenje *mesh* topologija kroz povećan propusni opseg između svih uređaja. Ovo se postiže tako što se deo saobraćaja balansira preko svih puteva u *mesh* mreži. SPB

konsoliduje više postojećih funkcionalnosti, na primer STP, MSTP, RSTP u jedan protokol stanja linka. SPB je dizajniran da virtuelno eliminiše ljudsku grešku tokom konfiguracije i očuva *plug-and-play* prirodu koju je uspostavio Ethernet, kao protokol drugog sloja [58].

- TRILL (*eng. Transparent Interconnection of Lots of Links*) je protokol izdat od strane IETF za omogućavanje korišćenja više puteva za prenos podataka. TRILL primenjuje protokole rutiranja mrežnog sloja na sloj linka, i koristeći znanje o celoj mreži, podržava više puteva za prenos podataka na drugom sloju. Smanjuje kašnjenje i povećava ukupnu iskorišćenost mrežnog propusnog opsega. TRILL ima za cilj da zameni STP. STP je napravljen da bi sprečio petlje u mreži, omogućavajući samo jedan put između mrežnih portova ili svičeva. Kada delovi mreže padnu, alternativni put se aktivira i ovaj proces može da izazove neprihvatljiva kašnjenja u mreži data centra. TRILL je napravljen da reši ovaj problem koristeći IS-IS protokol trećeg sloja na uređajima drugog sloja. Ovo omogućava uređajima drugog sloja da rutiraju ethernet okvire. Zagovornici TRILL protokola tvrde, da ako se ukloni STP i aktivira više puteva na drugom sloju, kompanije će moći bolje da prebacuju virtuelne mašine kroz mrežu data centara. Takođe će biti dostupan i veći propusni opseg za zahtevnije aplikacije kao što su RTC (*eng. Real-Time Communications*) i za prenos saobraćaja skladišta kroz ethernet mrežu pomoću FCoE i iSCSI. Dodatno, TRILL će omogućiti da svičevi budu isplativiji, jer omogućava kompanijama da koriste više linkova u mrežama data centra. Daje mogućnost svičevima da izbalansiraju opterećenje saobraćaja preko više linkova drugog sloja. Ipak, neki stručnjaci se ne slažu sa tim da će TRILL rešiti probleme vezane za STP i da je uopšte potrebna zamena STP protokola. Postoji tvrdnja da TRILL može dovesti do slabijih tehnika dizajna mreže data centra i da su bolji pristupi na primer SPB (*eng. Shortest Path Bridging*) ili MLAG (*eng. Multichassis Link Aggregation*) [59].

6. PROBLEMATIKA I TRENDOWI U DATA CENTRIMA

Data centri su osnova današnje ekonomije. Njihov rast je neprekidan, kako raste potreba za skladištenjem i prenosom sve veće količine podataka. Ipak data centri su jedni od najvećih potrošača električne energije. Iako se misli da najveći udeo u potrošnji električne energije imaju ogromni data centri velikih kompanija kao što su Google, Apple ili Microsoft, najveći potrošači električne energije u slučaju data centara su male serverske sobe malih ili srednjih preduzeća. U Americi serverske sobe učestvuju sa oko 50% u potrošnji ukupne električne energije koja ide na data centre. Polovina te potrošene energije je bespotrebno utrošena zbog nepažnje i nedostatka inicijative da se data centri učine efikasnijim. Potreba da se bolje iskoristi električna energija koju koriste data centri ostaje od kritične važnosti, ako se uzme u obzir porast potrošnje energije koja ide na data centre. Veliki napredak je postignut u efikasnosti data centara u oblasti efikasnosti ustanova i opreme koju koriste. Ipak malo je uređeno po pitanju iskorišćenosti servera. Najveće kompanije koje pružaju *Cloud Computing* usluge, uglavnom imaju efikasne data centre, ali drugi tipovi data centara predstavljaju problem [60].

- i) *Cloud Computing predstavlja novu praksu u data centrima. Predstavlja opšti naziv za sve što uključuje pružanje hostovanih usluga preko Interneta. Omogućava kompanijama da koriste računarske resurse preko Interneta, a da ne moraju da imaju svoju računarsku infrastrukturu. Ove usluge su podeljene u tri kategorije: Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS) and Software-as-a-Service (SaaS). Oni se mogu klasifikovati kao privatni, javni i hibridni.*

Najznačajniji problem je serverska neefikasnost u data centrima, koji troše energiju 24/7, a obavljaju veoma malo operacija. Razlozi za to su:

- dimenzionisanje prema vršnom opterećenju: operatori data centra instaliraju više opreme da mogu savladati godišnje vršno opterećenje, ne isključuju opremu tokom perioda kada je opterećenje data centra značajno ispod vršnog opterećenja,
- mala rasprostranjenost virtuelizovane tehnologije, koja omogućava konsolidaciju opterećenja na manjim serverima, na svim serverima,
- „Comatose” serveri: veliki broj servera koji se više ne koriste i dalje troše energiju 24/7, jer ih niko ne uklanja ili nije ni svestan da se više ne koriste [60].

Višekorisnički data centri su data centri u kojima korisnici zakupljuju prostor i energiju za rad njihove računarske opreme, radije nego da imaju svoj data centar. Ovo predstavlja problem, jer odluke koje prave korisnici po pitanju efikasnosti njihove IT opreme nema direktan uticaj na njihov račun. Tako da nema finansijskog motiva da povećaju efikasnost svoje opreme. Tehnička rešenja kao što su virtuelizacija, upravljanje potrošnjom servera i pojedinačna odgovornost korisnika, posebno u višekorisničkim data centrima, za troškove IT opreme i postrojenja u kome su smešteni data centri, dobro su poznati, ali nisu iskorišćeni sistematski [60].

Postoje sjajni primeri data centara velike efikasnosti, ali najveći problem u ovom sektoru predstavljaju tri njegova najveća segmenta:

- male serverske sobe malih i preduzeća srednje veličine,
- data centri kompanija,
- višekorisnički data centri.

U daljem delu će se detaljnije sagledati problemi i predstaviti dalji koraci u razvoju data centara.



Slika 6.1: Savremeni data centar – Google [61]

6.2. Izazovi energetske efikasnosti

Kao što je već navedeno, najveći potrošači u industriji data centara su:

- male serverske sobe malih i preduzeća srednje veličine,
- data centri kompanija,
- višekorisnički data centri.

Postoji nekoliko faktora na osnovu kojih se procenjuje efikasnost data centra.

6.2.1. PUE (Power Usage Effectiveness)

Operatori data centara imaju priliku da smanje upotrebu energije koja odlazi na IT i infrastrukturu za hlađenje. PUE je mera koliko efikasno računarska oprema u data centru koristi energiju. $PUE = \frac{\text{Total Facility Energy}}{\text{IT Equipment Energy}}$. Ovo je najpopularnija metoda za računanje energetske efikasnosti, ali ova metoda ima problem. Ne računa klimatske uslove u gradovima u kojima se nalaze data centri. Na primer, ne mogu se ovim metodom uporediti dva data centra od kojih se jedan nalazi u Moskvi, a drugi u Sidneju. Hladnija klima zahteva manje sisteme za hlađenje. Sistemi za hlađenje imaju udeo od oko 30% utrošene energije postrojenja. Dodatno, mana je što je procenjen PUE besmislen, ako IT ne radi punim kapacitetom. PUE je korisna, jer obezbeđuje

nekoliko bitnih stavki. Kompanije koriste ovaj način proračuna jer merenje može da se ponavlja, da bi ustanovile svoju efikasnost tokom sezonskih promena i u toku vremena. Druga bitna stvar je da kompanije mogu da procene kako potencijalna poboljšanja tj. promene u data centrima pristupa utiču na njenu celokupnu potrošnju. Ako im je PUE mali mogu ga koristiti u marketinške svrhe, jer što je veći PUE to je data centar manje efikasan. Prosečan PUE kojim se zadovolje kompanije je oko 1,65 PUE, dok data centri u najvećim kompanijama imaju PUE 1,1 [60].



Slika 6.2.1: Kabliranje u data centrima [61]

6.2.2. Iskorišćenost servera

Iskorišćenost servera i dalje ostaje mala. Možda je ovo jedna od najčešće pominjanih tema kada se govori o energetskej efikasnosti u data centrima. Kao što je već rečeno, većina servera u data centrima radi 24/7, čekajući da primi podatke ili zahtev za prenos. Kada se konstruiše data centar ubacuje se veći broj servera da bi se prevazišlo preopterećenje saobraćaja do kog može doći. Ostatak vremena ti serveri i u modu mirovanja i dalje troše zavidnu količinu energije. Ovo ostaje za sada nerešen problem, jer iskorišćenost kao mera se retko prati i prijavljuje. Iako se vrši virtuelizacija servera da se smanji njihova neiskorišćenost, prosečna iskorišćenost je i dalje između 12-18%. Postoji više razloga za neiskorišćenost data centara:

- rezervisanje prekomerne količine IT resursa,
- ograničeno raspoređena virtuelizacija, uprkos svim njenim prednostima,
- „Comatose” serveri,
- neprimenjivanje rešenja za upravljanje energijom servera,
- praksa da se uvek gleda početna cena, a ne ukupna cena vlasništva,
- nedostatak standardne mere za iskorišćenost servera.



Slika 6.2.2: Microsoft serveri u Dublinu [61]

6.2.3. Podela troškova

Dvadeset procenata IT odeljenja plaća račun za struju u data centrima. Nedostatak podsticaja po ovom pitanju je jedna od glavnih prepreka za postavljanje efikasnih mera u data centrima. *Hyper-scale cloud service* provajderi su uglavnom rešili ovaj problem. To nije slučaj sa višekorisničkim data centrima, jer veze između višekorisničkih provajdera i njihovih korisnika sprečavaju u ovom segmentu data centara rešavanje problema. Vlasnik data centra plaća račun za utrošenu električnu energiju, dok korisnici zakupljuju prostor u koji smeštaju svoju IT opremu, tako da nemaju motivaciju da investiraju u efikasnost. Ovaj problem najviše pogađa male i srednje organizacije. Problem je podela troškova, koji su sve veći kako raste potražnja za data centrima. Isti ovi problemi se odnose i na višekorisničke data centre. Pored toga postoje još dva razloga zbog čega višekorisnički data centri ne mogu da ostvare bolju energetska efikasnost: konkurentski prioriteti da bi se zadobili kupci i izazovi pri implementiranju podsticajnih programa efikasnosti koji odgovaraju višekorisničkom data centru [60].

6.3. Preporučene akcije i trendovi

Da bi se prevazišli problemi navedeni u prethodnom delu poglavlja mnogo strana mora da se aktivira. U nastavku je dat pregled preporučenih akcija, koje bi trebalo uraditi [60].

Federalne i državne vlade:

- uspostaviti politiku prema višekorisničkim data centrima, tako da odrede cene plaćanja vlasniku na osnovu prostora i energije koje korisnici koriste u njima,
- postaviti minimalne, obnovljive portfolio standarde za velike data centre,
- napraviti obrazac tako da se višekorisnički data centri prebace na *green* data centre i tu naplaćuju usluge,
- zahtevati da se uspostavi mera iskorišćenosti za data centre,
- razdvojiti komunalne prihode od prodaje električne energije za data centre.

Komunalna električna preduzeća:

- sprovesti efikasne programe podsticaja za energetske efikasnost vezane za data centre, kao što su programi za bolju iskorišćenost opreme i programe koji su usmereni na višekorisničke data centre.

Hyper-scale Cloud Service provajderi:

- nastaviti sa pristupima po pitanju unapređivanja infrastrukture i IT efikasnosti,
- pokušati da data centri koriste što više obnovljivih izvora energije,
- otkriti meru za iskorišćenost.

Provajderi višekorisničkih data centara:

- podesiti tarifne modele, koji bi podstakli energetske efikasnost od strane kupaca,
- usvojiti i rasporediti DCIM alate da bi povećali energetske karakteristike i poboljšali izveštavanje,
- otkriti meru energetske performansi i ugraditi je u informacije za obračun.

Korisnici višekorisničkih data centara:

- u novim ugovorima, pregovarati za cenu modela na osnovu stvarnog prostora i korišćenja energije, kao i da istraže mogućnost da pregovaraju o postojećim uslovima,
- razviti i uspostaviti temelje o ugovoru za *green* data centre,
- zahtevati meru za energetske performanse i te performanse uključiti u cenu iznajmljivanja.

Mala i srednja preduzeća:

- sagledati uticaj data centara na životnu sredinu, koristeći saradnju odeljenja za zaštitu sredine i IT odeljenja,
- angažovati vrhunske menadžere da usklade IT sektor i energetska postrojenja kako bi slali energiju kroz data centre,
- razmotriti modularne data centre radi poboljšanja operacija i usluga.

Industrijska udruženja:

- izdati priručnik najbolje prakse, standarda, metodologije za unapređenje energetske efikasnosti,
- staviti fokus na višekorisničke data centre i obezbediti najbolje načine poslovanja za njih,
- razviti nacrt ugovora koji bi višekorisničke data centre približio *green* data centrima.



Slika 6.2.1: Solarni data centar [61]

Jedan od najaktuelnijih trendova današnjice vezan za data centre je *green* data centar. *Green* data centar je prostor za skladištenje, upravljanje i širenje podataka u kojoj mehanički, električni i računarski sistemi su dizajnirani za maksimalnu energetske efikasnost i minimalan uticaj na životnu sredinu, jer standardni data centri izbacuju velike emisije gasova pri svom radu. Izgradnja *green* data centara obuhvata napredne tehnologije i strategije. Nekoliko primera: upotreba materijala niske emisije gasova, reciklaža otpada, instalacija katalizatora na rezervne generatore, korišćenje alternativnih tehnologija energije kao što je upotreba toplotnih pumpi. Ovi data centri predstavljaju budućnost celokupnog sektora data centara [60].



Slika 4.2.2: *Green* data centar [61]

7. ZAKLJUČAK

Data centri danas i sutra po pitanju poslovanja nisu isto. Preduzeća velika i mala se takmiče kako da što bolje iskoriste svoje data centre i da ih unaprede. Činjenica je da je tehnologija postala alat nadmetanja u današnjem poslovanju i prenos i čuvanje podataka je od kritične važnosti. Zato je stalna trka u unapređivanju data centara tako aktuelna.

Sušтина je da se data centar učini što efikasnijim za rad po prihvatljivim cenama. Ne može se reći da neka tehnologija koja se primenjuje u data centrima može da potisne sve ostale, jer svaka ima svoje prednosti i mane. Zaključak koji se izvlači iz ovog rada je da su data centri nezaobilazna komponenta funkcionisanja savremenog doba, ali da to ima svoju cenu. Cena koju zahtevaju za rad na primer po pitanju utrošene energije postaje neodrživ teret za većinu kompanija. Ali potražnja za skladištenjem i prenosom podataka je sve veća. Predstavljene su komponente infrastrukture i njihovu zahtevi, mere poboljšanja njihovog rada, kao i topologije mreže i protokoli koji moraju biti ispunjeni za efikasan rad data centara. Zaključak je da ovde ima još mesta za poboljšanjem i da nijedna topologija se ne može izdvojiti kao dominantna. Ipak budućnost za smeštanjem podataka i njihovim čuvanjem leži u data centrima što je opisano u poslednjem poglavlju. Nezamislivo veliki broj usluga koje koristimo u svakodnevnom životu se odvija preko data centara. Samim tim nastavlja se sa težnjom stvaranja savremenog data centra koji će ispuniti sve kriterijume kao što su cena, bezbednost podataka, dostupnost itd. Glavne karakteristike budućih data centara biće: brz odziv na potrebe korisnika, manja potrošnja, veća efikasnost, povećana pouzdanost, jednostavan dizajn, neutralnost u odnosu na proizvođače opreme, otvoreni standard.

LITERATURA

- [1] SAVREMENI DATA CENTRI, www.viser.edu.rs
- [2] Darwyn F. Kelley, *Digital History*, <http://www.old-computers.com/history/detail.asp?n=61>
- [3] Angela Bartels, *Data center evolution: 1960 to 2000*, <http://www.rackspace.com/blog/datacenter-evolution-1960-to-2000/>
- [4] *Data center*, https://en.wikipedia.org/wiki/Data_center
- [5] *Enterprise resource planning*, https://en.wikipedia.org/wiki/Enterprise_resource_planning
- [6] *CRM*, https://en.wikipedia.org/wiki/Customer_relationship_management
- [7] Dušan Đokić, *ŠTA JE CRM I ŠTA ĆE TO MENI?*, <http://blog.extreme.rs/2011/12/08/sta-je-taj-crm-i-sta-ce-to-meni/>
- [8] *What is data centre tiering and why should I consider a tier III data centre?*, http://www.coreix.net/resources/faqs_tiering
- [9] Steve Strutt, *DATA CENTER EFFICIENCY AND IT EQUIPMENT RELIABILITY AT WIDER OPERATING TEMPERATURE AND HUMIDITY RANGES*, <https://datacenters.lbl.gov/sites/all/files/WP50-Data%20Center%20Efficiency%20and%20IT%20Equipment%20Reliability%20at%20Wider%20Operating%20Temperature%20and%20Humidity%20Ranges.pdf>
- [10] Data Centre Huddle, *Data center 101*, <http://www.dchuddle.com/data-center-101/>
- [11] *CRAC vs CRAH*, <http://www.dchuddle.com/2011/crac-v-crah/>
- [12] Margaret Rouse, *tower server*, <http://whatis.techtarget.com/definition/tower-server>
- [13] Olivia Hammonds, *3 Benefits Of Using Server Racks*, <http://www.3benefitsof.com/3-benefits-of-using-server-racks/>
- [14] *Blade server*, https://en.wikipedia.org/wiki/Blade_server
- [15] *SERVERSKI SISTEMI I VIRTUELIZACIJA*, <http://www.telegroup.rs/ict-resenja-proizvodi/serverski-sistemi-i-virtuelizacija/34/serverski-sistemi-i-virtuelizacija/>
- [16] *OSI model*, https://sh.wikipedia.org/wiki/OSI_model#1. Fizi.C4.8Dki_sloj
- [17] *Storage Considerations in Data Center Design*, <http://snia.org/sites/default/files/StorageConsiderationsinDataCenterDesign.pdf>
- [18] *Data Center Network – Top of Rack (TOR) vs End of Row (EOR) Design*, <http://www.excitingip.com/2802/data-center-network-top-of-rack-tor-vs-end-of-row-eor-design/>
- [19] Yang Liu, Jogesh K Muppala, Malathi Veeraraghavan, Dong Lin, and Mounir Hamdi, *Data Center Networks: Topologies, Architectures and Fault-Tolerance Characteristics*, Springer Science & Business Media, 2013.

- [20] Yan Zhang and Nirwan Ansari, "On Architecture Design, Congestion Notification, TCP Incast and Power Consumption in Data Centers, " *Communications Surveys & Tutorials, IEEE*, 15(1): 39 – 64, 11 January 2012.
- [21] Chuanxiong Guo, Haitao Wu, Kun Tan, Lei Shi, Yongguang Zhang, and Songwu Lu, *DCell: A Scalable and Fault-Tolerant Network Structure for Data Centers*, <http://ccr.sigcomm.org/online/files/p75-guoA.pdf>
- [22] Chuanxiong Guo, Guohan Lu, Dan Li, Haitao Wu, Xuan Zhang, Yunfeng Shi, Chen Tian, Yongguang Zhang, Songwu Lu, *BCube: A High Performance, Server-centric Network Architecture for Modular Data Centers*, <http://research.microsoft.com/pubs/81063/comm136-guo.pdf>
- [23] MDCube, <http://www.slideshare.net/fangvv/ss-8846170>
- [24] Haitao Wu, Guohan Lu, Dan Li, Chuanxiong Guo, Yongguang Zhang, *MDCube: A High Performance Network Structure for Modular Data Center Interconnection*, <http://research.microsoft.com/pubs/103077/conext02-wu.pdf>
- [25] Dan Li, Chuanxiong Guo, Haitao Wu, Kun Tan, Yongguang Zhang, Songwu Luy, *FiConn: Using Backup Port for Server Interconnection in Data Centers*, <http://research.microsoft.com/pubs/79810/main.pdf>
- [26] Guohui Wang, David G. Andersen, Michael Kaminsky, Konstantina Papagiannaki, T. S. Eugene Ng, Michael Kozuch, Michael Ryan, *c-Through: Part-time Optics in Data Centers*, <http://www.cs.rutgers.edu/~badri/552dir/papers/datacenter/cthrough10.pdf>
- [27] Nathan Farrington, George Porter, Sivasankar Radhakrishnan, Hamid Hajabdolali Bazzaz, Vikram Subramanya, Yeshaiahu Fainman, George Papen, and Amin Vahdat, *Helios: A Hybrid Electrical/Optical Switch Architecture for Modular Data Centers*, <http://cseweb.ucsd.edu/~vahdat/papers/helios-sigcomm10.pdf>
- [28] Kai Chen, Ankit Singla, Atul Sing, Kishore Ramachandran, Lei Xu, Yueping Zhang, Xitao Wen, Yan Chen, *OSA: An Optical Switching Architecture for Data Center Networks with Unprecedented Flexibility*, <http://www.cse.ust.hk/~kaichen/courses/spring2013/comp6611/papers/osa-nsdi12.pdf>
- [29] <http://opticalengineering.spiedigitallibrary.org/article.aspx?articleid=1887713>
- [30] John Kim, William J. Dally, Steve Scott, Dennis Abts, "Technology-Driven, Highly-Scalable Dragonfly Topology," *Computer Architecture, 2008. ISCA '08. 35th International Symposium on Computer Architecture*, 21-25 June 2008.
- [31] William J. Dally, *Technology-Driven, Highly-Scalable Dragonfly Topology*, http://www.ece.rochester.edu/~parihar/pres/Pres_DragonFly.pdf
- [32] László Gyarmati, Tuan Anh Trinh, *Scafida: A Scale-Free Network Inspired Data Center Architecture*, <http://www.sigcomm.org/sites/default/files/ccr/papers/2010/October/1880153-1880155.pdf>
- [33] Tanenbaum, Andrew S, *Computer networks*, 4-th edition, *Prentice Hall*, 2003.
- [34] *Domain Name System (DNS) Definition*, <http://searchnetworking.techtarget.com/definition/domain-name-system>
- [35] *File Transfer Protocol*, https://en.wikipedia.org/wiki/File_Transfer_Protocol
- [36] *OSPF*, <https://sh.wikipedia.org/wiki/OSPF>

- [37] *IS-IS*, <https://en.wikipedia.org/wiki/IS-IS>
- [38] *Interior Gateway Routing Protocol*, https://en.wikipedia.org/wiki/Interior_Gateway_Routing_Protocol
- [39] *Unified communications*, https://en.wikipedia.org/wiki/Unified_communications
- [40] *VoIP*, https://en.wikipedia.org/wiki/Voice_over_IP
- [41] *Real-time Transport Protocol*, https://en.wikipedia.org/wiki/Real-time_Transport_Protocol
- [42] *Internet Control Message Protocol*, https://en.wikipedia.org/wiki/Internet_Control_Message_Protocol
- [43] *OpenFlow*, <https://en.wikipedia.org/wiki/OpenFlow>
- [44] *Hot Standby Router Protocol*, https://en.wikipedia.org/wiki/Hot_Standby_Router_Protocol
- [45] *IPv4*, <https://en.wikipedia.org/wiki/IPv4>
- [46] *IPv6*, <https://en.wikipedia.org/wiki/IPv6>
- [47] Margaret Rouse, *IPsec (Internet Protocol Security)*, <http://searchmidmarketsecurity.techtarget.com/definition/IPsec>
- [48] A Stateless Transport Tunneling Protocol for Network Virtualization (STT), <https://tools.ietf.org/pdf/draft-davie-stt-01.pdf>
- [49] *Multiprotocol Label Switching*, https://en.wikipedia.org/wiki/Multiprotocol_Label_Switching
- [50] *Virtual Router Redundancy Protocol*, https://en.wikipedia.org/wiki/Virtual_Router_Redundancy_Protocol
- [51] *User Datagram Protocol*, https://en.wikipedia.org/wiki/User_Datagram_Protocol
- [52] *UDT: UDP based Data Transfer Protocol*, <http://udt.sourceforge.net/doc/udt-sc08-poster.pdf>
- [53] *UDP-based Data Transfer Protocol*, https://en.wikipedia.org/wiki/UDP-based_Data_Transfer_Protocol
- [54] *Point-to-Point Protocol*, https://en.wikipedia.org/wiki/Point-to-Point_Protocol
- [55] *Link Layer Discovery Protocol*, https://en.wikipedia.org/wiki/Link_Layer_Discovery_Protocol
- [56] *Transmission Control Protocol*, https://en.wikipedia.org/wiki/Transmission_Control_Protocol
- [57] Aaron Balchunas, *Spanning Tree Protocol*, <http://www.routeralley.com/guides/stp.pdf>
- [58] *Spanning tree protocol*, https://en.wikipedia.org/wiki/Spanning_Tree_Protocol
- [59] *Transparent Interconnection Of Lots Of Links (TRILL) Definition*, <http://searchnetworking.techtarget.com/definition/Transparent-Interconnection-of-Lots-of-Links-TRILL>
- [60] Josh Whitney, Pierre Delforge, Eric Masanet, George Peridas, Maria Stamas, Nick Jimenez, Pat Remick, John Clinger, Simon Mui Ted Brown, *Scaling Up Energy Efficiency Across the Data Center Industry: Evaluating Key Drivers and Barriers*, <https://www.nrdc.org/energy/files/data-center-efficiency-assessment-IP.pdf>
- [61] <http://www.datacenterknowledge.com/the-top-10-data-center-images-of-2012/>