Računalni oblaci kao dio servisno orijentirane arhitekture

Brumec, Slaven

Doctoral thesis / Disertacija

2011

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Organization and Informatics Varaždin / Sveučilište u Zagrebu, Fakultet organizacije i informatike Varaždin**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:211:891056

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-03-29**

SVEUČILIŠTE U ZAGREBU
FAKULTET ORGANIZACIJE I INFORMATIKE
VARAŽDIN

Repository / Repozitorij:

Faculty of Organization and Informatics - Digital Repository

DIGITALNI AKADEMSKI ARHIVI I REPOZITORIJI

SVEUČILIŠTE U ZAGREBU

FAKULTET ORGANIZACIJE I INFORMATIKE VARAŽDIN

SLAVEN BRUMEC

RAČUNALNI OBLACI KAO DIO SERVISNO ORIJENTIRANE ARHITEKTURE

DOKTORSKA DISERTACIJA

VARAŽDIN, veljača 2011.

PODACI O DISERTACIJI

I. A u t o r

Ime i prezime	Slaven Brumec
Datum i mjesto rođenja	24. rujna 1976., Zagreb
Naziv fakulteta i datum diplomiranja	Fakultet elektrotehnike i računarstva, 26. travnja 2001.
Sadašnje zaposlenje	Softverski inženjer u tvrtki Koris

II. Disertacija

Naslov	Računalni oblaci kao dio servisno orijentirane arhitekture
Broj stranica, slika, tabela, priloga, bibliografskih podataka	XVI + 191 stranica, 27 tabela, 37 slika, 5 priloga i 115 bibliografskih referenci
Znanstveno područje, smjer i disciplina iz koje je postignut akademski stupanj	Društvene znanosti, informacijske znanosti, informacijski sustavi i informatologija
Mentor ili voditelj rada	prof.dr.sc. Neven Vrček
Fakultet na kojem je rad obranjen	Fakultet organizacije i informatike, Varaždin
Oznaka i redni broj rada	99

III. Ocjena i obrana

Datum prihvaćanja teme od Znanstveno-nastavnog vijeća	20. listopada 2009.
Datum predaje rada	1. veljače 2011.
Datum sjednice ZNV-a na kojoj je prihvaćena pozitivna ocjena rada	21. lipnja 2011.
Sastav Povjerenstva koje je rad ocijenilo	prof.dr.sc. Vjeran Strahonja, predsjednik prof.dr.sc. Neven Vrček, mentor i član prof.dr.sc. Mladen Vouk, član
Datum obrane rada	14. srpnja 2011.
Sastav Povjerenstva pred kojim je rad obranjen	prof.dr.sc. Vjeran Strahonja, predsjednik prof.dr.sc. Neven Vrček, mentor i član prof.dr.sc. Mladen Vouk, član
Datum promocije	

SVEUČILIŠTE U ZAGREBU

FAKULTET ORGANIZACIJE I INFORMATIKE VARAŽDIN

SLAVEN BRUMEC

RAČUNALNI OBLACI KAO DIO SERVISNO ORIJENTIRANE ARHITEKTURE

DOKTORSKA DISERTACIJA

VARAŽDIN, veljača 2011.

Predgovor

Unazad desetak godina često se čuje da napredak hardvera i softvera, to jest računarstva i informatike, nije bio ravnomjeran – hardver je napredovao geometrijskom progresijom, po Mooreovom zakonu, a softver jedva aritmetičkom. Onaj pak tko pobliže poznaje stanje u informatici može primijetiti da se velika sredstva i dalje troše na hardver – koji potom često nije iskorišten niti blizu mogućnostima, a ponekad niti uopće. Dodajmo tome i opažanje da je značajan dio proizvedenog poslovnog softvera, čak i suvremenog, i dalje relativno zatvoren te često također slabo ponovno iskoristiv (*reusable*).

Ukratko, unatoč ogromnom teorijskom i eksperimentalnom napretku u hardverskoj i softverskoj tehnologiji i industriji, poslovni korisnici prečesto se suočavaju sa skupim, glomaznim i zatvorenim poslovnim informacijskim sustavima, koji se izvode na skupom i nedovoljno iskorištenom hardveru. Sažeto rečeno, "*kupuju se 64-bitni strojevi s četverojezgrenim procesorom da bi se na njima igrao Tetris i pokretali programi u rangu složenosti Worda i Excela – u najboljem slučaju. Potom se rezultatima takve obrade uništavaju šume, jer se podaci ispisuju na papir kako bi ih negdje drugdje, netko drugi, opet upisivao u neki program.*" Nažalost, cijeli se ovaj nedjelotvorni postupak još i plaća!

Jasno, nije nam namjera kritizirati uredske alate. Radi se o sjajnim programima koji još od kraja sedamdesetih godina uvelike olakšavaju život i rad milijunima ljudi diljem svijeta. Kao takvi, za određene svrhe su nezamjenjivi i ostat će nezamjenjivi dokle god postoje računala u današnjem smislu. Problem nije u tim alatima, nego je gornji sažetak naveden samo kao ilustracija nepraktične uporabe *dobrih* rješenja.

Dakle, među odgovorima na gore navedene probleme često se promoviraju dva koncepta – servisno orijentirana arhitektura i oblačno računarstvo. Oba koncepta, iako se odnose na različite domene, imaju dvije bitne zajedničke značajke: trošenje resursa (softverskih odnosno hardverskih) prema korištenju te promicanje interoperabilnosti među sustavima različitih proizvođača, korištenjem standardnih komunikacijskih i mrežnih tehnologija. Drugim riječima, zajednička im je servisna paradigma uz koju se vezuje interoperabilnost, ponovna iskoristivost i ekonomičnost. Kako povezati te dvije tehnologije, to jest pod kojim je uvjetima probitačno SOA paradigmu uklopiti u oblačno računarstvo, primarni je interes ovoga rada.

Računalni oblaci su, kako ćemo ubrzo pokazati u ovom radu, komercijalno rješenje i nadgradnja računalnih gridova, prve jače proširene tehnologije korištenja računala u obliku usluge, koja je ipak uglavnom korištena na sveučilištima i znanstvenoj zajednici. Računalni oblaci omogućuju najam optimalno konfiguriranih i dobro održavanih serverskih farmi prema potrebi te bezbolno otpuštanje najmljenih resursa kada poslovne potrebe za njima nestane. Potencijalne koristi, ali i poneke opasnosti, takvog pristupa su višestruke te su zaslužile

detaljniji pregled i ozbiljnu znanstvenu analizu kroz ovaj rad, kako bi se razlučilo nekritičko prihvaćanje komercijalnih ponuda od stvarnih mogućnosti i konkretnih vrijednosti.

Ponuditelji usluga oblačnog računarstva (*cloud providers*) obećavaju izobilje računalnih resursa po pristupačnim cijenama za svakoga, „*serverske farme za mase, a ne klase*“ da parafraziramo Jacka Tramiela. Kalkulatore troškova i cjenike ima svaki oblačni poslužitelj na svojim stranicama, no ono što potencijalne korisnike oblačnog računarstva zanima jest što zapravo znače te cijene? Da li je to malo ili mnogo? Ima li taj iznos istu vrijednost za svaki obujam korištenja računalnih resursa? Moguće ulagače u računalne oblake zanima pak pod kojim uvjetima je isplativo ulagati u računalni oblak. Oni se pitaju o kojim parametrima ovisi rentabilnost oblaka? U ovom radu odgovorit ćemo na ta pitanja.

Kakva je bliska budućnost računalnih oblaka? Čini se da će zauzeti značajno mjesto na IT tržištu, iako ćemo obrazložiti zašto **neće** jednostavno nadomjestiti sve ostale računalne tehnologije. Koje su daljnje perspektive? Tu već zalazimo u znanstvenu fantastiku, ali – dva od tri najznačajnija oblaka proizišla su u većoj ili manjoj mjeri iz tehnologija korištenih u internetskim tražilicama s kraja 20. stoljeća. Dvojica velikana znanstvene fantastike, Arthur Clarke i Stephen Baxter, u zajedničkoj trilogiji zvanoj VREMENSKA ODISEJA zamislili su sveprisutne (*ubiquitous*) distribuirane računalne sustave, upravo kao daljnje potomke današnjih internetskih tražilica – računalnih oblaka. U duhu optimističnog transhumanizma kakvog zagovara, primjerice, poduzetnik i futurist Ray Kurzweil, ti sustavi, pomoćnici čovječji, postaju svjesni, dosežu status legalne osobe, elektroničkog djeteta ljudskog roda, i konačno, skupa s najvećim ljudskim umovima, u središnjem romanu trilogije, OLUJI SA SUNCA, postaju i spasitelji čovječanstva. Znanstvena fantastika? Zasigurno. Barem za sada, dok nam iz oblačnih podatkovnih centara još uvijek nije stigla rečenica na zaslon i u zvučnike: *"Dobro jutro, prijatelju, spreman sam za prvi sat."* Možda će ljudi jednom programirati oblake na način da će biti moguća i takva iznenađenja …

Sažetak

U istraživačkom fokusu ovog rada su komercijalni računalni oblaci. Provedena su mjerenja radi određivanja trajanja izvođenja aplikacije u oblaku te uspoređeni tehnički i ekonomski uvjeti njezinog izvođenja u oblaku s uvjetima izvođenja na vlastitim računalnim resursima. Povezivanjem rezultata vlastitih istraživanja s radovima drugih istraživača razvijena je metodika korištenja oblačnog računarstva. Tekst disertacije organiziran je na sljedeći način.

U prvom poglavlju izrečena je motivacija, objašnjena znanstvena i društvena opravdanost istraživanja, postavljeni su ciljevi rada, definirane istraživačke hipoteze te opisane metode i tehnike koje će se koristiti u istraživanju. U drugom poglavlju definirana je taksonomija oblačnog računarstva te opisana funkcionalna i uporabna svojstva računalnih oblaka.

U trećem poglavlju opisane su tehnologije na kojima se zasniva oblačno računarstvo, s posebnim naglaskom na virtualizaciju i servisnu paradigmu korištenja. Izvedena je definicija oblačnog računarstva, opisana fizička realizacija računalnih oblaka, naznačene su moguće ekonomske i tehničke prednosti njihovog korištenja te tržišni potencijali. Četvrto poglavlje opisuje ekonomske aspekte uporabe računalnih oblaka, napose elastičnost i optimalnu nabavu. Formulirani su temelji ekonomike oblačnog računarstva i kratko analizirane neke aktualne komercijalne ponude. Definirane su varijable za opis svojstava složene aplikacije (pogodne za izvođenje u oblaku ili na vlastitim računalima) te postavljen koncept za mjerenje utjecaja tih varijabli na trajanje i troškove izvođenja aplikacije u oblaku.

Rezultati opsežnih mjerenja performansi ogledne aplikacije smještene u oblaku obrađeni su u petom poglavlju. Matematičkom obradom rezultata mjerenja utvrđena je zavisnost vremena izvođenja aplikacije ***T*** o nezavisnim varijablama koje opisuju njezino izvođenje na računalnim resursima te je tako dokazana druga hipoteza koja se odnosi na razvoj višedimenzionalne metrike potreba za računalnim servisima iz računalnog oblaka.

U šestom poglavlju postavljen je općeniti postupak (temeljen na metrici iz petog poglavlja te radovima E. Walkera) kojim je moguće objektivno procijeniti da li je isplativije neku aplikaciju pokretati u računalnom oblaku ili na vlastitim računalnim resursima. Razvijen je algoritam za odlučivanje o kupnji ili najmu, temeljen na integraciji saznanja iz prethodnih poglavlja.

U sedmom poglavlju sistematizirani su postupci za projektiranje hibridnog informacijskog sustava koji ima barem neke komponente smještene u oblaku. Ovi su postupci oblikovani u metodiku korištenja oblačnog računarstva (KOR). Pokazano je da se koncepti SOA i oblačnog računarstva temelje na istim komunikacijskim tehnologijama i komplementarnim paradigmama njihovog korištenja. Metodika KOR daje smjernice o tome kako i pod kojim uvjetima se SOA može uklopiti u računalni oblak.

U disertaciji su postavljene dvije temeljne hipoteze:

- H1: Metodika razvoja hibridnih SOA rješenja može se izgraditi temeljem normiranih metoda i tehnika za modeliranje, usklađivanje i opisivanje poslovnih procesa.
- H2: Moguće je postaviti višedimenzionalnu metriku potreba za računalnim resursima SOA servisa prije njihove izgradnje te stvarnih performansi nakon njihove izgradnje. Korištenje takve metrike će povećati sigurnost pri odlučivanju o tome treba li neki servis stalno ili povremeno pokretati u računalnom oblaku.

Istraživanje obuhvaćeno ovom disertacijom provedeno je slijedeći način:

1) Izabrana je ogledna aplikacija koja dobro aproksimira svojstva većine poslovnih aplikacija. Utvrđeno je 6 parametra koji definiraju način izvođenja aplikacije u oblaku ili na privatnim računalima, a za koje se pretpostavlja da utječu na vrijeme izvođenja aplikacije. Aplikacijski parametri su složenost algoritma ***a*** i broj CRUD akcija po upitu ***q***. Podatkovni parametri su opseg podataka koji se šalje u oblak radi obrade ***u*** te veličina baze podataka ***b***. Potrebe za računalnim resursima određuju broj ***r*** i snaga računala ***p***. Istraživanje je izvedeno tako da je tipska aplikacija preuređena za izvođenje u oblaku te je vrijeme njezinog izvođenja ***T*** (za razne vrijednosti parametara) uspoređeno s vremenom T_{ORG} kojeg očekuje korisnik u radu s aplikacijom. Tipska aplikacija je namijenjena prepoznavanju oblika i koristi se za identifikaciju osobe usporedbom njezine aktualne fotografije s više slika iz baze podataka. Aplikacijski i podatkovni parametri mogu se mijenjati u potrebnim granicama, a resursni parametri se odabiru kod izvedbe aplikacije u oblaku. Tako su ostvareni uvjeti za mjerenje ***T*** u zavisnosti o izazvanim promjenama 6 ispitivanih parametara, uz isključene utjecaje slučajnih varijabli.

2) Da bi se odredio matematički model za opis performansi tipske aplikacije, izvedeno je 16 mjerenja prema višefaktorskom parcijalnom planu pokusa tipa 2^(6-2). Rezultati mjerenja su obrađeni metodama multivarijantne analize te je na taj način određen matematički model za izračun vremena izvođenja aplikacije u zavisnosti o 6 istraživanih parametara koji glasi: $T=0,631+14,993\cdot a+0,135\cdot u+0,702\cdot q-0,840\cdot r-2,160\cdot a\cdot r$. Ustanovljeno je da su signifikantni parametari ***a***, ***u***, ***q*** i ***r*** te dvofaktorne interakcije ***a*** i ***r***. Dobiveni rezultati prikazani su grafički i detaljno prodiskutirani, čime je dokazana hipoteza 2.

3) Razvijen je matematički model za izračun pune cijene vlasništva (TCO) oblačnog centra, a njegova korektnost provjerena je proračunom troškova virtualnog podatkovnog centa i usporedbom rezultata tog izračuna s dostupnim podacima o strukturi troškova više izvedenih podatkovnih centara. Polazeći od radova E. Walkera razvijeni su matematičko-ekonomski modeli za izračun troškova CPU vremena i troškova diskovnih resursa za pohranu podataka.

Prvim modelom, u kojem se računa s povećanjem računalne snage po Mooreovom zakonu, pokazano je da trošak po jedinici procesorskog vremena zavisi o stupnju iskorištenosti

procesora. To upućuje na zaključak da je za tvrtke koje imaju vremenski promjenljive zahtjeve, povoljnije unajmiti CPU vrijeme nego graditi vlastite serverske farme čiji kapacitet mora zadovoljiti vršne zahtjeve. To vrijedi ako sigurnosni zahtjevi i političke okolnosti ne ograničavaju takvo pribavljanje računalnih resursa. Drugi model, u kojem se računa s empirijski potvrđenim smanjivanjem cijene memorija tijekom vremena, pokazuje da je najam resursa za pohranu podataka dobra opcija za pojedine korisnike, a da je uvijek povoljniji za mala i srednja poduzeća. Za organizacije s velikim podatkovnim potrebama je kupnja vlastitih resursa za pohranu podataka povoljnija opcija od unajmljivanja jednako velikih resursa.

Oba modela omogućavaju investitoru uvid u rentabilnost ulaganja, a korisniku usporedbu troškova najma potrebnih resursa prema troškovima vlasništva nad resursima iste snage. Uz ta dva matematička modela, te sistematiziranu komercijalnu ponudu oblačnih poslužitelja, izveden je algoritam odlučivanja o kupnji ili najmu. Korektnost algoritma provjerena je za tri virtualne aplikacije. Za svaku od njih izračunati su troškovi, koje bi imao korisnik izvodeći ih na računalnim resursima izabranih komercijalnih pružatelja ili na vlastitim resursima.

4) Kao sinteza istraživanja definirana je metodika KOR za uklapanje servisno orijentiranih aplikacija u računalne oblake. Metodika je izvedena kao proces te je formalizirana po BPM notaciji. Provodi se u 23 koraka te omogućuje objektiviziranje odluke o tome koje aplikacije informacijskog sustava tvrtke ili ustanove je isplativo izvoditi u oblaku, za koje treba nabaviti vlastite ICT resurse te kako sve to provesti na sistematičan način. Metodika KOR primijenjena je u izgradnji tipske aplikacije te se pokazalo da može opisati postupke za razvoj hibridnih aplikacija u oblaku. Budući da je metodika KOR provjerena kod pripreme tipske aplikacije za izvođenje u oblaku, smatramo da je dokazana hipoteza 1.

Zaključno, ovom su disertacijom ostvareni su sljedeći znanstveni rezultati:

1. Postavljena je metrika za uspješno mjerenje aplikacijske potrebe za računalnim resursima i diskovnim kapacitetom. Ta metrika je općenita i primjenljiva je na razne vrste aplikacija koje se mogu izvoditi na različitim tehničkim platformama. Temelji se na mjerenju vremena potrebnog da aplikacija obavi zadani posao, pri čemu je to vrijeme ovisno o signifikantnim nezavisnim varijablama koje opisuju aplikaciju.
2. Razjašnjen je odnos između računalnih oblaka i drugih oblika naprednog računarstva te pokazano da su računalni oblaci platforma na kojoj je moguće postaviti i koristiti servisno orijentirana rješenja.
3. Razvijena je metodika KOR kojom je određen proces razvoja hibridnih informacijskih sustava i omogućena kvantificirana procjena koje aplikacije informacijskog sustava tvrtke ili ustanove je isplativo smjestiti u oblak, a za koje treba nabaviti vlastite ICT resurse. Metodika uzima u obzir potrebu preoblikovanja već postojećih aplikacija u SOA servise.

Sadržaj

1 Uvod ... 1
1.1 Motivacija ... 1
1.2 Dosadašnja istraživanja ... 2
1.3 Ciljevi rada ... 4
1.4 Metodologija i plan istraživanja ... 5
2 Nastanak računalnih oblaka ... 8
2.1 Distribuirano računarstvo ... 8
2.2 Paralelno računarstvo ... 10
2.3 Uslužno računarstvo ... 11
2.4 Mrežno računarstvo ... 11
2.5 Virtualizacija ... 14
2.6 Oblačno računarstvo – sinteza napretka ... 14
2.7 Sažetak ... 18
3 Opis oblačnog računarstva ... 19
3.1 Virtualizacija ... 19
3.2 Servisna paradigma ... 21
3.3 Definicija oblačnog računarstva ... 24
3.4 Etimologija oblačnog računarstva ... 25
3.5 Vrste računalnih oblaka ... 26
3.6 Izvedba računalnih oblaka ... 28
3.7 Ekonomske prednosti unajmljivanja računalnih resursa ... 33
3.8 Tehničke prednosti računalnih oblaka ... 33
3.9 Rizici i nedostaci ... 35
3.10 Perspektive oblačnog računarstva ... 42
3.11 Sažetak ... 43
4 Uporabna svojstva i očekivani učinci oblačnog računarstva ... 45
4.1 Elastičnost ... 45
4.2 Optimalna nabava ... 49

4.3 Ostale ekonomske prednosti 50
4.4 Uvod u ekonomiku komercijalnog oblačnog računarstva 51
4.5 Usporedba komercijalnih oblačnih ponuda 54
4.6 Primjer izračuna cijene oblačnog najma 59
4.7 Nova korisnička očekivanja 62
4.8 KVS model i BASE zahtjevi 63
4.9 Sažetak - istraživački fokus 66
5 Mjerenje performansi aplikacija u oblaku 68
5.1 Svrha, cilj i predmet mjerenja 69
5.2 Zavisne i nezavisne varijable u istraživanom problemu 70
5.3 Nezavisne varijable 71
5.4 Zavisne varijable 72
5.5 Izbor i priprema aplikacije za mjerenje 73
5.6 Ogledna aplikacija 73
5.7 Način mjerenja i predistraživanje 76
5.8 Plan mjerenja 82
5.9 Izračun vremena odziva 93
5.10 Zaključci i preporuke o zaposjedanju resursa u oblaku 98
5.11 Osvrt na prvi dio hipoteze 2 103
5.12 Sažetak 104
6 Ekonomika oblačnog računarstva 106
6.1 Puna cijena vlasništva oblačnog centra 107
6.2 Usporedbena analiza cijene CPU vremena 114
6.3 Cijena podatkovne pohrane 126
6.4 Komentar i zaključak o pribavljanju podatkovnih resursa 133
6.5 Taksonomija i ekonomika oblačnih usluga 134
6.6 Struktura cijena usluga oblačnog računarstva 139
6.7 Razrješavanje dvojbe kupnja ili najam 141
6.8 Osvrt na drugi dio hipoteze 2 154

6.9 Sažetak 154
7 Metodika razvoja oblačnih rješenja 157
7.1 V-model 158
7.2 Životni ciklus IT proizvoda u oblaku 159
7.3 Grafički i opisni prikaz metodike KOR 161
7.4 Oblačno računarstvo i SOA 176
7.5 Osvrt na Hipotezu 1 178
7.6 Sažetak 179
8 Zaključak 180
9 Literatura 185

Popis slika

Slika 2.1: **Distribuirani i paralelni računalni sustav** 10
Slika 2.2: **Arhitektura grida** 13
Slika 2.3: **Povijesni pregled naprednog računarstva** 15
Slika 2.4: **Osnovna arhitektura računalnog oblaka** 16
Slika 2.5: **Primjer korisničkog upravljačkog web sučelja za Windows Azure** 17
Slika 2.6: **Funkcionalnosti oblaka** 17
Slika 3.1: **Shematski prikaz oblačnih usluga prema servisnoj paradigmi** 23
Slika 3.2: **Vanjski izgled i dimenzije oblačnog kontejnera** 28
Slika 3.3: **Serveri u podatkovnom centru** 29
Slika 3.4: **Serverska računala u stalku** 29
Slika 3.5: **Elektroenergetske komponente oblačnog centra** 30
Slika 3.6: **Shematski prikaz virtualne lokalne mreže** 38
Slika 3.7: **Predviđanje udjela oblačnog računarstva u ukupnoj IT potrošnji** 43
Slika 4.1: **Odnos potrebnih i raspoloživih računalnih resursa** 46
Slika 4.2: **Nabava računalnih resursa za pokriće vršnih opterećenja** 46
Slika 4.3: **Nesigurna prognoza rasta potrebe za računalnim resursima** 47
Slika 4.4: **Gubitak neposluženih korisnika (bez ostalih posljedica)** 47
Slika 4.5: **Gubitak neposluženih korisnika uz stvaranje lošeg publiciteta** 48
Slika 4.6: **CAP teorem** 64
Slika 5.1: **Arhitektura ogledne aplikacije koja se izvodi u oblaku** 76
Slika 5.2: **Koncept mjerenja vremena izvođenja ogledne aplikacije u oblaku** 79
Slika 5.3: **Raspored mjernih točaka za parcijalni višefaktorski plan pokusa tipa 2^{6-2}** 89
Slika 5.4: **Usporedba mjerenih i procijenjenih vrijednosti za T u točkama mjerenja** 96
Slika 5.5: **Zavisnost T o broju računala, za različite razine aplikacijskih zahtjeva** 99
Slika 5.6: **Zavisnost vremena T o interakcijama parametara a i q** 100
Slika 5.7: **Potrebni broj računala za razne parametre ogledne aplikacije** 101
Slika 5.8: **Međuzavisnost varijabli a, u i r za T=90 [sek] i q=50 [akcija/upit]** 102
Slika 6.1: **Struktura troškova podatkovnog centra** 112
Slika 6.2: **Odnos cijene CPU/sat kod korištenja vlastite opreme i najma u oblaku** 120
Slika 6.3: **Prihodi, troškovi i dobit za dva različita stupnja korištenja kapaciteta** η 123
Slika 6.4: **Optimalni period povrata investicije za razne** η **serverske farme** 124
Slika 6.5: **Najveće ostvarive vrijednosti za ROI u zavisnosti o** η 125
Slika 6.6: **Postupni najam diskova za osobnog korisnika je bolji od kupnje** 132
Slika 6.7: **Struktura troškova aplikacija Ap1, Ap2 i Ap3 po poslužiteljima** 152
Slika 7.1: **Životni ciklus razvoja informacijskog sustava prikazan V-dijagramom** 159
Slika 7.2: **Životni ciklus IT usluga u oblaku** prema [51] 160
Slika 7.3: **Grafički prikaz metodike KOR** 162

Popis tablica

Tablica 2-I: **osnovna usporedba oblačnih resursa** 18
Tablica 4-I: **Struktura troškova oblačnog podatkovnog centra** 52
Tablica 4-II: **Struktura cijene za EC2** 56
Tablica 4-III: **Cijene standardnih RDS, Simple DB i S3** 57
Tablica 4-IV: **Najam računala u Windows Azure usluzi** 58
Tablica 5-I: **Procjena parametara koji utječu na vrijeme izvođenja aplikacije** 71
Tablica 5-II: **Opseg variranja nezavisnih varijabli** 77
Tablica 5-III: **Vrijednosti nezavisnih varijabli i rezultati prethodnih mjerenja** 81
Tablica 5-IV: **Granice originalnih parametara mjerenja i njihove supstitucije** 85
Tablica 5-V: **Oznake parametara istraživanja i nezavisnih varijabli** 90
Tablica 5-VI: **Plan i rezultati mjerenja te izračun koeficijenata za T** 91
Tablica 5-VII: **Usporedba mjerenih i izračunatih vrijednosti za T [sek]** 95
Tablica 5-VIII: **ANOVA matematičkog modela za T_r** 97
Tablica 6-I: **Struktura troškova oblačnog podatkovnog centra** 108
Tablica 6-II: **Podaci za izračun odnosa kupnja/najam** 117
Tablica 6-III: **Cijena CPU sata računalnih resursa u vlasništvu i najmu** 119
Tablica 6-IV: **Proračun prihoda i troškova serverske farme** uz L=0,10 $/h 122
Tablica 6-V: **Parametri modela za proračun vrijednosti ΔNPV** 131
Tablica 6-VI: **Pregled ponuditelja i njihovih usluga po komponentama** 136
Tablica 6-VII: **Katalog Microsoftovih usluga** 137
Tablica 6-VIII: **Katalog Amazonovih usluga** 138
Tablica 6-IX: **Katalog Googleovih usluga** 139
Tablica 6-X: **Pregled parametara za izračun cijene** 139
Tablica 6-XI: **Formule za izračun cijene oblačnog zakupa** 140
Tablica 6-XII: **Izračun podataka za odluku kupnja-najam za tri virtualne aplikacije** 143
Tablica 6-XIII: **Podaci u slučaju kupnje računala za tri virtualne aplikacije** 145
Tablica 6-XIV: **Komponente cijene najma za $Ap1$, $Ap2$ i $Ap3$ po poslužiteljima** 151
Tablica 7-I: **Tablični prikaz metodike KOR** 174
Tablica 7-II: **Poslovni sadržaji metodike KOR** 175

Popis priloga

Prilog 5.1: Dopune u strukturi i programskom kôdu ogledne aplikacije

Prilog 5.2: Program za proračun koeficijenata za polinom T na temelju mjerenja izvedenih po parcijalnom višefaktorskom planu pokusa 26-2

Prilog 6.1: Program za proračun troškova podatkovnog centra

Prilog 6.2: Program za proračun cijene CPU vremena

Prilog 6.3: Program za proračun NPV za diskovne kapacitete

Popis kratica

Ovaj popis sadrži samo one kratice koje se općenito koriste u računarstvu, dok su specifične kratice, kao dio komercijalnih naziva za različite tehnologije i proizvode objašnjene u tekstu rada.

ACID *Atomicity, consistency, isolation, durability* – Svojstva relacijskih baza podataka.

API *Application programming interface* – Programska sučelja za korištenje funkcionalnosti jednog programa iz nekog drugog programa.

BASE *Basically available, soft state, eventual consistency* – Svojstva KVS baza podataka.

BPMN *Business Process Modeling Notation* – Industrijski standard za grafičko modeliranje i prikaz poslovnih procesa.

CAP *Consistency, availability, partition tolerance* – Svojstva podatkovne pohrane čije je proučavanje važno za odabir relacijske ili alternativne baze podataka.

CRUD *Create – Read – Update – Delete* – osnovne operacije nad bazom podataka (analogne INSERT, SELECT, UPDATE i DELETE naredbama u SQL jeziku)

EAV *Entity-Attribute-Value* – Naziv za jedan koncept nerelacijskih baza podataka, sinonimno s KVS.

ISACA *Information Systems Audit and Control Association* – Profesionalno udruženje za upravljanje informacijskih tehnologijama (IT Governance).

ISP *Internet Service Provider* – Pružatelj internetskih usluga.

ITIL *Information Technology Infrastructure Library* – Zamisli, pravila i preporuke za upravljanje informacijskim (IT) servisima i razvojem.

KVS *Key-Value-Store* – Naziv za jedan koncept nerelacijskih baza podataka.

RDBMS *Relational Database Management System* – Softver za upravljanje relacijskom bazom podataka.

REST *Representational State Transfer* – Softverska arhitektura na kojoj se temelji World Wide Web (WWW), temeljna arhitektura za izgradnju RESTful web servisa koji ne rade sa SOAP protokolom, nego komuniciraju HTTP-om i, tipično, vraćaju podatke u JSON formatu.

ROI *Return of Investment* – Povratak investiranih sredstava.

SLA *Service-Level Agreement* – Ugovor o održavanju ili usluzi.

SOA *Service-Oriented Architecture* – Pristup u projektiranju softvera i hardvera namijenjen omogućavanju i olakšavanju suradnje heterogenih aplikacija i računala.

TCO *Total Cost of Ownership* – Puna cijena vlasništva nad nekim proizvodom.

VLAN	*Virtual Local Area Network* – Privatna lokalna računalna mreža (LAN) uspostavljena preko vanjske mrežne opreme (na primjer, u vlasništvu oblačnog poslužitelja).
VPN	*Virtual Private Network* – Privatna lokalna računalna mreža (LAN) uspostavljena preko javne telekomunikacijske mreže
WS-BPEL	*Web Service - Business Process Execution Language* – Izvršni (*executable*) jezik za specifikaciju aktivnosti poslovnih procesa izvedenih u obliku web servisa.
WS-CDL	*Web Services Choreography Description Language* – Jezik temeljen na XML-u za opis poruka kojima međusobno surađuju sudionici nekog procesa.
WSCI	*Web Service Choreography Interface* – Jezik temeljen na XML-u za definiranje poruka koje web servisi međusobno razmjenjuju u okviru nekog procesa.
WSCL	*Web Services Conversation Language* – Jezik temeljen na XML-u namijenjen opisu programskih sučelja web servisa.
XML	*Extended Markup Language* – Jezik za opis podataka i dokumenata temeljen na tagovima.

1 Uvod

Ovim poglavljem bit će razmotreni čimbenici i motivacija koji su utjecali na nastanak ovoga rada te načini kako spoznaje prezentirane u njemu mogu biti od koristi znanstvenicima koji se bave ovom domenom, poslovnim stručnjacima i organizacijama (posebice malim i srednjim tvrtkama). Sadržaj poglavlja poslužit će kao uvod u oblačno računarstvo – primarnu temu ovog rada – te će se okvirno opisati postavljeni ciljevi istraživanja. Važno je unaprijed naglasiti dvije značajke ovog rada:

1. U istraživačkom fokusu su **komercijalni** računalni oblaci, a ne privatni niti akademski računalni oblaci izvedeni na sveučilištima (samostalno ili u suradnji s privatnim sektorom). To znači da neki zaključci izvedeni u ovom radu, posebice oni vezani za ekonomiku oblačnog računarstva, vrijede prije svega za komercijalne oblake.
2. Korištenje vlastitih računalnih resursa (dakle, onih u posjedu tvrtke, javne ustanove ili pojedinaca, a ne unajmljenih u komercijalnom oblaku) pretežito će se promatrati s gledišta malog i srednjeg poduzetništva čiji temeljni posao (*core business*) nije u IT domeni. Drugim riječima, uglavnom će se uspoređivati način korištenja i ekonomičnost upotrebe vlastitih računalnih resursa u odnosu na računalne resurse koje bi organizacija mogla osigurati najmom u komercijalnom računalnom oblaku. Implicitno to znači da zaključci vrijede i za privatne računalne oblake, koji se od javnih pretežno razlikuju po tome što imaju manji broj unaprijed poznatih korisnika. Problematika ulaganja u vlastiti oblačni podatkovni centar bit će ukratko prikazana u poglavlju 6.

1.1 Motivacija

Računalni oblaci (*computing clouds*) su nova tehnologija koja je konceptualno odavno zamišljena, ali tek unazad nekoliko godina ulazi u opširniju uporabu [14]. Put od zamisli do uporabivog ostvarenja odvijao se u više faza. Tijekom svake faze razvijene su neke od tehnologija na kojima se temelje računalni oblaci ili za čiji su rad računalni oblaci iznimno pogodna platforma. Zbog takvog inkrementalnog razvoja, spomenute tehnologije odnosno pojmovi koji ih opisuju – primjerice, mrežno računarstvo (*grid computing*), paralelno računarstvo (*parallel computing*), servisno orijentirana arhitektura (*service-oriented architecture* – SOA), softver kao servis (*Software as a Service* – SaaS) i drugi – često se zamjenjuju s oblačnim računarstvom (*cloud computing*) ili pak smatraju sinonimnima. Stoga smatramo da postoji potreba za ontološkom klasifikacijom tih pojmova te određivanjem taksonomskog prostora njihove primjene u svrhu razvoja suvremenih informacijskih sustava.

Dosadašnja istraživanja u području servisno orijentiranih arhitektura te raspoloživa tehnička rješenja za njihovu izvedbu odnosila su se uglavnom na:

- definiranje optimalnih protokola za razmjenu informacija među SOA servisima, kao što su REST opisan u [27, 80] i SOAP [107],
- opći razvoj složenih normi za koreografiju web servisa – WSCI [104], WS-CDL [106] i WSCL [105]) te
- na metode i tehnike SOA razvoja u okviru klasičnog informacijskog sustava, smještenog na privatnim serverima poslovnog sustava [8, 9, 26, 52, 53, 61, 70].

Smatramo da su računalni oblaci i SOA tehnološki komplementarni pojmovi u smislu da računalni oblaci predstavljaju povoljnu platformu za pokretanje i izvođenje SOA usluga. Budući da su istraživanja na tom polju relativno skromna (a ovdje su opisana u potpoglavlju 1.2), nadamo se ovim radom dati odgovarajući doprinos na polju integracije servisno orijentiranih arhitektura s konceptom računalnih oblaka.

Također, u današnje doba širenja računalnih oblaka postavlja se iznimno zanimljivo pitanje da li će nakon njihove znatnije ekspanzije i dalje postojati šira potreba za jakim osobnim računalima i privatnim serverima ili će biti dovoljna računala minimalne hardverske snage, takozvani tanki klijenti (*thin client*), čiji su tipični predstavnici bila *netbook* prijenosna računala i *nettop* stolna računala.

1.2 Dosadašnja istraživanja

Rijetko koje dosadašnje istraživanje je razmatralo servisno orijentiranu arhitekturu u okviru računalnih oblaka. Najznačajniji takav rad objavili su de Leusse i suradnici [57]. U tom radu raspravlja se o tehničkim aspektima smještaja (*deployment*) cjelokupnog servisno orijentiranog softverskog rješenja u oblak. Međutim, autori se ne bave ekonomskom isplativošću takvog poduhvata niti problematikom hibridnih arhitektura u kojima se neki servisi pokreću iz oblaka, a neki na klasičan način, s privatnih servera poslovnog sustava.

O samoj izgradnji SOA postoji opsežna literatura, primjerice autora M. Bell [9], T. Erl [26] ili E. Marks [61]. U drugim djelima, poput onih M. Havey [40] i S. White [112], projektiranje SOA rješenja je povezano sa suvremenom industrijskom normom za modeliranje poslovnih procesa – BPMN. U spomenutoj knjizi T. Erla razvijeni su neki tipični predlošci (*patterns*) za izgradnju SOA servisa, dok su u radu Y. Natisa [70] preporučeni najbolji postupci (*best practices*) pri njihovom razvoju. Međutim, nigdje nije istraživano niti posebno razmotreno projektiranje SOA za računalni oblak, niti pak su do danas za tu svrhu dopunjene norme BPMN ili WS-BPEL.

Glede tehnika izrade SOA korištenjem web servisa i *workflow* pogona (*workflow engine*) na raspolaganju je prilično opsežna tehnička literatura koja obrađuje najnovije specifične tehnologije pojedinih proizvođača. Za Microsoftove tehnologije, na primjer, među odlične takve priručnike spadaju oni što su ih objavili priznati stručnjaci A. Bahree [8] i S. Klein [53] gdje je izložen temeljit pristup izgradnje SOA servisa korištenjem .NET Windows Communication Foundation (WCF) tehnologije te knjiga koju je napisao T. Kitta [52] u kojoj se raspravlja o programiranju radnih tokova (*workflow*) korištenjem .NET Windows Workflow Foundation (WF) tehnologije.

Što se tiče povezanosti oblačnog računarstva i servisno orijentirane arhitekture, zanimljivo je primijetiti kako su komercijalne računalne oblačne platforme, primjerice Amazonov Elastic Cloud i Microsoftov Windows Azure, nastale uglavnom empirijskim putem kao način skaliranja raznih softverskih rješenja, najčešće vlastitih (*in-house*), uključivo i onih vezanih za SOA.

Općeniti pristup objedinjavanju SOA načela sa suvremenim informacijskim tehnologijama istraživali su K. Kontogiannis i suradnici [54] te Vitharana [97]. Nove paradigme razvoja i korištenja softvera kao usluge istraživali su A. Dan, R. Johnson i A. Arsanjani u radu [21], gdje autori spominju kako su takva rješenja iznimno primjenjiva u okruženju računalnih oblaka. Općenitu problematiku migracije iz postojećih, naslijeđenih (*legacy*) sustava na SOA razmatrali su L. O'Brien, P. Breber i J. Gray [70], izloživši glavnu ideju da je prijelaz na servisno orijentirane arhitekture evolucijski proces koji ne obezvređuje postojeća rješenja, nego ih potvrđuje i dopunjuje.

Metrike uspješnosti korištenja SOA razmatrao je S. Gerić [34]. O ekonomskoj isplativosti SOA koncepta raspravlja se u radu [11] kojeg su na konferenciji AMCIS 2008 objavili J. vom Brocke, O. Thomas i C. Sonnenberg, opisavši metriku za ocjenu ekonomske isplativosti SOA rješenja nazvanu Service Oriented Process Controlling (SPOC) namijenjenu mjerenju financijskih i nefinancijskih dobitaka od uvođenja SOA. Ti se učinci, međutim, ne dovode u ovisnost o infrastrukturi gdje je smješteno SOA rješenje (računalni oblak ili privatni serveri).

Općenito, može se primijetiti da je izučavanje isplativosti korištenja računalnih oblaka za sada u začecima. Poznate su, naravno, cijene koje zaračunavaju oblačni poslužitelji (*cloud providers*), ali nije sasvim jasno kada je i pod kojim uvjetima korištenje njihovih usluga isplativo krajnjemu korisniku. Da bi se pak odgovorilo na to pitanje, potrebno je imati uvid u strukturu cijene samih oblačnih usluga što je, međutim, podatak glede kojega oblačni poslužitelji zapravo baš i nisu transparentni.

1.3 Ciljevi rada

Ovaj rad ima dva glavna cilja:

1. Postaviti cjelovitu metodiku razvoja hibridnih SOA rješenja, djelomično smještenih u računalnom oblaku, a djelomično na privatnim računalima. Takva bi metodika trebala biti oslonjena na postojeće metodike razvoja klasičnih informacijskih sustava, ali i nadopunjena specifičnim metodama uklapanja oblačnih resursa. Prema dosadašnjim saznanjima takva metodika ne postoji, dok je općeniti pristup razvoja servisno orijentirane arhitekture pomoću računalnog oblaka slabo istražen.
2. Proučiti isplativost korištenja oblačnog računarstva i razviti metriku za odabir oblačnih odnosno vlastitih računalnih resursa čiji bi glavni rezultat trebao biti trošak korištenja izražen kao funkcija vremena izvođenja. Neka istraživanja na tom polju su obavljena, primjerice u radu [24, 108], koji se ekstenzivno bavi cijenom oblačnog CPU sata te [109] gdje se govori o isplativosti podatkovnog smještaja (*data storage*) u oblaku. Međutim, doprinos ovoga rada će biti objedinjavanje tih izračuna i njihovo stavljanje u ovisnost o tipu aplikacija to jest aplikacijskim svojstvima poput složenosti algoritma, učestalosti korisničkog pristupa, i tako dalje.

Očekujemo da će od ostvarenja prvoga cilja posebnu korist imati softverski arhitekti i razvojni inženjeri jer će dobiti smjernice za izgradnju i upravljanje SOA rješenjima čiji su servisi djelomično smješteni u računalnom oblaku, a djelomično na privatnim računalima. Metodika će obuhvatiti i postojeće SOA predloške (*SOA patterns*).

Ostvarenje drugoga cilja od posebnog je interesa voditeljima informatike (*Chief Information Officer* – CIO) stoga što će, koristeći spomenutu metriku, moći donositi potkrijepljene odluke o angažmanu računalnih oblaka i načinu korištenja svih dostupnih resursa, privatnih ili oblačnih, za izvođenje servisa u složenom informacijskom sustavu, posebice onog izloženog vremenski promjenljivim zahtjevima za računalnom snagom. Učinci tog korištenja bit će prikazani u funkcionalnoj ovisnost o infrastrukturi, privatnoj odnosno oblačnoj, te će biti jasno određeno kada, pod kojim uvjetima i u kojoj količini je isplativo najmiti oblačne računalne resurse odnosno kupiti vlastite.

Sa znanstvenog aspekta ovim će se radom razjasniti odnos između računalnih oblaka i drugih oblika naprednog računarstva, primjerice spomenutog mrežnog i distribuiranog, te uspostaviti taksonomija oblačnih računalnih usluga.

Hipoteze koje će se dokazivati u ovome radu glase:

H1: Metodika razvoja hibridnih SOA rješenja može se izgraditi temeljem normiranih metoda, tehnika i notacija za modeliranje, usklađivanje i opisivanje poslovnih procesa.

H2: Moguće je postaviti višedimenzionalnu ponderiranu metriku potreba za računalnim resursima SOA servisa prije njihove izgradnje te stvarnih performansi nakon njihove izgradnje. Korištenje takve metrike će povećati sigurnost pri odlučivanju o tome treba li neki servis stalno ili povremeno pokretati u računalnom oblaku.

Kroz prva tri poglavlja rada bit će istraženo i definirano oblačno računarstvo kao istraživačka domena, objašnjena geneza ove računalne paradigme te identificirane ključne tehnologije na kojima se temelji oblačno računarstvo. Dokazivanje postavljenih hipoteza bit će provedeno u opsežnijoj, drugoj polovici ovog rada, od petog do sedmog poglavlja.

1.4 Metodologija i plan istraživanja

Da bi se dokazala prva hipoteza (H1), formulirat će se poslovni zahtjevi i arhitektura takozvanih hibridnih informacijskih sustava. Pod tim pojmom podrazumijevat će se takvi informacijski sustavi čiji se neki servisi pokreću iz oblaka, a neki s vlastitih računala. Odabir jedne odnosno druge platforme za njihovo pokretanje bit će temeljen na prethodno razvijenim metrikama za određivanje količine potrebnih računalnih resursa.

Koraci razvoja hibridnog rješenja bit će povezani u cjelovitu metodiku koja će biti prikazana kao proces u kojem sudjeluju najmanje dva ključna čimbenika: korisnici i pružatelji usluga oblačnog računarstva. Razvoj ove metodike čini **projektni** dio rada, temeljen na spoznajama koje su proizašle iz **eksperimentalnog istraživanja**, provedenog mjerenjem na oglednoj aplikaciji smještenoj u izabranom komercijalnom oblaku. Ta aplikacija prilagođena je za izvođenje u oblaku u **izvedbenom** dijelu rada. Za formalni prikaz metodike koristiti će se metoda modeliranja poslovnih procesa (BPM - *Business Process Modeling*), u skladu s notacijom BPMN v.2, koja danas predstavlja industrijski standard za modeliranje, usklađivanje i opis poslovnih procesa. Smatramo da će time biti dokazana i prva hipoteza.

Da bi se dokazala druga hipoteza (H2), bit će potrebno povezati dva prividno različita znanstvena i stručna područja – tehničko i ekonomsko.

U tehničkom dijelu posla bit će odabrano i predstavljeno ogledno softversko rješenje (aplikacija) iz poslovno interesantne domene. Također će biti objašnjeni postupci kako tu aplikaciju treba pripremiti na nekoj od raspoloživih tehnologija da bi se mogla izvoditi na resursima u oblaku. Ovdje su korištene Microsoftove tehnologije .NET i Windows Azure. Ovo ogledno softversko rješenje će se koristiti za mjerenje performansi obavljanja poslova u

komercijalnom oblaku (temeljenom na virtualnim računalima) te određivanje signifikantnih nezavisnih varijabli o kojima zavise te performanse. Ispitivane nezavisne varijable bit će: snaga i broj računala na kojima se pokreće korisničko rješenje, složenost algoritma aplikacije, opseg mrežnog prometa između aplikacije koja se izvodi u oblaku i korisnika (količina podataka poslanih u oblak – *upload* i količina podataka preuzetih iz oblaka – *download*), veličina baze podataka s kojima radi aplikacija te broj CRUD (*create, read, update, delete*) akcija nad tom bazom.

Varijable ili parametri koji opisuju aplikaciju utječu na količinu računalnih resursa potrebnih za izvođenje aplikacije, bez obzira jesu li ti resursi unajmljeni u oblaku ili raspoloživi na vlastitim računalima. Količina tih resursa utječe pak na aplikacijske performanse. Najvažnija performansa je svakako vrijeme izvođenja aplikacije u oblaku (a posljedično tome i troškovi unajmljivanja oblačnih resursa potrebnih za njezino izvođenje) pa će se to vrijeme u ovom istraživanju tretirati kao zavisna varijabla.

Zaključci izvedeni na temelju obrade rezultata mjerenja će se uopćiti za sve aplikacije istog tipa. Iz toga slijedi da će se ispitivanjem i matematičkim modeliranjem odnosa navedenih varijabli zapravo postaviti metrika za određivanje količine potrebnih računalnih resursa.

Takva metrika još uvijek, međutim, nije dovoljna za razrješenje dvojbe da li i kada računalne resurse nabaviti te koristiti na klasičan način ili ih unajmiti u komercijalnom oblaku. Ta dvojba riješit će se ispitivanjem strukture cijene (*pricing*) računalnih resursa, radi objektivne procjene troškova izvođenja iste aplikacije u oblaku u odnosu na vlastite računalne i podatkovne resurse. To ispitivanje obuhvatit će elemente klasičnih financijskih modela koji razmatraju opcije kupnje ili najma bilo kojega osnovnog sredstva, ali i dodatne elemente specifične za računarstvo, poput Mooreovog zakona. Ispitivanjem strukture cijene izvest će se egzaktne formule izračuna troškova računalnih resursa – onih najmljenih u komercijalnom oblaku te vlastitih koji se koriste na klasičan način (a ne u sklopu privatnog oblaka) – kao funkcije više nezavisnih varijabli poput vremena izvođenja aplikacije, broja angažiranih računala, broja procesorskih jezgri po računalu, godišnjih fiksnih troškova pogona podatkovnog centra, stupnja iskorištenosti računala, i drugih.

Nakon što budu istražene tehničke međuzavisnosti varijabli koje određuju ponašanje neke aplikacije određenog tipa tijekom izvođenja, moći će se izračunati troškovi pokretanja vlastitog rješenja. To će biti moguće ako se tehnički parametri specifične aplikacije budu mogli uvrstiti kao varijable u formule za izračun troškova korištenja vlastitih odnosno oblačnih računalnih resursa. Tako izračunati troškovi za obje opcije – unajmljivanje resursa u komercijalnom oblaku ili korištenje vlastitih resursa na klasičan način – dati će vrijednosti čijom usporedbom će biti moguće donijeti odluku treba li za izvođenje neke specifične

aplikacije unajmiti računalne i pohrambene resurse u oblaku ili je bolje te potrebe realizirati kupnjom vlastitih resursa. U slučaju izbora najma potencijalni korisnik oblačnog računarstva moći će izabrati takvog pružatelja usluga (*cloud provider*) koji je ekonomski najpovoljniji za traženu vrstu i obim usluga.

Cijeli je ovaj postupak analize, izbora najpovoljnijeg rješenja i postavljanje neke specifične aplikacije u oblak formaliziran u sedmom poglavlju kao metodika za razvoj oblačnih rješenja, čime će biti dokazana druga hipoteza ovog rada.

2 Nastanak računalnih oblaka

Posljednjih desetak godina je pojam oblačnog računarstva (*cloud computing*) postao nova marketinška krilatica informatičke industrije, *buzzword*, nadovezavši se i pomiješavši pri tom s mnogim drugim, često starijim pojmovima koji su označavali tehnologije koje imaju manje ili više dodirnih točaka s današnjim računalnim oblacima [48, 102].

U sklopu rasprava i prezentacija oblačnog računarstva vrlo često se taj pojam miješa s drugima [5], poput:

- distribuirano računarstvo (*distributed computing*)
- paralelno računarstvo (*parallel computing*)
- uslužno računarstvo (*utility computing*)
- mrežno računarstvo (*grid computing*)
- virtualizacija (*virtualization*).

Naročito je prisutno poistovjećivanje sa sintagmama:

- *Software as a Service* (SaaS)
- *Infrastructure as a Service* (IaaS)
- *Platform as a Service* (PaaS)
- i drugim "*…as a Service*" pojmovima.

Stoga je razjašnjavanje navedenih pojmova te precizno definiranje oblačnog računarstva važno za iznalaženje prostora za njegovu uspješnu primjenu jer je *"ponekad uopće teško reći da li neka usluga doista spada u oblačno računarstvo ili se tek radi o staroj, samo preimenovanoj usluzi."* [87].

2.1 Distribuirano računarstvo

Pojam distribuiranog računarstva izvorno se odnosio na skup umreženih, ali neovisnih računala s mogućnošću međusobne komunikacije porukama. Ta su računala mogla biti programirana tako da odrađuju jednu ili više zajedničkih zadaća. Distribuirani računalni sustavi imaju tri tipične značajke koje se sreću u današnjem računarstvu općenito, pa tako i u računalnim oblacima [35]:

- Čitav distribuirani sustav je izveden tako da je otporan na kvarove svojih komponenti, pojedinačnih računala, u smislu da takvi kvarovi ne uzrokuju zastoj u radu, nego tek privremeno smanjenje ukupnih performansi obavljanja poslova, proporcionalno broju

pokvarenih komponenti. Drugim riječima, distribuirani sustav nema jednu točku zastoja (*single point of failure*).

- Komponente distribuiranog računalnog sustava mogu biti različitih vrsta, to jest svako računalo u distribuiranom sustavu može biti temeljeno na drukčijem hardveru i koristiti različiti softver.
- Sastav distribuiranog sustava – količina i vrsta u njega uključenih računala – ne mora biti unaprijed poznat i može se mijenjati tijekom vremena.
- Pojedinačne komponente distribuiranog računalnog sustava su, načelno, specijalizirane za određene poslove i nemaju cjelovitu sliku o sustavu kojega su dio, niti im je ona potrebna za obavljanje njihovih zadaća.

Distribuirani računalni sustavi ostvareni su još 1970-ih godina kada su razvijene dvije ključne tehnologije potrebne za njihovo funkcioniranje:

1. operacijski sustavi s mogućnošću komuniciranja s vanjskim svijetom putem poruka (*message queue*)
2. mrežni standardi poput Etherneta, nužni za uspješno umrežavanje računala i razmjenu poruka među njima.

Primjena distribuiranih sustava je vrlo opsežna [60]:

- komunikacije – fiksne i mobilne mreže te Internet
- mrežni servisi poput World Wide Weba ili p2p mreža
- mrežne igre i internetske zajednice (*on-line communities*)
- mrežna pohrana podataka i internetske tražilice
- nadzorno-upravljački sustavi, od automobilske elektronike, preko industrijskih traka, do "let po žici" (*fly by wire*) zrakoplovnih sustava
- zahtjevni znanstveni proračuni i istraživanja.

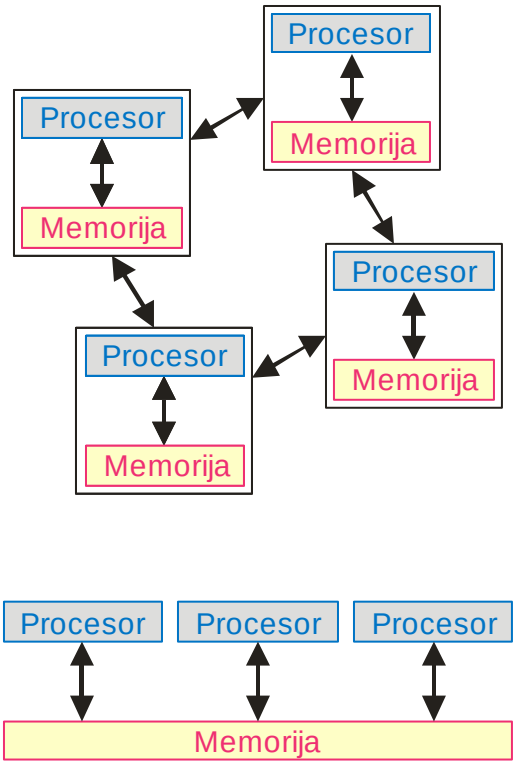
Može se reći da su načela distribuiranog računarstva ugrađena u temelje mnogih primjena računala, pa tako i u računalnim oblacima. Čak su i današnja osobna računala, posebice ona s višejezgrenim procesorima, dovoljno moćna da mogu simulirati distribuirani sustav. Drugim riječima, **računalni oblaci su distribuirani sustavi, no nije svaki distribuirani sustav istovremeno i računalni oblak**.

2.2 Paralelno računarstvo

Pojam paralelnog računarstva označava uporabu distribuiranog sustava u svrhu paralelnog, višeračunalnog rješavanja nekog računalno intenzivnog problema. Kako je naznačeno u prethodnom odlomku, distribuirani sustav može biti izveden i kao jedno fizičko računalo, pa tako i paralelni sustav može biti sadržan u jednoj hardverskoj komponenti. Primarna razlika paralelnog računalnog sustava u odnosu na distribuirani je u razini integracije njegovih komponenti:

- u distribuiranom računalnom sustavu, svaka komponenta ima vlastiti procesor i radnu memoriju, a komunikacija između pojedinih komponenata se obavlja izmjenom poruka;
- u paralelnom sustavu sve komponente imaju pristup zajedničkoj, dijeljenoj memoriji (*shared memory*).

Razlika između distribuiranog i paralelnog računalnog sustava ilustrirana je slikom 2.1.



Slika 2.1: ***Distribuirani i paralelni računalni sustav***

Može se reći da je paralelni računalni sustav samo uže vezana (*tightly coupled*) izvedba distribuiranog sustava [77]. Iz toga slijedi zaključak kako **računalni oblaci mogu biti korišteni kao platforme za paralelno računarstvo, no nije svaka takva platforma računalni oblak**.

2.3 Uslužno računarstvo

Pojam uslužnog računarstva označava korištenje i naplatu računalnih resursa po utrošku, kao komunalnih potrošnih resursa poput struje, vode, telefonije ili plina. U takvom modelu korištenja računarstva nisu potrebna početna kapitalna ulaganja jer se računalni resursi unajmljuju prema potrebi.

Ideja uslužnog računarstva nije nova – prvi ju je iznio John McCarthy 1961. godine, a nedugo zatim počeo primjenjivati IBM u svojim *mainframe* računalima. Pravo širenje takvog oblika računarstva počelo je upravo početkom 1990-ih godina s pojavom računalnih gridova.

Uslužno računarstvo je primarni **model korištenja** gridova i računalnih oblaka. Temeljna ideja je ponuditi računalne resurse sa sljedećim svojstvima [46]:

- raspoloživost (*availability*) koja mora biti jednaka ili veća nego što se može postići u korporativnim okruženjima
- mogućnost aplikacijske integracije – računalni resursi moraju biti tako organizirani i izvedeni da je na njima moguće izraditi i pogoniti složene informacijske sustave te integrirati postojeća rješenja, uz korištenje standardnih metoda, tehnika i protokola (HTTP, XML, i slično)
- fleksibilnost – plaćanje po korištenju, uz što manje administriranja i formalnosti.

2.4 Mrežno računarstvo

U funkcionalnom pogledu, mrežno računarstvo, u daljnjem tekstu **grid**, označava distribuirani računalni sustav čija je namjena ujediniti računala, mreže, baze podataka i druge resurse, poput znanstvene opreme, u jedinstveni sustav za rad na rješavanju određenih problema te postizanje željenog cilja. Očekuje se pri tome da će biti prevladana organizacijska ograničenja nad pojedinim komponentama takvog sustava [13]. Grid ima slijedeće karakteristike [44, 55, 62]:

1. Komponente grida (računala, diskovi i ostali hardver) nisu podvrgnute centraliziranom nadzoru s jednog kontrolnog mjesta, što znači da svaki logički element grida – čvor (*node*) – ima vlastito upravljanje resursima (*resource management*) te ne postoje vrhovni upravljački mehanizmi s takvim zadaćama.
2. Za komunikaciju unutar grida te grida kao cjeline prema okruženju koriste se standardni, otvoreni i općeprihvaćeni protokoli.
3. Grid nudi mogućnost obavljanja netrivijalnih računalnih zadaća.

Potrebno je razjasniti ta svojstva, napose stavke 1 i 3:

- U stavci 1, pod *"centraliziranim nadzorom s jednog kontrolnog mjesta"* se podrazumijeva takvo upravljanje distribuiranim sustavom gdje postoji centralizirani server koji ima pod nadzorom cijeli sustav te zna sve o svim korisničkim zahtjevima prema tom sustavu te upravlja svim njegovim komponentama. Drugim riječima, to svojstvo se **ne** odnosi na pitanja nadzorne organizacije ili organizacija, to jest problematiku tko upravlja sustavom ili mu je vlasnik. Distribuirani sustav može imati jednu takvu nadzornu organizaciju i svejedno se kvalificirati kao grid. Ako pak su komponente distribuiranog sustava ipak podvrgnute centraliziranom nadzoru s jednog kontrolnog mjesta, onda govorimo o **računalnoj skupini** *(computer cluster*), a ne o gridu.
- U stavci 3, pod netrivijalnim računalnim zadaćama se podrazumijevaju računalno intenzivni i opsežni znanstveni, matematički, istraživački problemi, ali i komercijalni problemi poput organskokemijskih istraživanja strukture, interakcije i ponašanja bjelančevina (*protein folding*), ekonomskih i financijskih simuliranja i modeliranja makroekonomskih sustava, prognoze vremena i klime, istraživanja potresa pa čak i takve zadaće poput obrade podataka s radio-teleskopa u svrhu možebitnog pronalaženja vanzemaljskih civilizacija.

Računalni gridovi pojavili su se tijekom 90-ih godina, upravo radi rješavanja složenih, velikih i računalno intenzivnih problema koji su do tada bili rješavani korištenjem superračunala. Budući da su takva superračunala bila iznimno skupa, mrežnim računarstvom se riješio problem cijene na način da su korišteni "obični" računalni resursi raznih ustanova združenih gridom.

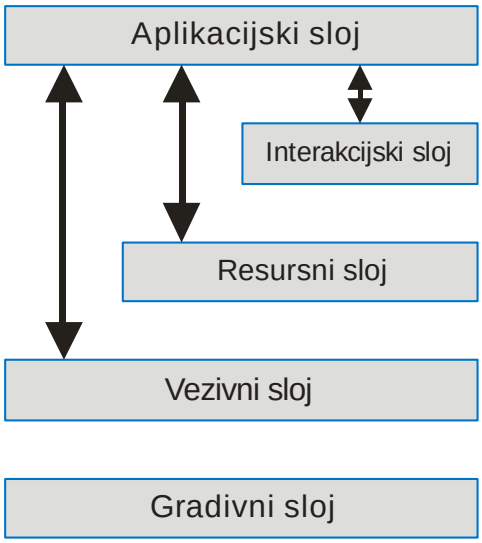
Može se reći da je grid virtualno superračunalo (*supercomputer*) **namijenjeno rješavanju specifičnih zadataka**, no od klasičnih superračunala kao što je IBM Roadrunner razlikuje ga slabija povezanost komponenti, njihova hardverska i softverska heterogenost te često i fizička, geografska raspršenost.

Važno je napomenuti da aplikacije namijenjene pokretanju u gridu moraju biti programirane specifično za takvu radnu okolinu te, načelno, nisu lako prenosive (*portable*) između raznih gridova, kao niti između grida i drugih platformi poput superračunala ili osobnih računala [55, 62].

Upravljanje gridom provodi se putem virtualnih organizacija, logičkih ustrojstava čiji računalni resursi mogu biti dijeljeni i upravljani kao da su unutar jedne te iste organizacije. Sam grid ima slojevitu arhitekturu:

- Gradivni sloj (*Fabric Layer*) – odnosi se na hardversku i softversku infrastrukturu poput podatkovne pohrane (*storage*), repozitorija programskog koda, umrežene znanstvene instrumente, i sličnog.
- Vezivni sloj (*Connectivity Layer*) – obuhvaća sigurnosnu infrastrukturu te komunikacijske protokole.
- Resursni sloj (*Resource Layer*) – definira protokole za objavu, otkrivanje, praćenje i naplatu računalnih operacija na resursima grida te nudi servise za pristup i upravljanje samim resursima.
- Interakcijski sloj (*Collective Layer*) – omogućuje interakcije među raznim vrstama resursa u gridu, primjerice konkretne aplikacije i sustave za raspoređivanje (*scheduling*) aplikacijskih servisa. Takve se zadaće provode izlaganjem odgovarajućih API-ja sloju u obliku portala, razvojnih alata i sučelja za grid servise.
- Aplikacijski sloj (*Application Layer*) – obuhvaća korisničke aplikacije u gridu.

Arhitektura grida prikazana je slikom 2.2.



Slika 2.2: ***Arhitektura grida*** *(prema [32])*

Danas se u svijetu koristi više gridova. U Sjedinjenim Državama najpoznatiji su Teragrid [62], financiran od Nacionalne znanstvene zaklade (*National Science Foundation*), te Grid3, zajednički poduhvat Amerike i Južne Koreje, korišten u istraživanjima visokoenergetske fizike, astronomiji i biologiji. Najznačajniji europski grid je CERN-ov, postavljen s ciljem uobličavanja istraživačke mreže za fundamentalnu fiziku, bioinformatiku i druge računalno intenzivne potrebe. Dalekoistočne zemlje i Australija imaju vlastite slične projekte ili pak sudjeluju u postojećim gridovima.

2.5 Virtualizacija

Općenito rečeno, virtualizacija je postupak odvajanja softverske usluge od hardvera koji omogućava tu uslugu [98]. Softverska usluga može biti pojedini program ili čitav operacijski sustav, dok razdvajanje softverske usluge od hardvera podrazumijeva mogućnost premještanja softverske usluge s jedne hardverske platforme (računala) na drugu bez ometanja ili prekida rada i bez dodatne prilagodbe programskog kôda softverske usluge. Virtualizacija je pobliže opisana u poglavlju 3.1 jer je ta tehnologija jedan od kamena temeljaca oblačnog računarstva.

U praksi, virtualizacija omogućuje pokretanje više virtualnih računala, svakog pod vlastitim operacijskim sustavom, na jednom fizičkom računalu. Interesantna je za mnoge informatičke djelatnosti, naročito one vezane za razvoj i testiranje softvera. Virtualizacija je posebice važna za funkcioniranje računalnih oblaka stoga što omogućuje jednostavno kopiranje virtualnih računala s jednog fizičkog računala na drugo, na način koji nije mnogo složeniji od kopiranja datoteka, te pokretanje tako kopiranih virtualnih računala na novom fizičkom računalu bez značajnijeg dodatnog konfiguriranja. Na taj način može se lako povećati iskorištenost procesorskih i memorijskih resursa fizičkog računala i posljedično smanjiti potreba za količinom fizičkih servera, troškova njihovog smještaja, hlađenja i potrošnje električne energije [17].

Virtualizacija se sve učestalije koristi u tvrtkama koje imaju velike računalne resurse [82] radi ekonomičnijeg upravljanja (*consolidation*) tim resursima i njihovog boljeg iskorištenja [64]. **Iako je virtualizacija nezaobilazan čimbenik oblačnog računarstva, nije svaka skupina virtualiziranih računalnih resursa ujedno i računalni oblak**.

2.6 Oblačno računarstvo – sinteza napretka

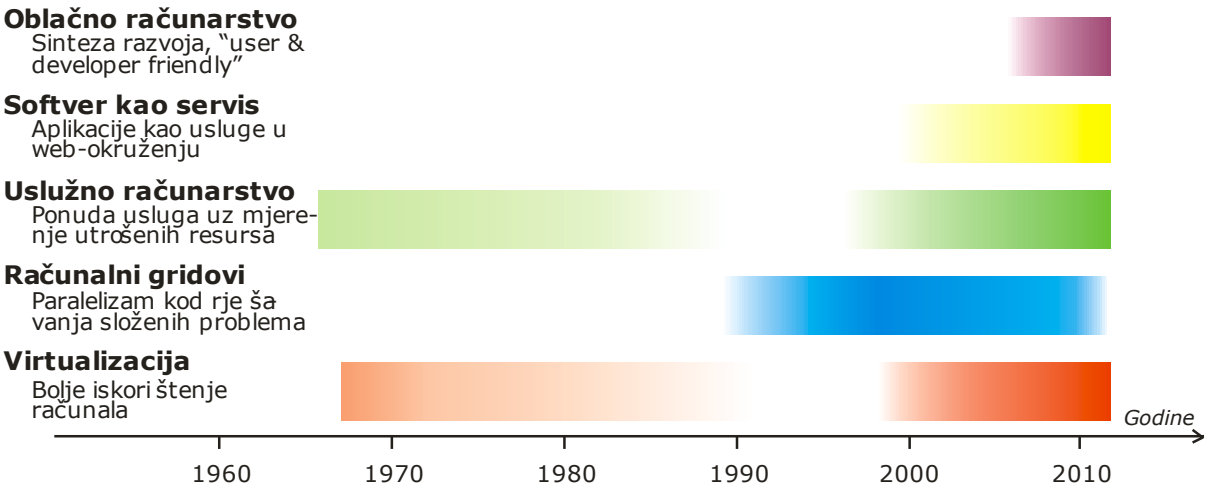
Mrežno i oblačno računarstvo temelje se na istim tehničkim i funkcionalnim načelima [17, 32, 45, 55, 62], a razlikuju se u korištenju od strane vanjskog korisnika, u načinima kako je to korištenje omogućeno te scenarijima primjene samog sustava. Kada govorimo o *'načinima kako je korištenje sustava omogućeno vanjskim korisnicima'*, dolazimo na spomenute, '*...as a Service*' sintagme koje su tipična odlika oblačnog računarstva i ključni element razlike grida i računalnog oblaka. Te se razlike očituju u slijedećim svojstvima:

- Gridovi se pretežno koriste za rješavanje specifičnih problema za čije je rješavanje potreban znatni paralelizam [79] ili visokoperformansno računarstvo (*high performance computing*) [17]. Oblaci su pak višenamjenski računalni sustavi prilagođeni pružanju raznih servisa. Stoga, prema nekim analizama [32, 62], postoji

tendencija učestalijeg korištenja gridova u akademskom svijetu, a oblaka u korporativnom.

- Gridovi su redovito složeni od manjih sustava u vlasništvu raznih organizacija, dok su oblaci obično u vlasništvu jedne organizacije (bez obzira radi li se o javnom oblaku kojeg ta organizacija potom iznajmljuje drugim korisnicima ili pak o privatnom oblaku za vlastite potrebe).
- Korištenje komercijalnih oblaka se temelji isključivo na plaćanju prema obimu korištenja (*pay per use*). Korištenje gridova (ali i akademskih oblaka) može se obavljati i s drugim oblicima naplate, primjerice ustupanjem vlastitih IT resursa u grid.
- Aplikacije u gridu moraju biti specifično skrojene, dok oblak pruža standardne softverske platforme za softverski razvoj. Stoga oblak podržava i uobičajene scenarije uporabe poput tradicionalnih višeslojnih web aplikacija.

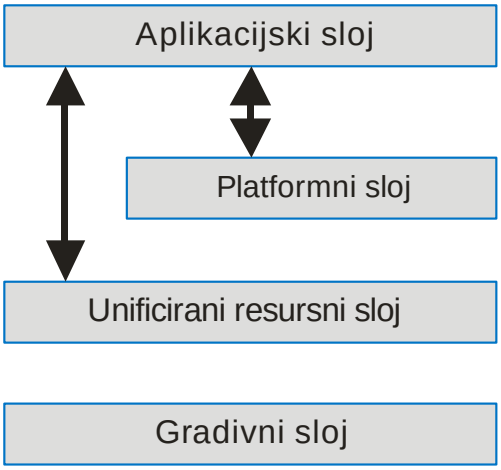
Životni ciklus od početaka, distribuiranog računarstva, preko gridova do računalnih oblaka može se prikazati slikom 2.3. Vodoravni stupci predstavljaju vremenski raspon korištenja pojedinih tehnologija, a intenzitet boje označava učestalost i raširenost korištenja.



Slika 2.3: ***Povijesni pregled naprednog računarstva***

Današnji stupanj razvoja, prikazan slikom 2.3, su računalni oblaci. Oni su zapravo razvijeni kako bi se zaobišle neke nepogodnosti uočene pri korištenju grida. To je, u prvom redu, složenost razvoja i korištenja softvera, čemu je uzrok prilično slabo apstrahiranje hardvera i operacijskog sustava grida te nestandardiziranost programskih sučelja (API) i razvojno-upravljačkih alata i metoda. U drugom redu, problem grida jest otežano simuliranje klasičnih radnih okruženja poput običnih višeslojnih aplikacija

Drugim riječima, programska sučelja u gridu su previše niske razine [62] – i upravo tu problematiku rješavaju računalni oblaci koji, u usporedbi s gridom, imaju daleko temeljitije riješenu apstrakciju hardverskih i softverskih resursa. To je vidljivo na slici 2.4 koja prikazuje opću arhitekturu računalnog oblaka.

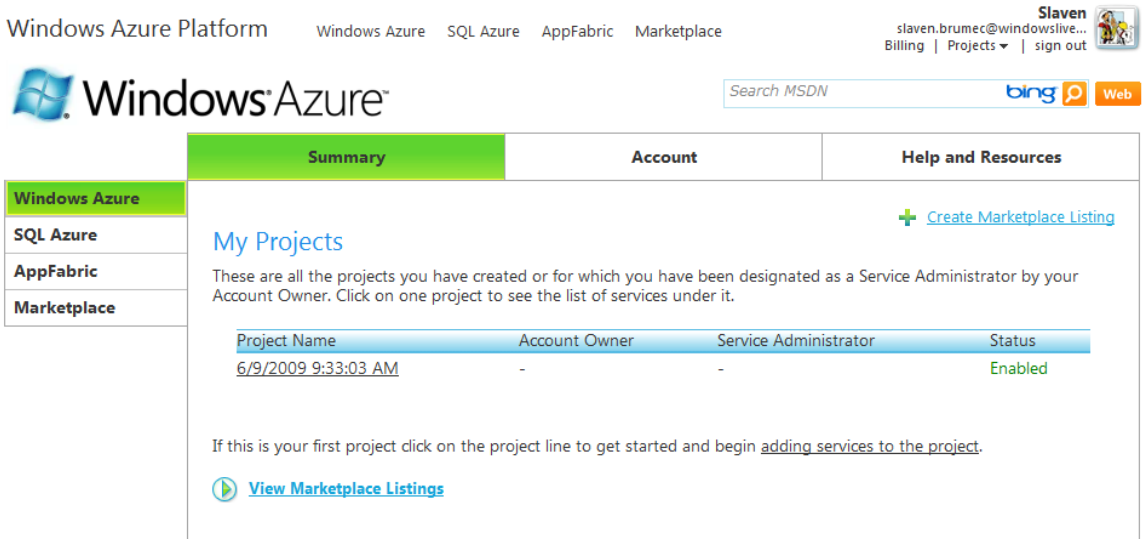


Slika 2.4: ***Osnovna arhitektura računalnog oblaka*** *(prema [32])*

Neke komponente oblaka su nazivom i funkcionalnošću jednake onima u gridu: gradivni sloj (*Fabric Layer*) i aplikacijski sloj (*Application Layer*). Primarna razlika oblaka spram grida je u slijedeća dva sloja:

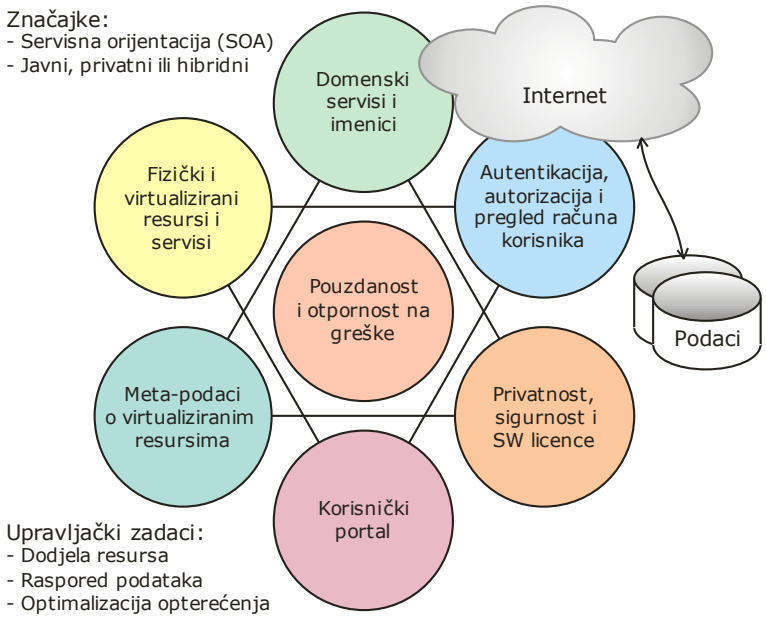
- Platformni sloj (*Platform Layer*) zamjenjuje resursni i interakcijski sloj na način da nudi standardizirane upravljačke i razvojne alate. Te se funkcionalnosti tipično izlažu u sklopu web portala za razvojne inženjere i upravitelje računalnih resursa te kroz dodatke za uobičajene razvojne alate, poput "oblačnih" alata za Visual Studio i SDK (*Software Development Kit* – razvojni alati) za .NET platformu. Primjer korisničkog upravljačkog sučelja za platformni sloj Windows Azure oblaka prikazan je slikom 2.5 (stanje u listopadu 2010., novo sučelje se može naći na http://windows.azure.com).
- Unificirani resursni sloj (*Unified Resource Layer*) zamjenjuje vezivni sloj na način da umjesto programskih sučelja niske razine sadrži apstrahirane softverske i hardverske resurse poput čitavih virtualnih računala i mrežnih diskova. Taj sloj, zapravo, čine virtualna računala pod nekim uobičajenim operacijskim sustavom.

Primjer početnog korisničkog sučelja za upravljanje poslovima u tipičnom oblaku prikazan je na slici 2.5, a odnosi se na Windows Azure.



Slika 2.5: ***Primjer korisničkog upravljačkog web sučelja za Windows Azure***

Osnovni računalni resursi u oblaku su tipizirani, prije svega zbog zahtjeva ciljanih grupa korporativnih korisnika, kao što su: višestruka namjena, skalabilnost, rad s univerzalnim web aplikacijama, korištenje resursa u skladu s potrebama i plaćanje po korištenju. Ilustracija glavnih funkcionalnosti koje računalni sustav mora ispunjavati, da bi mogao biti smatran oblakom, prikazana je slikom 2.6. Detaljnija definicija oblaka i opis njegovih funkcionalnosti prezentirani su u poglavlju 3.



*Slika 2.6:****Funkcionalnosti oblaka*** *(prema [101])*

Osnovne karakteristike unificiranog resursnog sloja u trenutku pisanja ovoga rada (2010 - 2011) prikazane su tablicom 2-I za računalne oblake dva značajna pružatelja usluga (*provider*), Microsoft i Amazon, s komercijalnim imenima Windows Azure odnosno Amazon Elastic Cloud – EC2. Tablica je orijentacijska, namijenjena stjecanju uvida u temeljne

cjenovne i hardverske karakteristike dvije tipične komercijalne oblačne ponude. Radi prikladnije usporedbe u tablici je prikazan samo dio Amazonove ponude, temeljen na virtualnim računalima pod Windows operacijskim sustavima.

Tablica 2-I: ***osnovna usporedba oblačnih resursa*** *(prema [2, 68, 90])*

	Proc. jezgre	*RAM [GB]*	*Platforma [bit]*	*OS [Win Server]*	*Satnica [$/h]*
	Windows Azure				
mala instanca	1	1,7	64	2008 R2	0,12
srednja instanca	2	3,5	64	2008 R2	0,24
velika instanca	4	7	64	2008 R2	0,48
XXL instanca	8	15	64	2008 R2	0,96
	Amazon Elastic Cloud				
mala instanca	1	1,7	32	2003	0,12
velika instanca	4	7,5	64	2003	0,48
XXL instanca	8	15	64	2003	0,96

Ova okvirna usporedba ne pokriva sve funkcionalne i uporabne mogućnosti oba oblaka niti objašnjava detalje koji se odnose na cijene korištenja računalnih resursa (čija je osnovna jedinica [$/h]), nego služi isključivo u orijentacijske svrhe. Te funkcionalne i uporabne mogućnosti uključuju, uz angažirane računalne kapacitete, još i opseg pohranjenih podataka, načine podatkovne pohrane (relacijski ili neki drugi), broj podatkovnih transakcija, opseg prometa između oblaka i vanjskog svijeta i drugo. Ta tematika bit će detaljno obrađena u sljedećim poglavljima ovog rada.

Nakon ove kratke komparativne analize dviju sličnih tehnologija, grida i oblaka, a prije izricanja jednoznačne definicije oblaka, potrebno je razjasniti neke temeljne tehnologije računalnih oblaka, što će biti učinjeno u idućem poglavlju.

2.7 Sažetak

U ovom poglavlju objašnjeni su pojmovi distribuiranog računarstva (*distributed computing*), paralelnog računarstva (*parallel computing*), uslužnog računarstva (*utility computing*) te mrežnog računarstva poznatijeg pod nazivom *computing grid*. Razjašnjen je odnos ovih tehnoloških koncepata prema računalnim oblacima. Pokazano je kako su sve opisane tehnologije s jedne strane preteča oblačnog računarstva, dok s druge strane ujedno spadaju u svojstva kojima se računalni oblaci opisuju. Također su razmotrena osnovna funkcionalna i uporabna svojstva računalnih oblaka, koncipiranih tako da udovolje zahtjevima suvremenih korisnika. Polazeći od ovih temeljnih spoznaja moguće je detaljnije istražiti unutrašnji ustroj, funkcionalnost i uporabu računalnih oblaka.

3 Opis oblačnog računarstva

Za upoznavanje svojstava računalnih oblaka i njihovo ispravno definiranje potrebno je detaljnije razmotriti najvažnije tehnologije na kojima se oblačno računarstvo temelji, a to su hardverska virtualizacija (*virtualization*) i servisna paradigma kao način korištenja ugrađenih informacijsko-komunikacijskih tehnologija. Te tehnologije nisu specifične za oblake (iako se često poistovjećuju s oblacima) i zapravo se koriste u mnogim područjima informatike i računarstva, no toliko su važne za funkcioniranje računalnih oblaka da bi oni bez njih bili nezamislivi.

3.1 Virtualizacija

Virtualizacija je postupak kojim se od korisnika skriva pravi hardver, softver i fizičke karakteristike računala te mu se daje na uporabu virtualno računalo čiji se rad **simulira** na jednom ili više skrivenih, stvarnih, fizičkih računala [17, 64, 98]. Stupanj tog skrivanja varira, no od 2005. godine su procesori i računala dovoljno jaki da je izvediva potpuna virtualizacija (*full virtualization*) to jest moguće je potpuno skrivanje cjelokupnog fizičkog računala od korisnika. Razlozi za primjenu virtualizacije mogu biti slijedeći [7]:

- Izvođenje programa nekompatibilnih s hardverom ili operacijskim sustavom fizičkog računala. Virtualno računalo koje simulira traženi hardver ili operacijski sustav može biti konfigurirano u tu svrhu.
- Testiranja novih korisničkih programa i sistemskog softvera, koja se mogu provoditi na virtualnom računalu te na taj način spriječiti moguće negativne efekte novoga softvera na fizičkom računalu.
- Bolje iskorištenje hardverskih resursa (*hardware utilization*). Računala najčešće rade s daleko manjim procesorskim i memorijskim iskorištenjem od maksimuma. Pokretanjem **više virtualnih računala na jednom fizičkom** daleko bolje se iskorištavaju računalni resursi, to jest postiže se veće iskorištenje fizičkoga računala.
- Lakše upravljanje računalnim konfiguracijama. Za mnoge namjene, poput obrazovnih (računalni laboratoriji), razvojnih (kada su razvojna okruženja stara ili bi nepoželjno utjecala na fizičko računalo) i produkcijskih (brzo umnožavanje željenih računala) korisno je i poželjno moći seliti virtualno računalo s jednog na drugi skup fizičkih računala, baš kao da se radi o kopiranju običnih datoteka. Takvo baratanje virtualnim računalima ključno je za funkcionalnu elastičnost, što je bitno svojstvo oblaka opisano kasnije u ovome radu.

Zbog svega toga, **virtualna računala postala su kamen-temeljac oblačnog računarstva** [15, 17]. Ona su temeljna mjerna jedinica za najam oblačnih resursa. Drugim riječima, pri angažmanu tih resursa obavezno se određuje broj (količina) potrebnih virtualnih računala. Upotreba tehnologije virtualizacije daje više pozitivnih učinaka:

- Bitno je olakšan i ubrzan oporavak od greški (*fault tolerance*) stoga što se virtualna računala mogu čuvati (*back-up*) kao obična datoteka – često zvana slikom (*image*) virtualnog računala – i stavljati u pogon postupkom ne mnogo zamršenijim od kopiranja datoteke.
- Fizičko smanjenje prostora za smještaj računala je važna posljedica boljega iskorištenja hardverskih resursa to jest činjenice da jedno fizičko računalo može pokretati više virtualnih.

Fizička platforma (računalo, moguće i operacijski sustav) na kojoj se pokreću virtualna računala moraju imati dogovarajući softver za upravljanje virtualnim računalima to jest za dinamičku raspodjelu fizičkih resursa među njima. Takav se softver zove nadglednik (*hypervisor*) [7, 96]. Nadglednik može biti izveden kao:

- običan korisnički program koji zahtijeva standardni operacijski sustav za svoj rad (na primjer sada već stari Microsoft Virtual PC koji je zahtijevao Windows ili višeplatformni VMware Server) ili
- kontrolni program nalik operacijskom sustavu koji se pokreće izravno na hardveru matičnog računala, poput VMware ESX.

Računalni oblaci također imaju nadglednički softver.

S obzirom na hardverske i softverske preduvjete izvođenja i tehniku izvedbe moguće je razlikovati tri temeljna oblika virtualizacije: potpunu virtualizaciju, hardverski potpomognutu virtualizaciju i paravirtualizaciju:

- Potpuna virtualizacija omogućuje posve neometanu seobu slika virtualnih računala s jednog fizičkog računala na drugo stoga što se virtualno računalo pokreće na fizičkom kroz softversku simulaciju. U takvom obliku virtualizacije, operacijski sustav virtualnog računala funkcionira kao da se nalazi na fizičkom računalu (*bare metal*) to jest "ne zna" da je pokrenut u okviru virtualnog računala, niti mu treba posebna prilagodba za takvo pokretanje.
- Hardverski potpomognuta virtualizacija (*hardware assisted virtualization*) uporabno je slična potpunoj virtualizaciji u smislu da omogućuje neometanu selidbu i pokretanje virtualnih računala pod standardnim izdanjem operacijskog sustava. Za pokretanje virtualnog računala dodatno se oslanja na hardversku podršku virtualizaciji, ugrađenu

u procesor. Takva hardverska podrška (koja se od otprilike 2005. godine ugrađuje u većinu Intelovih i AMD-ovih procesora) omogućuje nešto bolje performanse virtualiziranog računala upravo zbog sklopovske izvedbe nekih važnih virtualizacijskih funkcionalnosti poput binarnog prevođenja (*binary translation*). Izvedba tih sklopovskih rješenja nije u fokusu ovoga rada te neće biti dalje razmatrana.

- Paravirtualizacija (*paravirtualization*) se razlikuje od oba prethodna modela virtualizacije po tome što virtualna računala moraju biti pokretana pod modificiranom verzijom nekog standardnog operacijskog sustava, koja je specijalizirana za pokretanje virtualnog računala. Paravirtualizacija može donijeti zamjetne performansne prednosti [98], no prenosivost (*portability*) tako virtualiziranih računala je slaba stoga što jezgra (*kernel*) njihovog operacijskog sustava mora biti izmijenjena, što pak može uvećati troškove za posebnu potporu i održavanje.

Može se zaključiti da hardverska virtualizacija nije isključivo svojstvo računalnih oblaka jer je izvediva čak i na malim osobnim računalima, no bez nje komercijalno oblačno računarstvo praktički ne bi bilo moguće jer bi, kako će biti detaljnije objašnjeno u idućim poglavljima ovog rada, naprosto bilo preskupo.

3.2 Servisna paradigma

Oblačno računarstvo postoji zbog lakšeg i ekonomičnijeg pružanja računalnih usluga. Povezano je sa **servisnom paradigmom** (koja se najčešće iskazuje frazom "*…as a Service*") čija je temeljna ideja da se računalni resursi (hardverski i softverski) unajmljuju i plaćaju po utrošku umjesto kapitalnih ulaganja u hardver i kupnje softvera.

Osim pretpostavljene veće ekonomičnosti i lakše proširivosti takvog rješenja, koristi za potencijalne korisnike očituju se u možebitnom smanjenju operativnih troškova korištenja računalnih resursa, stoga što se čitava briga oko održavanja (*support*) hardvera i ažuriranja (*update*) softvera i operacijskog sustava prebacuje (*outsourcing*) na vanjske dobavljače (*vendor*). Kod takvih rješenja je informatička potpora temeljnom poslu organizacije (*core business*) moguća uz manje vlastitog informatičkog osoblja. Time su i kapitalni troškovi znatno manji stoga što se hardver i softver ne moraju unaprijed kupovati. Budući da je takva paradigma zapravo i pokretač oblačnog računarstva, kako je to opisano u prethodnim poglavljima, logično je da računalni oblaci služe općenito kao platforme za najam i hardvera i softvera. Postoji pet osnovnih vrsta usluga koje s obzirom na sadržaj unajmljivanja, prema [15, 33, 49, 71], mogu biti:

- *Application as a Service* – **AaaS**
- *Software as a Service* – **SaaS**

- *Platform as a Service* – **PaaS**
- *Infrastructure as a Service* – **IaaS**
- *Hardware as a Service* – **HaaS**

AaaS označava ponudu gotovih programa u obliku usluge. Korisnik više ne mora kupovati ili licencirati softver, primjerice uredske pakete kao što su MS Office ili Open Office, potom ih postavljati na vlastita računala te brinuti za ažuriranje zakrpama (*patch*) i novim verzijama, nego su mu sve funkcionalnosti takvih programa omogućene u oblaku i učinjene dostupnim preko interneta. Primjer takve usluge su uredski alati Google Docs ili CRM (*Customer Relationship Management*) softver tvrtke Salesforce.

SaaS je usluga slična AaaS-u jer također obuhvaća ponudu gotovih programa u obliku usluge, no (za razliku od AaaS pristupa) u okviru SaaS-a korisnik oblačnog računarstva ima veću slobodu glede odabira i konfiguriranja softvera kojeg želi koristiti. Primjer SaaS-a bi mogao biti dodatno prilagođeni Salesforceov CRM softver, skrojen po mjeri konkretnog (a ne generičkog) korisnika, ili pak specifične mogućnosti Google Docs ili Gmail usluga za korisnike koji ih plaćaju (spram običnih, "besplatnih" korisnika).

AaaS i SaaS zapravo nisu u fokusu ovoga rada. Naime, takve usluge su interesantne korisnicima koji trebaju gotova rješenja za jednostavnije poslove, dok je fokus ovoga rada na korištenju oblačnog računarstva kao platforme za specijalizirani namjenski softver, namijenjen pojedinoj, specifičnoj, maloj ili srednjoj tvrtki.

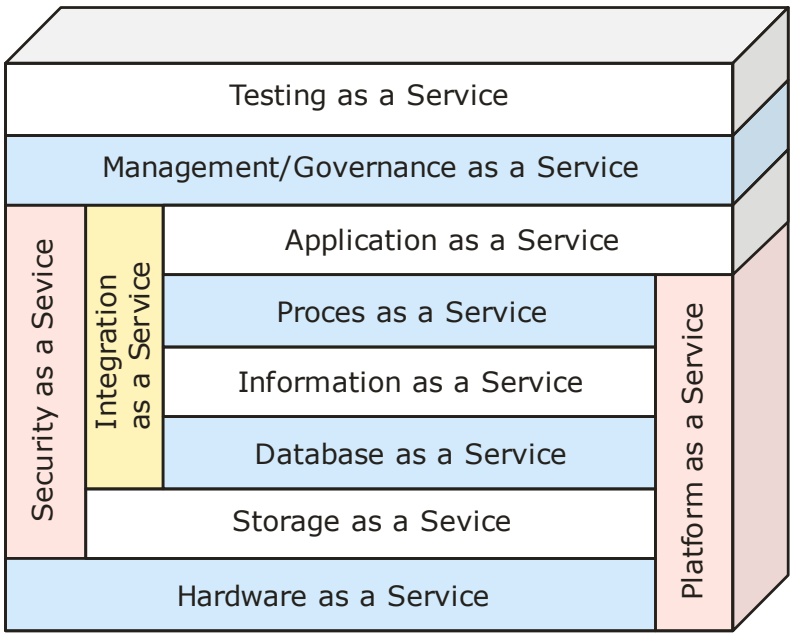
PaaS označava ponudu razvojnog okruženja i aplikacijske platforme u obliku usluge. Radno okruženje i aplikacijsku platformu čine razni uslužni programi (*utilities*), programske biblioteke (*libraries*), softverski servisi, i slično. Na primjer, u Windows Azure oblaku u sklopu PaaS-a nudi se zapravo .NET tehnologija za razvoj i pokretanje softvera, Internet Information Service (IIS) web server te sustavi za pohranu poput Table, Blob i Queue Storagea, temeljenih na *Key-Value-Store* (KVS) konceptu, ili pak relacijska baza podataka SQL Azure. Kao i u AaaS-u, korisnik ne mora više unaprijed kupovati, nabavljati i održavati klasične verzije tih tehnologija (primjerice SQL Server sustav za upravljanje relacijskim bazama podataka), niti planirati proširenje ako mu ono ustreba, nego je briga o svemu tome prebačena na oblačnog poslužitelja.

IaaS označava ponudu temeljnog softvera (operacijskog sustava i upravljačkih programa) i hardvera – računalnih i pohrambenih kapaciteta – u obliku usluge. Umjesto kupovanja, računala, diskovi, ruteri i druga oprema mogu se unajmiti kroz oblak izbjegavajući na taj način kapitalne troškove.

HaaS označava ponudu isključivo hardvera, kolokvijalno rečeno "golog metala" (*bare metal*) u obliku usluge, pri čemu se (za razliku od IaaS-a), korisniku prepušta konfiguriranje tog hardvera u kohezivnu infrastrukturu na način da korisnik instalira i podesi operacijske sustave i upravljačke programe na unajmljenom hardveru.

Mogućnost najma hardvera u obliku usluge (HaaS) značajna je za akademske oblake koji trebaju podržati nastavne [102] i potencijalno vrlo specifične istraživačke aktivnosti. Ova vrst usluge manje je značajna za komercijalno oblačno računarstvo koje je više orijentirano na poslovnu informatiku to jest ponudu računalnih usluga korisnicima kojima informatika nije temeljni posao (*core business*) te koji stoga ne žele ili ne znaju podešavati "goli metal" u skladu s vlastitim potrebama. Zato je komercijalno oblačno računarstvo više orijentirano na iznajmljivanje infrastrukture kao servisa (IaaS) što je posebno obrađeno u taksonomiji komercijalnih oblačnih usluga u poglavlju 6. Zbog svih tih razloga, u ovom radu (orijentiranom na proučavanje uvjeta korištenja komercijalnog oblačnog računarstva) uglavnom se promatra IaaS, a ne HaaS.

Shematski prikaz oblačnih usluga i podjele na IaaS, PaaS i SaaS prikazan je slikom 3.1. Iz te slike vidljiv je međusobni tehnički odnos SaaS-a, PaaS-a i IaaS-a. Taj prikaz odnosi se, zapravo, na poslužiteljevo viđenje tih paradigmi. Ako poslužitelj želi ponuditi SaaS, mora imati odgovarajuću platformu na svojoj infrastrukturi. Implicitno, korisnik zakupom SaaS-a (u komercijalnom oblaku) također zakupljuje platformu i hardver, iako ne mora znati ništa o njima niti ih igdje izravno plaća (nego samo neizravno kao troškove već uračunate u SaaS). Ako pak pružatelj usluga oblačnog računarstva želi ponuditi PaaS na kojem korisnici mogu sami razvijati svoj softver, onda mora imati i odgovarajuću infrastrukturu.



Slika 3.1: ***Shematski prikaz oblačnih usluga prema servisnoj paradigmi***

Neki autori [48, 50, 59] spominju i mnoge druge servisne paradigme, još najmanje 8 dodatnih (uz 5 već navedenih):

1. *Storage as a Service* – najam diskova iz oblaka za pohranu podataka.
2. *Database as a Service* – najam relacijske ili neke druge baze podataka u oblaku.
3. *Information as a Service* – mogućnost razmjene informacija (poruka) kroz oblak u okviru definiranog programskog sučelja (API) kao na primjer – web servisi.
4. *Process as a Service* – usluga povezivanja i komunikacije raznih procesa kroz oblak, primjerice korištenje oblaka kao servisne sabirnice (*service bus*).
5. *Integration as a Service* – usluge integracije softverskih i hardverskih rješenja u oblaku s vanjskim rješenjima na lokalnim računalima ili rješenjima u drugim oblacima.
6. *Security as a Service* – usluge osiguravanja podataka i aplikacija.
7. *Management as a Service* – usluga upravljanja oblačnim resursima i rješenjima smještenim u oblaku.
8. *Testing as a Service* – mogućnosti testiranja raznih rješenja, oblačnih i lokalnih, iz oblaka, putem oblačnog softvera najmljenog kao usluge.

Detaljnijom analizom može se, međutim, doći do zaključka da ovih 8 dodatnih servisnih paradigmi zapravo predstavljaju specijalizaciju usluga unutar temeljnih tipova kao što su SaaS, PaaS ili IaaS. U platformu kao servis spadaju baze podataka (točka 2), u hardver kao servis spada pohrana (točka 1), a sve ostale navedene stavke spadaju ili u platformu ili u aplikaciju kao servis. Na primjer, u Windows Azure oblaku, *process as a service* je osiguran kao usluga u sklopu platforme zvane AppFabric.

Može se zaključiti da **servisna paradigma nije nužno vezana za oblačno računarstvo, no oblačno računarstvo je primarna i najdjelotvornija platforma za njenu primjenu.**

3.3 Definicija oblačnog računarstva

Zbog ispreplitanja velikog broja pojmova vezanih za oblačno računarstvo, u ovom su radu najprije određena funkcionalna svojstva računalnih oblaka, temeljem kojih se može dati njihova egzaktna definicija.

Najjednostavnija takva definicija je ona prema kojoj je oblak bilo kakav računalni servis kojeg pruža vanjska ustanova (vanjska – gledano iz korisničke perspektive), a koji se plaća prema utrošku (*pay as you go*) [22]. Takva je definicija, međutim, previše općenita i neodređena jer bi se jednako mogao definirati i računalni grid, a uz to opisuje svojstvo isključivo komercijalnog računalnog oblaka.

Definiciju računalnog oblaka komplicira i činjenica da je to tehnologija čiji se sami temelji još uvijek razvijaju [111], no velika većina istraživača, primjerice [69, 102, 110], korisnika i stručne javnosti [15, 49, 50, 71, 72] računalni oblak definira na slijedeći način:

Računalni oblak je skup mrežnih servisa namijenjenih pružanju raznih računalnih usluga (poput digitalne pohrane podataka ili izvođenja softverskih rješenja). Pružanje tih usluga odlikuje se slijedećim osobinama:

- **Samoposlužno je i na zahtjev** (*on-demand self-service*) – korisnik može nabavljati računalne usluge automatski, bez potrebe za stalnom interakcijom s osobljem pružatelja usluga oblačnog računarstva.
- **Mrežni pristup** (*network access*) – dostupnost usluga ostvaruje se temeljem standardnih informacijskih i komunikacijskih tehnologija i protokola te je podržana preko raznovrsnih uređaja (primjerice osobnih računala i mobitela).
- **Virtualiziranost resursa** (*resource virtualization*) – utemeljenost na virtualiziranim hardverskim i softverskim resursima pomoću kojih se dobivaju pružene usluge. Virtualiziranost resursa omogućuje njihovo udruživanje (*resource pooling*) u svrhu posluživanja mnogih korisnika. Resursi se raspoređuju na način da korisnik ne zna niti mora znati točnu mikrolokaciju unajmljenog resursa, no može ga znati ili čak zadati na makro razini, primjerice, država, grad ili neki konkretni podatkovni centar. Važno je napomenuti da, unatoč jakom naglasku na virtualizaciji, neki oblaci, posebice akademski poput VCL-a [69, 102] omogućuju i najam fizičkih računala gdje su operacijski sustav i korisnički programi postavljeni izravno na hardver (*bare-metal load*).
- **Brza elastičnost** (*rapid elasticity*) odnosno skalabilnost (*scalability*) – opseg usluga i resursa može se brzo proširivati ili sužavati u skladu s trenutnim poslovnim potrebama korisnika.
- **Naplata prema potrošnji** (*pay as you go*) – korisnik ne mora najmljene usluge i resurse plaćati unaprijed, nego onoliko koliko ih koristi u zadanoj obračunskoj jedinici. Ovo svojstvo valja uzeti s rezervom jer se isključivo koristi samo u komercijalnim oblacima, dok ostali (privatni ili akademski oblaci) mogu imati drukčije modele naplate.

3.4 Etimologija oblačnog računarstva

Oblačno računarstvo, *cloud computing*, je zapravo prilično nespretan pojam kojeg mnogi, uključivo i engleske govornike [7], smatraju gotovo žargonom. Pojam potječe iz običaja da su

računalni gridovi te internet prikazivani u raznovrsnoj dokumentaciji kao oblačić [71] okružen raznim klijentskim računalima u koga ti klijenti šalju i iz njega primaju podatke i posao. Oblačić je simbolizirao apstrahiranost tih središnjih serverskih farmi, primopredajnika posla i podataka, te činjenicu da klijente nije zanimalo što se tamo događa, nego samo rezultati tih događanja. Kada se počelo razvijati oblačno računarstvo, upravo na temeljima gridova i internetskih tehnologija, oblačne podatkovne centre se nastavilo crtati kao oblačiće. I tako je rođen pojam oblačnog računarstva koji je ostao do dana današnjeg te se proširio i u hrvatski jezik, uglavnom preko stručnih konferencijskih izlaganja.

Temeljni prigovor tome nazivu, uz onaj da nema nikakve veze sa funkcionalnošću i opisom računalnih oblaka, jest u njegovoj proizvoljnosti jer bi, da su se internet i gridovi crtali kao kvadrati, elipse, oceani ili crne kutije, danas glasio "kvadratno", "elipsasto", "oceansko" ili "crnokutijasto" računarstvo. To je sve točno. Također, mnogi su predlagali nazivlje poput distribuiranog ili mrežnog odnosno gridnog računarstva, no u prethodnom poglavlju ovoga rada pokazano je da su ti nazivi već zauzeti, rezervirani za tehnologije koje, doduše, imaju veze s oblačnim računarstvom, ali nisu oblačno računarstvo.

Najuvjerljiviji pokušaj suvislijega naziva za tehnologiju računalnih oblaka je visoko virtualizirana infrastruktura, *highly virtualized infrastructure*, no on se nije ukorijenio, vjerojatno zbog toga što je predugačak te što, za jedan opisni naziv, zapravo prilično slabo opisuje sve aspekte oblačnog računarstva, jer je njime obuhvaćen uglavnom samo hardver kao servis (HaaS). Zato smo u ovom radu ostali na pojmu **oblačnog računarstva** kao najboljemu.

3.5 Vrste računalnih oblaka

Računalni oblaci mogu se, s obzirom na to tko im može pristupiti, kategorizirati u nekoliko vrsta [15, 28, 29, 33, 49, 71].

- Privatni oblaci (*Private Clouds*), vrlo često nazivani korporativnima (*Corporate Cloud*)
- Potrošački oblaci (*Consumer Clouds*)
- Komercijalni oblaci (*Public Clouds*)
- Mješoviti (hibridni) oblaci (*Hybrid Clouds*)
- Akademski oblaci (*Academic Clouds*)

Privatni ili korporativni oblaci su podatkovni centri (*data centers*) u vlasništvu jedne ustanove, namijenjeni vlastitom korištenju, a temeljeni su na istim tehnologijama kao i javni oblaci. Privatni oblaci korisni su za balansiranje računalnih potreba između raznih odjela jedne

ustanove te izbjegavanja umnožavanja računalnih resursa koje se događa kada svaki odjel ima vlastitu računalnu infrastrukturu koja mora osigurati dovoljno računalne snage i kod svih vršnih opterećenja (*peak demand*). U privatnom oblaku računalni resursi se, u skladu s trenutnim vršnim opterećenjem, dodjeljuju onom odjelu koji ih trenutno najviše treba.

Potrošački oblaci su najrašireniji, u potpunosti odgovaraju najjednostavnijoj funkciji oblaka kao pružatelja aplikacijskih usluga (AaaS). Na takvoj platformi pojedinci i skupine, fizičke ili pravne osobe, mogu internetskim pristupom koristiti uslužne aplikacije i pohrambene resurse poput društvenih mreža, blogova, e-pošte, pohrane podataka ili slika, i tako dalje. Facebook, Twitter, Gmail, Google Docs, Windows Live, Dropbox i mnogi drugi servisi primjeri su aplikacija koje se pokreću u potrošačkom oblaku.

Pojam komercijalnih oblaka odnosi se na oblačno računarstvo temeljeno na specifičnoj oblačnoj platformi (kao što je Windows Azure ili Amazon Elastic Cloud) s podržanim razvojem i smještajem aplikacija, internetskim pristupom, podatkovnom pohranom i drugim mogućnostima. Najnoviji razvoj u ovom području omogućava ostvarenje privatnog oblaka, što je za sada sadržaj ponude kod nekih poslužitelja (poput Amazona). Drugim riječima, komercijalni oblaci redovito (uz softver kao servis) nude i platformu kao servis. Glavna ekonomska značajka komercijalnih oblaka je naplata prema obujmu korištenja (*pay as you go*). Budući da su u fokusu ovog rada komercijalni oblaci, u idućim poglavljima bit će detaljnije istražena isplativost najma oblačnih resursa spram korištenja vlastitih IT resursa na klasičan način

Mješoviti (hibridni) oblaci su kombinacije privatnih i javnih oblaka ili javnih oblaka i lokalnih računala. Takav koncept nastao je kao odgovor na određene sigurnosne brige oko smještaja podataka te korisničku težnju za "svojim aplikacijama na vlastitim računalima". Razlog uporabe hibridnih oblačnih arhitektura, odnosno povremenog iznajmljivanja javnih oblačnih resursa, jest potreba za fleksibilnim i jeftinim odgovorom na vršna opterećenja (*peak demand*) [7]. Zbog toga se hibridni oblaci često kolokvijalno označavaju kao izbojno računarstvo (*surge computing*), uz istovremeno zadržavanje neovisnosti softverskog rješenja o tehnološkim specifičnostima pojedinog oblaka. Za softverska rješenja temeljena na hibridnim oblacima ponekad se, kao na primjer u [28], koristi naziv **oblačno orijentirana poslovna arhitektura** (*Cloud Oriented Business Architecture – COBA*). COBA spada u glavne teme ovoga rada stoga što spaja SOA pristup s tehnologijama oblačnog računarstva i internetskim (web) tehnologijama općenito.

Akademski oblaci koriste se na sveučilištima te su često izvedeni u suradnji s privatnim sektorom [69, 100, 102]. Primarna im je namjena služiti u istraživačke svrhe radi eventualne primjene tako stečenih saznanja u gospodarstvu [84] te kao podrška nastavnim aktivnostima.

Po tehničkim značajkama akademski oblaci nalik su komercijalnima. Upravljanje resursima u akademskim oblacima također ima dodirnih točaka s upravljanjem resursima u komercijalnim oblacima – na primjer, u obje vrste oblaka nastoji se postići što veće iskorištenje (*utilization*) pojedinačnih fizičkih računala [100]. U akademskim oblacima visoko iskorištenje je nužno radi postizanja što većih ušteda (u IT opremi i električnoj energiji), dok je u komercijalnim oblacima visoka iskoristivost potrebna radi ostvarivanja što veće isplativosti oblačnih podatkovnih centara (što će biti opširno obrađeno u idućim poglavljima ovog rada).

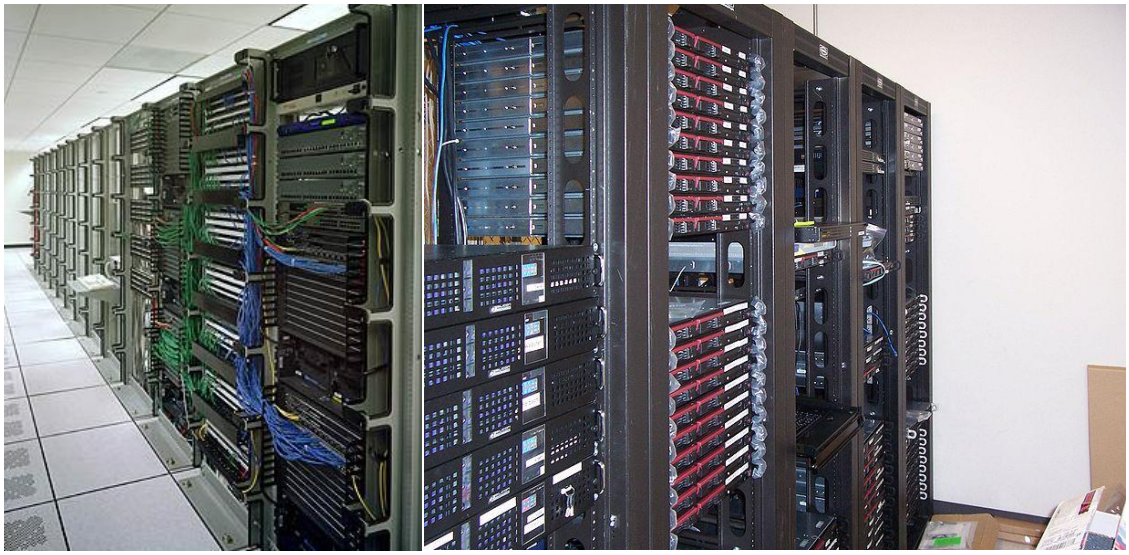
3.6 Izvedba računalnih oblaka

Računalni oblaci mogu se fizički realizirati kao: (1) serverski prostori u čvrstim građevinama, (2) kao mobilni sustavi smješteni u kontejnerima ili (3) kombinirano – kontejneri smješteni u čvrstim građevinama. Sve te izvedbe se, s obzirom na njihovu temeljnu funkcionalnost, nazivaju podatkovnim centrima (*data center*). Gradivni blokovi za izvedbu (2) ili (3) su kontejneri, vanjskog izgleda poput klasičnih brodskih (*intermodal container*). Njihove dimenzije mogu biti razne, ali su standardizirane [114]. Svaki kontejner sadrži od nekoliko stotina do nekoliko tisuća računala (ovisno o dimenzijama kontejnera), potrebnu mrežnu opremu te vanjske energetske i komunikacijske priključke za integraciju kontejnera u širi sustav. Takvi se kontejneri kao cjeline smještavaju u prikladne građevine i povezuju, a mogu se, prema posebnim zahtjevima u pogledu uvjeta korištenja, izvesti kao potpuno mobilna rješenja, kao što je prikazano slikom 3.2.

Slika 3.2: ***Vanjski izgled i dimenzije oblačnog kontejnera*** *(preuzeto s http://sgi.com)*

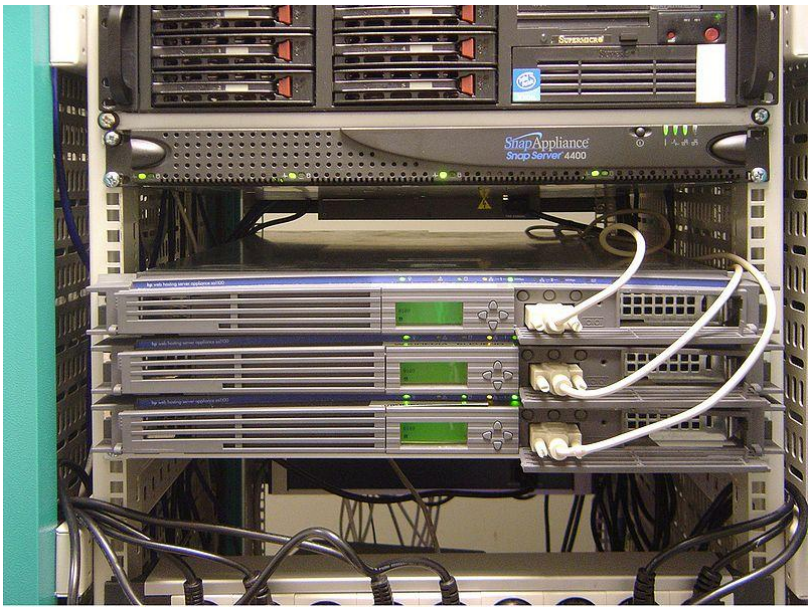
Računalni kontejneri izvorno su zamišljeni u tvrtki Sun [15], no danas ih koriste svi veliki podatkovni centri [66], a ne samo oblačni.

Bez obzira na vanjsku izvedbu oblačnih podatkovnih centara (kontejneri ili klasične serverske sale), njihova unutrašnjost sadrži niz serverskih stalaka (*server rack*), kao u serverskim salama manjih tvrtki ili klasičnim podatkovnim centrima. Tipični izgled popunjenog stalka u podatkovnom centru prikazan je, iz dvije perspektive, na slici 3.3.



Slika 3.3: ***Serveri u podatkovnom centru*** *(preuzeto s http://keyitec.com)*

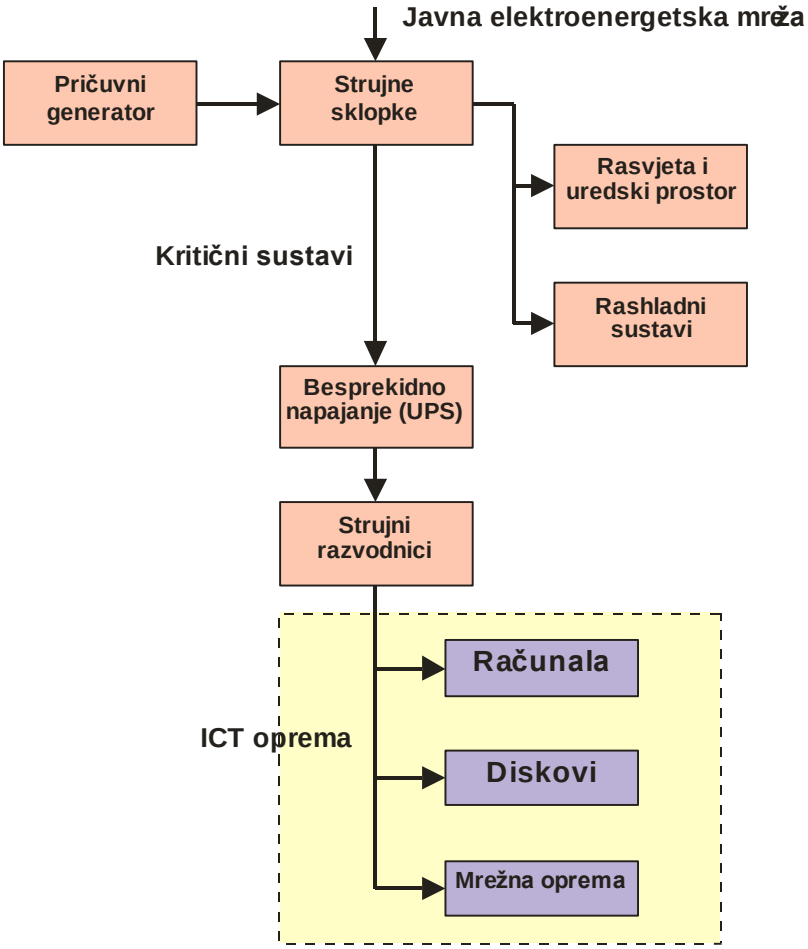
Širina stalaka, njihova unutarnja struktura te razmak između dviju polica su standardizirani (na primjer, uži stalci koji su češće u uporabi, imaju propisanu širinu od točno 482,6 *mm*). Stalci sadrže serverska računala, obično *blade* izvedbe te potrebna napajanja i mrežne utičnice, što je pregledno prikazano slikom 3.4.



Slika 3.4: ***Serverska računala u stalku*** *(preuzeto s http://snapshot.sysv.com)*

Stalak je temeljna jedinica mjere fizičkih računalnih kapaciteta što je korišteno kod izračuna troškova podatkovnog centra u poglavlju 6. Osim stalaka s računalima te mrežne opreme i

diskova – stavki vezanih za temeljni posao (*core business*) – oblačni centri (i podatkovni centri općenito) imaju i raznu drugu servisnu opremu: pričuvne dizel-generatore, mnoštvo električne opreme (sklopke, utikači, kablovi), uredske prostorije i mokre čvorove, besprekidna napajanja te, što je naročito bitno, rashladne uređaje. U idućem poglavlju bit će više riječi o svim troškovima oblačnog centra s naglaskom upravo na odnos između troškova komponenti koje se tiču temeljnoga posla spram troškova servisnih komponenti poput hlađenja. Shematski prikaz svih električnih komponenti oblačnog centra prikazan je prema [93] slikom 3.5.



Slika 3.5: ***Elektroenergetske komponente oblačnog centra*** *(prema [93])*

Pri izvedbi oblačnih centara mora se težiti smanjenju glavnih troškovnih komponenti. To su:

- Iskorištenost računala
- Elektroenergetska infrastruktura
- Potrošnja električne energije
- Mrežni troškovi

Većina običnih podatkovnih centara imaju relativno nisku **iskorištenost računala** od oko 10%. Prema dostupnoj literaturi [39] uzrok tome su slijedeći razlozi (poredani prema redoslijedu važnosti):

1. Virtualizacijski nedostaci uzrokovani mrežnom organizacijom – zaliha (*pool*) virtualnih računala ne može se uvijek optimalno i dinamički podijeliti prema potrebama, nego postoje ograničenja uzrokovana mrežnim razlozima, primjerice, konfiguracijom balansera opterećenja (*load balancer*).
2. Slabo aplikativno iskorištenje računala ili njegovih komponenti, uglavnom do 10%.
3. Nesigurnost prognoziranja potrebe za računalnim resursima.
4. Upravljanje rizicima u smislu nabave (*provisioning*) i pokretanja više resursa nego što je potrebno, za slučaj nepredviđenih povećanih zahtjeva i otpornosti na kvar većeg broja komponenti.

Glavni načini ublažavanja ovog troškovnog faktora u oblačnim centrima su:

- Povećanje agilnosti dodjele virtualnih računala korištenjem lokacijski nezavisnog adresiranja (u smislu neovisnosti IP adrese virtualnog računala od njegove fizičke lokacije u oblačnom centru).
- Ujednačenje mrežne protočnosti (*bandwidth*) i počeka (*latency*) svakog pojedinog virtualnog računala.
- Sigurnosna i aplikacijska izolacija virtualnih računala tako da tehnički ispad ili sigurnosni proboj na jednome ne uzrokuje lančanu reakciju na drugima.

Iskorištenost računala je najbitniji čimbenik profitabilnosti i isplativosti oblačnog računarstva, što će biti vrlo opširno istraženo u poglavljima 5 i 6. Povećanje iskoristivosti instaliranih računala je od presudnog značaja za uspješno poslovanje svakog pružatelja usluga oblačnog računarstva.

Elektroenergetske infrastrukturne komponente poput generatora, transformatora i besprekidnih napajanja (UPS) su skupe jer ne spadaju u široko dostupnu (*commodity*) opremu. Za snižavanje njihovoga troška ključno je horizontalno skaliranje, s većim brojem jeftinijih (i slabijih) računala grupiranih u više geografski udaljenih oblačnih centara. Teži se tome da "jedinica ispada" postaje cijeli jedan oblačni centar (od više njih), to jest računalni oblak nastavlja funkcionirati i onda ako jedna cijela njegova komponenta (odnosno jedan oblačni centar) ispadne iz uporabe. Na taj način smanjuje se potreba za UPS-ovima koji mogu biti osnovnoga tipa, odnosno namijenjeni samo brzom i sigurnom isključivanju računala, te potreba za električnim generatorima jer se njihov kapacitet može ograničiti samo

na opsluživanje kontrolnih dijelova oblačnog centra neophodnih za njegovo brzo ponovno pokretanje ili pak brzo gašenje.

Google je ogledni primjer takve jednostavne zaštite fizičkih računala u oblaku [7]. Fizička računala koja čine Googleov oblak su sklopljena u vlastitoj režiji temeljem standardnih komponenti (*off the shelf*), bez klasičnih serverskih komponenti poput dvostrukih ventilatora ili pričuvnog napajanja. Jedina komponenta koja ih razlikuje od prosječnog, korisničkog, stolnog računala je obična baterija uklopljena u sustav za napajanje, koja osigurava rad računala u slučaju kvara napajanja upravo toliko vremena koliko treba da se njegov radni teret – virtualna računala – prebaci na drugo fizičko računalo. To je krajnje pojednostavnjenje principa spomenutog u prethodnom odlomku, stoga što su čak i jednostavni UPS-ovi zamijenjeni još jednostavnijim i jeftinijim baterijama.

Potrošnja električne energije promatra se kroz koeficijent iskorištenja. Koeficijent iskorištenja je kvocijent ukupne snage i snage za pokretanje informatičke opreme. Smatra se da dobro izveden oblačni centar ima koeficijent iskorištenja oko 0,6 [4]. Drugim riječima, u dobro izvedenom oblačnom centru oko 60% električne energije se troši na pogon računala, a ostalo otpada na hlađenje cijelog postrojenja (oko trećine) te na gubitke pri prijenosu (oko 8%). Uz prosječnu (nisku!) potrošnju od oko 180 W po računalu lako je izračunati da oblačni centar s 50-ak tisuća računala godišnje troši oko 134 milijuna kWh električne energije.

Najvažniji čimbenik smanjenja utroška struje je energetska proporcionalnost. To je svojstvo računala prema kojem je njegov utrošak energije proporcionalan radnom opterećenju (procesorskom zauzeću). Osim nabave takvih računala koja su što je više energetski proporcionalna, eksperimentira se i s „toplijim“ podatkovnim centrima u kojima se manje struje troši na hlađenje, to jest radno okruženje cjelokupnog hardvera je visokotemperaturno. No, budući da se stopa hardverskih kvarova znatno povećava s porastom radne temperature, potrebno je ili pronaći dobar omjer između energetske uštede i povećanih troškova za hardver ili pak utjecati na elektroničku industriju da se više orijentira prema visokotemperaturnim računalima i drugom elektroničkom sklopovlju.

Mrežni troškovi sastoje se od cijene mrežne opreme i cijene vanjskog umrežavanja (*wide area networking*) oblačnog centra koja nadmašuje cijenu mrežne opreme. Trošak vanjskog umrežavanja izrazito je podložan lokalnim ekonomskim, političkim, poreznim i drugim čimbenicima pa je izbor dugoročno jeftinog i stabilnog mjesta za oblačni centar najvažniji faktor smanjivanja tih troškova. Strategije optimalnog smještaja i međusobnog nadopunjavanja većih i manjih oblačnih centara mogu također doprinijeti smanjenju mrežnih troškova zbog bolje i ravnomjernije iskorištenosti komunikacijske infrastrukture.

3.7 Ekonomske prednosti unajmljivanja računalnih resursa

Kako je vidljivo iz objašnjenja u prethodnim poglavljima, računalni oblaci su, zapravo, sredstvo kojim se računalni resursi mogu ponuditi korisnicima u obliku javne komunalne usluge (*utility*), sasvim nalik ponudi struje, vode, plina ili telekomunikacija. U tom smislu, **prednosti korištenja računalnih oblaka uvelike su izražene već njihovom definicijom** te **servisnom paradigmom korištenja.** Radi potpunog razumijevanja konceptualnih prednosti oblačnog računarstva potrebno je spomenuti dvije glavne povoljne posljedice njegovih definicijskih svojstava: **elastičnost i optimalnu nabavu.** Ta svojstva su toliko bitna da će u ovome poglavlju biti samo konceptualno objašnjena, dok će njihovo temeljitije istraživanje biti provedeno u idućem.

Elastičnost znači da se korištenje računalnih resursa može precizno podešavati angažmanom pojedinačnih računala. Naime, većina računala općenito, pa tako i u podatkovnim centrima uključivo i oblačne, obično imaju ukupno iskorištenje između 5% i 20%. Istovremeno, usluge koje ta računala pružaju mogu biti (a najčešće i jesu) takva da imaju vršna opterećenja. Pri nabavi vlastitog hardvera ta se opterećenja moraju uzeti u obzir. Zbog toga se kupac dovodi u situaciju da je u običnim razdobljima, kada vršnih opterećenja nema, kupljeni hardver nedovoljno iskorišten, što bitno povećava troškove pogona izražene cijenom efektivnog sata rada računala.

Elastičnost, poslovno definirana kao angažiranje računalnih resursa prema potrebi uz plaćanje po korištenju, je ključno svojstvo koje omogućuje **optimalnu nabavu** bez prevelikih kapitalnih ulaganja i otpuštanje računalnih resursa koje je potrebno imati za pokrivanje vršnih opterećenja. Potrebna elastičnost u planiranju i upravljanju računalnim resursima je praktički nemoguća u uvjetima nabavljanja opreme za potrebe samo jedne organizacije, jer bi nabavljeni hardver morao pokrivati i kratkotrajna vršna opterećenja, trebalo bi ga staviti u pogon u vrlo kratkom vremenu te imati nevjerojatno aktivno tržište rabljenog hardvera.

3.8 Tehničke prednosti računalnih oblaka

Pod tehničkim prednostima oblačnoga računarstva smatraju se sve one koje su strogo informatičke, računalne i komunikacijske naravi. Drugim riječima, one koje olakšavaju upravljanje računalima i računalnom mrežom, unapređuju razvoj softvera i poboljšavaju performanse informacijskog sustava. Prema [49, 71, 87, 92], u računalnim oblacima najvažnije tehničke prednosti su slijedeće:

- Oblačna softverska rješenja se postavljaju (*deploy*) brzo i jednostavno.

- Poticanje razvoja (oblačnog) softvera temeljenog na industrijskim standardima te posljedično posredno povećanje interoperabilnosti (*interoperability*) među softverskim rješenjima raznih proizvođača.
- Vlastito informatičko osoblje se može usredotočiti na ICT inovacije, a održavanje hardvera i infrastrukturnog softvera (poput operacijskog sustava ili uredskih alata) prepustiti pružatelju usluga oblačnog računarstva.
- Olakšano je raspršenje podatkovnih spremišta i softverskih rješenja na razne geografske i virtualne lokacije, što poboljšava informatičku sigurnost (jer „*nisu sva jaja u istoj košari*“). Posljedično, raspršenjem je također lakše obraniti se od napada uskraćivanjem usluge (*Denial of Service* – DoS).
- Olakšana je rezervacija resursa namijenjenih mamljenju (*honey pot*) potencijalnih napadača na informacijski sustav.

Za oslikavanje tih prednosti oblačnog računarstva potrebno je opisati tipove aplikacija za čije je izvođenje računalni oblak posebice dobra platforma. Mnogi autori [5, 16, 29, 47] posebice navode takve aplikacije koje karakterizira pojava vršnih opterećenja i mogućnost paralelizacije, no mišljenja smo da svaki informacijski sustav za čiji rad postoji potreba za više usklađenih računala povezanih u serversku farmu (ili čak čitav poduzetnički podatkovni centar) može biti isplativo pokretati u računalnom oblaku. Ostvarenje takve mogućnosti je jedan od motiva ovoga rada i bit će detaljno istraženo u kasnijim poglavljima, s ciljem izrade praktičnih smjernica kada odabrati oblak, a kada vlastita računala.

Prva skupina aplikacija posebice pogodnih za oblačno računarstvo jesu one namijenjene brzom podatkovnom rudarenju (*data mining*) i drugim vrstama podatkovne analize koja treba biti obavljena u relativno kratkom vremenu ili trenutno (*real-time*). Takve aplikacije su:

- Trgovačko odlučivanje o stavljanju određene robe u prodaju ili sniženju cijene, u skladu s podacima o uzorcima kupovanja u zadnje vrijeme.
- Odlučivanje u trgovini (elektroničkoj ili običnoj) i posudionicama o tome što ponuditi kupcu ili posuđivaču temeljem njegovih prethodnih kupnji ili posudbi. Realni primjer takvog sustava je Amazonova e-knjižara.

Navedene aplikacije mogu imati vršna opterećenja i mogu se paralelizirati, jer algoritmi koji obavljaju podatkovnu analizu mogu biti distribuirani na više računala, odnosno svako računalo obavlja algoritam nad zasebnim podatkovnim podskupom.

Druga skupina aplikacija pogodnih za izvođenje u oblaku su one namijenjene odlučivanju u realnom ili kratkom vremenu, kao na primjer:

- Kreditna ustanova mora u nekim slučajevima izraditi brzi proračun kreditnog boniteta za nekog klijenta.
- Osiguravajuća ustanova može zatrebati brzi proračun rizika za svakog pojedinog tražitelja osiguranja.

Ovakve aplikacije obično se mogu paralelizirati, a vrlo često imaju i vršna opterećenja, primjerice u proljeće kada počinje stambena gradnja i dodjela stambenih kredita.

Treća skupina za oblak pogodnih aplikacija su one gdje se provodi nekakvo pozadinsko (*batch*) procesiranje s povremenim predvidljivim ili nepredvidljivim vršnim opterećenjima:

- Obrada e-narudžbi tijekom predbožićne kupovne groznice.
- Priljev slika na fotografske web portale i zahuktavanje rasprava nakon završetka ljetnih godišnjih odmora.
- Obrada poreznih prijava nakon kraja poreznog razdoblja.
- Marketinške kampanje koje obuhvaćaju slanje velikog broje e-poruka i elektroničkih letaka.
- Periodičko generiranje elektroničkih geografskih karata sukladno novim satelitskim mjerenjima i snimkama.
- Identifikacija osoba temeljem njihove slike, otiska prsta, šarenice ili drugih biometrijskih podataka, posebice tijekom turističke sezone.
- Klasično korisničko računarstvo, *desktop computing*, za obrazovne aktivnosti gdje treba brzo organizirati i staviti na raspolaganje veliki broj sasvim namjenski konfiguriranih virtualnih računala sa specifičnim programima [102].

Uz sve navedene primjere, mora se naglasiti da i "običnije" aplikacije mogu biti pogodne, ekonomski ili tehnički, za smještaj u oblak ako su takvih svojstava da ih je naprosto jeftinije izvoditi u računalnom oblaku. Analiza tih svojstava i isplativosti izvođenja u oblaku provedena je u petom i šestom poglavlju ovoga rada.

3.9 Rizici i nedostaci

Oblačno računarstvo, kao i svaka tehnologija, ima svoje dobre i loše strane. U poslovnom pogledu ono je novi koncept, ne toliko u smislu korištenih tehnologija (jer se, na kraju krajeva, gotovo sve oblačne ponude temelje na već postojećim tehnologijama i platformama) nego u smislu optimalnog korištenja računalnih kapaciteta. Kao kod svakog novog tehnološkog koncepta, glavni rizici korištenja oblačnog računarstva uglavnom su posljedica

nedovoljne zrelosti same platforme, relativno kratkih iskustava ponuditelja i nepripremljenosti korisnika. Te će rizike, zasigurno, i proizvođači i korisničke zajednice vremenom ublažiti ili bitno smanjiti. Za sada, ovi se rizici i nedostaci mogu, prema [5, 7, 46, 59, 87], tematski grupirati ovako:

- Sigurnost i privatnost – pretežito se odnosi na sigurnost smještaja podataka u oblaku. Aplikativna i komunikacijska sigurnost je također predmet izučavanja, no te sigurnosne stavke nisu specifične za oblačno računarstvo, nego se odnose na sve mrežne i web aplikacije.
- Ovisnost o isporučitelju (*vendor lock-in*) – odnosi se na bojazan da bi postavljanje vlastitih softverskih rješenja u oblak jednog pružatelja usluga moglo, zbog tehničkih specifičnosti pojedinih oblačnih platformi, uzrokovati dugoročnu posvemašnju zavisnost o tom isporučitelju, to jest nemogućnost jednostavne promjene pružatelja oblačne usluge.
- Raspoloživost usluge – odnosi se na stvarnu raspoloživost i pouzdanost najmljenih oblačnih resursa, havarijski scenarij te vrijeme potrebno za nastavak rada ako se desi opći pad sustava. Treba napomenuti da osiguranje tražene raspoloživosti usluge nije u isključivom djelokrugu isporučitelja oblačne usluge, jer je komunikacija prema oblaku i iz oblaka u nadležnosti lokalnih internetskih servisnih isporučitelja (*internet service providers* – ISP).
- Isplativost – nisu sasvim jasni uvjeti pod kojima je isplativo koristiti računalni oblak. Upravo ovaj aspekt oblačnog računarstva će biti detaljno istražen u ovom radu.

Osim navedenih stavki koje će biti pobliže objašnjene u idućim poglavljima, treba imati na umu da, kako god se razvijalo, **oblačno računarstvo nikada neće biti potpuna zamjena za sve ostale informacijske tehnologije** iz barem dva razloga:

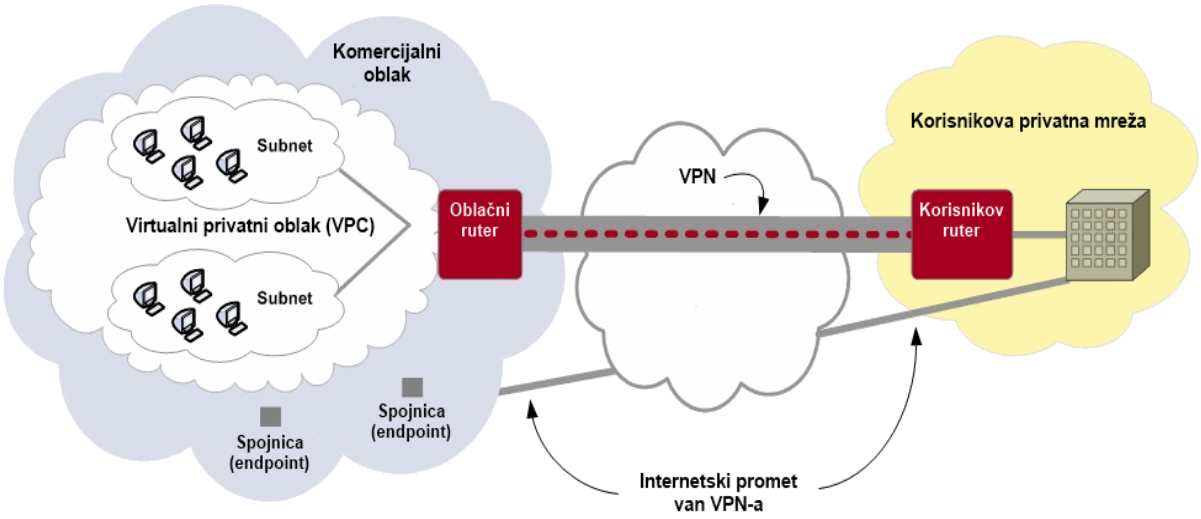
1. Utemeljenost na virtualizaciji s jedne strane, te nužnost da oblačni centri učinkovito odgovore na široku lepezu potreba za računalnim resursima vrlo različitih korisnika, imaju kao posljedicu univerzalnost virtualiziranog hardvera. Drugim riječima, virtualna računala u oblačnim centrima nisu zadnja riječ tehnike, niti imaju jak hardver kakav se očekuje od, primjerice, vertikalno skaliranih servera u poduzetničkim centrima, ili u grafičkih radnih stanica, ili za igraća računala. To nije samo po sebi nedostatak stoga što oblačno računarstvo ionako nije namijenjeno specifičnim svrhama za koje se koriste "jaka" računala. Međutim, takva ograničenja potencijalni korisnici ipak moraju imati u vidu, kako ne bi računali da će s unajmljenim resursima moći zadovoljiti funkcionalnosti za koje oblačni resursi naprosto nisu namijenjeni.

2. Internetske tehnologije su kamen temeljac oblačnog računarstva. Oblačne aplikacije jesu i bit će web aplikacije, iako će vremenom "čisti" web – JavaScript i HTML – biti sve više i više dopunjavan daleko moćnijim klijentskim tehnologijama i bogatim internetskim aplikacijama (*rich internet application* – RIA) čiji je rad moguć uz odgovarajuće dodatke za web preglednike, poput Silverlighta ili Flexa. Mišljenja smo da mogućnosti web aplikacija nikada neće u potpunosti doseći mogućnosti klasičnih izvršnih programa, pogotovo ne onih koji zahtijevaju opsežni pristup hardverskim resursima i njihovo maksimalno korištenje. Iako je interakcija tih klasičnih, stolnih (*desktop*) aplikacija sa internetom, pa tako i s oblacima, potpuno izvediva i u uporabi već dugi niz godina, uvijek će postojati zadaće koje će se moći obaviti samo na lokalnim računalima. Takve zadaće ne moraju nužno biti igre – grafički rad, programiranje robota i upravljanje složenim tehnološkim procesima definitivno spadaju u poduzetničke aplikacije.

Određeni paradoks oblačnog računarstva očituje su u tome što je uzrok glavne bojazni glede **sigurnosti i privatnosti** njegovog korištenja istodobno i glavna potencijalna prednost. Najam računalnih resursa, djelomično ili potpuno odricanje od njihovog posjedovanja te – posljedično - djelomični ili potpuni izmještaj vlastitih podataka i programa u oblak na fizička računala u tuđem vlasništvu, mogu kod korisnika osloboditi ljudske potencijale za njegove ključne poslovne aktivnosti (*core business*).

Razumljiva briga glede te činjenice može se, međutim, riješiti načinima koji su već isprobani u informatici i telekomunikacijama:

- Uspostavljanjem virtualne lokalne mreže (*virtual local area network* – VLAN), po uzoru na virtualne privatne mreže (VPN) u telekomunikacijama, može se zaštititi protok podataka unutar oblaka te između oblaka i korisnika. U trenutku pisanja ovoga rada (2010 – 2011), Amazon ima takvu uslugu zvanu Amazon Virtual Private Cloud (Amazon VPC), za sada u probnoj fazi te namijenjenu samo ograničenom skupu korisnika. Shematski prikaz takvog spajanja s oblakom vidljiv je na slici 3.6.
- Kriptiranje (*encryption*) podataka prije njihove pohrane u oblak. Doista, takav se pristup već koristi [1] za vrlo osjetljive medicinske podatke. U kriptiranje podataka spada i kriptiranje virtualnih diskova (*file system*) te pričuvnih podatkovnih kopija [78].



Slika 3.6: ***Shematski prikaz virtualne lokalne mreže***

Optimizam glede rješavanja ovih problema temelji se na tome što su one već uspješno riješene u drugim konceptima primjene suvremenih informacijsko-komunikacijskih tehnologija. Ozbiljniji problemi su, međutim, političke naravi.

Naime, zakonska regulativa mnogih zemalja traži da osjetljivi podaci o vlastitim građanima budu na računalima fizički smještenima na tlu matične zemlje. Osim promjene takve regulative, odgovor na to može biti odgovarajuće širenje oblačnih podatkovnih centara, no isporučitelj bi ih morao u konkretnoj zemlji napraviti dovoljno da robusnost čitavog (državnog) podoblaka bude jednaka kao i matičnog globalnog oblaka. S obzirom na velike troškove postavljanja podatkovnih centara, malo je vjerojatno da bi bilo koji globalni isporučitelj požurio s takvim rješenjem, dok je mnogo vjerojatnije da bi takva zemlja bila zaobiđena u oblačnom računarstvu, barem glede aplikacija za obradu povjerljivih podataka o vlastitim građanima, ili bi pak morala sama graditi svoju oblačnu infrastrukturu, bilo izravnim državnim angažmanom, bilo u suradnji s privatnim sektorom.

Nadalje, podaci smješteni u oblaku čiji su podatkovni centri na tlu druge države mogu biti podložni tamošnjem lokalnom zakonodavstvu koje potencijalno može izdati sudski nalog za predaju oblačnih podataka lokalnim vlastima, primjerice zbog stečaja oblačnog isporučitelja, ali i zbog potpuno "negospodarskih" razloga kao što je špijunaža i kontrašpijunaža [78]. Tome se, doduše, može doskočiti snažnim kriptiranjem podataka. Međutim, povećanjem snage kriptiranja povećava se i procesorsko opterećenje, što onda traži veći broj zakupljenih virtualnih računala te intenzivniji mrežni promet, što na kraju poskupljuje korištenje oblaka.

Može se reći da je sigurnost podataka smještenih u oblaku *conditio sine qua non*, te da niti jedan oblačni poslužitelj neće moći biti tržišno uspješan bez dobrih rješenja za tu problematiku, no daljnja, detaljnija razmatranja tog polja korištenja i primjene računalnih oblaka ne spadaju o doseg ovoga rada.

Ovisnost o isporučitelju opravdan je razlog za brigu svakog potencijalnog ili konkretnog korisnika oblačnih usluga. Povijest računarstva prepuna je bitki za prevlast tijekom kojih je omiljeno oružje gotovo svakog proizvođača računala bio vlasnički (*proprietary*) i zatvoreni softver i hardver. Iako se može argumentirati kako takav proizvođački pristup možda jest početno opravdan radi povrata ulaganja, gotovo uvijek je bio trajno korišten radi ostvarenja lakšeg profita, bez ravnopravne konkurencije. Teško je, međutim, vjerovati da danas, nakon svega što je učinjeno u normiranju svojstava računalne opreme, komunikacijskih protokola i načina pružanja usluga, ijedan oblačni isporučitelj, ma kako moćan bio, može uspješno ponoviti taj već viđeni i na kraju uvijek odbačeni scenarij.

Trenutna situacija je takva da, zapravo, izbor bilo kojeg oblačnog isporučitelja jamči barem malo ovisnosti o njemu, no isto tako isporučitelji znaju da moraju omogućiti slobodu korisnicima oblaka ili će biti istisnuti iz tog posla. U tom smislu, važni su otvoreni standardi te inicijative poput „Otvorenog oblačnog manifesta“ (*Open Cloud Manifesto*) na web adresi http://www.opencloudmanifesto.org.

Dobar način za smanjenje ovisnosti o isporučitelju usluga oblačnog računarstva, ali i za povećanje interoperabilnosti između korisnika, svakako je normiranje formata datoteke za spremanje virtualnih računala [7]. Napori na tom polju doveli su do *Open Virtualization Format* (OVF) specifikacije, standarda kojeg usvaja sve više oblačnih poslužitelja (uključivo i Microsoft) i proizvođača virtualizacijskog softvera poput VMwarea. Bitno je napomenuti da je OVF format zapravo apstraktna specifikacija – ne postoji virtualno računalo spremljeno kao OVF datoteka. OVF standard samo definira kakvi trebaju biti drugi, vlasnički formati da bi se virtualna računala mogla nesmetano seliti s jednog na drugo fizičko računalo, pod različitim nadglednicima, na način da se virtualnoračunalne datoteke lako i jednoznačno mogu pretvoriti iz jednog vlasničkog formata u drugi. Tri najraširenija vlasnička formata su Microsoftov VHD, VMwareov VMDK i Amazonov AMI, iako su, zapravo, vrlo slični.

Potrebno je standardizirati i pristup podacima. Ta problematika ne tiče se samo oblačnog računarstva, nego je bitna za interoperabilnost općenito. U tom smislu specificiran je *Open Data Protocol* (OData) – način kako podatke iz raznih izvora (relacijskih baza podataka, datoteka, web stranica) učiniti pristupačnima i čitljivima nezavisno od platforme koja ih sadrži. SQL Azure, relacijska baza podataka u Azure oblaku, podržava OData specifikaciju.

Sam oblak može biti izveden u potpunosti kroz *open source* tehnologije u kojoj su i virtualni datotečni formati i nadglednički sustav upravljanja oblakom slobodni softver. Primjer takvog sustava je Eucalyptus [73], otvoreni IaaS sustav interoperabilan s Amazonovim oblakom, namijenjen stvaranju privatnih i hibridnih oblaka, u kojem nadglednički program može kombinirati privatne oblačne resurse s javnim Amazonovim.

Raspoloživost oblaka je složena varijabla koju čine dvije stavke:

- Internetska povezivost preko komercijalnih internetskih poslužitelja (*Internet Service Provider* – ISP), ako se koristi javni ili hibridni oblak.
- Raspoloživost i kvaliteta same oblačne usluge.

Cijena komercijalne internetske veze je svakako vrlo značajna varijabla koju treba uzeti u obzir za račun troškova koji pokazuje isplativost najma oblačne infrastrukture spram nabavke vlastite opreme, no daljnje rasprave o tome izlaze van okvira ovoga rada.

Pojmovi raspoloživosti i kvalitete oblačne usluge prije svega se odnose na nadzor nad unajmljenim resursima u oblaku. Na klasična pitanja, poput *"Gdje su moji serveri?"* i *"Gdje su moji podaci?"* na koja se moglo odgovoriti sa *"U serverskoj sali"* i *"Na ovome disku"* sada se odgovara sa, primjerice, *"Negdje u jugoistočnoj Aziji"* ili *"Negdje u Virginiji"*. Ukratko, prebacivanjem svojeg biznisa u oblak svaka tvrtka zapravo povećava svoju izloženost i zavisnost o nekoj drugoj tvrtki. Naravno, oblačni poslužitelji postoje zbog toga da pružaju usluge svojim korisnicima, no može se dogoditi da jedan poslužitelj poslovno propadne ili ga preuzme druga tvrtka kojoj se neka dosadašnja oblačna usluga učini neisplativom te bude ukinuta, ograničena ili za nju prestane potpora. Svi korisnici koji su koristili takvu uslugu mogu zbog toga pretrpjeti veliku poslovnu štetu.

Najbolja obrana od takvih događaja jesu otvoreni standardi opisani u prethodnim odlomcima gdje je bilo riječi o ovisnosti o poslužitelju, no mnogi [20] pozivaju i na standardizaciju mehanizama za upravljanje životnim ciklusom proizvoda u oblaku, njihovim licenciranjem i drugim upravljačkim postupcima. Smatramo također da će postojeće ITIL (*Information Technology Infrastructure Library*) specifikacije ubrzo biti dopunjene preporukama o specifičnostima pružanja usluga oblačnog računarstva.

Isplativost korištenja računalnih oblaka mora se promatrati s dva stajališta: korisničkog i poslužiteljskog. Naime, oblačni računalni poslužitelji obećavaju izobilje računalnih resursa po pristupačnim cijenama, koje su okvirno prikazane u tablici 2.1. Precizne kalkulatore troškova i cjenike svaki oblačni poslužitelj ima na svojim stranicama. No, ono što svakog potencijalnog oblačnog korisnika zanima jest:

- Što zapravo znače te cijene? Da li je to malo ili mnogo?
- Ima li za korisnika taj iznos istu vrijednost za svaki obujam korištenja računalnih resursa?

Drugim riječima, potencijalni korisnik računalnih usluga se pita: *"Da li oblačni poslužitelji daju pravu protuvrijednost usluge za ono što naplaćuju?"* Radi potpunog odgovora na to pitanje neophodno je provesti objektivni izračun stvarne cijene (*pricing*) oblačnih usluga koju

pružatelji usluga oblačnog računarstva, dakako, ne objavljuju u javnosti zbog razumljivih poslovnih razloga.

Svaki potencijalni ulagač u oblačno računarstvo mora za svoje ulaganje provesti račun rentabilnosti ulaganja, identificirati parametre o kojima ta rentabilnost ovisi, odnosno dobiti argumentirane odgovore na sljedeća pitanja:

- Da li cijene, poput onih prikazanih tablicom 2.1, omogućuju zaradu, ako da – pod kojim uvjetima i kakvu?
- Kakav je povrat ulaganja, osobito ako se ulaganje u izgradnju oblačnog centra usporedi s ulaganjima u druge djelatnosti?

Ta pitanja su interesantna i s korisničke perspektive stoga što argumentirani odgovori na njih daju naslutiti mogućnost rata cijenama (*price wars*), to jest procijeniti trendove u kretanju cijena oblačnog najma u budućnosti.

Tehnički (ali i financijski) uvjeti pod kojima je isplativo sve, ili samo neke funkcionalnosti informacijskog sustava smjestiti u oblak od primarnog su interesa za sve korisnike informacijskih tehnologija a time i potencijalne korisnike komercijalnog oblačnog računarstva. Traženje egzaktnog odgovora na ta pitanja je glavna motivacija ovoga rada i tema koja će biti istraživana kroz njegov daljnji tijek.

Općenito se može pretpostaviti da je oblačno računarstvo nesumnjivo djelotvorno u smislu racionalnijeg korištenja računalnih resursa, djelotvornije od vlastitog podatkovnog centra. S druge strane, veća djelotvornost, to jest bolje iskorištenje resursa, ne mora značiti i manji ukupni trošak korištenja tih resursa! Neki autori [59] upozoravaju na potrebu za cjelovitim, holističkim pristupom pri izračunu ukupne cijene informacijskih tehnologija. Takvim pristupom, u njihovu ukupnu cijenu korištenja uračunavali bi se slijedeći čimbenici:

- Integracija raznih postojećih rješenja ili postojećih rješenja s novima.
- Usklađivanje (*compliance*) softvera i hardvera s internim, nacionalnim ili međunarodnim propisima.
- Prijenos podataka pri radu.
- Pohrana podataka za rad i sigurnosne kopije.
- Oporavak od greške (*recovery service*).
- Rad osoblja.
- Gubitak tijekom nedostupnosti informacijskog sustava (*outage management*).
- Rješavanje specifičnih poslovnih problema.

- Obračunavanje softverskih pretplata i licenci.
- Postizanje tražene sigurnosti.
- Vrijednost kapitalnih ulaganja.

Tek kada se u proračun troškova uključe svi ovi elementi, moći će se procijeniti da li je u svakom pojedinačnom slučaju unajmljivanje oblačnih resursa povoljnija opcija od kupnje vlastitih resursa ili obrnuto. Komercijalni oblačni poslužitelji imaju relativno ujednačen skup kriterija kojima se opisuju usluge oblačnog računarstva i obračunava pristojba za naplatu pruženih usluga. Na primjer, za Windows Azure, ovi su kriteriji sljedeći:

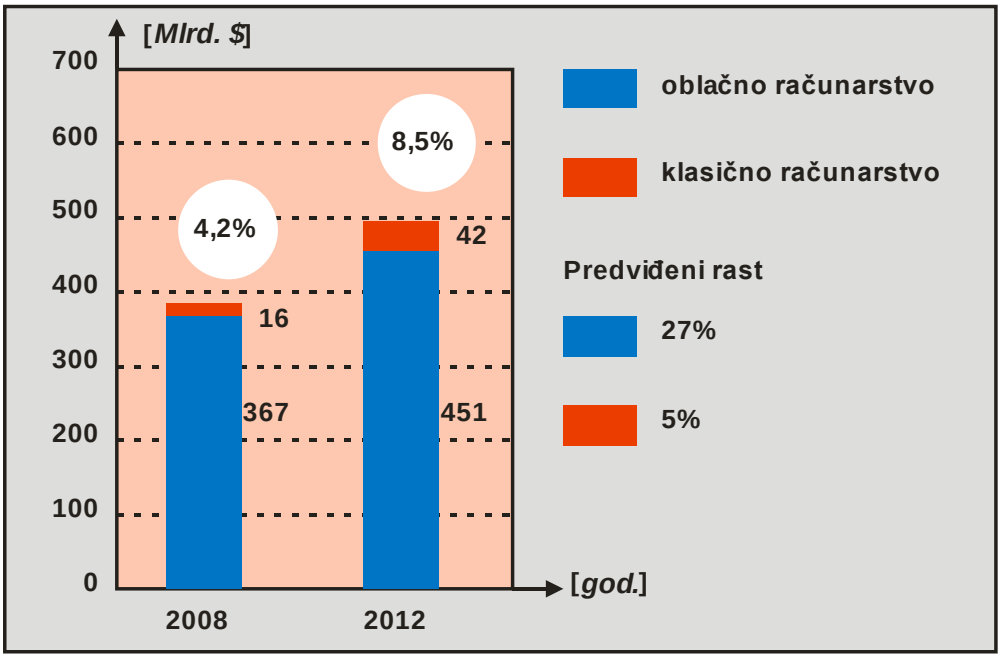
- količina i snaga zakupljenih računala te vremensko trajanje svakog pojedinog zakupa
- opseg podatkovne pohrane, bilo u relacijskoj bazi podataka SQL Azure, bilo u *key-value-store* (KVS) bazi podataka (*Table*, *Blob* i *Queue Storage*)
- broj transakcija u KVS bazi podataka u vremenu ili učestalost (frekvencija) KVS transakcija i
- opseg podataka koji se šalje u oblak radi obrade i opseg izlaza podataka iz oblaka prema korisniku.

Uključivanje svih ovih elemenata u obračun ukupne cijene koštanja te usporedba troškova koji nastaju izvođenjem aplikacije u oblaku s troškovima koji bi nastali korištenjem vlastitih računalnih resursa znači doprinos holističkom pristupu koji se sugerira u [59] a proveden je u petom i šestom poglavlju ovog rada.

3.10 Perspektive oblačnog računarstva

Oblačno računarstvo novi je koncept i dodatna tehnologija u paleti informacijskih i komunikacijskih tehnologija koje su danas raspoložive. Taj koncept neće uvijek zamijeniti postojeće, ali će se znatno proširiti i integrirati s postojećim modelima korištenja računarstva. Slikom 3.7 prikazana su, prema [46], predviđanja svjetskog širenja oblačnog računarstva u sljedećem kratkoročnom razdoblju (u milijardama dolara).

I drugi istraživači imaju slična predviđanja [100]: "*Očekuje se da će u idućih 10 godina servisno orijentirana rješenja biti glavno sredstvo isporuke informacijskih funkcionalnosti osobama i organizacijama, i to softverskih aplikacija, web servisa, stolnog računarstva za osobe i poslovne svrhe te visokoperformansnog računarstva.*"



Slika 3.7: ***Predviđanje udjela oblačnog računarstva u ukupnoj IT potrošnji***

Oblačno računarstvo će u narednom razdoblju biti od posebnog značaja za dvije kategorije korisnika:

- Manjim tvrtkama koje će tek biti osnovane ili tvrtke koje imaju malu ICT infrastrukturu te ne žele ili ne mogu provesti velika kapitalna ulaganja u njezino povećanje. To će im omogućiti da raspoloživ kapital investiraju u svoje temeljne poslovne aktivnosti.
- Srednjim i velikim tvrtkama koje žele da njihova IT infrastruktura podnese vršna opterećenja bez dodatnih kapitalnih ulaganja ili pak samo tijekom razvoja i testiranja softvera trebaju dodatne računalne resurse. Koncept oblačnog računarstva će omogućiti takvim tvrtkama da kontroliraju svoje ICT troškove.

Što se više budu širili i razvijali otvoreni standardi za oblačnu interoperabilnost, to će se koncept oblačnog računarstva više koristiti. Problem sigurnosti, kao rizik koji se još uvijek povezuje s ovom tehnologijom, bit će riješen na sasvim zadovoljavajući način jer se, na kraju krajeva, radi samo o čisto tehničkom pitanju.

3.11 Sažetak

U ovom poglavlju izrečena je jasna definicija oblačnog računarstva kao skupa mrežnih servisa namijenjenih pružanju raznih računalnih usluga (od digitalne pohrane podataka do izvođenja zahtjevnih aplikacija) kojeg odlikuju slijedeće osobine: dostupnost temeljem standardnih informacijskih i komunikacijskih tehnologija, utemeljenost na računalnoj virtualizaciji, proširivost, naplata prema korištenju te korištenje i skaliranje na zahtjev.

Opisane su servisne paradigme IaaS, PaaS, SaaS, HaaS, AaaS i ostale koje se praktično pojavljuju te je obrazloženo da se one najbolje koriste upravo kroz računalne oblake, zbog općenite isplativosti korištenja oblačnih računalnih resursa. Ta isplativost kao ekonomska kategorija se očituje se kroz dva svojstva računalnih oblaka: elastičnost u zaposjedanju računalnih resursa i optimalnu nabavu. Elastičnost znači da se računalni resursi mogu angažirati kada su potrebni i otpustiti kada se ta potreba smanji. Optimalna nabava znači da ne treba nabavljati one računalne resurse koji služe samo pokrivanju vršnih opterećenja.

Analizirana su osnovna svojstva aplikacija koje posebice dobro mogu koristiti prednosti računalnih oblaka. Zajedničko tim aplikacijama jest mogućnost paralelizacije izvođenja i/ili podrška djelatnostima u kojima se pojavljuju vršna opterećenja. Opisane su osnovne tehničke karakteristike podatkovnih centara – mjesta u kojima su fizički smješteni računalni oblaci. Utvrđene su prednosti i nedostaci uporabe računalnih oblaka te procijenjeni pravci daljnjeg razvoja i predviđanja o tržišnom širenju. Naznačeni su problemi koji će se rješavati u ovom radu, među kojima je najbitniji stvarna isplativost računalnih oblika s korisničkog i poslužiteljevog gledišta.

4 Uporabna svojstva i očekivani učinci oblačnog računarstva

U ovom poglavlju bit će pobliže opisana uporabna svojstva oblačnog računarstva s naglaskom na razlike u odnosu na klasične serverske farme u privatnim podatkovnim centrima. Učinci koji slijede iz tih uporabnih svojstava stvaraju primarnu razliku računalnih oblaka spram klasičnih tehnologija. Tehnologija oblačnog računarstva, kako je to opisano u prethodnom poglavlju, sama po sebi ne sadrži u tehničkom smislu neka nova rješenja koja već nisu ugrađena u klasične serverske farme. Oblačno računarstvo zapravo znači nov i efikasniji način korištenja takvih farmi.

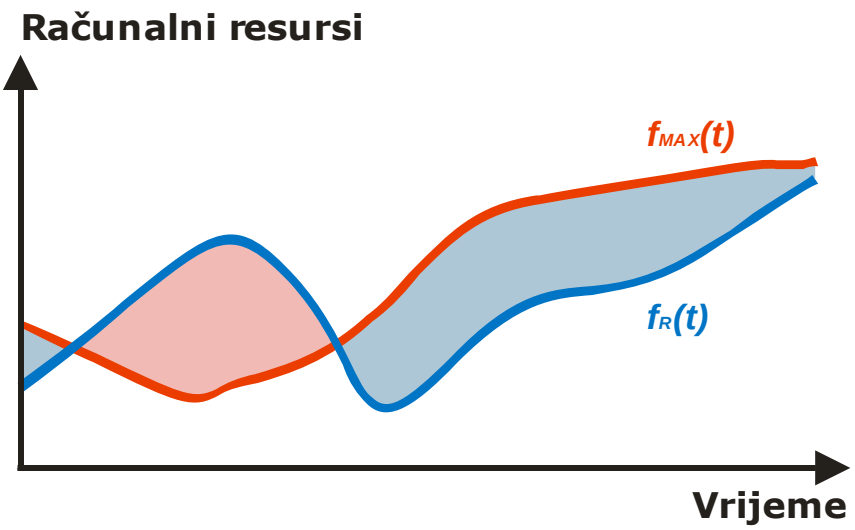
4.1 Elastičnost

Kako je ukratko rečeno u prethodnom poglavlju, elastičnost je svojstvo korištenja računalnih resursa u oblaku na način da se angažman tih resursa može vrlo precizno podešavati prema poslovnim potrebama, to jest povećati kada se pojavi veća potreba za njima, a smanjiti kada te potrebe nestane [94]. To je posebice bitno za aplikacije koje imaju vršna opterećenja, ali je potencijalno važno i za sve one slučajeve kada vlasnik aplikacije tu aplikaciju želi učinkovito koristiti, a da se istodobno ne opterećuje kapitalnim ulaganjima u računalnu opremu.

Da bi se podrobnije opisala svojstva i učinci elastičnosti, prvo treba zamisliti dva slučaja nabave računalnih resursa: nabavu za pokrivanje vršnih opterećenja i nabavu uz zanemarivanje vršnih opterećenja. Problematika ta dva slučaja vizualno će se prikazati grafom slijedećih svojstava:

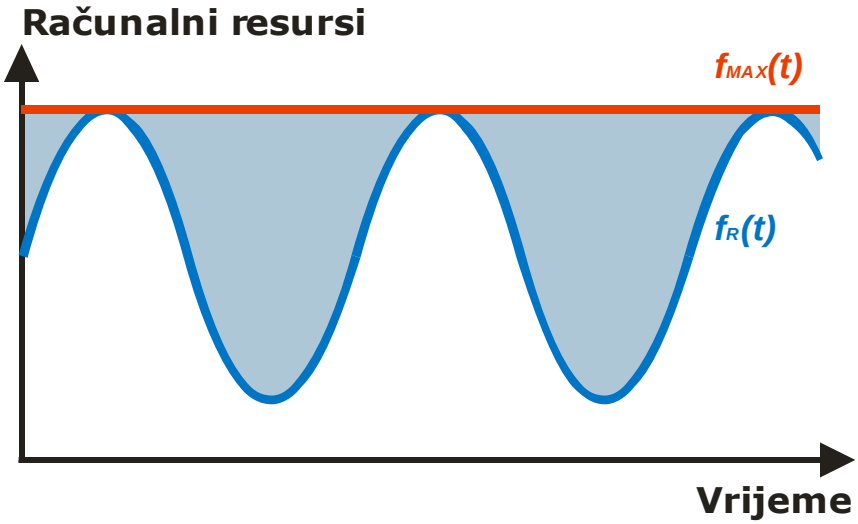
- Na apscisi je vrijeme izraženo u danima.
- Na ordinati su dvije funkcije (izražene kao funkcija vremena): funkcija potrebe za računalnim resursima $f_R(t)$ čiji je graf nacrtan plavom bojom i funkcija maksimalne raspoloživosti računalnih resursa $f_{MAX}(t)$ čiji je graf nacrtan crvenom bojom.
- Osjenčane površine pokazuju višak ili manjak računalnih resursa: višak je sve ono što je iznad $f_R(t)$, a ispod $f_{MAX}(t)$ i osjenčano je plavim, a manjak je sve ono što je ispod $f_R(t)$, a iznad $f_{MAX}(t)$ i osjenčano je crvenom bojom.

Ovi su temeljni odnosi prikazani su orijentacijskim grafom na slici 4.1. Taj graf ne prikazuje, dakle, niti jedan konkretan slučaj ovih odnosa, nego služi kao tumač za ispravno razumijevanje grafova konkretnih slučajeva.



Slika 4.1: ***Odnos potrebnih i raspoloživih računalnih resursa***

Kada se računalna oprema nabavlja za pokrivanje stalnih potreba, ali i vršnih opterećenja, onda odnos potrebnih i raspoloživih računalnih resursa izgleda prema [5] kao slici 4.2.

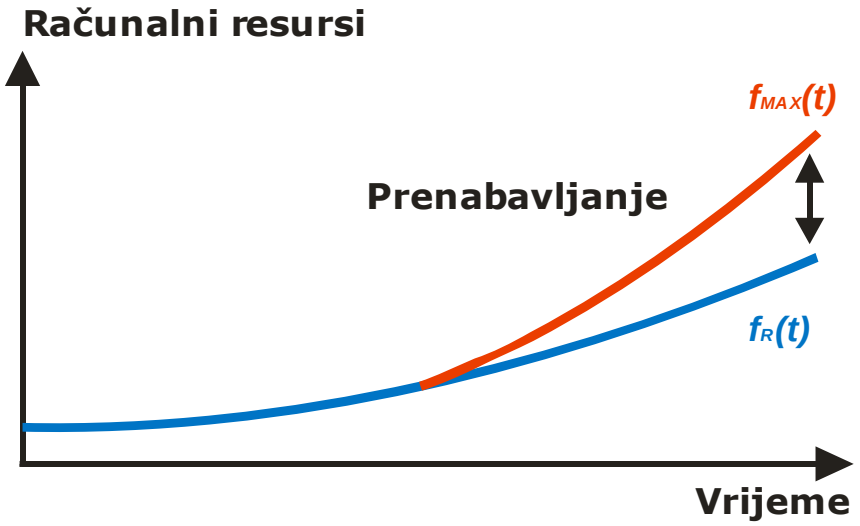


Slika 4.2: ***Nabava računalnih resursa za pokriće vršnih opterećenja***

Sa slike je vidljivo da je značajan dio vlastitih računalnih resursa veći dio vremena neiskorišten. Može se reći da predstavljaju neiskorišteno kapitalno ulaganje. Takvo stanje je prirodna posljedica nabave računala i drugog hardvera u količini koja je izražena cjelobrojnim višekratnikom jediničnog resursa, odnosno nije moguće kupiti samo dio jediničnog resursa. Drugim riječima, vlastito nabavljanje računalnih resursa ima slabu elastičnost.

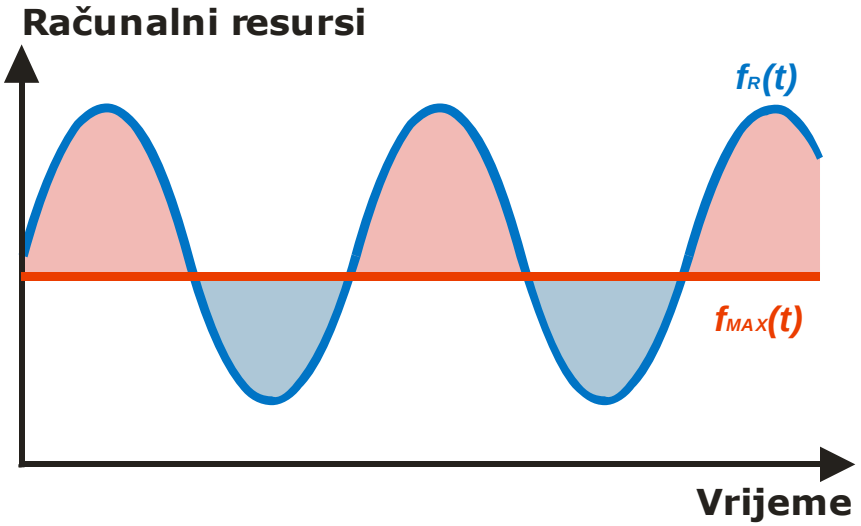
Poseban problem kod ovog načina nabavljanja jest netočna prognoza rasta potrebe za računalnim resursima. Da bi se isključio utjecaj nepouzdane prognoze nabavlja se u količinama koje ne odgovaraju očekivanoj srednjoj potrebi već najvećoj mogućoj potrebi. Taj fenomen je to izraženiji što je planski horizont dulji, što je prikazano slikom 4.3. U tom slučaju još se dodatno povećavaju ulaganja u resurse koji ne će biti u potpunosti iskorišteni tijekom

većeg dijela svog životnog ciklusa, odnosno dolazi do prekapacitiranja zbog prekomjernih količina nabavljanja (prenabavljanje ili *overprovisioning*). Takvo se ponašanje može registrirati kod velikog broja voditelja informatike (CIO), osobito u državnoj upravi i javnim službama.

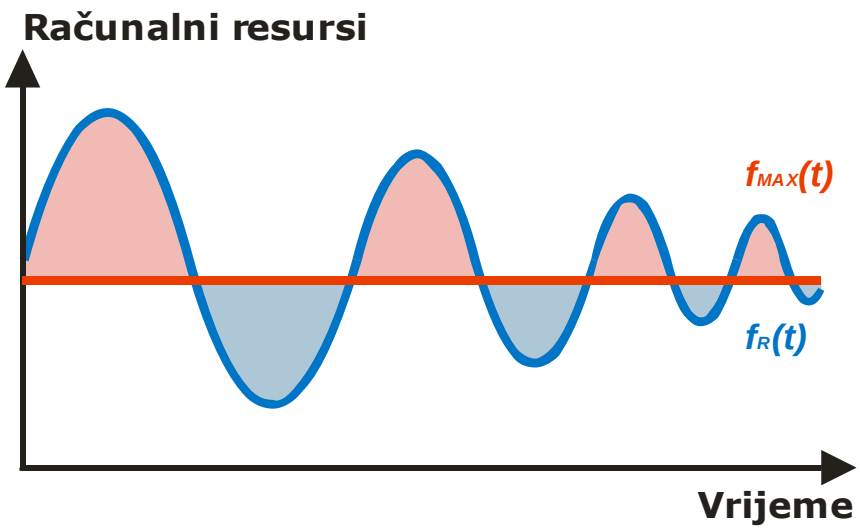


Slika 4.3: ***Nesigurna prognoza rasta potrebe za računalnim resursima***

Ako pak se problem rješavanja vršnih opterećenja zanemari, a računa isključivo prema očekivanim prosječnim godišnjim potrebama za računalnim resursima, dolazi do fenomena podnabavljanja (*underprovisioning*) koji rezultira potkapacitiranošću [5]. To će prema slici 4.4 prouzročiti gubitak zbog neposluženih korisnika. Ako takvo stanje potraje dulje, onda će neposluženi korisnici tražiti druge mogućnosti za pribavljanje nedovoljnih računalnih resursa, kako je to prikazano slikom 4.5, što neminovno smanjuje poslovnu reputaciju pružatelja računalnih usluga.



Slika 4.4: ***Gubitak neposluženih korisnika (bez ostalih posljedica)***



Slika 4.5: ***Gubitak neposluženih korisnika uz stvaranje lošeg publiciteta***

Za razliku od cijene prenabavljanja, cijenu podnabavljanja je teže izmjeriti. U jednostavnom slučaju, kakav je prikazan slikom 4.4, određen broj korisnika naprosto neće moći u svom radu koristiti aplikaciju koja im je potrebna. Ako je u pitanju aplikacija koja nema alternativu, primjerice neki korporativni program, to će rezultirati povećanim nezadovoljstvom korisnika – radnika i uprave – zbog smanjene produktivnosti. No, u drugom slučaju, tipično za javne aplikacije poput socijalnih mreža koje imaju alternativu i konkurenciju, dogodit će se slučaj prikazan slikom 4.5 – ne samo povećano nezadovoljstvo korisnika, nego i njihovo napuštanje aplikacije i prelazak konkurenciji.

Svi navedeni primjeri ilustriraju slabu elastičnost računalnih resursa ako se oni nabavljaju za vlastite potrebe. Može se zaključiti kako bi povećana elastičnost bila iznimno dobra mjera uštede za slučajeve koji su prikazani slikama 4.2 i 4.3, jer bi ukinula gubitke prenabavljanja. Istovremeno, povećana elastičnost je praktički poslovna nužnost (*business requirement*) za drugi i treći slučaj, prikazan slikama 4.4 i 4.5 jer ukida gubitke podnabavljanja. Računalni oblaci mogli bi biti idealno sredstvo osiguravanja željene elastičnosti. Elastičnost uz nabavljanje računalne opreme u vlasništvo je vrlo teško postići kako zbog nedostatka kapitala tako i stoga što bi trebalo osigurati potrebne stručnjake koji mogu nabavljene računalne resurse staviti u pogon u vrlo kratkom vremenu te istovremeno imati nevjerojatno aktivno tržište rabljenog hardvera. Da li je pak elastičnost dovoljno jak ekonomski argument za odabir računalnog oblaka umjesto nabave vlastitih računalnih resursa, bit će temeljitije istraženo u poglavlju 6.

4.2 Optimalna nabava

Kao posljedica razumijevanja činjenica o odnosima potrebnog i raspoloživog računalnog kapaciteta, izloženih u prethodnom potpoglavlju, proizlazi sljedeći zaključak: samoposlužnost na zahtjev (kao definicijsko i imanentno svojstvo računalnih oblaka) i elastičnost osiguravaju optimalnu nabavu – pravovremeni angažman računalnih resursa kada su potrebni i njihovo otpuštanje kada više nisu potrebni. To može dovesti do uštede čak i onda kada je cijena oblačnog najma, izračunata na temelju angažiranih resursa po vremenskoj jedinici, veća nego što je na isti način izračunata cijena rada na vlastitim računalima.

Na primjer, pretpostavimo da vršno opterećenje zahtijeva angažman 500 računala po satu, uobičajeno opterećenje 100, dok je pri prosječnom opterećenju zauzeto oko 300 računala po satu. Prosječni troškovi takve serverske farme u tom slučaju iznose 7200 [računalo·sati] dnevno. Ako se pak trebaju pokriti i vršna opterećenja, onda treba stalno imati na raspolaganju svih 500 računala uklopljenih u vlastitu serversku farmu, pri čemu su neka od tih računala dulje vrijeme, tijekom uobičajenog opterećenja, neaktivna – ali i tada uzrokuju troškove! U takvom slučaju stalno se plaća 12.000 [računalo·sati] dnevno, odnosno 1,67 puta više od onoga koliko prosječno treba.

Iz toga slijedi da bi najam u komercijalnom oblaku mogao biti isplativiji od korištenja vlastitih računala [6] na klasičan način čak i ako je cijena tamošnjih računalnih resursa u vremenu kroz neki standardni amortizacijski period (obično 3 do 5 godina) do 1,67 puta veća od amortizirane cijene vlastitih ulaganja u hardver kroz taj isti period! Pri tome treba imati na umu da je situacija lošija u slučaju nepredviđenih vršnih opterećenja stoga što nabavka i postavljanje vlastite opreme može potrajati, a neiskorištenost te opreme u periodima normalnog opterećenja i dalje postoji. Sve te prednosti dobivaju pravo značenje kada ih smjestimo u matematičke okvire. Navedeno rezoniranje može se matematički izraziti [5] sljedećom nejednadžbom:

$$korVrijeme_{OBLAK} \cdot (prihod - trošak_{OBLAK}) \geq korVrijeme_{PRIVAT} \cdot (prihod - \frac{trošak_{PRIVAT}}{iskorištenje})$$

Varijable nejednadžbe i njihova značenja su:

- $korVrijeme_{OBLAK}$ – korisničko vrijeme rada u oblaku
- $korVrijeme_{PRIVAT}$ – korisničko vrijeme rada u vlastitom podatkovnom centru
- $trošak_{OBLAK}$ – troškovi korištenja oblaka
- $trošak_{PRIVAT}$ – troškovi korištenja vlastitog podatkovnog centra

- $iskorištenje$ – prosječno iskorištenje računalnih resursa u vlastitom podatkovnom centru

Dakle, ako vrijednosti varijabli u nekom pojedinačnom slučaju zadovoljavaju gornju opću nejednadžbu, mogli bi biti ispunjeni ekonomski kriteriji isplativosti za korištenje usluga oblačnog računarstva. Jasno, ekonomski kriteriji ne moraju biti jedini koji se postavljaju za korištenje računalnih resursa. Sigurnosni, politički te, na kraju krajeva, tehnički uvjeti (to jest stvarne mogućnosti raspoloživih oblačnih platformi, komercijalnih, privatnih i akademskih) su često od presudne važnosti. Osim toga, gornja nejednadžba je općenito ekskluzivna, jer isplativost korištenja oblaka definira u smislu da se oblak ili uopće ne koristi, ili da se koristi isključivo, bez značajnijeg učešća vlastitih računala. Takvom konceptu korištenja računalnih oblaka nedostaje hibridni pristup. Usavršavanje pokazatelja isplativosti oblačnog računarstva, kao što je gornja nejednadžba, te iznalaženje takvih aplikacija koje je moguće izvoditi u oblaku, uz smanjenje cijene obračunske jedinice za utrošak resursa, primarni su ciljevi ovoga rada koji će biti ostvareni u sljedećim poglavljima.

4.3 Ostale ekonomske prednosti

Ostale ekonomske prednosti oblačnog računarstva posljedica su intrinzičnih definicijskih svojstava te prethodno opisane elastičnosti i optimalne nabave [15, 83, 95]:

- Smanjenje kapitalnih troškova – zbog plaćanja po utrošku i korištenja virtualizacije u oblaku korisnici računalnih resursa ne moraju unaprijed trošiti velika financijska sredstva za nabavu hardvera i softvera. Drugim riječima, kapitalni troškovi koji u slučaju nabave vlastitih računalnih resursa (i njihovog korištenja na klasičan način) nastaju prije početka njihovog korištenja, a također su i rizični, raspodjeljuju se na postepene i nadzirane operativne troškove najma u komercijalnom oblaku.
- Smanjenje operativnih troškova pogona računalne infrastrukture zbog visoke automatizacije kod angažiranja računalnih resursa iz oblaka.
- Olakšana uporaba masivnog paralelizma pri obavljanju vrlo zahtjevnih računalnih poslova zbog virtualizacije oblačnih resursa, izrazito distribuirane građe oblaka i tomu prilagođenog korištenja oblačnih resursa u obliku zakupa određenog broja virtualnih računala.

4.4 Uvod u ekonomiku komercijalnog oblačnog računarstva

U prethodnom dijelu razmotreni su neki temeljni ekonomski aspekti korištenja računalnih oblaka, kao što je plaćanje računalnih resursa po potrošnji umjesto kapitalnog ulaganja u vlastitu računalnu opremu. Utvrđeno je da bi takav oblik ulaganja u informatičke resurse mogao biti iznimno zanimljiv malim i srednjim tvrtkama, koje obično nemaju potrebne ljudske i financijske resurse za kapitalna ulaganja u hardver i softver, ali i velikim firmama za slučajeve vršnih opterećenja, kao što je prikazano u prethodnim točkama ovog poglavlja. Naime, možda je najvažnija (te uglavnom neizrečena) prednost oblačnog računarstva to što se njime, uz suhoparne "računalne resurse" – virtualna računala sa sistemskim softverom te manje ili više uslužnih programa – zapravo unajmljuje prethodno konfigurirana serverska farma u koju su ti resursi uklopljeni! Za krajnjeg korisnika oblačni resursi imaju istu funkcionalnost kao i vlastita serverska farma, za čiju uspostavu korisnik računalnog oblaka nije utrošio ni vrijeme, ni ljudske resurse. Na taj način tvrtke mogu uštedjeti stoga što se njihovo osoblje može baviti temeljnim poslom, a većina sistem-inženjerskih poslova, uključivo i velik broj poslova usuglašavanja i instalacije raznih hardverskih i softverskih komponenti, može se prebaciti (*outsourcing*) na vanjskog izvođača kojemu je pak to temeljni posao za kojega se specijalizirao i obavlja ga s maksimalnim kompetencijama.

No, kako je napomenuto u trećem poglavlju, svatko tko razmišlja o prebacivanju vlastite računalne infrastrukture u komercijalni oblak mora moći izračunati da li mu je tako nešto uopće isplativo spram kupnje, odnosno kapitalnog ulaganja, u vlastite računalne resurse (te njihovog korištenja u sklopu klasičnih serverskih farmi). Potencijalni korisnik oblačnog računarstva mora biti u stanju objektivno i egzaktno usporediti komercijalne cijene iz ponude pružatelja usluga oblačnog računarstva s troškovima nabave i održavanja vlastite opreme istih mogućnosti. U tim procjenama važnu ulogu imaju neke stavke koje možda jesu poznate inženjeru računarstva ili ekonomistu, ali ne moraju biti poznate prosječnom korisniku.

Prvo i osnovno načelo korektne ekonomske analize isplativosti najma komercijalnih računalnih oblaka spram kapitalnog ulaganja u vlastita računala i softver jest da se vrijednost novca mijenja tijekom vremena. Sto kuna danas, njihova sadašnja vrijednost (*present value*), vrijedi više od sto kuna u budućnosti (*future value*). Razlozi zašto treba računati s vremenskom promjenom efektivne vrijednosti novca su inflacija, kao i mogućnost da isti novac uložen u neki drugi poduhvat može ostvariti veći prinos. Uobičajeno je da se prinos od ulaganja u neki poduhvat uspoređuje s prinosom koji bi se mogao ostvariti od bankovnih kamata na isti iznos.

U financijskim modelima toka novca mora se, općenito gledajući, uzimati u obzir i amortizacija nabavljene opreme, odnosno smanjivanje njezine stvarne vrijednosti tijekom

tehnološkog vijeka trajanja. Ako se ti modeli odnose na računalnu opremu, onda treba računati i s djelovanjem Mooreovog zakona. Temeljna formulacija tog zakona kojeg je 1965. godine izrekao Gordon E. Moore, suosnivač Intela, glasi [67]: *"Količina tranzistora koju je moguće postaviti na zadanu površinu integriranog kruga uz konstantnu cijenu udvostručuje se svake dvije godine."* Budući da količina tranzistora na čipu okvirno određuje snagu čipa te, posredno, samog računala koje se gradi od čipova, Mooreov zakon često se izgovara na preformulirani, ali suštinski točan način: *"Snaga računala uz konstantnu cijenu (i dimenzije) udvostručuje se svake dvije godine."* To znači da se uz cijenu početnog ulaganja u računalnu opremu mora uračunavati i dogradnja serverske farme ili vlastitih računala kako bi i u budućnosti bio na raspolaganju isti računalni kapacitet kakav će u tom trenutku biti moguće unajmiti. Takva je dogradnja za slučaj najma računalnih resursa briga oblačnog poslužitelja, implicitno već uračunata u cijenu oblačnih usluga. Međutim, da bi se ispravno procijenilo da li je ta cijena prevelika u odnosu na vlastitu nabavu i dogradnju, mora se uzimati u obzir efekt koji daje Mooreov zakon.

Da bi se cjelovito razmotrio ekonomski aspekt oblačnog računarstva te analizirala njegova isplativost, moraju se poznavati stvarni troškovi fizičke izgradnje oblačnog podatkovnog centra (*cloud data center*), naročito u usporedbi sa sličnim podatkovni centrom u vlasništvu kompanije (*enterprise data center*), koje ga neće koristiti za iznajmljivanje već radi potpore u obavljanju svojih temeljnih aktivnosti.

Prema [38] amortizirani troškovi oblačnog podatkovnog centra, s oko 50 tisuća računala i ukupnim operativnim troškom od oko 120 milijuna dolara godišnje, strukturirani su kao u tablici 4-I. Prikazani udjeli pojedinih komponenata troška u ukupnom trošku izračunati su uz pretpostavku da je prosječna cijena jednog računala oko 3.000 $, trošak kapitala 5% godišnje, a da se oprema amortizira (u zavisnosti od vrste) u razdoblju od tri do pet godina.

Tablica 4-I: ***Struktura troškova oblačnog podatkovnog centra***

Amortizirana cijena	*Komponente*	*Podkomponente*
~45%	računala	računalne komponente
~25%	infrastruktura	hlađenje, rasvjeta, i drugo
~15%	električna energija	napajanje hardvera
~15%	mrežna oprema	ruteri, linkovi, mrežni promet

Troškovi ljudskog rada u ovakvim se izračunima temelje na empirijski provjerenim podacima da jedan administrator održava oko 100 računala u korporativnom podatkovnom centru, a oko 1000 računala u oblačnom podatkovnom centru. Stoga je trošak osoblja u tablici 4-I zanemaren jer iznosi ispod 5% ukupnih troškova. Takvi razmjeri smanjenja osoblja mogući su stoga što je oblačni centar izvorno zamišljen i izveden uz najveću moguću automatizaciju i

standardizaciju, kojom se otklanja i znatan dio ljudskih greški, dok su klasični korporativni podatkovni centri samo djelomično automatizirani.

Nadalje, oblačni centri iskorištavaju do maksimuma ekonomiju razmjera (*economy of scale*). Ekonomija razmjera je sintagma za pojavu smanjenja prosječne cijene proizvoda ili usluge što se više povećava količina tog isporučenog proizvoda ili usluge.

Naime, svaki proizvod ili usluga ima fiksne troškove neovisne o poslovnim aktivnostima. Na primjer, oblačni centar od 50 tisuća računala ima neki fiksni trošak izgradnje i održavanja infrastrukture, koji je označen s C_{FIX}. Prosječni fiksni trošak (u našem poslovnom žargonu se često naziva hladnim pogonom), označen s C_{AVG}, je trošak po jednom proizvodu – računalu. Broj proizvoda – računala – označen je s r. Iz svega ovoga slijedi jednadžba:

$$C_{AVG} = \frac{C_{FIX}}{r}$$

Jednadžba pokazuje osnovno svojstvo svih ekonomija razmjera, pa tako i oblačnih centara, prema kojem je masovna nabava jeftinija po proizvodu nego što je to slučaj s manjim nabavama. Drugim riječima, oblačni centri mogu u konačnici koštati manje nego skup poduzetničkih centara istoga računalnog kapaciteta.

Osim spomenute masovne nabave, smanjenje cijene postiže se i standardizacijom hardvera. To je pojava koja nije tipična samo za računalne oblake, nego i općenito pri nabavi vlastitih računalnih resursa. Praktički svi oblačni poslužitelji nude računala temeljena na x86 procesorskom standardu (uključivo i kompatibilni 64-bitni x86-64) uz izbor od jednog ili dva operacijska sustava – Linux ili Windows.

Nadalje, značajan dio pune cijene vlasništva (*total cost of ownership* – TCO) oblačnog centra čine troškovi električne energije, trenutno između 15% i 20% ukupnih troškova [93]. Oblačni centri s vrlo velikim brojem računala, koji mogu biti izvedeni i kao mobilni, mogu se smjestiti u regijama jeftinije električne energije.

Kod raspodjele opterećenja u oblačnim centrima se danas koristi horizontalno skaliranje (*scale-out*), za razliku od vertikalnog (*scale-up*) skaliranja u većini poduzetničkih centara. U horizontalnom skaliranju, poslovno opterećenje (*workload*) se dijeli na veći broj relativno jednostavnih računala. Pri vertikalnom skaliranju opterećenje se dijeli na manji broj skupljih i moćnijih računala. Vertikalno skaliranje je često i jedino moguće kada je fizički prostor namijenjen podatkovnom centru dimenzijski ograničen.

Svojedobno je prednost horizontalnog skaliranja pokazao Google sa svojim prvim produkcijskim gridovima namijenjenima pokretanju vlastite tražilice, u kojima se koristio običan hardver s police [89]. Takav pristup imaju i danas [7, 113], kao i Microsoft za svoje oblačne centre [38].

Detaljni troškovi i struktura cijene oblačnog podatkovnog centra bit će pobliže analizirani u poglavlju 6 ovoga rada gdje će biti uspoređeni s odgovarajućim cijenama kapitalnoga ulaganja u vlastite resurse, a radi analize prednosti i nedostataka svakog od tih pristupa korištenja računalnih resursa. To će omogućiti objektivnu procjenu kada i pod kojim uvjetima se nekom korisniku isplati koristiti komercijalne računalne oblake. Napominjemo da će ovakve analize biti uključene u postupak dokazivanja hipoteze 2 a njihovi rezultati korišteni kod izrade smjernica za isplativost ulaganja u oblačno računarstvo.

Za daljnju detaljizaciju ekonomike oblačnog računarstva bit će korisno razmotriti ponudu tri pružatelja komercijalnih usluga oblačnog računarstva koji su najdulje prisutni na ovom tržištu:

- Microsoft – Azure oblak, čiji se detaljni cjenik usluga može naći na web stranici http://microsoft.com/windowsazure/pricing
- Amazon – Amazon Web Services (AWS) oblačne usluge, detaljni cjenik se nalazi na http://aws.amazon.com
- Google – Google AppEngine oblak, čiji se detaljni cjenik oblačnih usluga nalazi na web stranici http://appengine.google.com

4.5 Usporedba komercijalnih oblačnih ponuda

Prikupljeni podaci o vrstama i cijenama oblačnih usluga te usporedba koja slijedi odnosi se na 2010. godinu. Bitno je sljedeće:

- Većina oblačnih poslužitelja, uključivo Microsoft i Amazon, temelje naplatu najmljenih računala na ukupnoj **količini procesorskih jezgri** u svim zakupljenim računalima. Zauzetost procesora u tim računalima, složenost aplikacija koje se na njima izvode i iskorištenost tih aplikacija [19, 42] nisu parametri koji ulaze u obračun. Ostala svojstva zakupljenih računala, poput kapaciteta memorije i diskova, proporcionalna su količini procesorskih jezgri, kako se može zaključiti već iz tablice 2-I. Osnovna vremenska jedinica za obračun je sat. Drugim riječima, korisnik zakupljuje računala tijekom određenog vremena, a pružatelj usluge oblačnog računarstva mu obračunava ukupnu količinu procesorskih jezgri u zakupljenim računalima.
- Manji broj oblačnih poslužitelja, primjerice Google, naplatu najmljenih računala temelje na stvarnom **vremenu rada** zakupljenih procesora. Drugim riječima, mirovanje (*idle*) zakupljenog procesora se **ne** naplaćuje, za razliku od Microsoftovog i Amazonovog pristupa. Korisnik izravno zakupljuje samo procesore, dok se ostala pripadna računalna svojstva, poput kapaciteta memorije i diskova, dodjeljuju

implicitno, ali proporcionalno količini zakupljenih procesora (u tom dijelu to je u suštini slično Microsoftovom i Amazonovom pristupu).

U daljnjem tekstu, često će se za najmljena računala sinonimno koristiti pojmovi *"računalo"* i **"računalna instanca"** (*computing instance*). Radi preglednosti, prikazane su cijene za klasični, *pay as you go* najam računalnih instanci i pohrambenih resursa, iako neki oblačni poslužitelji, primjerice Amazon, imaju i mogućnost zakupa rezervirane količine računalnih i drugih resursa u oblaku na nekoliko godina, takozvanu *reserved instances* ponudu. Međutim, takve usluge nisu od značaja za razmatranje u ovom radu jer više odgovaraju klasičnom kapitalnom ulaganju u vlastite resurse, s obzirom da se naplaćuju unaprijed (*one-time, up-front payment*). Nadalje, većina oblačnih poslužitelja, posebice Amazon, nudi mnoštvo raznih manjih usluga od kojih neke čak i ne spadaju u oblačno računarstvo, ali se nude i prodaju zajedno s oblačnim uslugama. Detaljnija analiza tih vrsta usluga nalazi se u poglavlju 6.

4.5.1 Amazon

Amazon svoj oblak pokreće iz tri fizička oblačna centra: dva u Americi, u Virginiji i Kaliforniji, te jednog u Europskoj Uniji, u Irskoj. Najmiti se mogu razne veličine računalnih instanci u okviru usluge zvane Amazon Elastic Compute Cloud – EC2. Usluga kod Amazona može biti:

- Standardna, dalje podijeljena na mikro, malu, veliku i jako veliku – *micro*, *small*, *large* i *extra large*.
- S puno RAM-a – *Double Extra Large*, *Quadruple Extra Large*. U daljnjem tekstu i tablici 4-II te će instance biti označene HM (*High Memory*) nastavkom.
- S povećanim brojem procesorskih jezgri to jest većom procesorskom snagom – *Medium*, *Extra Large*. U daljnjem tekstu te će instance biti označene HP (*High Performance*) nastavkom.

Nadalje, svaka instanca može biti temeljena na Windows Server operacijskom sustavu ili na Linuxu. Windows računala su skuplja. Cijena najma ponajviše ovisi o geografskom smještaju podatkovnog centra to jest o lokalnoj cijeni električne energije, a prikazana je tablicom 4-II. Niža se vrijednost odnosi se na najmanju cijenu (u sjevernoj Virginiji), a viša na Irsku.

Tablica 4-II: ***Struktura cijene za EC2***

Veličina instance	*Cijena – Linux [$/h]*	*Cijena – Windows [$/h]*
Micro	0,02 – 0,025	0,03 – 0,035
Small (zadana)	0,085 – 0,095	0,12
Large	0,34 – 0,38	0,48
Extra Large	0,68 – 0,76	0,96
Extra Large – HM	0,5 – 0,57	0,62
Double Extra Large - HM	1 – 1,14	1,24
Quadruple Extra Large – HM	2 – 2,28	2,48
Medium – HP	0,17 – 0,19	0,29
Extra Large HP	0,68 – 0,76	1,16

U sklopu Amazonovog EC2 postoje usluge za mapiranje IP adresa, automatsko skaliranje, balans opterećenja i komunikaciju porukama (Simple Queue Service – SQS) koje se posebno naplaćuju.

U ponudi je relacijska pohrana podataka, nazvana Amazon Relational Database Service – RDS [3], temeljena na MySQL sustavu za upravljanje bazama podataka, te dvije varijante nerelacijske pohrane: Simple DB i Simple Storage Service – S3. Zajedničko naplatama svih tih vrsta pohrane jest:

- Obračunava se količina pohranjenih podataka.
- Naplaćuje se ulazni i izlazni podatkovni transfer (*data transfer*) – količina podataka poslanih u oblak i preuzetih iz oblaka nakon obrade.

Razlike u obračunu cijene raznih vrsta Amazonove podatkovne pohrane su sljedeće:

- Naplata Simple DB temelji se još i na količini računala najmljenih za upravljanje uslugom. Procesorska snaga analogna je 1,7 GHz Intel Xeon procesoru.
- Naplata S3 uopće nema nikakvu računalnu niti procesorsku komponentu, nego količinu CRU (*create, read, update*) operacija nad bazom podataka. Brisanja su besplatna.
- Naplata RDS uzima u obzir i procesorsku snagu, kao u Simple DB, i količinu CRU operacija nad bazom podataka.

Nadalje, cijene pohrane mogu znatno varirati u odnosu na sigurnosnu konfiguraciju. Na primjer, u sklopu RDS pohrane podaci se mogu držati u jednom podatkovnom centru ili u više njih. S3 pohrana može biti najmljena uz standardno jamstvo raspoloživosti od 99,99%, što znači da će usluga biti nedostupna maksimalno 4,32 minute mjesečno (za 30-dnevni mjesec) odnosno praktički uvijek. Uz jamstvo raspoloživosti Amazon navodi i jamstvo za

trajnost čuvanja podataka (*durability*) od 99,999999999% (http://aws.amazon.com/s3). Osim toga, cijena pohrane po jediničnoj mjeri pada s ukupnom veličinom pohrane. Zbog svega toga su u tablici 4-III prikazane cijene standardne usluge za sjevernu Virginiju i Irsku.

Tablica 4-III: ***Cijene standardnih RDS, Simple DB i S3***

	Veličina pohrane [$/GB · mj]	*Prijenos podataka [$/GB]*	*Količina CRU-ova [CRU/1000]*	*Računalo [$/sat]*
Simple DB	0,25 – 0,275	0,1 – 0,15	-	0,14 – 0,154
S3	0,055 – 0,165	0,08 – 0,15	0,001 – 0,0011 (R) 0,01 – 0,011 (C,U)	-
RDS	0,1 – 0,11	0,1 – 0,15	0,0001	0,11 – 2,96

4.5.2 Google

Google se ponešto razlikuje glede obračuna korištenja oblačnih resursa stoga što se računalne instance nigdje eksplicitno ne unajmljuju. Unajmljuje se i naplaćuje ukupno efektivno procesorsko radno vrijeme pri čemu se AppEngine brine o skaliranju unutar zadanih ograničenja. Općenito govoreći, Google ima daleko najjednostavniju naplatu oblačnih resursa i to ne samo od poslužitelja koji su ovdje analizirani, nego i općenito.

- Procesorska snaga (ekvivalentna x86 procesoru od 1,2 GHz) plaća se 0.1 [$/h], pri čemu se vrijeme mirovanja, kako je već rečeno, ne naplaćuje.
- Pohrana podataka je zasad moguća samo u nerelacijskom obliku po cijeni 0,15 [$/GB·mj]. Ulaz podataka u oblak naplaćuje se 0,1 [$/GB].
- Izlaz podataka iz oblaka naplaćuje se 0,12 [$/GB].

Uz to, svaka Googleova oblačna usluga ima i besplatnu kvotu: 6,5 računalnih sati, 500 MB pohrane te po 1 GB ulaznog i izlaznog prometa.

4.5.3 Microsoft

Microsoftov pojam Windows Azure (www.microsoft.com/windowsazure) ima dvostruko značenje:

- oblačni nadglednički operacijski sustav ili
- Microsoftova usluga oblačnog računarstva.

Projektni i eksperimentalni dio ovoga rada bit će izveden na Windows Azure pa je stoga potrebno nešto detaljnije [16, 50] upoznati ponudu te platforme, cijene i osnovna tehnička svojstva kako bi se ti podaci mogli staviti u kontekst svega do sada izrečenoga o oblačnom računarstvu i njegovoj ekonomici. Analizirane cijene su iz svibnja 2010. godine i odnose se na općenitu ponudu u Europi i Americi, a mogu biti modificirane (snižene) u okviru raznih posebnih ponuda kojima Microsoft nastoji privući korisnike (primjerice BizSpark program).

Postoji i besplatna ponuda namijenjena malim razvojnim tvrtkama i učenju. Microsoft u sklopu Azure oblaka nudi tri grupe oblačnih usluga:

- Windows Azure – uključuje najam računala, nerelacijsku pohranu podataka i komunikaciju porukama
- SQL Azure - relacijska baza podataka u oblaku temeljena na SQL Serveru i
- AppFabric – servisna sabirnica za povezivanje raznih sustava, lokalnih i oblačnih, kroz Windows Azure oblak.

Windows Azure računala imaju karakteristike ukratko opisane tablicom 2-I. Osnovno računalo koje se može najmiti u Azure oblaku ima slijedeća svojstva:

- Windows Server 2008 R2 operacijski sustav
- x86-64 jednojezgreni procesor na 1,7 GHz
- 1,7 GB RAM-a i
- 250 GB diskovnog prostora

Cijena najma računala u računalnom oblaku Windows Azure računa se prema tablici 4-IV.

Tablica 4-IV: ***Najam računala u Windows Azure usluzi***

Veličina instance	*Cijena [$/h]*
Extra small	0,05
Small (zadana)	0,12
Medium	0,24
Large	0,48
Extra Large	0,96

Korisnik koji zakupi Azure računalne oblačne resurse može nesmetano iz vlastitih programa, bez ikakvog posebnog dodatnog naručivanja, koristiti neku od sljedećih nerelacijskih KVS (*key-value-store*) pohrana koje se, međutim, posebno naplaćuju:

- Azure Table Storage – tablična pohrana, donekle nalik klasičnim tablicama u relacijskim bazama podataka.
- Blob Storage – pohrana velikih binarnih objekata (slika, XML datoteka i tako dalje).
- Queue Storage – pohrana poruka za komunikaciju između oblačnih programa s korisničkim sučeljem, zvanih *web role*, i oblačnih programa bez korisničkog sučelja, zvanih *worker role*, koji obično služe za pozadinsko procesiranje.

Više o nerelacijskoj KVS pohrani će biti rečeno u kasnijim dijelovima ovoga poglavlja. Ta pohrana ima dvije cjenovne komponente:

- količinu pohranjenih podataka, koja se naplaćuje po cijeni od 0,15 [$/GB·mj]
- količinu CRUD operacija nad podacima (za razliku od Amazona, brisanja se naplaćuju) koja se naplaćuje 0,01 [$/mj·10000] (riječima – 0,01 dolar na 10.000 transakcija mjesečno).

Relacijska SQL Azure pohrana se naplaćuje 9,99 [$/GB·mj].

Korištenje servisne sabirnice i pripadnih usluga pod nazivom AppFabric naplaćuje se 1,99 dolara po 100 tisuća transakcija. Konekcije na AppFabric naplaćuju se po sljedećim tarifama:

- 3,99 dolara po konekciji za čisto plaćanje po utrošku
- 9,95 dolara za 5 konekcija
- 49,75 dolara za 25 konekcija
- 199 dolara za 100 konekcija i
- 995 dolara za 500 konekcija.

Komunikacija s oblakom – Windows Azure uslugom, SQL Azure pohranom i AppFabric servisnom sabirnicom – naplaćuje se po različitim tarifama za ulaz podataka u oblak i izlaz podataka iz oblaka:

- 0,1 [$/GB] za ulaz podataka u oblak i
- 0,15 [$/GB] za izlaz podataka iz oblaka.

4.6 Primjer izračuna cijene oblačnog najma

Za ilustraciju korištenja cjenika, u ovome će poglavlju biti proveden izračun hipotetskog najma oblačnih resursa u Windows Azure oblaku.

Kao što je navedeno u prethodnom pregledu, Microsoft, poput većine drugih oblačnih poslužitelja, naplatnu jedinicu temelji zapravo na ukupnoj količini **procesorskih jezgri** u svim zakupljenim računalima, bez obzira na stvarnu zauzetost tih jezgri odnosno pripadnih računala [19, 42]. Ostala pak svojstva zakupljenih računala, poput kapaciteta memorije i diskova, proporcionalna su količini procesorskih jezgri, kako se može zaključiti već iz tablice 2-I.

Drugim riječima, ako se u Azure oblaku zakupi 5 virtualnih računala s po dvije procesorske jezgre na 2 dana i na njih smjesti neki program, naplaćuje se vrijeme tijekom kojih je tih 10

procesorskih jezgri radilo – 20 [jezgra·dana] – čak i ako u tom periodu aplikaciji nije pristupio niti jedan korisnik, pa čak i onda ako je aplikacija nedostupna (*suspended*) te samo smještena u oblaku!

4.6.1 Računalna obračunska jedinica – ROJ

U Azure cjenicima se najam virtualnih računala u oblaku označava sa *compute*. U daljnjoj razradi izračuna cijene oblačnog najma za taj pojam koristit će se, kao i do sada, riječi *računala*, *računalni resursi* ili *računalne instance*. Za naplatne jedinice, u prethodnom odlomku privremeno označene jedinicom mjere nazvane [jezgra·dani], koristit će se izraz *računalne obračunske jedinice* to jest skraćenica **ROJ**.

Jedno računalo može, dakle, obuhvaćati jednu ili više ROJ – u Windows Azureu to može biti 1, 2, 4 i 8 ROJ a u Amazon Elastic Cloudu 1, 4 ili 8 ROJ. Nadalje, unajmljeno računalo donosi, osim procesora s određenim brojem jezgri, i druge resurse – memoriju i disk. Kako je vidljivo iz tablice 2-I, odnos između tih svojstava i broja jezgri je proporcionalan. Na primjer, ne može se zakupiti računalo s jednojezgrenim procesorom i 3,5 GB RAM-a niti računalo s četverojezgrenim procesorom i 6 GB RAM-a.

Radi dodatnog pojašnjenja, zamislimo situaciju gdje korisnik troši dvije ROJ. To može biti moguće na dva načina:

1. Unajmio je dva manja računala s po jednim jednojezgrenim procesorom ili
2. Unajmio je jedno srednje računalo, s po jednim dvojezgrenim procesorom.

Obračunska jedinica korištenja računalnih resursa nije slučajno nazvana pomalo općenito, "računalna obračunska jedinica" (ROJ), umjesto, primjerice, "oblačna obračunska jedinica". Naime, ako treba uspoređivati učinak i cijene najma u oblaku u usporedbi s nabavljanjem vlastitih računala, ta temeljna obračunska jedinica mora biti ista za oba slučaja, što je naglašeno ponešto neutralnim nazivom.

4.6.2 Cijene računala

Zamislimo da je korisnik u oblaku unajmio jednu srednju računalnu instancu i četiri male (6 ROJ), na dva dana. Osnovna mjerna jedinica vremenskog trajanja zakupa instance su sati [h]. Umnožak sati i ROJ daje ukupnu cijenu korištenja računalnih resursa u oblaku C_r:

$$C_r = 6 \cdot ROJ[{}^{\$}/_{h}] \cdot 48\ [h]$$

Novčana jedinica je američki dolar $\$$. Ako je cijena jednog ROJ 0,12 [$/h], onda je po isteku navedenog najma korisnik dužan oblačnom poslužitelju 34,56 $.

4.6.3 Cijena nerelacijske pohrane podataka

Sve tri vrste nerelacijske pohrane podataka u Windows Azureu – *Table*, *Blob* i *Queue* – naplaćuju se prema veličini pohrane $\boldsymbol{b}$, po jednakoj tarifi od 0,15 [$/GB·mj] i količini transakcija $\boldsymbol{q}$, po jednakoj tarifi od 0,01 [$/mj·10000] (oznake $\boldsymbol{b}$, $\boldsymbol{q}$, $\boldsymbol{u}$ i $\boldsymbol{d}$ korištene su, s istim značenjem u poglavlju 5), pri čemu u transakcije spadaju operacije pisanja, čitanja, ažuriranja i brisanja podataka (*create-read-update-delete* – CRUD). Pretpostavimo da korisnikova aplikacija u oblaku tijekom jednog mjeseca ima sljedeću količinu sadržaja i transakcija:

- 3 TB sadržaja u blobu, 300.000 transakcija,
- 48 GB sadržaja u tablicama, 40.000 transakcija i
- prosječno 50 MB poruka, 450.000 transakcija

Ukupna pohrana iznosi: $b = 3120{,}049\ GB$.

Ukupan broj transakcija iznosi: $q = 790000$.

Ukupna cijena pohrane podataka C_b iznosit će:

$$C_b = \left(p \cdot 0{,}15 \left[\frac{\$}{GB \cdot mj}\right] + X \cdot \left[\frac{\$}{mj \cdot 10^4}\right]\right) \cdot 1[mj]$$

Dulje napisano, bez prethodnih supstitucija za ukupnu veličinu pohrane i ukupan broj transakcija, formula glasi:

$$C_b = \left((3072[GB] + 48[GB] + 0{,}049[GB]) \cdot 0{,}15 \left[\frac{\$}{GB \cdot mj}\right] + (300000 + 40000 + 450000) \cdot 0{,}01 \left[\frac{\$}{mj \cdot 10^4}\right]\right) \cdot 1[mj]$$

Korisnik će oblačnom poslužitelju za ovakvu uslugu biti dužan 475,91 $\$$.

4.6.4 Cijena relacijske pohrane podataka

Za SQL Azure čija je cijena 9,99 [$/GB·mj] lako je izračunati troškove relacijske pohrane podataka. Na primjer, za opseg podataka od 2 GB i period čuvanja od 3 $mjeseca$, cijena C_{rb} iznosi

$$C_{rb} = 2[GB] \cdot 9{,}99 \left[\frac{\$}{GB \cdot mj}\right] \cdot 3[mj]$$

Za navedene parametre, trošak relacijske pohrane iznosi 59,94 $\$$.

4.6.5 Cijena podatkovnog prometa

Komunikacija izvana prema oblaku, ulazna $\boldsymbol{u}$ – *upload*, naplaćuje se 0,1 [$/GB], a iz oblaka prema van, izlazna $\boldsymbol{d}$ – *download*, 0,15 [$/GB]. Tarifa je jednaka za sve vrste komunikacije, kako prema ROJ-evima, tako i prema relacijskoj te nerelacijskoj pohrani, a odnosi se na Europu i sjevernu Ameriku. Stoga je izračun cijene razmjene podataka s oblakom najjednostavniji. Na primjer, ako je u oblak ušlo 4400 MB podataka ($u = 0{,}446\ GB$), a iz njega izašlo 1583 MB ($d = 1{,}546\ GB$) onda je cijena $C_{I/O}$:

$$C_{I/O} = u + d = 0{,}446[GB] \cdot 0{,}1\left[\frac{\$}{GB}\right] + 1{,}546[GB] \cdot 0{,}15\left[\frac{\$}{GB}\right]$$

Cijena ulaza i izlaza iznosi 0,2765 $\$$. Bitno je napomenuti da se **ne** radi o cijeni korištenja Interneta radi pristupa oblaku, jer su te tarife u domeni lokalnih internetskih poslužitelja.

4.6.6 Daljnja istraživanja mjerenjem

U daljnjim poglavljima ovoga rada namjera nam je:

1. Postulirati varijable koje opisuju svojstva složene aplikacije (koja se pokreće iz oblaka ili s lokalnih računala), poput složenosti algoritma, veličine baze podataka i sličnih.
2. **Izmjeriti** kako te varijable utječu na performanse aplikacije koja se izvodi u oblaku te na parametre oblačnog zakupa, temeljem kojih oblačni poslužitelji zaračunavaju cijenu izvođenja složene aplikacije korisnicima.

Ti postulati i mjerenja odnosit će se na računalne oblake općenito, a ne samo na jednu njegovu izvedbu kao što je Windows Azure. Važno je napomenuti da će neke varijable iz točke 2 ovoga plana biti semantički jednake komercijalnim parametrima oblačnih poslužitelja, no za neke druge od tih parametara pokazat će se da su zapravo funkcionalno ovisni i aplikacijskim svojstvima. Na primjer, pokazat ćemo da je količina ROJ-eva funkcionalno ovisna o složenosti aplikacijskog algoritma.

4.7 Nova korisnička očekivanja

Oblačno računarstvo je paradigma koja i korisnicima, a ne samo razvojnim stručnjacima i upravama poduzeća, obećava neke promjene glede korištenja računalnih resursa [39]:

- Očekivanje boljih performansi to jest mogućnost obavljanja složenih te resursno i vremenski zahtjevnih zadaća (obrada velikih količina podataka, izvođenje složenih proračuna, provođenje opsežnih podatkovnih transfera i tako dalje) i u zenitu radnog vremena (*peak hours*), a ne samo u za to određenim vremenskim razdobljima manje opterećenosti sveukupnih računalnih resursa informacijskog sustava, tipično tijekom noći. Svojstva elastičnosti i optimalne nabave izjednačuju troškove pokretanja jednog

računala tijekom 1000 sati i 1000 računala tijekom jednog sata. U oblaku, praktički nema financijskih nepogodnosti pri odabiru druge spomenute varijante.

- Zbog razloga objašnjenih u prethodnome odlomku, može se očekivati da će se jedan dio pozadinskih poslova početi obavljati u stvarnom vremenu, poput ažuriranja stanja skladišta, izračuna potreba raspuštanjem velikih sastavnica, i tako dalje.

4.8 KVS model i BASE zahtjevi

Danas praktički ne postoji program, niti uporaba računala općenito, gdje ne bi bilo potrebe za trajnijim pohranjivanjem podataka. Međutim, ne mora svaka pohrana podataka nužno biti i relacijska. U većine oblačnih poslužitelja, početna, zadana (*default*) baza podataka je neka temeljena na nerelacijskom sustavu i teoretski zasnovana na *Key-Value-Store* (KVS) modelu. Alternativni naziv tog modela je *Entity-Attribute-Value* (EAV).

KVS model spada u širu kategoriju takozvanih No-SQL baza podataka (http://nosql-database.org/) čiji je naziv, zapravo, prilično nesretno odabran – bolji bi bio No-ACID [30], gdje je ACID kratica nužnih svojstava (*atomicity*, *consistency*, *isolation*, *durability*) relacijskih baza. Drugim riječima, KVS model ne pokorava se ACID zahtjevima, nego slijedi drugi niz zahtjeva poznat pod skraćenicom BASE – *basically available, soft state, eventual consistency* [99]. BASE zahtjevi i KVS model pojavili su se u informacijskim tehnologijama krajem 90-ih godina 20. stoljeća, razvojem velikih internetskih sustava poput Amazona, Googleovih, Microsoftovih i Yahoo servisa te širenjem socijalnih aplikacija poput Facebooka. Svi ti sustavi imaju dva bitna svojstva:

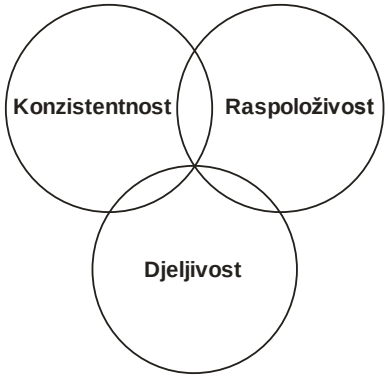
- barataju daleko većom količinom podataka nego li bilo koji poslovni informacijski sustavi, makar i najsloženiji i
- zbog ogromnog broja korisnika moraju omogućiti daleko veću raspoloživost tih podataka nego klasični informacijski sustavi.

Drugim riječima, takvi sustavi moraju biti **visoko distribuirani** da bi mogli odgovoriti na potrebe **stotina milijuna korisnika**. Raspoloživost i proširivost (*scalability*) u smislu raspodjele smještaja podataka i obrade korisničkih zahtjeva na mnogo računala, umjesto na samo jednom ili njih nekoliko, su primarni zahtjevi koji se postavljaju pred takve sustave

U takvom izrazito distribuiranom softverskom sustavu korištenje relacijskih softvera za upravljanje bazama podataka (*Relational Database Management System* – RDBMS) postaje neadekvatno, što je posljedica takozvanog CAP teorema [103]. Naime, svaki informacijski sustav mora u što je moguće većoj mjeri zadovoljiti tri zahtjeva:

- konzistentnost (*consistency*) – svojstvo da svi korisnici vide iste podatke čak i u uvjetima istovremenog ažuriranja
- raspoloživost (*availability*) – svojstvo prema kojem svi korisnici u svakom trenutku imaju na raspolaganju tražene podatke i
- djeljivost (*partition tolerance*) – mogućnosti smještaja podataka na mnogo računala te uključivanja mnogo računala u obradu tih podataka.

CAP teorem pak kaže da je kod distribuiranog sustava moguće postići **potpuno** ispunjenje samo neka dva od ukupno tri ta zahtjeva [36]. Dakle, raspoloživost i djeljivost može biti postignuta samo ako se **djelomično odustane** od konzistencije. Jasno, potpuno odustajanje od konzistencije je također neprihvatljivo. Stoga se uvodi odgođena (*eventual*) konzistencija koja se, dakle, ne postiže trenutno, u vrijeme izvršavanja određene transakcije, nego na drugi način, primjerice periodički pokretanim pozadinskim procesiranjem. Svojstva CAP teorema prikazana su na slici 4.6.



Slika 4.6: ***CAP teorem***

Prema slici 4.6 KVS baze podataka omogućavaju raspoloživost i djeljivost, uz žrtvovanje trenutne konzistentnosti, dok relacijske baze podataka omogućuju konzistentnost i raspoloživost, uz žrtvovanje jednostavnog particioniranja.

Još jednom treba napomenuti da to ne znači da se relacijske baze podataka uopće ne mogu particionirati i istovremeno održati konzistentnima i raspoloživima [65], nego ih je teže učiniti takvima, odnosno pri tome će se smanjiti performanse nekog od preostala dva svojstva. Stoga se često pojednostavnjeno zaključuje da relacijske baze podataka, navodno, nisu horizontalno proširive, no treba imati na umu da su pojmovi proširivosti odnosno raspoloživosti prilično neegzaktni [30]. Proširivost može značiti, primjerice:

1. pripremu za 500 tisuća korisničkih posjeta dnevno što je ogromna posjećenost koju ima samo mali broj web mjesta ili
2. desetke i stotine milijuna posjeta dnevno.

Primjer prvog [41] slučaja je programerska zajednica Stack Overflow – čiji web portal sasvim lijepo radi pod klasičnom relacijskom bazom podataka SQL Server 2008. Primjeri drugoga slučaja su već spomenuti Amazonovi, Googleovi, Microsoftovi, Facebookovi i drugi mrežni servisi koji doista traže alternativne pristupe pohrani i obradi podatka poput KVS modela, jer im je zahtjev raspoloživosti na prvome mjestu.

Također, za one djelatnosti gdje je trenutna konzistentnost (za razliku od kasnije) nužna, poput bilježenja financijskih transakcija u bankarstvu, relacijske baze podataka su sada **nezamjenjive** te će ostati i dalje nezamjenjive. Stoga će se i dalje razvijati i koristiti metode njihovog skaliranja, kako vertikalnog (uporabom jačih računala), tako i horizontalnog (dijeljenjem). Djelatnosti za koje je trenutna konzistentnost nužna imaju slijedeća svojstva:

- Podaci su visoko povezani (u relacijama, pomoću vanjskih ključeva).
- Konzistentnost i integritet podataka su od velike poslovne važnosti.

Za razliku od takvih djelatnosti, KVS baze podataka koriste se u aplikacijama i uslugama slijedećih svojstava:

- Podaci su slabo povezani.
- Trenutni integritet podataka nije pretjerano bitan.

Jednostavno rečeno, da bi se spremio i prikazao Facebookov *status update*, zapisi zvani cvrkutima (*tweet*) na Twitteru ili pak signatura na Google Talku – nije potrebna relacijska baza podataka. Istovremeno, takva je baza podataka čak i uteg, jer ju je daleko teže skalirati za milijune i milijune korisnika nego KVS bazu podataka.

Da bi se potpuno razumio KVS model, najbolje ga je usporediti s relacijskim modelom to jest otkriti što nedostaje u KVS modelu, a postoji u relacijskom. Ukratko, u KVS modelu nema referencijalnog integriteta što znači da nema vanjskih (stranih) ključeva (*foreign key*) koji povezuju pojedine zapise te je nemoguće napisati bilo kakav JOIN upit. Pojašnjenja radi, u relacijskim bazama podataka moguće je napisati JOIN upit čak i između dviju relacijskih shema (*table*) koje nisu međusobno formalno referencijalno povezane vanjskim ključevima. Nasuprot tome, u KVS bazama podataka nikakva vrsta JOIN naredbe naprosto ne može postojati. Osim toga, KVS baze podataka nemaju okidače (*trigger*) niti pohranjene procedure (*stored procedure*). One se koriste kao čista podatkovna pohrana, bez ugrađene aplikacijske logike. Također, filtri (ili WHERE uvjeti) u KVS modelu obično se ne mogu tako široko postaviti kao u relacijskom jer, na primjer, nedostaju agregatne funkcije. Bilo kakva okidačka, aplikacijska, JOIN ili neka druga logika nad podacima u KVS bazi podataka može se izvesti jedino izvana, u programskom kodu, nakon dobavljanja svih potrebnih podatkovnih setova.

KVS model nije vezan isključivo za oblačno računarstvo, nego za sve okolnosti korištenja informacijskih tehnologija kod kojih je nužna velika distribuiranost podatkovne pohrane i poslova, odnosno svuda gdje neki posao treba obavljati na velikim serverskim farmama, a ne na nekom omanjem broju računala. KVS model može se, također, primijeniti u mnogim okolnostima gdje složenost i velike mogućnosti relacijskih baza podataka naprosto nisu potrebne, odnosno gdje su relacijske baze prejake (*overkill*). Svaki relacijski sustav za upravljanje bazama podataka svakako može vrlo lako simulirati KVS model jer je on jednostavniji i primitivniji od relacijskog. Budući da poslužitelji oblačnog računarstva žele osigurati široku primjenu svojih platformi, jasno je da u njih uključuju i KVS baze podataka. Te baze podataka jednostavnije su od relacijskih pa se stoga i nude kao početna, jeftinija opcija podatkovne pohrane. Za očekivati je da će se korištenje KVS baza podataka povećavati s opsegom korištenja računalnih oblaka, kako zbog njihove manje cijene u odnosu na relacijske, tako i zbog toga što njihova prava snaga dolazi do izražaja upravo u velikim serverskim farmama poput računalnih oblaka.

4.9 Sažetak - istraživački fokus

O ovom poglavlju obrađene su specifičnosti uporabe i korištenja računalnih oblaka u odnosu na klasične serverske farme, s ekonomskog, korisničkog i razvojnog gledišta. Poseban je naglasak stavljen na elastičnost oblaka, svojstvo koje omogućuje zaposjedanje i otpuštanje računalnih resursa u skladu s trenutnim poslovnim potrebama, te – posljedično - optimalnu nabavu koja oslobađa korisnike od rizika kapitalnog ulaganja.

Utvrđeno je kako svatko tko razmišlja o uporabi oblaka, bilo kao korisnik, bilo kao pružatelj usluga oblačnog računarstva, mora izračunati uvjete isplativosti zakupa oblačnih resursa odnosno postavljanja oblaka i davanja tih resursa u najam. Korisnik mora imati način usporedbe komercijalne cijene i resursne ponude oblačnih poslužitelja s troškovima nabave i održavanja vlastite opreme istih mogućnosti. U tim procjenama važnu ulogu ima, osim klasičnih ekonomskih postavki, poput sadašnje i buduće vrijednosti te amortizacije, još i Mooreov zakon, koji je specifičan samo za računalne tehnologije.

Potencijalni pružatelj usluga oblačnog računarstva mora poznavati strukturu cijene oblačnih usluga, kako bi mogao formirati ponudu koja ga neće odvesti u gubitke i propast, ali će istovremeno biti dovoljno konkurentna da privuče određen broj oblačnih korisnika. U usporedbi s klasičnom, omanjom serverskom farmom, oblačnom poslužitelju ide na ruku ekonomija razmjera koja dolazi do izražaja u velikim računalnim centrima poput oblačnih, a opisuje pojavu smanjenja prosječne cijene proizvoda ili usluge što se više povećava količina tog isporučenog proizvoda ili usluge.

Prikazana je i okvirno uspoređena stvarna komercijalna ponuda tri značajna oblačna poslužitelja Amazona, Googlea i Microsofta, uz ilustrativni izračun troškova hipotetskog najma iz oblaka jednog od tih poslužitelja, Microsoftovog Azurea. Taj izračun troškova skupa s cijelim ovim poglavljem poslužili su za određivanje istraživačkog fokusa:

1. Odrediti varijable koje opisuju svojstva složene aplikacije koja se može smjestiti u oblak ili na lokalna računala.
2. Izmjeriti kako te varijable utječu na komercijalne parametre pod kojima se unajmljuju oblačni resursi, a kako na parametre kojima se određuje nabava vlastitih računala.

Ti postulati i mjerenja odnosit će se na sve računalne oblake. Temeljem tih mjerenja će se izvesti metodika za odabir računalnih resursa za informacijsku podršku u poduzeću ili ustanovi koja će predstavljati okosnicu za dokazivanje hipoteze 2 ovoga rada.

5 Mjerenje performansi aplikacija u oblaku

Većina objavljenih radova u području oblačnog računarstva ističe nekoliko bitnih značajki ove računalne paradigme pomoću kojih se objašnjavaju pozitivni poslovni učinci za potencijalne korisnike. Ugledna međunarodna organizacija ISACA[1] u svom dokumentu [49] navodi da korištenjem usluga oblačnog računarstva organizacije mogu:

- smanjiti ukupne troškove vlasništva (TCO)
- ostvariti veći povrat ulaganja (*return of investment* – ROI)
- povećati učinkovitost ICT infrastrukture potrebne za potporu svojem poslovanju
- smanjiti troškove poslovanja plaćanjem samo stvarno korištenih usluga.

U tom smislu definirane su poslovne značajke oblačne arhitekture kao što su: zadržavanje troškova ulaganja (*cost containment*), kratko vrijeme osiguranja potrebnih resursa (*immediacy*), uz njihovu pravovremenu raspoloživost (*availability*) u traženom opsegu (*scalability*), povećanje učinkovitosti vlastitog ICT sektora (*efficiency*) smanjivanjem opsega njegovih operativnih poslova u korist inovacije i razvoja, elastičnost (*resiliency*) u slučaju kvara informatičkih resursa ili visokih trenutnih opterećenja, i druge. Sve se ove značajke mogu koristiti kao vrlo uvjerljiv argument kada voditelj informatičke službe predlaže upravi poduzeća da se potrebni informatički resursi osiguraju korištenjem oblačnog računarstva. Međutim, za svakog se istraživača postavlja pitanje vrijede li ti argumenti uvijek ili pak samo uz neke pretpostavke te imaju li istu težinu bez obzira na informatičke potrebe koje ima organizacija i informatičke resurse potrebne da bi organizacija zadovoljila te potrebe.

Mnoga radna izvješća konzultanata, kao npr. [23] i [18] ili objavljeni radovi upućuju na to da praktične informatičke potrebe izrazito zavise o djelatnosti poduzeća ili organizacije. Tako Lindner u radu [58] kaže *"From an economic perspective, each of these services is based on a certain business model."* Sigurno je da su potrebni različiti informatički resursi mora li neka organizacija podržati *on-line* obradu narudžbi, planirati potrebe materijala raspuštanjem velikog broja složenih sastavnica, koristiti CRM softver, voditi platni promet za fizičke i pravne osobe ili izvoditi znanstvene proračune. Stoga će se u ovom poglavlju istražiti relevantni čimbenici o kojima zavisi učinkovitost korištenja informatičkih resursa koji su kao servis osigurani u oblaku. Istraživanje će biti provedeno mjerenjem relevantnih svojstava tipskih aplikacija koje će se izvoditi na informatičkim resursima u oblaku te usporedbom tih rezultata sa svojstvima koje bi postigla ista aplikacija izvedena na programskoj i sklopovskoj opremi u vlasništvu korisničke organizacije.

[1] ISACA – *Information Systems Audit and Control Association*

5.1 Svrha, cilj i predmet mjerenja

Osnovno svojstvo za iskazivanje koristi koju može ostvariti neka organizacija korištenjem određene aplikacije je vrijeme njezinog izvođenja T. Općenito, kraće vrijeme izvođenja aplikacije znači da će korisnik prije dobiti potrebnu informaciju o nekom stanju ili poslovnom događaju i tako za organizaciju stvoriti potencijal koji se može realizirati kao određena novčana vrijednost ili ekvivalentna vrijednost iskazana nekom drugom metrikom. Za ostvareni učinak E može se dakle reći da vrijedi:

$$E = f\left(\frac{1}{T}\right) \text{ pri čemu je vrijeme izvođenja definirano u intervalu } T=[T_{min}, T_{max}] \quad (5.1)$$

U gornjem izrazu treba obratiti pozornost na granične vrijednosti. Varijabla T_{max} predstavlja najveće prihvatljivo vrijeme izvođenja aplikacije, ono vrijeme nakon kojeg bi aplikacija bila bez praktične svrhe. Trajanje T_{min} pak ne može biti matematički po volji malo jer je ograničeno svojstvima aplikacije i informatičkim resursima na kojima se ona izvodi. Ovo vrijeme neka bude označeno s $T_{min,ICT}$. S druge strane, za svaku je aplikaciju moguće utvrditi neko vrijeme T_{ORG} koje zavisi o realnoj potrebi korisnika radi izvršavanja njegove uloge u organizaciji. Ta dva vremena koja se mogu pojaviti kao donja granica intervala u izrazu *(5.1)*, ne moraju biti jednaka. Općenito vrijedi $T_{min,ICT} \neq T_{ORG}$. Stoga se usklađivanje informatičkih resursa s realnim potrebama organizacije svodi na nalaženje takvog vremena izvođenja aplikacije za koje vrijedi: $T_{max} > T_{ORG} \approx T \geq T_{min,ICT}$. Upravo to je razlog zašto treba istražiti o čemu sve zavisi vrijeme T u slučaju da se informatički resursi ostvaruju korištenjem računala iz oblaka. Poznavanje tih čimbenika omogućilo bi korisniku da u računalnom oblaku, za potrebe svoje aplikacije, zaposjedne upravo toliko resursa iz računalnog oblaka koliko je potrebno za postizanje potrebnog T_{ORG}.

Iz analize gornjih odnosa također slijedi da je **nepotrebno inzistirati da se neka aplikacija uvijek izvodi u najkraćem mogućem vremenu**. Potrebno vrijeme izvođenja aplikacije mora biti približno jednako vremenu unutar kojeg korisnik koji se služi aplikacijom mora dobiti odgovor ili podatak kojeg je zatražio pokretanjem aplikacije. Stoga bi svako tehničko osiguranje vremena izvođenja aplikacije T koje bi bilo mnogo manje od T_{ORG} zapravo značilo samo nepotrebno angažiranje dodatnih računalnih resursa.

Drugi razlog za mjerenje je dokazivanje hipoteza postavljenih u ovom radu. Mjerenje će omogućiti uspostavljanje višedimenzionalne metrike potreba za računalnim resursima što je sadržaj hipoteze 2. Tako utvrđena metrika biti će zatim ugrađena u metodiku razvoja oblačnih i hibridnih SOA rješenja čija je mogućnost razvoja najavljena hipotezom 1.

5.2 Zavisne i nezavisne varijable u istraživanom problemu

Ako je vrijeme izvođenja aplikacija, kao što je prikazano u prethodnom odjeljku, glavni kriterij za procjenu učinaka oblačnog računarstva, onda će se daljnje istraživanje usmjeriti na određivanje sljedeće funkcijske zavisnosti:

$$T = f_t(x_1, x_2, ...x_k) \tag{5.2}$$

U gornjem su izrazu $x_1, x_2, ...x_k$ nezavisne varijable koje predstavljaju parametre o kojima zavisi vrijeme izvođenja T neke aplikacije u oblaku. Ovako iskazana zavisnost svojstvena je temeljnoj razini komercijalnog pružanja usluga, koja se označava akronimom IaaS.

Druga zavisnost, interesantna za istraživanje učinaka oblačnog računarstva, je trošak C kojeg ima korisnik ako svoju aplikaciju izvodi na resursima iznajmljenim u oblaku. Svi pružatelji usluga oblačnog računarstva uvijek naplaćuju vrijeme zauzeća računalnih resursa, a onda eventualno još i druge usluge (poput pohranjivanja velikih količina podataka, razmjene podataka između pružatelja i korisnika usluge ili pravo korištenja neke aplikacije u sustavu SaaS). Stoga se može uzeti da trošak C uvijek zavisi o svim onim čimbenicima, označenima s x_1 do x_k u izrazu *(5.2)*, o kojima zavisi T, ali da u općem slučaju u obračun troškova mogu biti uključeni i još neki drugi parametri, što zavisi o karakteristikama aplikacije koja se izvodi na oblačnoj platformi. Može se dakle zapisati da za trošak C aplikacije koja se izvodi u oblaku općenito vrijedi:

$$C = f_c(x_1, x_2, ...x_k, x_{k+1}, ...x_n) = f_c(T, x_{k+1}, ...x_n) \tag{5.3}$$

U gornjoj jednadžbi su x_{k+1} do $x_n$upravo oni faktori koji utječu na trošak C, a ne utječu na vrijeme izvođenja aplikacije T. Ovi se faktori odnose na sve usluge koje su iznad razine IaaS, a zavise o politici cijena koje vode pojedini isporučitelji usluga oblačnog računarstva.

U jednadžbama *(5.2)* i *(5.3)* nepoznati su oblici funkcijskih ovisnosti f_t i f_c, ali isto tako i nezavisne varijable x_i $(i = 1, 2, ..., n)$. Analitički postupci su ovdje teško provedivi pa će se ove zakonitosti utvrditi mjerenjem te odgovarajućom obradom izmjerenih vrijednosti. Da bi se nešto izmjerilo potrebno je redom:

1. odrediti parametre koji će se mjeriti
2. pripremiti objekt mjerenja
3. osmisliti takav plan mjerenja koji će dati korisne podatke za izračunavanje izraza *(5.2)* i *(5.3)*.

Za početnu identifikaciju parametara koje treba mjeriti korištena je metoda ekspertskih procjena. Formiran je jednostavni upitnik otvorenog tipa u kojem je 8 poznavatelja oblačnog

računarstva rangiralo parametre koji, prema njihovom mišljenju, najznačajnije utječu na trajanje izvedbe neke aplikacije u oblaku. Eksperti su svakom kriteriju mogli dati značaj 1 do 5, a uz 6 parametara navedenih u upitniku mogli su dopisati parametar koji nije naveden u upitniku, no smatraše ga smatraju ga važnim, i dati mu ocjenu. Rezultati tog predistraživanja su prikazani u tablici 5-I.

Tablica 5-I: ***Procjena parametara koji utječu na vrijeme izvođenja aplikacije***

Naziv parametra	*Oznaka*	*1.*	*2.*	*3.*	*4.*	*5.*	*6.*	*7.*	*8.*	*Φ*
Snaga angažiranih računala	p	1	2	5	1	4	5	2	4	3,00
Složenost algoritma	a	5	5	4	4	5	2	3	1	3,63
Opseg poslanih podataka	u		3		3		3			1,13
Veličina baze podataka	b	3	1	2	2	1		4	3	2,00
Broj akcija po upitu	q	4				3	1		2	1,25
Broj angažiranih računala	r	2	4	3	5	2	4	5	5	3,75
Opseg primljenih podataka	d							1		0,13
Brzina prijenosa podataka	v			1						0,13

Ovdje valja naglasiti da se prosječna vrijednost ranga Φ koja je navedena u zadnjem stupcu ne smije nikako drugačije tumačiti nego isključivo kao smjernica za to koje parametre treba detaljnije istražiti u daljnjem radu. Slova iza naziva parametara ($\boldsymbol{p}$, $\boldsymbol{a}$, $\boldsymbol{u}$, i tako dalje) su skraćene oznake pod kojom će se neki parametar referencirati dalje u radu.

Temeljem rangova iz tablice 5-I procjenjuje se da dalje treba istražiti utjecaj 6 parametara za koje vrijedi $\Phi \geq 1{,}0$ što je označeno u tablici sjenčanjem. U daljnjem istraživanju će se T i C smatrati zavisnim varijablama a parametri $\boldsymbol{r}$, $\boldsymbol{p}$, $\boldsymbol{a}$, $\boldsymbol{u}$, $\boldsymbol{b}$, $\boldsymbol{q}$ nezavisnim varijablama.

5.3 Nezavisne varijable

Prema [10] **nezavisne** varijable su one čije se vrijednosti mogu proizvoljno postavljati (unutar određenih granica) prije mjerenja, ili mijenjati prije uvrštavanja u funkcijski izraz radi izračunavanja. Za područje oblačnog računarstva, obuhvaćeno ovim istraživanjem, autor ih je rasporedio u tri grupe:

- Resursne – vezane za način angažiranja i korištenja oblačnih resursa
- Aplikacijske – opisuju složenost aplikacije koja je kandidat za izvođenje u oblaku i
- Podatkovne – odnose se na opseg podataka nad kojima se izvodi obrada.

Resursnim nezavisnim varijablama opisuje se angažman računalnih resursa u oblaku čija su tehnička svojstva opisana u prethodnom poglavlju. U istraživanom problemu moguće je prepoznati dvije nezavisne varijable ovog tipa:

- Broj angažiranih računala $\boldsymbol{r}$. U tehnici pripreme rada u oblaku ovo se proizvoljno bira iz vrlo široke domene, primjerice od jednog do tisuću računala.
- Snaga angažiranih računala $\boldsymbol{p}$. U tablici 5-II je naveden raspon konfiguracija računala koja se standardno može angažirati kod oblačnih poslužitelja.

Obje varijable proizvoljno bira potencijalni korisnik oblačnog računarstva, onako kako smatra da je najbolje za njega, te ih ugovara s pružateljem usluga oblačnog računarstva u granicama njegove ponude.

Aplikacijske varijable $\boldsymbol{a}$ i $\boldsymbol{q}$ opisuju posao koji korisnik namjerava izvoditi, odnosno složenost aplikacije koja je kandidat za izvođenje korištenjem oblačnog računarstva. Složenost aplikacije može se mjeriti na razne načine, poput broja funkcijskih točaka ili količine linija kôda, ali u ovom istraživanju raditi će se s onim aspektom složenosti o kojem ovisi vrijeme obavljanja aplikacije u oblaku, te koji se uopćeno može nazvati složenost algoritma $\boldsymbol{a}$. Složenost algoritma mjerit će se kao vrijeme izvođenja jedne transakcije, na računalu jedinične konfiguracije, uz nepromjenljive srednje vrijednosti svih ostalih varijabli. Varijabla $\boldsymbol{q}$ označavati će broj CRUD (*Create, Read, Update, Delete*) akcija koje se prosječno izvode po jednom upitu.

Pojam podatkovnih varijabli odnosi se na opseg podataka nad kojima se izvodi nekakva aplikacijska obrada. Taj opseg može se mjeriti na dva načina:

- opseg podataka koji se šalje u oblak radi obrade (*upload*) po jednom upitu koji će biti označen s $\boldsymbol{u}$
- veličina baze podataka $\boldsymbol{b}$ nad kojom se izvodi obrada te unutar koje treba pronaći one podatke koji su potrebni za izvršenje jednog upita.

5.4 Zavisne varijable

Zavisne varijable su one značajke nekog sustava čija je vrijednost funkcijski ovisna o vrijednosti jedne ili više drugih, nezavisnih varijabli. Drugim riječima, zavisne varijable su ona svojstva koja se mjere ili računaju za razne vrijednosti nezavisnih varijabli. Za problem koji se istražuje u ovome radu primarna zavisna varijabla će biti vrijeme izvođenja aplikacije u oblaku T, u smislu kako je to definirano izrazom *(5.2)*, jer je to vrijeme **relevantno** za korisnika aplikacije, bez obzira gdje se ona izvodila – na vlastitim ili iznajmljenim računalnim resursima.

5.5 Izbor i priprema aplikacije za mjerenje

Drugi korak u istraživanju neke pojave mjerenjem odnosi se, prema točki 2 u potpoglavlju 5.2, na pripremu objekta mjerenja. Objekt mjerenja u ovom radu je aplikacija koja će se izvoditi u oblaku. Za objektivno istraživanje funkcijskog odnosa, iskazanog izrazom *(5.2)*, potrebno je raspolagati takvim aplikacijama čija se svojstva, navedena u tablici 5-I kao relevantna (te označena kao $\boldsymbol{p}$, $\boldsymbol{a}$, $\boldsymbol{u}$, i tako dalje), mogu proizvoljno mijenjati, a da se mjerenja izvode pod nepromjenljivim uvjetima u odnosu na druge slučajne i nepoznate varijable koje nisu predmet istraživanja. Takve bi pretpostavke bile ispunjene onda kada bi se mogla pronaći jedna, jedinstvena aplikacija čije bi se karakteristike opisane aplikacijskim i podatkovnim varijablama mogle proizvoljno mijenjati unutar neke domene. Naime, ako bi se mjerenja u kojima variraju vrijednosti nezavisnih varijabli $\boldsymbol{r}$, $\boldsymbol{p}$, $\boldsymbol{a}$, $\boldsymbol{u}$, $\boldsymbol{b}$, $\boldsymbol{q}$ provela uvijek nad istom aplikacijom koja bi se opetovano izvodila u oblaku, onda bi se istraživanje moglo provesti pod strogo kontroliranim uvjetima u kojima je isključen slučajni utjecaj drugih nekontroliranih pojava na vrijeme T. Ovakav pristup mjerenju kao metodologiji znanstvenog istraživanja opisan je u opsežnoj literaturi, primjerice [43].

Opsežnim pretraživanjem karakteristika velikog broja aplikacija, praktično primjenjivanih u praksi radi potpore u različitim ljudskim aktivnostima, te pretraživanjem na internetu, pronađena je, dopunjena i pripremljena za izvođenje u oblaku aplikacija koja odgovara traženim zahtjevima. Ta se aplikacija bavi prepoznavanjem oblika, a koristi se za identifikaciju određene osobe na temelju njezine aktualne fotografije i usporedbe s velikim brojem fotografija pohranjenih u bazi podataka. U daljnjem tekstu ta će aplikacija biti referencirana kao **ogledna aplikacija** (OA). Radi razumijevanja kako se unutar nje mogu mijenjati vrijednosti parametara $\boldsymbol{r}$, $\boldsymbol{p}$, $\boldsymbol{a}$, $\boldsymbol{u}$, $\boldsymbol{b}$, $\boldsymbol{q}$ potrebno je nešto detaljnije opisati njezine funkcionalnosti.

5.6 Ogledna aplikacija

Ogledna aplikacija se temelji na algoritmima koji su razvijeni u sklopu OpenCV [75] projekta (*Open Computer Vision*). Radi se o otvorenoj biblioteci, izvorno zasnovanoj u Intelu, koja sadrži više od 500 algoritama namijenjenih rješavanju problematike računalnog vida. Biblioteka je izvedena u C i C++ programskim jezicima radi što šire višeplatformnosti. Među mnogim podržanim funkcionalnostima su i algoritmi za otkrivanje i raspoznavanje objekata općenito, pa tako i lica.

Ogledna aplikacija je razvijena u .NET tehnologiji, u programskom jeziku C#, u potpunosti u upravljanom (*managed*) kodu. Da bi to bilo moguće, korišten je .NET omotač (*wrapper*) oko OpenCV biblioteke nazvan Emgu CV [25]. Emgu CV je također otvoreno rješenje koje se

može koristiti na više platformi. To rješenje radi pod operacijskim sustavom Windows uz .NET platformu, ali i pod Linuxom, uz Mono platformu koja je zapravo .NET prenesen (*ported*) na Linux sustave. Temeljni koraci pri korištenju ogledne aplikacije su sljedeći:

- Rad OA odvija se nad bazom podataka ljudskih portreta pa je zato preduvjet za njezin rad formiranje referentne baze. U ovom slučaju je konkretna baza podataka preuzeta sa sveučilišta u Essexu koja sadrži portrete 400 osoba, pri čemu su neke osobe snimljene u više različitih varijanti [86].
- Korisnik OA sustavu šalje sliku za prepoznavanje putem klijentskog programa s korisničkim sučeljem (ili preko web aplikacije). Korisnik može biti automatizirani softver, ne nužno i ljudsko biće. Uz sliku za prepoznavanje, korisnik šalje i upute koje sadrže informaciju treba li OA sustav radi prepoznavanja pretražiti čitavu svoju bazu podataka lica ili samo neki njen dio.
- Ogledna aplikacija čita sve slike iz referentne baze za koje je dobila nalog u prethodnom koraku i uspoređuje ih s poslanom aktualnom slikom tako da se pokrene algoritam za prepoznavanje lica. Konačni rezultat rada aplikacije je izbor jedne slike iz referentne baze, najsličnije onoj koja je poslana radi prepoznavanja, ili nijedne, ako su sve slike iz referentne baze previše različite od poslane. Korisniku se vraća slika primljena radi prepoznavanja i slika koja je njoj najsličnija.

Matematička i programska struktura algoritma za prepoznavanje lica su izvan domene ovog rada, ali se ovdje ukratko navodi na čemu se algoritam temelji i kako radi:

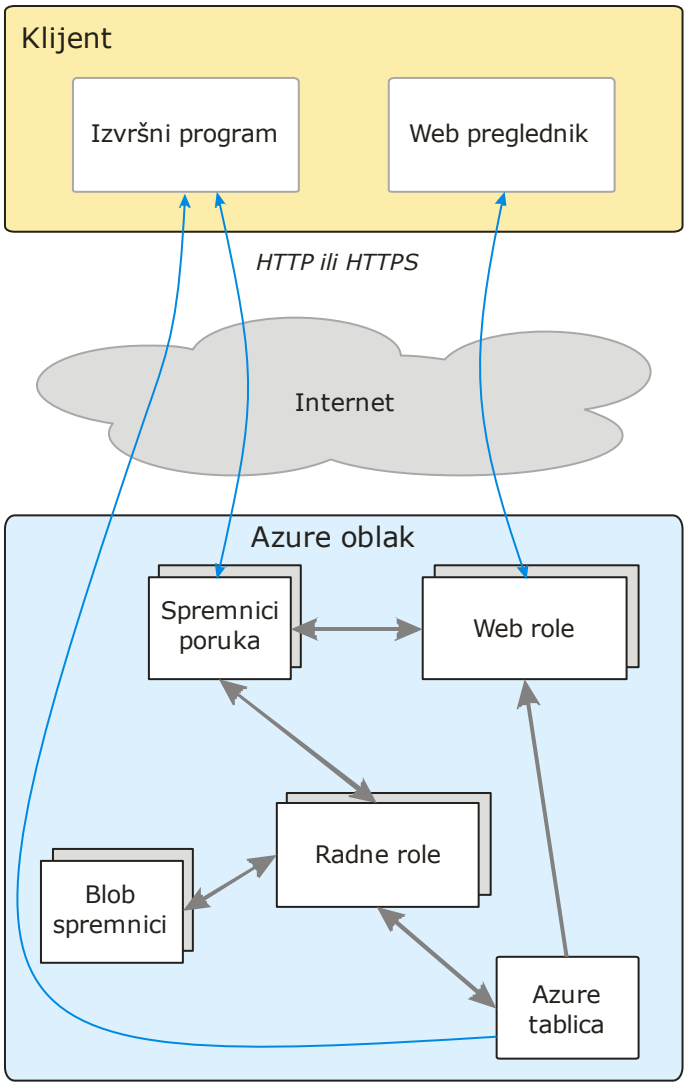
1. Algoritam učitava sliku po sliku iz pridružene mu referentne baze podataka i pretvara ju u bitmapu (*bitmap*). Sve slike moraju biti istih dimenzija, a ako nisu, moraju se programski učiniti takvima. Isto se čini s poslanom, aktualnom slikom.
2. Za svaku bitmapu iz prethodnog koraka se izračunava karakteristična matrica metodom *principal component analysis*. To je razlog zašto sve slike moraju biti istih dimenzija.
3. Za svaku matricu iz prethodnog koraka računaju se svojstvene vrijednosti (*eigenvalue*) i uspoređuju sa svojstvenom vrijednošću karakteristične matrice poslane slike.
4. Slika iz referentne baze podataka, čija matrica ima svojstvenu vrijednost najbližu i dovoljno blisku svojstvenoj vrijednosti matrice poslane aktualne slike, proglašava se ispravnim rezultatom prepoznavanja i isporučuje korisniku.

Programski kôd aplikacije koji poziva i povezuje pojedine dijelove složenog algoritma za prepoznavanje lica razvijen je za potrebe ovog rada i izgleda ovako:

```
Emgu.CV.Image<Bgr, Byte> poslanaSlika =
      new Image<Bgr, Byte>(new Bitmap(memorijskiIzvor));
Emgu.CV.Image<Gray, Byte>[] images = new Image<Gray,
Byte>[blobList.Count];
for (int i = 0; i < blobList.Count; i++)
{
      blobList[i], msgParts[1]);
      blob = blobClient.GetBlobReference(blobList[i]);
      streamDB = new MemoryStream();
      blob.DownloadToStream(streamDB);
      streamDB.Seek(0, SeekOrigin.Begin);
      Bitmap bitmap = new Bitmap(streamDB);
      images[i] = new Image<Gray, byte>(bitmap);
}
termCrit = new MCvTermCriteria(0.001);
// Simuliranje usložnjavanja algoritma for petljom
for (int i = 0; i < Convert.ToInt32(msgParts[2]); i++)
{
      EigenObjectRecognizer recognizer =
            new EigenObjectRecognizer(
                  images,
                  blobList.ToArray<string>(),
                  5000,
                  ref termCrit
            );
      identification = recognizer.Recognize(imgToRecognize.Convert<Gray,
      Byte>());
}
```

Sklopovska i programska arhitektura koja omogućava izvođenje ogledne aplikacije na računalnim resursima u oblaku (upravljanima korištenjem Microsoftove platforme Azure) prikazana je na slici 5.1. Program za prepoznavanje slika pokrenut je kroz jednu ili više instanci radne (*worker*) role. Svaka instanca radne role jest, zapravo, jedno virtualno računalo. Drugim riječima, program za prepoznavanje slika pokreće se na jednom ili više računala u oblaku. Web rola je pak korisnička web aplikacija – jedno od dva alternativna korisnička sučelja i također se pokreće na jednom ili više virtualnih računala (povezanih u serversku farmu u oblaku što je obrazloženo u ranijim poglavljima). Druga korisnička aplikacija je običan izvršni (.exe) program koji s oblakom također komunicira preko Interneta. Korisničke aplikacije šalju u *ulazni* spremnik poruka nalog za prepoznavanje poslane slike. Taj spremnik poruka periodički čitaju radne role i po primitku naloga pokreću raspoznavanje slike spram slika u bazi podataka smještenoj u blob-spremnicima. Svaka radna rola obrađuje samo slike iz "svojega" blob-spremnika. Nakon završetka raspoznavanja slike, svaka radna rola sprema podatke o duljini obavljanja posla u Azure tablicu te u *izlazni* spremnik poruka upisuje obavijest da li je sliku nalik poslanoj pronašla u svojem blob-spremniku i, ako jest,

koja je to. Klijentski program periodički pak provjerava izlazni spremnik te, kada su sva prepoznavanja obavljena, bilježi ukupno vrijeme i prikazuje korisniku slike nalik poslanoj.



Slika 5.1: ***Arhitektura ogledne aplikacije koja se izvodi u oblaku***

5.7 Način mjerenja i predistraživanje

U prethodnom odlomku je ukratko opisana funkcionalnost ogledne aplikacije čije se aplikacijske i podatkovne značajke mogu mijenjati u dovoljnom širokom rasponu, kako bi se mogla mjeriti promjena zavisne varijable T o promjenama nezavisnih varijabli $\boldsymbol{a}$, $\boldsymbol{u}$, $\boldsymbol{b}$, $\boldsymbol{q}$. Za te promjene biti će potrebna određena intervencija u nekim aplikacijskim parametrima i kôdu aplikacije. Promjene resursnih varijabli, opisanih parametrima $\boldsymbol{r}$ i $\boldsymbol{p}$, za koje se također pretpostavlja da utječu na T (što isto treba utvrditi mjerenjem), mogu se provesti daleko lakše jer su nezavisne o aplikaciji i podacima. Radi se o parametrima koji se inače moraju jednoznačno navesti kod svakog zaposjedanja resursa u oblaku. Opseg variranja nezavisnih varijabli za koje će se mjerenjem utvrđivati vrijeme izvođenja ogledne aplikacije je prikazan u tablici 5-II.

Tablica 5-II: ***Opseg variranja nezavisnih varijabli***

Grupe nezavisnih varijabli *(1)*	***Naziv parametra*** *(2)*	***Skraćeni naziv*** *(3)*	***Mjerna jedinica*** *(4)*	***Domena [min; max]*** *(5)*	***Način određivanja vrijednosti nezavisne varijable*** *(6)*
1) Način angažiranja i korištenja resursa	Broj angažiranih računala	r	cijeli broj	[1; 5]	Izbor iz ponude pružatelja usluge
	Snaga angaži-ranih računala	p	cijeli broj	[1; 4]	Izbor iz ponude pružatelja usluge
2) Složenost aplikacije	Složenost algoritma	a	sek/trans	[1; 10]	Multiplikator vremena identifikacije slike na jediničnoj konfiguraciji
	Broj akcija po upitu	q	Broj akcija /upit	[10; 200]	Broj CRUD akcija na bazom podataka po jednom upitu
3) Opseg podataka nad kojima se izvodi obrada	Opseg poslanih podataka	u	kB	[5,4;540]	1 slika za identifikaciju; 100 slika za identifikaciju
	Veličina baze podataka	b	MB	[1,2; 5,8]	Broj slika u bazi koja se pretražuje

U stupcima 4 i 5 tablice 5-II utvrđene su mjerne jedinice i granice mogućeg variranja za svaku značajku ogledne aplikacije koja će se u mjerenju promatrati kao nezavisna varijabla. U stupcu 6 daje se sažeti prikaz načina kako su ostvarene tehničke promjene radi postavljanja donje i gornje granice nezavisnih varijabli. Resursne varijable (r i p) mijenjaju se kod izvođenja aplikacije izborom parametara, u okviru mogućnosti koje pruža davatelj usluge. Da bi se varirale vrijednosti nezavisnih varijabli iz grupe 2 i 3 bilo je potrebno izvesti određene dopune u strukturi i programskom kôdu ogledne aplikacije, što je vidljivo u prilogu 5.1. U sljedećim odlomcima kratko je opisan način kako je ostvareno variranje nezavisnih varijabli i kako su određene donja i gornja granica variranja.

Broj angažiranih računala (r). Kod zaposjedanja resursa u oblaku ovaj se parametar bira u granicama (na primjer, od jednog do tisuću računala) koje omogućava pružatelj usluge. Za mjerenje provedeno u okviru ovog istraživanja donja granica zaposjedanja je bilo jedno računalo, a gornja 5 računala. Ova gornja granica bila je određena ponudom Microsoft-a zajednici razvojnih stručnjaka (*Community Technical Preview*) koja je vrijedila za Hrvatsku tijekom probnog rada na platformi Azure[2].

Snaga angažiranih računala (p). Analizom ponuda većeg broja pružatelja usluga oblačnog računarstva može se utvrditi da se računala na njihovim serverskim farmama relativno malo razlikuju po snazi. Obično je najslabije računalo koje se može angažirati sastavljeno od dvojezgrenog procesora na 1,7 GHz i 3,5 GB RAM-a te umreženo uz brzinu mreže od 1 $Gbit/sek$. Zbog toga će se to u ovom radu smatrati donjom granicom snage računala i označiti relativnom snagom 1. Najsnažnije računalo ima četverojezgreni procesor na 1,7 GHz i 7 GB

[2] U vrijeme pisanja ovog rada za korisnike iz Hrvatske još uvijek nije bio moguć komercijalni zakup oblačnih resursa na platformi Azure.

RAM-a, ali je također umreženo uz brzinu od 1 $Gbit/sek$. Takvo računalo će se označiti relativnom snagom 4.

Složenost algoritma aplikacije (a). Ovaj parametar je definiran kao procesorsko vrijeme, potrebno za izvođenje jedne transakcije ili upita na računalu jedinične konfiguracije i uz nepromjenljive srednje vrijednosti svih ostalih varijabli. Iskazivati će se u [sek/transakcija]. Za variranje ovog parametra trebalo je oglednu aplikaciju preurediti. Najjednostavniji način za promjenu složenosti algoritma ogledne aplikacije je ponavljanje glavne programske petlje kroz koju se taj algoritam izvršava, to jest da se za jednom učitanu sliku više puta ponavljaju koraci 3 i 4 iz opisa algoritma ogledne aplikacije u točki 5.3. U funkcionalnom smislu to znači da se za jednu učitanu sliku pronalazi ne samo jedna najsličnija već više "najsličnijih" slika. U ovom radu se kao donja granica varijacije smatra vrijeme za nalaženje jedne najsličnije slike, i ta će granica biti iskazana kao relativna vrijednost 1, a kao gornja granica nalaženje 10 "najsličnijih" slika, i ta će relativna vrijednost biti označena vrijednošću 10.

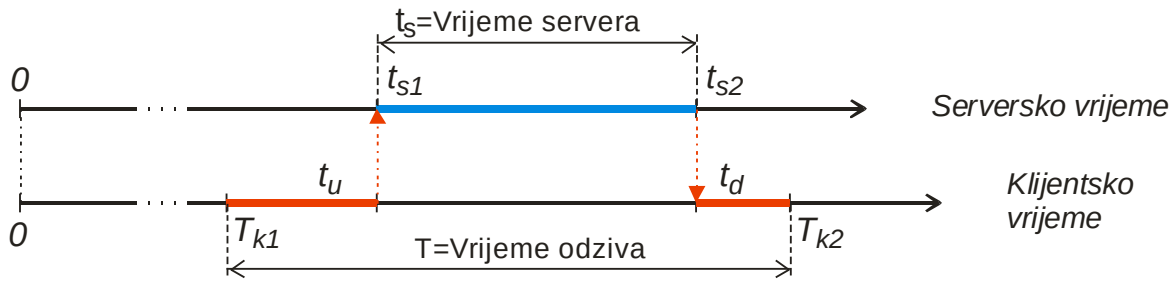
Opseg poslanih podataka (u). Jedinični zahtjev za oglednu aplikaciju je slanje jedne slike u oblak radi identifikacije. S obzirom da su slike u .jpg formatu, a slika je standardne rezolucije 180x200, to je donja granica za opseg podataka koji se šalju u oblak oko 5,4 kB. Gornja granica za ovu varijablu je 540 kB, što će kroz oglednu aplikaciju biti realizirano kao zahtjev da se na unajmljenim resursima u oblaku izvede sukcesivna identifikacija za 100 poslanih slika.

Veličina baze podataka u obradi (b). Za rad ogledne aplikacije bilo je na raspolaganju 240 slika (svaka zauzima memorijski prostor od 5,4 kB) za isto toliko različitih osoba. Za pohranjivanje tih slika potrebno je oko 1,3 MB memorije u bazi podataka, odnosno aplikacija radi najmanje nad tim opsegom podataka, pa je ta vrijednost odabrana kao donja granica za ovu nezavisnu varijablu. To odgovara poslovnoj bazi podataka nad kojom radi MRP-algoritam za manje poduzeće (6.000 zapisa po 200 bajtova). Gornja granica je postavljena na 5,8 MB, što odgovara veličini poslovne baze podataka nad kojom radi MRP algoritam za srednje poduzeće: 29.000 zapisa po 200 bajtova). Kroz oglednu aplikaciju će se to realizirati kao baza podataka od 1160 slika, koje predstavljaju istih 240 osoba, ali je svako lice snimljeno u više različitih varijanti.

Broj akcija po upitu na bazu podataka (q). Pod akcijom se ovdje podrazumijeva broj čitanja, upisivanja ili ažuriranja koji se izvodi nad bazom podataka po jednom upitu. U tom smislu različite aplikacije imaju različite zahtjeve prema bazi podataka. Tako se, na primjer, evidencija prometa skladišta tipično realizira kroz tri transakcije: *čitati* (pročitati matični zapis sa starim stanjem zaliha skladišne stavke koja je imala promet), *ažurirati* (ponovno upisati matične podatke iste stavke s novim stanjem zaliha) i *upisati* (prometnu stavku). S druge pak

strane, zadavanje primarnih potreba gotovog proizvoda radi izračuna potreba materijala raspuštanjem sastavnice može se realizirati najmanje sa svega 10-tak akcija nad bazom podataka (za vrlo jednostavan proizvod) ili do nekoliko tisuća akcija (ako se potrebe računaju za vrlo složen proizvod). U oglednoj aplikaciji ovaj će se uvjet oponašati čitanjem i upisivanjem ("akcijama") sadržajno beskorisnih (*dummy*) ali funkcionalno potpuno adekvatnih zapisa u bazi podataka slika, pri čemu se broj akcija određuje vanjskim parametrom. Kao donja granica postavljena je vrijednost 10 akcija po upitu, a gornja 200 akcija po upitu.

Praktična provedba mjerenja vrijednosti koje poprima zavisna varijabla T za različite vrijednosti nezavisnih varijabli $\boldsymbol{r}$, $\boldsymbol{p}$, $\boldsymbol{a}$, $\boldsymbol{u}$, $\boldsymbol{b}$ i $\boldsymbol{q}$ provedena je na način kako je prikazano na slici 5.2. Mjeri se takozvano vrijeme odziva (*response time*), koje je prema [31] definirano kao vremensko razdoblje između trenutka kada je korisnik aplikacije zatražio njezino izvođenje u računalnom sustavu, do trenutka kada je primio odgovor. Za ovo mjerenje[3] razvijen je vlastiti servisni program koji na klijentskom računalu bilježi vrijeme kad je izabrana slika koja će biti poslana u oblak radi identifikacije te vrijeme kada je iz oblaka primljen odgovor, s pronađenom sličnom slikom iz referentne baze podataka ili porukom da nije pronađena slika dovoljno velike sličnosti.



Slika 5.2: ***Koncept mjerenja vremena izvođenja ogledne aplikacije u oblaku***

Prema gornjoj slici se vidi da se vrijeme odziva T sastoji od vremena potrebnog za slanje podataka prema serveru t_u, vremena rada (jednog ili više) servera u oblaku t_s i vremena potrebnog za vraćanje rezultata obrade iz oblaka do klijenta t_d, to jest vrijedi:

$$T = T_{k2} - T_{k1} = t_u + t_s + t_d \qquad (5.4)$$

Na slici 5.2 valja također uočiti da se vrijeme mjeri u dva potpuno različita vremenska sustava: na klijentskom računalu, čija su fizička lokacija i lokalno vrijeme poznati, te na nekim serverima u oblaku, čija su fizička lokacija i lokalno vrijeme (u nepoznatoj vremenskoj zoni) za korisnika na klijentskoj strani potpuno nepoznati. Zato je ovdje trebalo primijeniti sustav mjerenja koji se temelji na registraciji događaja u apsolutnom vremenu, u kojem je

[3] S obzirom da se ovdje traži mjerenje s točnošću od jedne stotinke sekunde, korištena je mjerna jedinica *tick* što je jedan od načina mjerenja vremena u .net tehnologiji. 1 [*tick*] = 100*10^{-9} [*sek*] ili 1 [*msek*]= 10000 [*tick*]

vremenska jedinica $tick$. U tom apsolutnom vremenu (čiji je početak označen s 0 na slici 5.2) odbrojavanje za sva računala počinje na dan 01.01.0001. godine u 0 sek (od tog događaja do trenutka kada je provedeno mjerenje odbrojeno je približno $6{,}343077 \cdot 10^{17}$ tickova).

Da bi se dobio uvid da li odabrani faktori (kada variraju u prethodno definiranim granicama) signifikantno utječu na vrijeme izvođenja aplikacije u oblaku T te da bi se odabrala pravilna strategija njihovog detaljnijeg istraživanja, provedena su prethodna mjerenja vremena izvođenja ogledne aplikacije u oblaku, to jest provedeno je predistraživanje s ciljem da se utvrdi višedimenzionalni prostor u kojem se mogu kretati vrijednosti svih istraživanih varijabli.

Svako mjerenje zavisne varijable izvodi se za točno postavljene vrijednosti nezavisnih varijabli, koje se moraju precizno zapisati radi kasnije obrade rezultata mjerenja. Prema konvencijama iz teorije mjerenja i planiranja eksperimenata [10] neke specifične vrijednosti nezavisnih varijabli se označavaju skraćeno kako bi se izbjeglo višekratno ispisivanje postavljenih vrijednosti nezavisnih varijabli koje su realni brojevi s više decimala i najčešće izraženi u različitim mjernim jedinicama. Skraćena oznaka mjerenja se zapisuje tako da se postavljanje svake nezavisne varijable na donju graničnu vrijednost označava s ***-1*** a postavljanje na gornju graničnu vrijednost se označava kao ***+1***. Ovakvo označavanje (ili još kraće, samo sa znakovima + i -) nije proizvoljno, već je povezano s načinom obrade rezultata mjerenja te će se koristiti u točki 5.5. Shema postavljanja nezavisnih varijabli za planirana mjerenja je riječima objašnjena u stupcu 3 i formalno zapisana, u skladu s prethodnom notacijom, u stupcu 2 tablice 5-III.

U okviru prethodnog istraživanja provedeno je ukupno 11 mjerenja, a rezultati su prikazani u stupcima 4 i 5 tablice 5-III. Prvo mjerenje je izvedeno za slučaj da su sve nezavisne varijable na svojim donjim graničnim vrijednostima. U mjerenjima broj 2 do 7 samo je po jedna nezavisna varijabla postavljena na gornju graničnu vrijednost (oznaka $+1$), dok su sve ostale na donjim graničnim vrijednostima (oznaka -1). U mjerenju broj 8 sve su nezavisne varijable postavljene na gornje granične vrijednosti ($+1$). Zadnja tri mjerenja, izvedena su za slučaj da su vrijednosti svih nezavisnih varijabli postavljene na srednju vrijednost između donje i gornje granice. Rezultat ovih mjerenja će se koristiti radi ocjene mjerne greške i adekvatnosti matematičkog modela za istraživanu pojavu, što je detaljnije obrađeno u poglavlju 5.6

Rezultati mjerenja zapisani su u zadnja dva stupca tablice 5-III. Predzadnji stupac sadrži ukupno izmjereno vrijeme koje je predmet istraživanja jer za korisnika predstavlja vrijeme izvođenja aplikacije T, odnosno vrijeme odziva prema slici 5.2. Zadnji stupac sadrži vrijeme zaposjedanja jednog ili više procesora u oblaku t_s. Podatak o tom vremenu automatski dostavlja korisniku pružatelj usluga oblačnog računarstva, zajedno s rezultatima obrađenog servisnog zahtjeva. Razlika vremena $T - t_s$ troši se za komunikaciju između klijenta i oblaka.

Tablica 5-III: ***Vrijednosti nezavisnih varijabli i rezultati prethodnih mjerenja***

Redni broj (1)	***Postavljanje parametara (2)***						***Parametri na gornjoj granici (3)***	***Rezultat T [sek] (4)***	***Serversko vrijeme t_s [sek] (5)***
1.	-1	-1	-1	-1	-1	-1	***Nijedna***	**21,64**	19,01
2.	+1	-1	-1	-1	-1	-1	***Broj angažiranih računala (r)***	**11,70**	4,89
3.	-1	+1	-1	-1	-1	-1	***Snaga angažiranih računala (p)***	**18,30**	15,01
4.	-1	-1	+1	-1	-1	-1	***Složenost algoritma (a)***	**156,27**	153,87
5.	-1	-1	-1	+1	-1	-1	***Opseg poslanih podataka (u)***	**24,51**	21,17
6.	-1	-1	-1	-1	+1	-1	***Veličina baze podataka (b)***	**346,95**	344,01
7.	-1	-1	-1	-1	-1	+1	***Broj akcija po upitu (q)***	**37,97**	19,29
8.	+1	+1	+1	+1	+1	+1	***Sve***	**74,45**	66,57
9.	0	0	0	0	0	0	Svi parametri su na srednjoj vrijednosti između donje i gornje granice. Mjerenje se 3 puta ponavlja, a rezultati će se koristiti radi procjene adekvatnosti matematičkog modela u poglavlju 5.8.	152	
10	0	0	0	0	0	0		161	
11.	0	0	0	0	0	0		156	

Kvalitativnom analizom rezultata iz tablice 5-III mogu se izvući sljedeći zaključci, na temelju kojih će organizirati daljnje istraživanje:

1. Rezultati mjerenja su konzistentni jer je uvijek $T > t_s$ što je u skladu sa slikom 5.2.
2. Kada su sve nezavisne varijable na donjoj granici onda vrijeme servera $t_s = 19{,}01\ sek$ odgovara vremenu izvođenja ogledne aplikacije na jednom računalu u sjedištu korisnika. Kada pak se za izvođenje angažira svih 5 računala u oblaku (mjerenje pod rednim brojem 2 u kojem je nezavisna varijabla ***Broj angažiranih računala (r)*** postavljena na gornju granicu), onda se vrijeme servera smanjuje gotovo četiri puta, na $t_s = 4{,}89\ sek$, a skladu s tim se smanjuje i vrijeme izvođenja T. To je očekivani rezultat jer ukazuje na snažan utjecaj ove varijable na T (premda to ne mora biti korisniku uvijek neposredno vidljivo zbog zbrajanja serverskog i komunikacijskog vremena) i u skladu je s izrazom *(5.4)*. Zbog toga treba u daljnjem radu posebno istražiti utjecaj varijable ***r*** na T, kao i utjecaj interakcija varijable ***r*** i drugih varijabli.
3. Promjenom većine nezavisnih varijabli, osim varijabli ***Opseg poslanih podataka (u)*** i ***Snaga angažiranih računala (p)***, značajno se mijenja vrijednost zavisne varijable T, a veličinu i smjer te promjene valja detaljnije istražiti daljnjim mjerenjima.
4. Promjena nezavisne varijable ***Opseg poslanih podataka (u)*** povećava vrijeme odziva T, ali ne značajno, pa signifikantnost tog utjecaja treba potvrditi ili odbaciti daljnjim istraživanjem.

5. Nezavisna varijabla ***Snaga angažiranih računala (p)*** ponešto smanjuje vrijeme odziva T, ali nije sigurno da je to smanjenje statistički signifikantno pa pojavu također treba detaljnije istražiti.
6. Snažan utjecaj na ukupno vrijeme odziva T imaju također nezavisne varijable ***Složenost algoritma (a)*** i ***Veličina baze podataka (b)***.

Na kraju se može zaključiti da ekspertska procjena (čiji su rezultati prikazani u tablici 5-I) i prethodno istraživanje (provedeno mjerenjem ogledne aplikacije koja koristi računalne resurse u realnom oblaku jednog ponuđača) daju konvergentne zaključke u pogledu izbora nezavisnih varijabli za koje se opravdano može pretpostaviti da utječu na vrijeme odziva T. Na takav su način, kroz dva odvojena postupka, identificirane istovjetne nezavisne varijable $x_i\,(i = 1, 2, \ldots, n)$ iz izraza *(5.2)* kao parametri $\boldsymbol{p}$, $\boldsymbol{a}$, $\boldsymbol{u}$, $\boldsymbol{b}$, $\boldsymbol{q}$ i $\boldsymbol{r}$ iz tablice 5-II. Stoga će stoga daljnji rad biti usmjeren na istraživanje njihovog utjecaja na T, odnosno utvrđivanja sljedeće funkcijske zavisnosti:

$$T = f(p, a, u, b, q, r) \qquad (5.5)$$

Iz teorije pokusa je poznato da je kod potpuno nepoznate prirode izučavane pojave, zavisnoj od k nezavisnih varijabli, opravdano za početak pretpostaviti da se funkcijska ovisnost može iskazati polinomom k-tog reda gdje se pojavljuje k linearnih članova za sve nezavisne varijable, sve njihove interakcije r-tog reda (r=2 do k) kojih ima ukupno $C_r^k = \frac{k!}{[r!\cdot(k-r)!]}$, [10] jedna interakcija k-tog reda te konstantni član polinoma. Takav polinom ima, dakle, ukupno 2^n članova pa je za određivanje svih njegovih koeficijenata potrebno 2^n podataka. Ovi će se podaci dobiti detaljnim mjerenjima prema posebno razvijenom planu što je obrađeno u sljedećim točkama ovog rada.

5.8 Plan mjerenja

U prethodnim odlomcima je pokazano da odzivna funkcija T zavisi o varijablama $\boldsymbol{p}$, $\boldsymbol{a}$, $\boldsymbol{u}$, $\boldsymbol{b}$, $\boldsymbol{q}$ i $\boldsymbol{r}$. Priroda te funkcijske zavisnosti, za koju se pretpostavlja da se može opisati polinom k-tog reda, do ovog trenutka nije poznata te će se njezin tijek (to jest svi koeficijenti pretpostavljenog polinoma) izračunati u daljnjem postupku na temelju rezultata mjerenja. Za određivanje te ovisnosti može se prema [10] pretpostaviti da T, iako može linearno zavisiti od svake pojedinačne nezavisne varijable, može također zavisiti i o interakcijskim efektima više nezavisnih varijabli te da stoga f ne mora biti polinom prvog reda. Kvalitativnim razmatranjem dolazi se do zaključka da, na primjer, vrijeme izvođenja aplikacije u oblaku T ne zavisi jednako o broju angažiranih računala $\boldsymbol{r}$ za slučaj kada je složenost aplikacije $\boldsymbol{a}$ mala kao i za

slučaj kada je složenost aplikacije velika. Stvarna ovisnost se može utvrditi samo dovoljno velikim brojem mjerenja.

Ako se pretpostavi da bi za utvrđivanje svake zavisnosti trebalo mjeriti barem u tri mjerne točke, da bi svako to mjerenje ponoviti barem 3 puta (radi otklanjanja slučajnih grešaka u mjerenju), zatim da u gornjem problemu ima ukupno 6 nezavisnih varijabli te da mjerenja treba provesti za svaku kombinaciju promjena nezavisnih varijabli, trebalo bi izvesti ukupno $3 \cdot 3^6 = 2187$ mjerenja! Takav ekstenzivni postupak mjerenja praktično je neprovediv zbog više razloga:

- Mjerenje bi bilo dugotrajno (oko 550 sati, ako bi se za pripremu i izvedbu svakog mjerenja utrošilo prosječno samo 15 minuta) pa prema tome i skupo.
- Obrada tako velikog broja izmjerenih rezultata bila bi suviše složena.
- Pouzdanost rezultata bila bi vrlo mala jer tijekom mjerenja nije moguće zadržati iste uvjete, to jest ne bi se mogao isključiti utjecaj slučajnih varijabli (primjerice, da su tijekom mjerenja isključivana i ponovnog uključivana neka računala u oblaku).

Smanjivanje ukupnog broja mjerenja može se postići na tri načina:

1. Smanjivanjem broja mjerenja s tri na dvije mjerne točke po svakoj kombinaciji nezavisnih varijabli. To je dozvoljeno u početku istraživanja jer se s mjerenjem T za dvije različite vrijednosti nezavisne varijable x_i može odrediti linearna zavisnost T o x_i. Ako bi se kasnije pokazalo da pretpostavka o linearnoj zavisnosti nije održiva, mogu se izvesti dopunska mjerenja. U istraživanom problemu na taj će se način broj potrebnih mjerenja smanjiti s $3^6 = 729$ na $2^6 = 64$, dakle 11 puta!
2. Prikladnim izborom jedne mjerne točke radi utvrđivanja greške mjerenja (umjesto ponavljanja mjerenja u svakoj mjernoj točki), čime se broj potrebnih mjerenja smanjuje još dva puta.
3. Izostavljanjem mjerenja koja bi dala podatke za izračun koeficijenata regresijskog polinoma čija je vrijednost zanemariva, odnosno manja od mjerne greške.

Različiti načini provedbe ovakve optimalizacije broja mjerenja detaljno su razrađeni u literaturi iz teorije planiranja pokusa, kao što je [10]. Za istraživanje kakvo je potrebno provesti u ovom radu odabran je stoga takozvani parcijalni višefaktorski plan mjerenja (*Fractional Factorial Design)*. Ovakav je plan mjerenja prikladan u slučaju kada se istražuje pojava koja zavisi od velikog broja nezavisnih varijabli (tri ili više, a u ovom istraživanju ih se pojavljuje šest), a vrsta funkcijske zavisnosti nije niti približno poznata. U ovoj se teoriji svaka nezavisna varijabla tretira kao faktor ili čimbenik, a plan mjerenja se skraćeno označava kao plan oblika 2^{k-r}. Ovaj oblik skraćenog zapisivanja ujedno objašnjava i najvažnija svojstva:

- Mjerenja se izvode samo za donju $X_{i,min}$ i gornju $X_{i,max}$ *(i=1, 2, ..., n)* graničnu vrijednost svake nezavisne varijable (u ovom slučaju su to $\boldsymbol{p}$, $\boldsymbol{a}$, $\boldsymbol{u}$, $\boldsymbol{b}$, $\boldsymbol{q}$ i $\boldsymbol{r}$). Za n varijabli čija se vrijednost mjeri samo u dvije mjerne točke broj svih mogućih kombinacija mjernih točaka iznosi 2^k i takav plan mjerenja se u literaturi naziva potpuni faktorski plan.
- Radi lakšeg kasnijeg računanja uzima se da je $x_{i,min} = -1$ i $x_{i,max} = +1$. Da bi ta pretpostavka bila ispunjena, potrebno je provesti određene supstitucije, uzimajući u obzir stvarne granične vrijednosti nezavisnih varijabli X_i, kako su definirane u stupcu 5 tablice 5-II. Potrebne linearne supstitucije za sve istraživane parametre sažeto su prikazane u tablici 5-IV, a izvedene su prema sljedećem općem izrazu:

$$x_i = -\frac{2}{d-g} \cdot X_i + \frac{d+g}{d-g}$$

  U gornjem izrazu su d i g donja i gornja granica varijabli X_i (to jest $\boldsymbol{r}$, $\boldsymbol{p}$, $\boldsymbol{a}$, $\boldsymbol{u}$, $\boldsymbol{b}$, $\boldsymbol{q}$) u realnom intervalu *[d;g]*, dok je supstitucija x_i u granicama *[-1;+1]*. Koeficijenti smjera (m) i slobodni članovi (n) linearne transformacije dobiju se rješavanjem jednadžbi: $1=g \cdot m+n$ i $-1=d \cdot m+n$ za svaku nezavisnu varijablu.
- Od svih mogućih kombinacija vrijednosti k nezavisnih varijabli mjerenja se ne izvode u svim mjernim točkama (kojih ima ukupno 2^k), već u 2^r puta manje mjernih točaka, koje se izabiru prema posebnoj shemi. Ako je, na primjer, vrijednost $r=2$, onda to znači da će se izvesti samo četvrtina mjerenja u odnosu na broj mjerenja koja traži potpuni faktorski plan, za $r=3$ samo osmina mjerenja, i tako dalje.

Tablica 5-IV: ***Granice originalnih parametara mjerenja i njihove supstitucije***

Naziv parametra	$[X_{i,min};X_{i,max}]$	***Supstitucija***	$[x_{i,min};x_{i,max}]$
Snaga angažiranih računala (p)	[1; 4]	$x_1 = \frac{2}{3} \cdot p - \frac{5}{3}$	[-1;+1]
Složenost algoritma (a)	[1; 10]	$x_2 = \frac{2}{9} \cdot a - \frac{11}{9}$	
Opseg poslanih podataka (u)	[5,4;540]	$x_3 = \frac{2}{534,6} \cdot u - \frac{545,4}{534,6}$	
Veličina baze podataka (b)	[1,2; 5,8]	$x_4 = \frac{2}{4,6} \cdot b - \frac{7}{4,6}$	
Broj akcija po upitu (q)	[10; 200]	$x_5 = \frac{2}{190} \cdot q - \frac{210}{190}$	
Broj angažiranih računala (r)	[1; 5]	$x_6 = \frac{1}{2} \cdot r - \frac{3}{2}$	

Teoretsko obrazloženje za mogućnost bitnog smanjenja broja mjerenja temelji se na pretpostavci da je za upoznavanje prirode neke pojave najčešće dovoljno tražiti funkcijsku zavisnost zavisne varijable o svim nezavisnim varijablama (ili faktorima) kojih ima k te o svim mogućim dvofaktornim interakcijama kojih ima $k!/[r! \cdot (k-r)!]$, to jest 15 u ovom radu gdje se promatra 6 nezavisnih varijabli. Višefaktorne interakcije tri ili više faktora se pri tome zanemaruju što se prema [10] pokazalo opravdanim u brojnim provedenim istraživanja. U tom se slučaju funkcijska zavisnost može predstaviti matematičkim modelom u obliku regresijskog polinoma drugog reda koji ima oblik:

$$T_r = T_0 + \sum_{i=1}^{k} \beta_i \cdot X_i + \sum_{i<j}^{k} \beta_{ij} \cdot X_i \cdot X_j + \varepsilon \qquad (5.6)$$

U matematičkom modelu *(5.6)* vrijednost ε predstavlja ukupnu grešku za koju se teoretski procijenjena vrijednost vremena odziva T_r razlikuje od stvarne vrijednosti te pojave T. Uzroci ove greške mogu biti slučajna odstupanja tijekom mjerenja ali i neadekvatnost matematičkog modela[4] koji, pretpostavlja se, uključuje samo linearne utjecaje nezavisnih varijabli i njihove dvojne interakcije.

Matematički model *(5.6)* je određen ako je poznato $1+k+k\cdot(k-1)/2$ koeficijenata polinoma na desnoj strani. Vrijednosti koeficijenata mogu se izračunati na temelju isto toliko rezultata mjerenja ako se (uz minimalni broj ponavljanja i primjenom prikladnih statističkih metoda) osigura nepristrana procjena signifikantnosti tako izračunatih koeficijenata, odnosno utvrdi da je greška ε manja od dozvoljene. S obzirom da vrijedi:

[4] Ako je linearni model neadekvatan prelazi se na kvadratni model koji ima k koeficijenata više. Ako se mjerenja provedu po Box-Wilson metodi [10], tada se za izračun koeficijenata kvadratnog modela koriste sva mjerenja provedena radi određivanja koeficijenata za model prvog reda i svega $2k$ dodatnih mjerenja.

$$1+k+\frac{k\cdot(k-1)}{2}<2^k \text{ za svaki } k>2 \qquad (5.7)$$

proizlazi da se za svaki ozbiljni istraživački problem koji ima $k>2$ nezavisnih varijabli, regresijski polinom oblika *(5.6)* može odrediti na temelju manje od 2^k mjerenja. Ta "ušteda" u broju mjerenja je to veća što je veći broj nezavisnih varijabli u istraživanom problemu. Za slučaj koji se istražuje u ovom radu, gdje se (prema tablici 5-III) javlja $k=6$ nezavisnih varijabli, dovoljno bi bilo provesti 22 mjerenja da bi se mogli odrediti svi koeficijenti regresijskog polinoma oblika *(5.6)* u kojem se pojavljuje konstantni član, 6 članova za 6 nezavisnih varijabli i 15 članova za 15 mogućih dvofaktornih interakcija.

Prije praktične provedbe mjerenja potrebno je, poznavajući prirodu istraživane pojave, procijeniti da li baš **sve** višefaktorne interakcije mogu utjecati na vrijednost zavisne varijable T (teoretska rasprava o tome nalazi se u [10]). Tako, na primjer, nije za očekivati da bi se mogao pojaviti bilo kakav pozitivni sinergijski efekt zbog interakcije varijabli ***Snaga angažiranih računala (p)*** i ***Opseg poslanih podataka (u)***, odnosno da bi njihov kombinirani utjecaj (kad su obje na svojoj gornjoj vrijednosti) mogao uzrokovati veću promjenu zavisne varijable od zbroja njihovog pojedinačnog utjecaja. Naravno, moguć je i negativni sinergijski učinak, ako bi istovremena pozitivna promjena dviju varijabli uzrokovala manju promjenu zavisne varijable od zbroja njihovog pojedinačnog utjecaja.

Isključivanje nekih interakcija omogućava da se koeficijenti preostalih članova polinoma mogu izračunati od podataka iz manje mjerenja. Ako se izvede 16 (ili 2^{6-2}) mjerenja, moguće je iz dobivenih podataka izračunati 15 koeficijenata za polinom oblika kao u izrazu *(5.6)*, odnosno koeficijente uz 6 linearnih članova i 9 dvofaktornih interakcija. To može dati zadovoljavajući rezultat ako iz proračuna ispadnu koeficijenti za one interakcije koje su po veličini zanemarive. Budući da ekspert koji poznaje istraživano područje može predvidjeti zanemarive interakcije (što je pokazano u prethodnom odlomku), preostaje pronaći takav plan mjerenja koji će dati podatke za izračun željenih koeficijenata, ali ne i podatke za izračun koeficijenata čija je utjecajnost unaprijed procijenjena kao zanemariva. Izbor takvih mjernih točaka koje pripadaju višefaktorskom parcijalnom planu pokusa tipa 2^{r-k} je detaljno objašnjen u [10].

U ovom istraživanju primijenit će se plan mjerenja za takozvanu četvrtrepliku. Drugim riječima, od moguće 64 $(=2^6)$ kombinacije vrijednosti nezavisnih varijabli mjerenje će se izvesti samo za $1/4$ $(16=2^{6-2})$ mjernih točaka. Očekuje se da će tako postavljen plan mjerenja dati podatke za izračun koeficijenata uz sve linearne članove polinoma i one

članove polinoma koji stoje uz sve nezanemarive dvofaktorne interakcije[5]. Plan mjerenja se u tom slučaju može izvesti prema postupku koji je postavio F. Yates. Koraci Yatesovog postupka detaljno su opisani u [10] te ćemo ovdje napisati samo sažetak:

1. Svim nezavisnim varijablama treba pridružiti oznake u skladu s konvencijom iz teorije planiranja pokusa. Po toj konvenciji, svaki se plan mjerenja opisuje shemom u kojoj se može pojaviti toliko slova koliko ima ukupno nezavisnih varijabli. Ako se u problemu pojavljuje $k=6$ nezavisnih varijabli (kao u ovom istraživanju) onda će se koristiti redom slova a, b, c, d, e i f. Oznaka plana mjerenja oblikuje se tako da se za svaku nezavisnu varijablu postavljenu na gornju granicu ispiše slovo koje odgovara rednom broju te varijable, a to se slovo ispušta kad je varijabla na donjoj granici. Tako, na primjer, shema mjerenja bdf znači da su druga, četvrta i šesta varijabla postavljene na gornju granicu dok su sve ostale na donjoj granici. Plan mjerenja u kojem su sve nezavisne varijable na donjoj granici se označava oznakom 1. Plan mjerenja označen kao $abcdef$ znači da je svih 6 varijabli na gornjoj granici.

2. Izraditi kompletni plan mjerenja koji ima 2^k mjernih točaka – 64 mjerne točke u ovom istraživanju. Prva mjerna točka ima oznaku 1 (što znači da su sve varijable na donjoj granici). Druga mjerna točka određuje se tako da se uvede prvo slovo, dakle a. Iduća mjerna točka određuje se uvođenjem sljedećeg slova, dakle b. Svaka nova mjerna točka nastaje kombiniranjem netom uvedenog slova sa svim prethodnim planovima koji su već označeni slovima, dakle slijedi ab. Novi planovi dobivaju se ponavljanjem zadnja dva koraka sve dok se ne iskoriste sva slova. Za ovo istraživanje kompletni plan bi izgledao: $1\ a\ b\ ab\ c\ ac\ bc\ abc\ d\ ad\ bd\ abd\ cd\ acd\ bcd\ abcd\ e\ ae\ be\ abe\ ce\ ace\ .....\ bcdef\ abcdef$. Ovako dobiveni redoslijed mjerenja zove se standardni Yatesov red. Treba naglasiti da se mjerenja mogu izvesti redoslijedom koji najbolje odgovara prirodi istraživane pojave, ali se dobiveni rezultati za daljnju obradu moraju upisati u redom kako je definirano Yatesovim redom.

3. Kompletan plan mjerenja od 64 mjerne točke treba podijeliti na četiri simetrična parcijalna plana (četvrtreplike) od kojih svaki ima 16 mjernih točaka. Ne ulazeći dublje u objašnjavanje svojstva simetričnosti, niti u teoretske dokaze za opravdanost ovog postupka koje su detaljno izložene u [10][6], može se utvrditi da je njegov krajnji

[5] Ovdje valja upozoriti na još neke konvencije iz teorije planiranja pokusa. Malim slovima a, b, c, d, e, f označava se shema mjerenja (to jest razine na kojoj su nezavisne varijable u nekom mjerenju), ali isto tako i vrijednost zavisne varijable (*yield, response*) izmjerena kod kombinacije nezavisnih varijabli zadanih shemom. Velikim slovima A, B, C, D, E, F biti će označene vrijednosti linearnih koeficijenata regresijskog polinoma koji opisuje T, procijenjene iz podataka dobivenih mjerenjem. Kombinacija velikih slova predstavlja izračunate koeficijente uz članove polinoma koji iskazuju sinergističke učinke interakcija nezavisnih varijabli na T. Tako će s BDF biti označen izračunati koeficijent trofaktorne interakcije druge, četvrte i šeste nezavisne varijable.

[6] Neke knjige iz teorije planiranja pokusa, poput [76], imaju katalog s više simetričnih parcijalnih planova.

cilj dobiti takve planove mjerenja u kojima se svaka varijabla nalazi jednaki broj puta na gornjoj i donjoj razini. Za slučaj da se istražuje 6 varijabli to praktično znači da će svaka četvrtreplika imati 16 različitih mjerenja i da će se u svakoj od njih, svako pojedino slovo (od a do f), pojaviti 8 puta (to jest onda kad će varijabla koju to slovo predstavlja biti postavljena na gornju granicu). Jedna takva četvrtreplika, nazvana glavnom replikom, izgledala bi ovako: $1\ ab\ acf\ bcf\ bce\ ace\ ef\ abef\ cd\ abcd\ bdf\ adf\ bde\ ade\ cdef\ abcdef$. Glavna replika odabire se posebnim postupkom koji se temelji na određivanju višefaktorne interakcije čije se vrijednosti smatraju zanemarivim s obzirom na prirodu problema. Ta se interakcija u literaturi naziva ortogonalni kontrast [76] ili *defining relation* [10], a mogla bi se prevesti kao **ključ replike**. Ako se u planu pokusa sa 6 varijabli vrijednost interakcije $ABCD$ smatra zanemarivom, onda se smatra da je to ključ replike i označava kao $I=ABCD$, gdje I operator identičnosti. Ključem replike mogu se isključiti neka od mjerenja na temelju čijih rezultata se izračunava vrijednost interakcije pa se tako dolazi do parcijalnog plana pokusa[7]. U takvom parcijalnom planu pokusa će se za izračun više različitih koeficijenata koristiti isti podaci, što se za gornji primjer zapisuje kao $I=ABCD=ABEF=CDEF$. To se u [10] naziva *aliased effects* te znači da će se neki izračunati koeficijenti odnositi na zajednički utjecaj više interakcija nezavisnih varijabli.

4. U ovom koraku treba napraviti skraćeni Yatesov red za podskup od 4 slova iz skupa od 6 slova koja predstavljaju 6 varijabli istraživane pojave. Ovaj podskup se formira privremenim ispuštanjem 2 slova (*dead letters*), a zatim se za preostala 4 slova radi privremeni plan pokusa na način kako je to predviđeno u koraku 2. Ako su, na primjer, iz skupa $\{a\ b\ c\ d\ e\ f\}$ ispuštena slova a i c, onda za preostali podskup slova $\{b\ d\ e\ f\}$ skraćeni Yatesov red izgleda onako kako je to upisano u prvom stupcu tablice 5-VI. Takav red ima 16 članova – toliko koliko ima parcijalni faktorski plan za problem sa 6 varijabli (2^{6-2}). Valja uočiti da je ovaj plan također simetričan s obzirom na nezavisne varijable koje sadrži.

5. U zadnjem koraku se skraćeni Yatesov red proširuje vraćanjem privremeno ispuštenih slova kako bi novi plan mjerenja obuhvatio sve istraživane nezavisne varijable. Ovo proširivanje ne smije poremetiti simetričnost plana pokusa (što znači da se svako "vraćeno" slovo uvodi u samo polovinu članova). Izvodi se tako da se glavnoj replici dodaju ispuštena slova na mjestima gdje su ona u jednoj od sporednih replika. Tako se dobiva prošireni Yatesov red, upisan u drugom stupcu tablice 5-VI,

[7] Za izbor pravog ključa replike nema pravila. Dobar je onaj koji će omogućiti korektnu provedbu četvrtog i petog koraka ovog algoritma što znači da bi trebalo provjeriti nekoliko mogućih izbora. Stoga se preporuča da se za parcijalni plan pokusa određenog tipa pregledaju gotovi planovi iz kataloga. Tako je i za potrebe ovog rada korišten provjereni plan pokusa tipa 2^{6-2} iz kataloga koji se nalazi u [10].

koji predstavlja parcijalni višefaktorski plan po kojem treba provesti mjerenje. Slova dodana stupcu 2 zapisana su u zagradama. Valja naglasiti da redoslijed nezavisnih varijabli, kao i činjenica da su oznake nekih varijabli u zagradama, ne mijenja smisao sheme mjerenja. Stoga zapis $bd(ac)$ znači isto što i $abcd$, to jest da pri tom mjerenju treba prve 4 nezavisne varijable postaviti na gornju granicu dok će varijable e i f ostati na donjoj graničnoj vrijednosti.

Na slici 3 prikazan je raspored mjernih točaka prema proširenom Yatesovom redu. Simetričnost plana mjerenja se vidi tako što je svaki parametar 8 puta postavljen na donju i 8 puta na gornju granicu. Takvo osmerostruko ponavljanje mjerenja svakog parametra omogućava statističku procjenu njegove signifikantnosti.

			D_D				D_G			
			E_D		E_G		E_D		E_G	
			F_D	F_G	F_D	F_G	F_D	F_G	F_D	F_G
A_D	B_D	C_D	1			ef				
		C_G					cd			$cdef$
	B_G	C_D						bdf	bde	
		C_G		bcf	bce					
A_G	B_D	C_D						adf	ade	
		C_G		acf	ace					
	B_G	C_D	ab			$abef$				
		C_G					$abcd$			$abcdef$

Slika 5.3: ***Raspored mjernih točaka za parcijalni višefaktorski plan pokusa tipa 2^{6-2}***

Iz teorije planiranja pokusa je poznato da rezultati ovako izvedenih mjerenja omogućavaju procjenu vrijednosti koeficijenata regresijskog polinoma kojim se opisuje utjecaj svih 6 nezavisnih varijabli na zavisnu varijablu. Međutim, zbog toga što su poznati rezultati mjerenja samo jedne četvrtreplike, nije moguće izračunati pojedinačne vrijednosti svih 65 članova regresijskog polinoma, već samo takozvane pomiješane utjecaje (*confounded effects*). Ne ulazeći u detalje teoretske analize iz [10], ovdje se samo navodi zaključak: na temelju podataka iz ovako postavljenog mjerenja moguće je izračunati one "mješavine" koeficijenata koje su navedene u stupcu 3 tablice 5-VI. Zapis u tom stupcu treba (za drugi redak tablice) čitati ovako: moguće je procijeniti vrijednost koeficijenta B koji pokazuje utjecaj druge nezavisne varijable (b), ali ta vrijednost uključuje i utjecaj dviju trofaktornih i jedne petfaktorne interakcije. Međutim, kako je već ranije spomenuto da su u praktičnim istraživanjima ovog tipa realni utjecaji interakcija trećeg i svih viših redova zanemarive, to se može smatrati da B predstavlja dobru procjenu linearnog utjecaja druge varijable, odnosno da je B linearni

koeficijent drugog člana regresijskog polinoma. Pregledom cijelog stupca 3 može se vidjeti da isti zaključak vrijedi i za koeficijente A, C, D, E i F, to jest da se temeljem rezultata ovako postavljenog mjerenja mogu izračunati koeficijenti linearnog utjecaja svih 6 nezavisnih varijabli na zavisnu varijablu jer ti koeficijenti, doduše, sadrže utjecaje višefaktornih interakcija, ali su one zanemarive).

Nešto drugačije zaključivanje vrijedi za procjenu utjecaja dvofaktornih interakcija AF, BF, CF, DF i EF. Iz stupca 3 se vidi da su one pomiješane s po još jednom dvofaktornom interakcijom (na primjer, AF dolazi s BE u šestom retku tablice) i nekim interakcijama višeg reda. Za interakcije višeg reda već je utvrđeno da se mogu zanemariti, ali što je s miješanjem dviju dvofaktornih interakcija? Za pravilan odgovor na ovo pitanje treba se vratiti na prirodu nezavisnih varijabli koje se ovdje istražuju. Ako se, naime, mjerenje organizira tako da se parametru ***Broj angažiranih računala (r)*** u tablici 5-II pridruži oznaka f iz sheme mjerenja, onda će koeficijenti AF, BF, CF, DF i EF procjenjivati jačinu utjecaja interakcije tog parametra s drugih pet parametara, a to je upravo ono što se od ovog istraživanja očekuje. Ostale **dvofaktorne** interakcije koje se miješaju s navedenim uključuju druge parametre i ne očekuje se da one značajno utječu na vrijeme odziva T, ali **linearni** utjecaj tih parametara je, naravno, važan i on će ovakvim planom mjerenja biti istražen. Iz ovih razmatranja slijedi, dakle, da je mjerenjem prema parcijalnom faktorskom planu tipa 2^{6-2} moguće utvrditi funkcijsku zavisnost vremena izvođenja ogledne aplikacije u oblaku, pri čemu treba parametrima iz tablice 5-II pridružiti oznake nezavisnih varijabli redom kako je provedena supstitucija (tablica 5-IV) što je pregledno prikazano u tablici 5-V.

Tablica 5-V: ***Oznake parametara istraživanja i nezavisnih varijabli***

Naziv parametra koji se istražuje	*Oznaka parametra kao nezavisne varijable*	*Oznaka parametra u matematičkom modelu*
Snaga angažiranih računala (p)	a	x_1
Složenost algoritma (a)	b	x_2
Opseg poslanih podataka (u)	c	x_3
Veličina baze podataka (b)	d	x_4
Broj akcija po upitu (q)	e	x_5
Broj angažiranih računala (r)	f	x_6

Mjerenje vremena odziva T za oglednu aplikaciju smještenu u oblak (opisanu u potpoglavlju 5.3), izvedeno prema parcijalnom višefaktorskom planu mjerenja tipa 2^{6-2}, dalo je rezultate koji su upisani u stupac 4 tablice 5-VI. Izmjereni rezultati kreću se u vrlo širokom rasponu od 16 do 323 sekunde (što je na prvi pogled potpuno stohastička raspodjela vremena) te ih je

potrebno interpretirati na prethodno izložen način i matematički obraditi radi utvrđivanja utjecaja istraživanih faktora na vrijeme odziva T.

Mjerenje za slučaj kada su svi istraživani parametri na donjoj granici je opisano shemom koja ima oznaku *1*. Rezultat tog mjerenja je upisan u prvi redak stupca 4. Plan mjerenja označen kao *abcdef* znači da je svih 6 varijabli na gornjoj granici što odgovara mjerenju koje je zapisano u zadnjem retku istog stupca[8]. U ostalim poljima stupca 4 upisane su izmjerene vrijednosti zavisne varijable T za one kombinacije donjih i gornjih vrijednosti istraživanih parametara koje su određene proširenim Yatesovim redom.

Tablica 5-VI: ***Plan i rezultati mjerenja te izračun koeficijenata za T***

Skraćeni Yates-ov red (1)	*Prošireni Yates-ov red (2)*	*Procijenjeni utjecaji (3)*	*Rezultat mjerenja T [sek] (4)*	*1 (5)*	*2 (6)*	*3 (7)*	*4 (8)*	*kvadrat odstupanja (9)*	*Signifikantnost k_s (10)*	*Vrijednost signifikantnih koeficijenata (11)*
1	1		24	143	444	1448	2.489			155,56
b	b(a)	B+3+3+5	119	301	1004	1041	613	23.485,56	91,99	38,31
d	d(c)	D+3+5+3	103	575	267	462	33	68,06	0,27	2,06
bd	bd(ac)	BD+AC+4+4	198	429	774	151	69	297,56	1,17	4,31
e	e(ac)	E+5+3+3	232	197	190	12	1.067	71.155,56	278,70	66,69
be	be(c)	BE+4+AF+4	343	70	272	21	63	248,06	0,97	3,94
de	de(a)	DE+4+4+CF	134	313	85	50	-29	52,56	0,21	-1,81
bde	bde	BDE+3+3+3	295	461	66	19	103	663,06	2,60	6,44
f	f(ac)	F+5+3+3	73	95	158	560	-407	10.353,06	40,55	-25,44
bf	bf(c)	BF+4+AE+4	124	95	-146	507	-311	6.045,06	23,68	-19,44
df	df(a)	DF+4+4+CE	18	111	-127	82	9	5,06	0,02	0,56
bdf	bdf	BDF+3+3+3	52	161	148	-19	-31	60,06	0,24	-1,94
ef	ef	EF+6+AB=CD	149	51	0	-304	-53	175,56	0,69	-3,31
bef	bef(a)	BEF+5+A+3	164	34	50	275	-101	637,56	2,50	-6,31
def	def(c)	DEF+5+3+C	205	15	-17	50	579	20.952,56	82,07	36,19
bdef	bdef(ac)	BDEF+4+AD+BC	256	51	36	53	3	0,56	0,00	0,19

Na temelju rezultata ovako postavljenog mjerenja može se najprije odrediti koji od istraživanih parametara statistički signifikantno utječu na vrijeme izvođenja ogledne aplikacije u oblaku, a nakon toga će se pomoću tih parametara oblikovati izraz *(5.6)* koji matematički opisuje istraživanu pojavu.

Procjena signifikantnosti temelji se analizi varijance (ANOVA) i testiranju nulte hipoteze H_0: *koeficijent izračunat na temelju mjerenja je premali za zaključak da odgovarajući*

[8] Znak *1* u stupcu 2 tablice V-VI nije broj već oznaka sheme mjerenja. Isto tako, slova *abcdef* označavaju razinu nezavisnih varijabli, a ne kratice parametara koji se istražuju. Ako se parametar ***p*** postavlja na gornju granicu onda će to biti iskazano kao ***a*** u shemi mjerenja, a ako se složenost algoritma kod mjerenja postavlja na gornju granicu, onda je to u shemi mjerenja zapisano kao ***b*** (jer se radi o drugoj varijabli).

parametar utječe na vrijeme odziva. Da bi se potvrdila ili odbacila H_0 potrebno je izračunati procijenjene vrijednosti 6 linearnih koeficijenata s kojima istraživani parametri utječu na vrijeme odziva te utjecaj njihovih 9 dvofaktornih interakcija. Teoretska obrazloženja za te izračune se nalaze u [10]. Za praktičnu provedbu opsežnog računskog postupka najbolje je koristiti Yatesov algoritam (ako su mjerne točke određene proširenim Yatesovim redom).

Yatesov algoritam izvodi se tako da se u tablici s upisanim rezultatima mjerenja po proširenom Yatesovom redu formira pomoćni stupac (stupac 5 u tablici 5-VI) u čije se prvo polje odozgo upisuje zbroj rezultata prva dva mjerenja, u drugo polje zbroj rezultata trećeg i četvrtog mjerenja, u treće polje zbroj rezultata petog i šestog mjerenja i tako redom, dok se u osmo polje ne upiše zbroj rezultata petnaestog i šesnaestog mjerenja. Nakon toga se sljedeća polja pomoćnog stupca popunjavaju vrijednostima razlika dvaju uzastopnih rezultata mjerenja i to tako da se u deveto polje upisuje razlika rezultata drugog i prvog mjerenja, u deseto polje razlika četvrtog i trećeg mjerenja i tako redom, sve dok se u šesnaesto polje pomoćnog stupca ne upiše razlika rezultata šesnaestog i petnaestog mjerenja. Valja uočiti da se razlika dobiva uvijek tako da se od vrijednosti iz parnog rednog broja mjerenja oduzima vrijednost od prvog nižeg neparnog rednog broja mjerenja. Na ovaj način popunjen je prvi pomoćni stupac, to jest stupac 5 u tablici 5-VI.

Cijeli ovaj postupak izvodi se četiri puta (toliki broj puta koliko iznosi potencija kojom je određen broj mjerenja u dvofaktornom pokusu, ovdje je to $2^{6-2} = 2^4$), ali tako da se za svaki sljedeći pomoćni stupac uzimaju vrijednosti iz prethodnog pomoćnog stupca. Na taj način su u tablici 5-VI formirana četiri pomoćna stupca. Iz teorije planiranja pokusa [10] je poznato da ovako formirani rezultati predstavljaju šesnaest puta uvećanu procjenu koeficijenata koji su navedeni u adekvatnom polju stupca 3. Tako na primjer peto polje odozgo u stupcu 8 ima vrijednost *1067* što znači da je *E=66.69 (=1067/16)* procijenjena vrijednost koeficijenta s kojim nezavisna varijabla e (to jest parametar ***Broj akcija po upitu a***) utječe na T. Vrijednosti signifikantnih koeficijenata upisane su u stupac 11 tablice 5-VI.

Iz teorije planiranja pokusa također je poznato da vrijednost u prvom retku (u tablici 5-VI je to *2489*) podijeljena sa 16 daje konstantni član za polinom oblika *(5.6)*. čija je vrijednost upisana u prvo polje stupca 11 tablice 5-VI.

Za procjenu signifikantnosti koeficijenata potrebno je vrijednosti iz stupca 8 podijeliti s 4 (kvadratni korijen iz broja mjerenja) i to kvadrirati. Rezultat te operacije su kvadrati srednje vrijednosti procjene koeficijenata što je za ovaj slučaj upisano u stupcu 9 tablice 5-VI. S druge pak strane, potrebno je izračunati zbroj kvadrata srednjih vrijednosti onih koeficijenata koji **ne** stoje uz linearne članove i odabrane dvofaktorne interakcije (*sum of squares error* – SSQ_{error}) i srednju vrijednost te sume (*mean of squares error* – MSQ_{error}). U ovom slučaju to

su koeficijenti u recima 4, 8, 12 i 16 (ispisani zeleno) čija srednja vrijednost iznosi $MSQ_{error} = 1.021{,}25$. Valja uočiti da ova statistička veličina ima stupanj slobode (*degree of freedom)* $df=4$. Ako se kvadrati srednje vrijednosti koeficijenata (stupac 9) podijele s MSQ_{error}, dobiju se koeficijenti signifikantnosti koji su u ovom slučaju upisani u stupac 10 i imaju $df=1$. Radi provjere H_0 ovi se koeficijenti signifikantnosti uspoređuju s graničnom vrijednošću po F-distribuciji, koja za $\alpha=0.05$ i stupnjeve slobode $(1,4)$ iznosi $k_F = 7{,}7086$. Za svaki koeficijent iz stupca 10 za kojeg vrijedi $k_s > k_F$ hipoteza H_0 se odbacuje, to jest koeficijent se smatra signifikantnim.

Cijeli ovaj postupak sadržan je u programskom rješenju izrađenom u Excelu, koje se nalazi u prilogu 5.2.

Uvidom u stupac 10 tablice 5-VI može se konačno, na temelju objektivno provedenih mjerenja, zaključiti da na istraživanu pojavu – vrijeme izvođenja T ogledne aplikacije u oblaku – dokazano značajan utjecaj imaju nezavisne varijable označene kao b, c, e, f i dvofaktorna interakcija bf. Prema tablici 5-VI to su redom parametri:

- ***Složenost algoritma (a)***
- ***Opseg poslanih podataka (u)***
- ***Broj akcija po upitu (q)***
- ***Broj angažiranih računala (r)***

Za parametre ***Snaga angažiranih računala (p)*** i ***Veličina baze podataka (b)*** ne može se sa sigurnošću tvrditi da njihovo variranje značajno utječe na promjenu zavisne varijable T. To znači da se izraz *(5.5)* može reducirati na oblik *(5.8)*.

$$T = f(a,u,q,r) \tag{5.8}$$

Zaključak da parametri $\boldsymbol{r}$, $\boldsymbol{a}$, $\boldsymbol{u}$ i $\boldsymbol{q}$ značajno utječu na T je polazna osnova za daljnje istraživanje kroz koje treba točnije odrediti veličinu utjecaja svakog parametra posebno te sinergistički učinak interakcije parametara $\boldsymbol{a}$ i $\boldsymbol{r}$.

5.9 Izračun vremena odziva

U prethodnoj točki određeni su signifikantni koeficijenti i zaključeno da vrijeme izvođenja T ogledne aplikacije u oblaku zavisi o promjeni nezavisnih varijabli b, c, e, f i dvofaktornoj interakciji bf. U skladu s tim konačno se može odrediti polinom prvog reda za kojeg se može pretpostaviti da dovoljno dobro opisuje pojavu unutar istraživane domene (definirane gornjim i donjima granicama nezavisnih varijabli prema tablici 5-II):

$$T_r = 155{,}6 + 38{,}3 \cdot x_2 + 36{,}2 \cdot x_3 + 66{,}7 \cdot x_5 - 25{,}4 \cdot x_6 - 19{,}4 \cdot x_2 \cdot x_6 \tag{5.9}$$

U gornjem izrazu su oznake za istraživanje parametre a, b, c, d, e i f zamijenjene redom s nezavisnim varijablama x_1, x_2, x_3, x_4, x_5 i x_6 (prema pridruživanju u tablici 5-V) kako bi se naznačilo da je polinom *(5.9)* određen za sve vrijednosti koje odgovarajući parametar može poprimiti unutar svoje donje i gornje granične vrijednosti.

Ako gornji polinom dobro opisuje istraživanu pojavu, onda bi vrijednosti za T u različitim mjernim točkama koje se mogu **izračunati** pomoću tog polinoma morale biti približno iste vrijednostima koje su ranije **izmjerene** u stvarnosti. Takav se proračun može izvesti ako se razine nezavisnih varijabli, u mjernim točkama definiranima u drugom stupcu tablice 5-VI, upišu u pravokutnu matricu $\boldsymbol{M}$ dimenzija $16x16$. Ako se koeficijenti polinoma *(5.9)* prikažu kao matrica $\boldsymbol{K}$ dimenzija $16x1$ (ili stupčani vektor), onda umnožak ta dva vektora daje stupčani vektor rezultata $\boldsymbol{R}$ dimenzija $16x1$ čiji elementi sadrže izračunate vrijednosti mjernih točaka, odnosno vrijedi matrična jednadžba: $\boldsymbol{M} \times \boldsymbol{K} = \boldsymbol{R}$. Ovaj račun je najlakše izvesti korištenjem funkcije *MMULT(array1;array2)* u Excelu što prema prilogu 5.2 izgleda ovako:

$$
\begin{array}{ll}
 & \\
1 & \Phi \\
ab & B \\
cd & D \\
abcd & \\
ace & E \\
bce & AF \\
ade & CF \\
bde & \\
acf & F \\
bcf & BF \\
adf & DF \\
bdf & \\
ef & EF \\
abef & AF \\
cdef & CF \\
abcdef &
\end{array}
\begin{bmatrix}
 & B & D & & E & AF & CF & & F & BF & DF & & EF & A & C & \\
1 & -1 & -1 & 0 & -1 & 1 & 1 & 0 & -1 & 1 & 1 & 0 & 1 & -1 & -1 & 1 \\
1 & 1 & -1 & 0 & -1 & -1 & 1 & 0 & -1 & -1 & 1 & 0 & 1 & 1 & -1 & 1 \\
1 & -1 & 1 & 0 & -1 & 1 & -1 & 0 & -1 & 1 & -1 & 0 & 1 & -1 & 1 & 1 \\
1 & 1 & 1 & 0 & -1 & -1 & -1 & 0 & -1 & -1 & -1 & 0 & 1 & 1 & 1 & 1 \\
1 & -1 & -1 & 0 & 1 & -1 & -1 & 0 & -1 & 1 & 1 & 0 & -1 & 1 & 1 & 1 \\
1 & 1 & -1 & 0 & 1 & 1 & 1 & 0 & -1 & -1 & 1 & 0 & -1 & -1 & 1 & 1 \\
1 & -1 & 1 & 0 & 1 & -1 & -1 & 0 & -1 & 1 & -1 & 0 & -1 & 1 & -1 & 1 \\
1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & -1 & -1 & -1 & 0 & -1 & -1 & -1 & 1 \\
1 & -1 & -1 & 0 & -1 & 1 & 1 & 0 & 1 & -1 & -1 & 0 & -1 & 1 & 1 & 1 \\
1 & 1 & -1 & 0 & -1 & -1 & -1 & 0 & 1 & 1 & -1 & 0 & -1 & -1 & 1 & 1 \\
1 & -1 & 1 & 0 & -1 & 1 & 1 & 0 & 1 & -1 & 1 & 0 & -1 & 1 & -1 & 1 \\
1 & 1 & 1 & 0 & -1 & -1 & -1 & 0 & 1 & 1 & 1 & 0 & -1 & -1 & -1 & 1 \\
1 & -1 & -1 & 0 & 1 & -1 & -1 & 0 & 1 & -1 & -1 & 0 & 1 & -1 & -1 & 1 \\
1 & 1 & -1 & 0 & 1 & 1 & -1 & 0 & 1 & 1 & -1 & 0 & 1 & 1 & -1 & 1 \\
1 & -1 & 1 & 0 & 1 & -1 & 1 & 0 & 1 & -1 & 1 & 0 & 1 & -1 & 1 & 1 \\
1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1
\end{bmatrix}
\times
\begin{bmatrix}
155{,}6 \\ 38{,}3 \\ 0 \\ 0 \\ 66{,}7 \\ 0 \\ 0 \\ 0 \\ -25{,}4 \\ -19{,}4 \\ 0 \\ 0 \\ 0 \\ 0 \\ 36{,}2 \\ 0
\end{bmatrix}
=
\begin{bmatrix}
20{,}4 \\ 135{,}8 \\ 92{,}8 \\ 208{,}3 \\ 226{,}1 \\ 341{,}6 \\ 153{,}8 \\ 269{,}3 \\ 80{,}8 \\ 118{,}5 \\ 8{,}4 \\ 46{,}1 \\ 141{,}8 \\ 179{,}5 \\ 214{,}1 \\ 251{,}9
\end{bmatrix}
$$

Kod određivanja elemenata matrice treba uočiti da je polinom *(5.9)* nastao mjerenjem u dvije točke (što je svojstveno planu mjerenja 2^n) pri čemu donju granicu svake varijable treba uključiti u proračun koeficijenata s vrijednošću *-1*, gornju granicu s vrijednošću *+1*, a interakcija dviju varijabli uključuje se s vrijednošću koja je njihov kao algebarski umnožak (pa za mjernu shemu *acf* u stupcima *A*, *C* i *F* stoji *+1*, ali u stupcu za procjenu interakcije *BF* stoji vrijednost *-1* jer vrijedi *BF →(-1)·(+1)=-1*. Stupci za one višefaktorne interakcije koje se ne razmatraju (to jest njihov mali doprinos regresijskom polinomu ulazi u ukupnu grešku

mjerenja) imaju vrijednost 0. Isto tako su u stupac $\boldsymbol{K}$ s vrijednošću 0 uključeni svi nesignifikantni koeficijenti. Matematički model za cijeli proračun je izveden u Excelu i nalazi se u prilogu 5.2. Rezultati proračuna su prikazani u tablici 5-VII. Sve izračunate vrijednosti zaokružene su na **cijeli broj** sekundi.

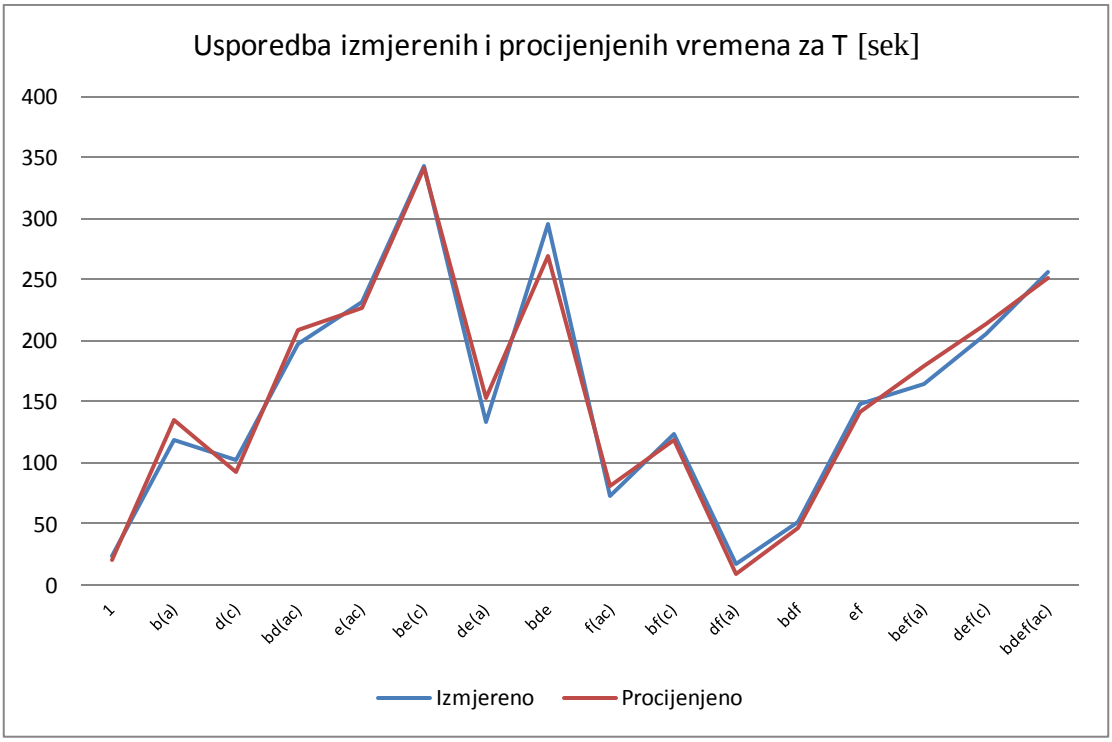
Tablica 5-VII: ***Usporedba mjerenih i izračunatih vrijednosti za T [sek]***

Izračunato $\boldsymbol{T_r}$	20	136	93	208	226	342	154	269	81	119	8	46	142	176	214	252
Mjereno $\boldsymbol{T}$	24	119	103	198	232	343	134	295	73	124	18	52	149	164	205	256
Mjerna točka	*1*	*ab*	*cd*	*abcd*	*ace*	*bce*	*ade*	*bde*	*acf*	*bcf*	*adf*	*bdf*	*ef*	*abef*	*cdef*	*abcdef*

Usporedbom izmjerenih vrijednosti T iz stupca 4 tablice 5-VI s vrijednostima T_r izračunatima po izrazu *(5.9)* vidi se vrlo visoka podudarnost tih dviju veličina u svim mjernim točkama. Ta je podudarnost kvalitativno evidentna u tablici 5-VII, a posebno na slici 5.4[9]. Iz toga slijedi da polinom *(5.9)* matematički dobro opisuje istraživanu pojavu, pa se svi zaključci o utjecaju istraživanih parametara $\boldsymbol{r}$, $\boldsymbol{p}$, $\boldsymbol{a}$, $\boldsymbol{u}$, $\boldsymbol{b}$ i $\boldsymbol{q}$ na vrijeme izvođenja ogledne aplikacije u oblaku T smiju donositi na temelju analize matematičkog modela za T_r.

Premda je podudarnost izmjerenih i izračunatih vrijednost u svim mjernim točkama vizualno jako dobra, potrebna je ipak egzaktna provjera adekvatnosti matematičkog modela za T_r. Provjera adekvatnosti modela i ovdje se temelji na analizi varijance, uz korištenje F-testa za testiranje nulte hipoteze koja glasi H_0: *odstupanja mjerenih i izračunatih vrijednosti u svim mjernim točkama nisu dovoljno velika za zaključak da matematički model nije adekvatan*. Za potvrđivanje ili odbacivanje H_0 potrebno je usporediti varijancu koja je posljedica nepodudarnosti modela i stvarne pojave te je usporediti s varijancom koju uzrokuju pogreške u mjerenju.

[9] Obje krivulje na grafu treba promatrati kao povezane rezultate mjerenja i izračuna poredane na apscisi po Yatesovom redu. To ne implicira neku promjenu rezultata u vremenu, već omogućava lakše uočavanje podudarnosti izmjerenih i procijenjenih vrijednosti, to jest kvalitete matematičke aproksimacije rezultata mjerenja jedne realne pojave.



Slika 5.4: ***Usporedba mjerenih i procijenjenih vrijednosti za T u točkama mjerenja***

Da bi se mogle utvrditi pogreške mjerenja potrebno je "kalibrirati mjerni mehanizam" tako da se više puta ponovi mjerenje u nekoj mjernoj točki. U teoriji se za to preporuča ponavljanje mjerenja u "nultoj točki." Pod tim pojmom podrazumijevaju se uvjeti kada su sve nezavisne varijable postavljene na srednju vrijednost između donje i gornje granice. Ta se mjerenja mogu nazvati kontrolnim, a njihovi rezultati će biti označeni s T_0. Iz ponovljenih mjerenja u nultoj točki računa se srednja vrijednost Φ_0 te se kao mjera za grešku mjerenja uzima kvadrat odstupanja pojedinačnih mjerenja od te srednje vrijednosti.

Rezultati kontrolnih mjerenja, izvedenih kad i mjerenja tijekom prethodnog istraživanja, prikazani su u zadnja tri retka tablice 5-III. Cijeli postupak analize varijance (ANOVA) temelji se na teoretskim obrazloženjima u [10] i [88]. U tablici 5-VIII su prikazani glavni koraci tog postupka. Za testiranje H_0 potrebno je odrediti Fisherov[10] koeficijent adekvatnosti f_{mF} za postavljeni matematički model koji u ovom slučaju ima vrijednost $f_{mF} = 15{,}23$ (donje desno polje tablice 5-VIII). Radi provjere H_0 ovaj se koeficijent uspoređuje s graničnom vrijednošću po F-distribuciji koja za $\alpha=0.05$ i stupnjeve slobode $df_e = 2$ i $df_m = 7$ iznosi $f_F = 19{,}353$. Ako je $f_{mF} < f_F$ onda se hipoteza H_0 potvrđuje, to jest u ovom se slučaju matematički model *(5.9)* smatra adekvatnim[11].

[10] Engleski statističar i genetičar Sir Ronald Fisher objavio je ovaj postupak (koji je kasnije uveden u sve radove o planiranju pokusa) u knjizi "*Statistical Methods for Research Workers*", objavljenoj još 1925. godine, a ovdje je preuzet iz [10].

[11] Ako bi model prvog reda bio neadekvatan, trebalo bi postaviti matematički model drugog reda. U teoriji se za to preporuča ortogonalni parcijalni višefaktorski plan pokusa tipa 2^n koji koristi sve podatke iz plana prvog reda i

Tablica 5-VIII: ***ANOVA matematičkog modela za T_r***

Izvor rasipanja	***Stupnjevi slobode***	***Zbroj kvadrata odstupanja***	***Varijanca***	***Koeficijent***
Koeficijenti regresijskog polinoma	$df_k=6$ (Šest koeficijenata polinoma, jedan stupanj slobode svaki)			
Nesignifi-kantni koeficijenti	$df_r=N-k-1=16-6-1=9$ (Iz ukupno $N=16$ mjerenja izračunato je $k=6$ koeficijenata)	$S_r=\sum_1^{16}(T-T_m)^2=2.208{,}13$		
Slučajne greške kod mjerenja	$df_e=n_0-1=3-1=2$ (Broj ponavljanja mjerenja u nultoj točki iznosi $n_0=3$)	$S_e=\sum_1^{3}(T_0-\emptyset_0)^2=40{,}67$	$s_e^2=\frac{S_e}{df_e}=20{,}33$	
Matematički model	$df_m=f_r-f_e=9-2=7$	$S_m=S_r-S_e=2.167{,}46$	$s_m^2=\frac{S_m}{df_m}=309{,}64$	$f_{mF}=\frac{s_m^2}{s_e^2}=15{,}23$

Budući da je u prethodnom postupku dokazano da regresijski polinom *(5.9)* adekvatno opisuje vrijeme trajanja izvođenja ogledne aplikacije u oblaku, zavisno o istraživanim parametrima, moguće je o tome donositi općenitije zaključke. Radi boljeg razumijevanja i tumačenja izvornog problema potrebno je s varijabli x_i ($i=1$ do 6), korištenih u matematičkoj obradi rezultata mjerenja i definiranih u domeni $[-1,+1]$, prijeći na parametre definirane svojim originalnim intervalima $[d,\ g]$. Zbog toga je potrebno u polinom *(5.9)*, umjesto nezavisnih varijabli x_2, x_3, x_5 i x_6 uvrstiti izraze za parametre $\boldsymbol{a}$, $\boldsymbol{u}$, $\boldsymbol{q}$ i $\boldsymbol{r}$ koji se izračunavaju rješavanjem supstitucijskih jednadžbi u tablici 5-IV:

$$a=4{,}5\cdot x_2+5{,}5\ ;\qquad u=267{,}3\cdot x_3+272{,}7\ ;\qquad q=95\cdot x_5+110\ ;\qquad r=2\cdot x_6+3$$

Uvrštavanjem ovih vrijednosti u izraz *(5.9)* konačno se dobije jednadžba koja opisuje trajanje izvedbe aplikacije u oblaku, u zavisnosti od vrijednosti realnih parametara:

$$T=0{,}631+14{,}993\cdot a+0{,}135\cdot u+0{,}702\cdot q-0{,}840\cdot r-2{,}160\cdot a\cdot r \qquad (5.10)$$

Treba naglasiti da gornji izraz vrijedi unutar istraživane domene definirane gornjom i donjom granicom nezavisnih varijabli s vrijednostima (prema tablici 5-IV) $a\in[1;10]$, $u\in[5{,}4;540]$, $q\in[10;200]$, $r\in[1{,}5]$, pri čemu $\boldsymbol{r}$ može biti isključivo cijeli broj.

Radi empirijske provjere adekvatnosti gornje jednadžbe u realnom prostoru mogu se izračunati vremena izvedbe aplikacije u oblaku za tri hardversko/softverske konfiguracije:

Svi parametri su na donjoj razini: $a=1;\ u=5{,}4;\ q=10;\ r=1 \rightarrow T=T_d=20{,}5$ *sek*

Svi parametri su u nultoj točki: $a=5{,}5;\ u=272{,}7;\ q=105;\ r=3 \rightarrow T=T_0=156{,}7$ *sek*

Svi parametri su na gornjoj razini: $a=10;\ u=540;\ q=200;\ r=5 \rightarrow T=T_g=254{,}2$ *sek*

traži $2n$ dopunskih mjerenja. U ovom istraživanju za to nije bilo potrebe jer se matematički model prvog reda pokazao adekvatnim.

5.10 Zaključci i preporuke o zaposjedanju resursa u oblaku

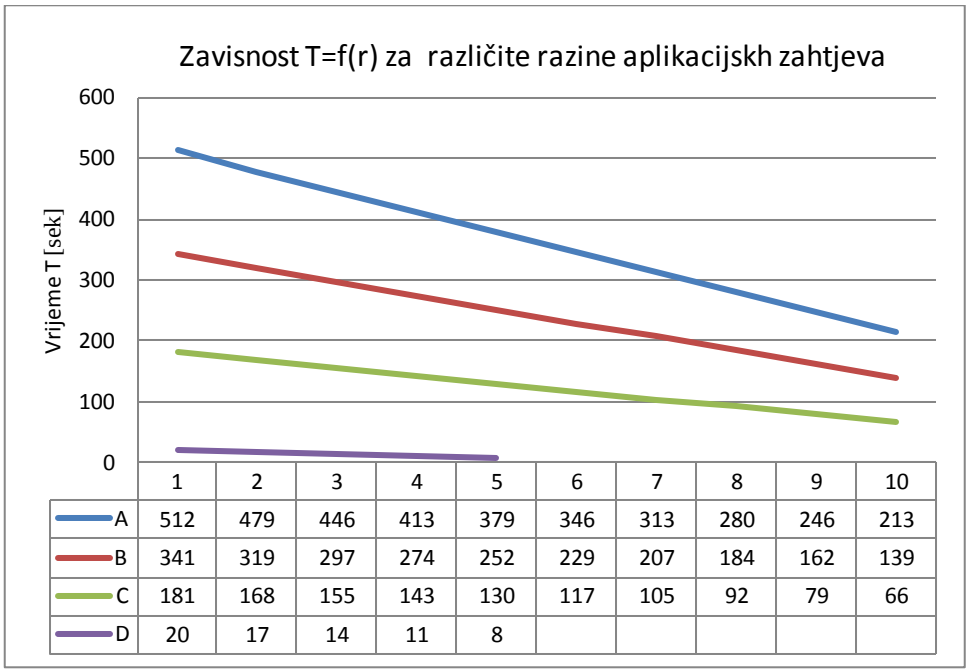
Od ukupno 6 istraživanih parametara za koje se u početku pretpostavljalo da utječu na T, dokazano je da samo četiri imaju značajan utjecaj: ***Složenost algoritma (a)***, ***Opseg poslanih podataka (u)***, ***Broj akcija po upitu (q)*** i ***Broj angažiranih računala (r)***. Pokazalo se također da preostala dva parametra ***Snaga angažiranih računala (p)*** i ***Veličina baze podataka (b)*** nemaju značajnog utjecaja. Ovdje će se najprije kratko prokomentirati nesignifikantni parametri a zatim detaljnije oni koji značajno utječu na T.

Konstatacija da se ne može sa sigurnošću utvrditi značajan utjecaj parametra ***Snaga angažiranih računala (p)*** na vrijeme odziva T je možda neočekivana, ali je logična ako se uoči da je raspon varijacije tehničkih performansi računala, ponuđenih kao donja i gornja granična konfiguracija, relativno mali u odnosu na mogući raspon varijacije drugih parametara koji opisuju aplikaciju (vidjeti opis parametra ***Snaga angažiranih računala*** u točki 5.4). Vjerojatno da sličnu spoznaju imaju i davatelji usluga oblačnog računarstva pa zbog toga u njihovoj standardnoj ponudi niti nema računala izrazito snažnih performansi – takva računala korisnici uglavnom kupuju i koriste za visoko specijalizirane zadaće koje se izvode na vlastitoj informatičkoj infrastrukturi.

Nesignifikantna utjecajnost parametra ***Veličina baze podataka (b)*** na vrijeme izvođenja aplikacije T može se također logično protumačiti ako se zna da nije važno da li se isti skup podataka traži (radi obrade) u maloj ili velikoj bazi podataka. Suvremeni sustavi za organizaciju, adresiranje i pretraživanje baza podataka (relacijskih, ali i drugih) imaju optimizirane algoritme i izvrsne tehničke performanse za pristup podacima pa vrijeme pronalaženja i upisivanja podataka ne zavisi o veličini normalnih baza podataka sve dok su one opsega do 1000 GB. Naravno, sa stanovišta vremena obrade nije svejedno da li se na obradu šalje malo ili mnogo podataka, ali taj čimbenik je obuhvaćen parametrom ***Opseg poslanih podataka (u)*** koji se pokazao signifikantnim. Također, nije nevažno koliki se broj CRUD akcija mora izvesti nad jednim skupom podataka, ali to je obuhvaćeno parametrima ***Broj akcija po upitu (q)*** i ***Složenost algoritma (a)*** koji su se također u ovom istraživanju pokazali signifikantnima.

Analizom strukture polinoma *(5.10)* se vidi da zavisna varijabla T raste ako se povećavaju vrijednosti parametara ***Složenost algoritma (a)***, ***Opseg poslanih podataka (u)*** i ***Broj akcija po upitu (q)*** od donje prema gornjoj granici. Nasuprot tome, ako se povećava parametar ***Broj angažiranih računala (r)*** u oblaku, tada će se vrijeme izvođenja ogledne aplikacije T očekivano smanjivati. Graf na slici 5.5 ukazuje na taj opći odnos između vremena T i broja angažiranih računala r. On je izračunat iz polinoma *(5.10)* za četiri različita slučaja: kada su a, u i q na donjoj granici (krivulja D), na srednjim vrijednostima (C), na gornjoj granici (B) i

ekstrapolirano za slučaj da su parametri $\boldsymbol{a}$, $\boldsymbol{u}$ i $\boldsymbol{q}$ povećani za 50% u odnosu na gornje granične vrijednosti (krivulja A). Krivulje nisu paralelne, što znači da će se ostvariti veće uštede u vremenu T ako se isti broj računala angažira kod većih aplikacijskih opterećenja (na primjer, 5 računala smanjuje T za 51 sek kod srednjeg te 133 sek kod vrlo velikog opterećenja).



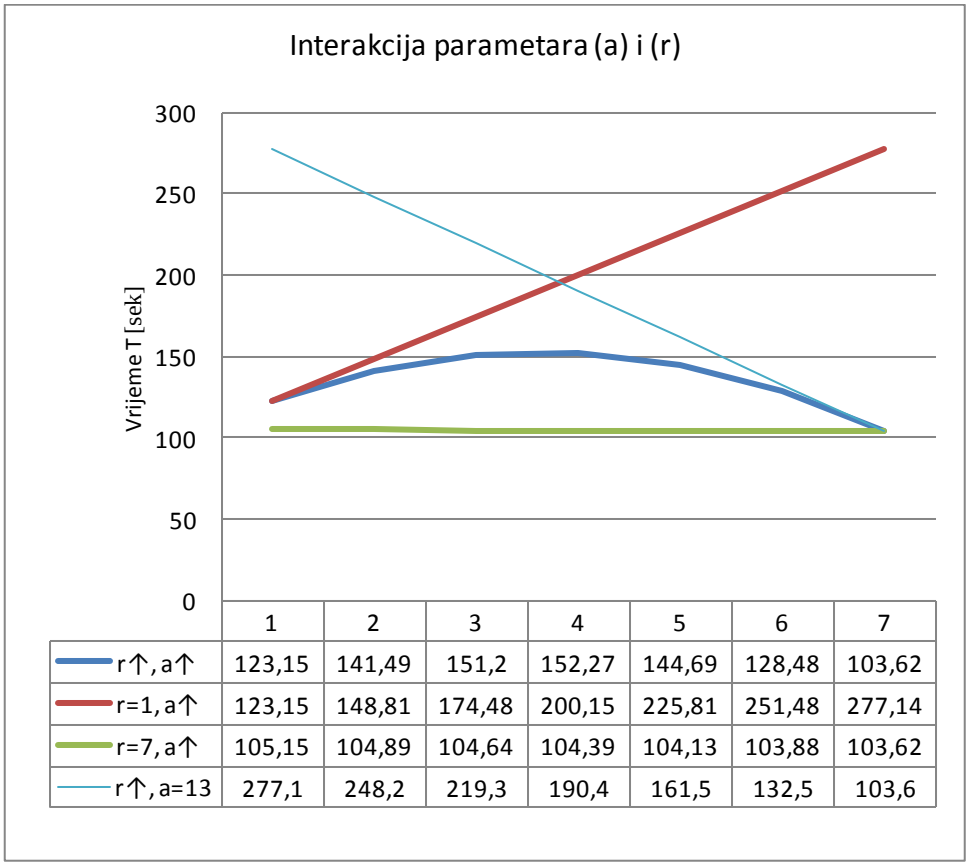
	1	2	3	4	5	6	7	8	9	10
A	512	479	446	413	379	346	313	280	246	213
B	341	319	297	274	252	229	207	184	162	139
C	181	168	155	143	130	117	105	92	79	66
D	20	17	14	11	8					

Slika 5.5: ***Zavisnost T o broju računala, za različite razine aplikacijskih zahtjeva***

Sinergistički učinak zbog interakcije varijabli $\boldsymbol{a}$ i $\boldsymbol{r}$ dosta je izražen. Iz jednadžbe *(5.10)* se vidi da će porast vremena T biti veći ako se povećava složenost algoritma $\boldsymbol{a}$ uz malo angažiranih računala, nego ako se složenost algoritma jednako povećava uz veliki broj angažiranih računala. Takva zakonitost je vidljiva na slici 5.6 na kojoj je prikazano kako se mijenja T u zavisnosti o odnosima parametara $\boldsymbol{a}$ i $\boldsymbol{r}$ ako su parametri $\boldsymbol{u}$ i $\boldsymbol{q}$ na nekoj srednjoj vrijednosti. Na apscisi grafa na slici 5.6 je definirano 7 točaka koje opisuju promjene parametra $\boldsymbol{r}$ kroz tih 7 točaka ($r=1, 2,...7$ što je označeno kao $\boldsymbol{r}\uparrow$) uz konstantnu vrijednost $\boldsymbol{a}$ ili promjenu parametra $\boldsymbol{a}$ u 7 točaka (relativna složenost $1, 3, 5, 7, 9, 11, 13$ što je označeno kao $\boldsymbol{a}\uparrow$) uz konstantnu vrijednost $\boldsymbol{r}$.

Vidi se da povećanje složenosti algoritma $\boldsymbol{a}$ uopće ne mijenja vrijeme T kod velikog broja računala (zelena krivulja), dok se za mali broj računala vrijeme T snažno povećava ako raste složenost algoritma $\boldsymbol{a}$ - crvena krivulja. Kod velike složenosti algoritma $\boldsymbol{a}$ vrijeme izvođenja T brzo se smanjuje povećanjem broj računala $\boldsymbol{r}$ angažiranih u oblaku (tanka plava krivulja). Vrlo je interesantna krivulja označena debelom plavom crtom čiju apscisu čini 7 parova uređenih tako da točka 1 znači najmanji broj računala uz najmanju složenost, a točka 7 najveći broj računala uz najveću složenost. Ona upućuje na to da se dobro vrijeme T postiže

ako se za malu složenost koristi malo računala, a za veliku mnogo računala (što je samo po sebi razumljivo), ali da je lošije ako se za neko srednje opterećenje koristi srednji broj računala.



	1	2	3	4	5	6	7
r↑, a↑	123,15	141,49	151,2	152,27	144,69	128,48	103,62
r=1, a↑	123,15	148,81	174,48	200,15	225,81	251,48	277,14
r=7, a↑	105,15	104,89	104,64	104,39	104,13	103,88	103,62
r↑, a=13	277,1	248,2	219,3	190,4	161,5	132,5	103,6

Slika 5.6: ***Zavisnost vremena T o interakcijama parametara a i q***

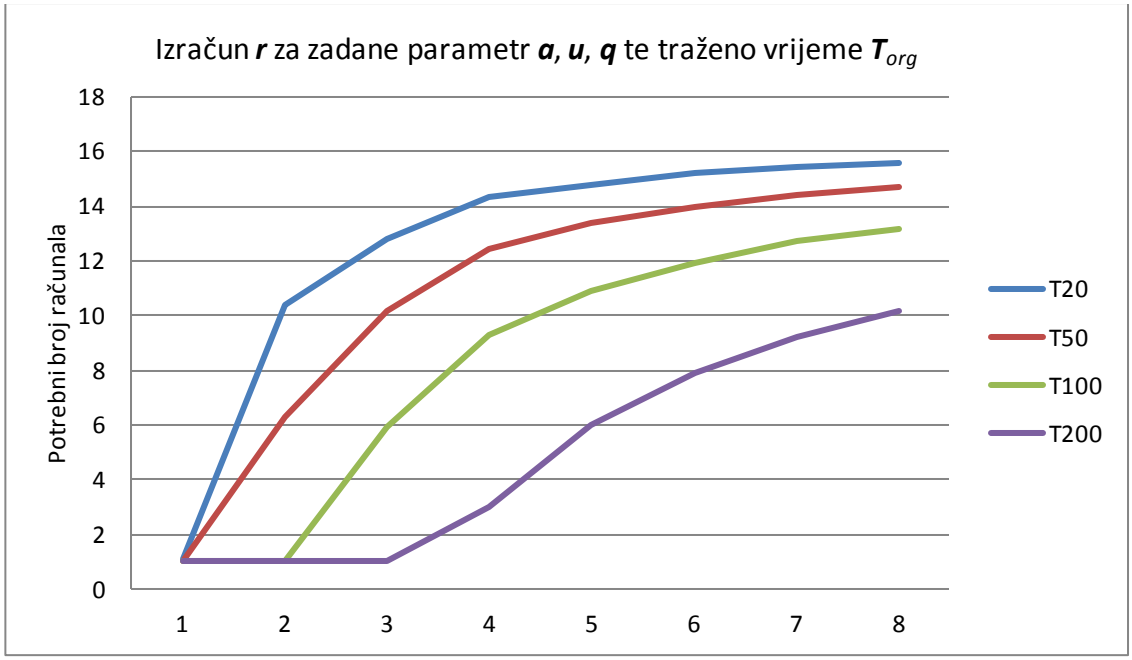
Na temelju cijele ove teoretske studije izrađen je sustav za automatizirano preporučivanje koliko računala $\boldsymbol{r}$ treba angažirati (u oblaku, ali to bi vrijedilo i za vlastite resurse) za izvođenje aplikacije koja ima određene karakteristike definirane parametrima $\boldsymbol{a}$, $\boldsymbol{u}$ i $\boldsymbol{q}$, a uz organizacijski prihvatljivo vrijeme njezinog izvođenja $T = T_{org}$. Da bi se to moglo izvesti treba polinom *(5.10)* preurediti tako da se $\boldsymbol{r}$ postavi kao zavisna varijabla.

$$r=\frac{14{,}993\cdot a+0{,}135\cdot u+0{,}702\cdot q-(T-0{,}631)}{2{,}160\cdot a+0{,}840} \qquad (5.11)$$

S ovim polinomom izrađen je program u Excelu koji se također nalazi u prilogu 5.2. Opći odnosi između nezavisnih varijabli i T koji proizlaze iz *(5.11)* vide se na slici 5.7 koja pokazuje broj računala koji bi bio potreban za izvođenje ogledne aplikacije s promjenljivim vrijednosti parametara $\boldsymbol{a}$, $\boldsymbol{u}$ i $\boldsymbol{q}$ za četiri različita organizacijska vremena T_{org}. Na apscisi tog grafa je definirano 8 točaka koje predstavljaju različite zahtjeve koje ima aplikacija, odnosno točke na apscisi predstavljaju uređene trojke parametara $\boldsymbol{a}$, $\boldsymbol{u}$ i $\boldsymbol{q}$ sa sljedećim vrijednostima:

a:	1	3	5	7	9	11	13	15
u:	5,4	113,4	221,4	329,4	437,4	545,4	653,4	761,4
q:	10	50	90	130	170	210	250	290
Točka na apscisi:	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**

Iz grafa se vidi da će za aplikacije s niskim zahtjevima (dakle s malim vrijednostima $\boldsymbol{a}$, $\boldsymbol{u}$ i $\boldsymbol{q}$) te s malim zahtjevima za vremenom odziva (T=200 [sek]) biti dovoljno samo jedno računalo. Potrebni broj računala raste s povećanjem aplikacijskih zahtjeva i smanjivanjem organizacijskog vremena T_{org} u kojem se aplikacija mora izvesti.

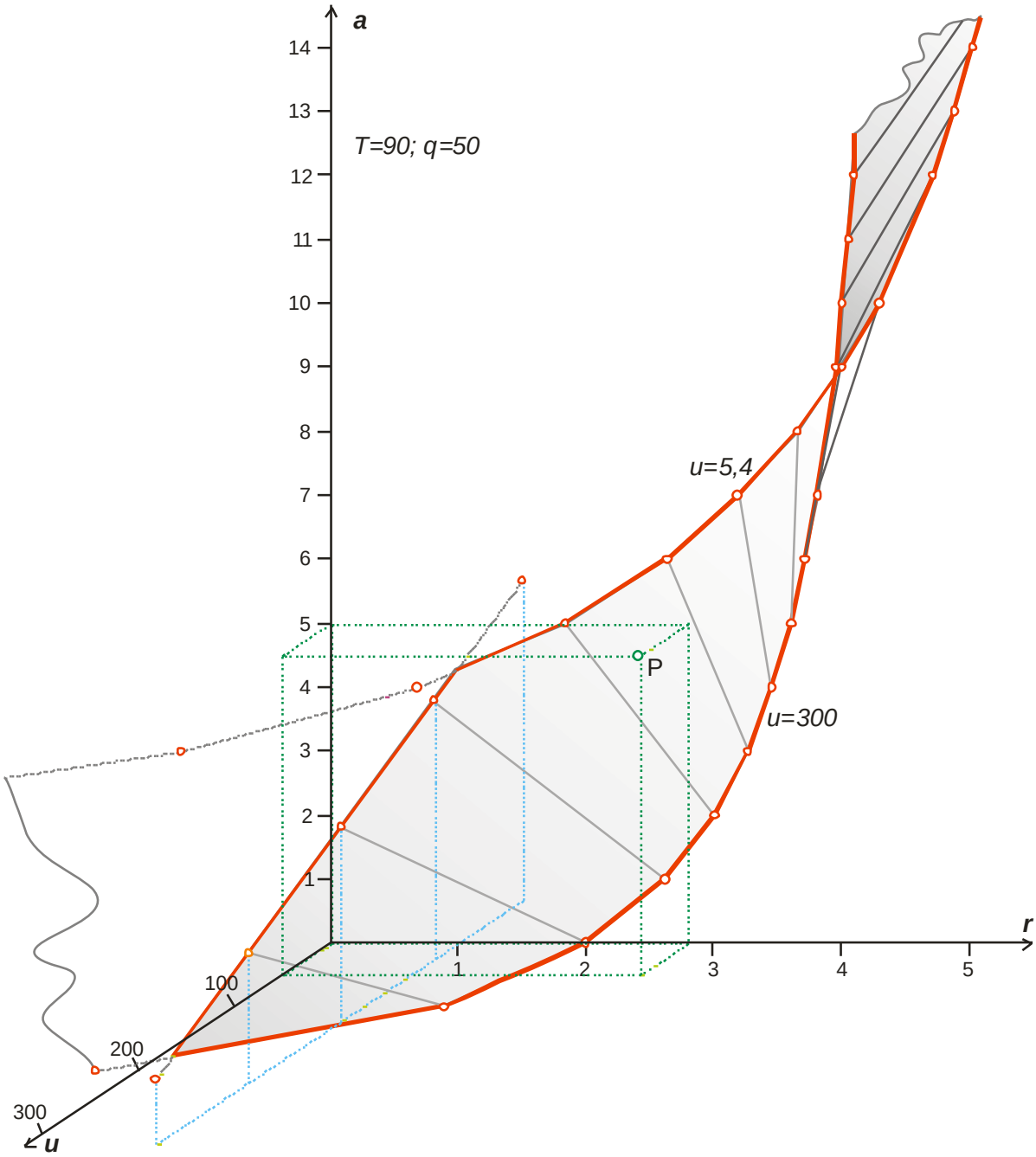


Slika 5.7: ***Potrebni broj računala za razne parametre ogledne aplikacije***

Korištenjem programa iz priloga 5.2 moguće je izvesti složene "što-ako" (*what-if*) analize (simulacije) i na takav način, za svaku aplikaciju opisanu parametrima $\boldsymbol{a}$, $\boldsymbol{u}$ i $\boldsymbol{q}$, utvrditi potreban broj računala za organizacijski prihvatljivo vrijeme njezinog izvođenja. Praktični slijed koraka kod provedbe takve simulacijske analize je sljedeći:

1. Procijeniti karakteristike aplikacije za koju se radi analiza te ih formalno definirati postavljanjem parametara $\boldsymbol{a}$, $\boldsymbol{u}$ i $\boldsymbol{q}$;
2. U skladu s upravo provedenim razmatranjima, a uz primjenu programskog rješenja iz priloga 5.2, izračunati potreban broj računala za nekoliko razina organizacijski prihvatljivog vremena;
3. Za svaki broj računala izračunati isplati li ih se najmiti u oblaku ili kupiti (što se rješava u sljedećem poglavlju).

S obzirom da poslovnim stručnjacima i informatičarima (koji bi željeli praktično iskoristiti ove rezultate radi specifikacije zahtjeva za uslugom oblačnog računarstva) može biti teško predočiti si odnose između ovdje istraženih varijabli u peterodimenzionalnom prostoru (kojeg čine T, $\boldsymbol{a}$, $\boldsymbol{u}$, $\boldsymbol{q}$ i $\boldsymbol{r}$) pa su ovi odnosi prikazani u trodimenzionalnom prostoru na slici 5.8.



Slika 5.8: ***Međuzavisnost varijabli a, u i r za T=90 [sek] i q=50 [akcija/upit]***

Prikaz na slici 5.8 je ostvaren tako da su fiksirane dvije varijable na najčešće očekivane vrijednosti u istraživanoj domeni određenoj oglednom aplikacijom. U prikazu za sliku 5.8 uzeto je: $T=90$ *[sek]* i $q=50$ *[akcija po upitu]*.

5.11 Osvrt na prvi dio hipoteze 2

Analiza površine na slici 5.8 pokazuje da broj računala $\boldsymbol{r}$ (koja treba unajmiti u računalnom oblaku nekog pružatelja usluga oblačnog računarstva) raste nelinearno s povećanjem aplikacijskih zahtjeva $\boldsymbol{a}$, ali je taj porast daleko veći i strmiji ako je to kombinirano s povećanjem opsega podataka $\boldsymbol{u}$ koji se šalje u oblak radi obrade (sve uz konstantno traženo vrijeme T i konstantni broj CRUD akcija $\boldsymbol{q}$).

Iako su svi prethodni zaključci izvedeni na temelju istraživanja ponašanja ogledne aplikacije, cijeli postupak ima šire metodološko značenje. To je zato što izabrana ogledna aplikacija dobro reprezentira široki skup aplikacija tipičnih za poslovne informacijske sustave. Drugim riječima, ogledna aplikacija može se smatrati njihovim modelom. Najvažnija se svojstva tipične poslovne aplikacije mogu opisati aplikacijskim parametrima $\boldsymbol{a}$ (složenost algoritma) i $\boldsymbol{q}$ (broj CRUD akcija po upitu), podatkovnim parametrom $\boldsymbol{u}$ (opseg podataka koji se šalje u oblak radi obrade) i traženim organizacijskim vremenom za njezino izvođenje T_{org}. Ako je aplikacija opisana na takav način onda se do kraja objektivizirano može odrediti potreban broj računala koja treba iznajmiti u oblaku što predstavlja (kako će biti pokazano u sljedećem poglavlju) za korisnika daleko najveći trošak korištenja usluga oblačnog računarstva.

Općenito gledano, grafički prikaz poput onoga sa slike 5.8 može se iskoristiti za brzo određivanje potrebnog broja računala koje treba unajmiti u oblaku: ako se, uz navedene vrijednosti za T i $\boldsymbol{q}$ na prikazanu površinu postavi točka $\boldsymbol{P}$ koja je određena vrijednostima parametara $\boldsymbol{a}$ (na aplikati) i $\boldsymbol{u}$ (na ordinati), onda se projekcijom točke s te površine na apscisu može očitati približno potreban broj računala $\boldsymbol{r}$. Tako se, na primjer, za točku $\boldsymbol{P}$, definiranu s $a=5$ i $u=50$, može očitati da je $r \approx 2{,}4$ to jest potrebno je unajmiti $r = 3$ računala (zaokruženo na prvi veći cijeli broj). Naravno, takav se nomogram može konstruirati i za bilo koje druge vrijednosti T i $\boldsymbol{q}$ iz istražene domene, a ne samo za one koje su navedene na nomogramu sa slike 5.8.

U ovom poglavlju je pokazano da se svi parametri kojima se opisuje bilo koja poslovna aplikacija mogu mijenjati u jednoj – oglednoj – aplikaciji. Takva je mogućnost osigurala nepromjenljive uvjete tijekom mjerenja, to jest pouzdane rezultate mjerenja iz kojih su isključeni nepoželjni utjecaji nekontroliranih varijabli što je pak temeljna pretpostavka za vjerodostojno zaključivanje o ponašanju istraživane pojave.

U slučaju da se neka druga aplikacija značajno razlikuje od ogledne ne bi se neposredno mogli primijeniti zaključci izvedeni na oglednoj aplikaciji, ali bi **metodološki postupak ostao isti**. Za takvu aplikaciju bilo bi potrebno izvesti mjerenja po shemi 2^{n-k} (obrazloženoj u prethodnim točkama ovog poglavlja) i ponoviti izračun radi određivanja polinoma oblika *(5.6)* koji će matematički opisivati istraživanu aplikaciju na sličan način kao što polinom *(5.10)*

opisuje ovdje analiziranu oglednu aplikaciju. Za provedbu tog postupka može se koristiti isti programski alat koji se nalazi u prilogu 5.2 ili bi taj program trebalo malo dopuniti za slučaj da se linearni model oblika *(5.6)* ne pokaže adekvatnim. U tom slučaju treba izvesti dopunska mjerenja i nešto složenije proračune, ali su smjernice za takve nadopune već navedene u odgovarajućim dijelovima ovog teksta.

Iz ovih razmatranja zaključujemo da je dokazan prvi dio hipoteze 1 koji glasi: *"Moguće je postaviti višedimenzionalnu ponderiranu metriku potreba za računalnim resursima SOA servisa prije njihove izgradnje..."*. Dakle, u ovom je poglavlju najprije ostvaren teoretski doprinos metodologiji organiziranja oblačnog računarstva, a zatim je pokazano kako su rezultati tog teoretskog razvoja mogu koristiti u praksi.

5.12 Sažetak

Svrha mjerenja bila je utvrditi oblik zavisnosti vremena izvođenja aplikacije T i nezavisnih varijabli koje opisuju njezino izvođenje na računalnim resursima u oblaku. Temeljem procjene eksperata utvrđeno je da valja istražiti tri grupe nezavisnih varijabli: aplikacijske, podatkovne i resursne. Utvrđeno je da aplikacijske varijable čine složenost algoritma $\boldsymbol{a}$ i broj CRUD akcija po upitu $\boldsymbol{q}$. Podatkovni aspekt opisan je opsegom podataka koji se šalje u oblak radi obrade $\boldsymbol{u}$ i veličinom baze podataka nad kojom se izvodi obrada $\boldsymbol{b}$. Resursne varijable su broj angažiranih računala u oblaku $\boldsymbol{r}$ i njihova snaga $\boldsymbol{p}$.

Kao objekt mjerenja izabrana je ogledna aplikacija koja dobro predstavlja veliki skup aplikacija tipičnih za poslovne informacijske sustave. Ogledna se aplikacija odnosi na prepoznavanje zadanih oblika u odnosu prema referentnim oblicima iz baze podataka. Ona se pokazala prikladnom za variranje aplikacijskih i podatkovnih varijabli u dovoljno širokom rasponu za istraživanje te je pripremljena za izvođenje na računalnim resursima u oblaku na Microsoftovoj platformi Azure. Tijekom probnog istraživanja provedeno je ukupno 11 mjerenja vremena T, pri čemu su svi parametri varirali između njihove donje i gornje očekivane vrijednosti. Pokazalo se da je srednja vrijednost za T, postignuta kad su svi parametri postavljeni na srednju vrijednost iznosi 156 sekundi te da varira od 11 do 367 sekundi za razne vrijednosti parametara. Na temelju saznanja iz predistraživanja postavljen je višefaktorski parcijalni plan tipa 2^{6-2} prema kojem je izvedeno 16 mjerenja. Rezultati mjerenja obrađeni su metodama matematičke statistike te je izračunat matematički model kojim se opisuje ponašanje ogledne aplikacije u oblaku, u granicama istraženih varijacija parametara, koji glasi: $T = 0{,}631 + 14{,}993 \cdot a + 0{,}135 \cdot u + 0{,}702 \cdot q - 0{,}840 \cdot r - 2{,}160 \cdot a \cdot r$

Korištenjem metode ANOVA potvrđena je adekvatnost ovog linearnog modela u kojem su se našla četiri signifikantna parametra ($\boldsymbol{a}$, $\boldsymbol{u}$, $\boldsymbol{q}$ i $\boldsymbol{r}$) te jedna njihova interakcija. Ovaj matematički

model ponašanja ogledne aplikacije u oblaku je interpretiran grafički i analitički s različitih aspekata. Na temelju toga su date smjernice i izrađen nomogram za praktično određivanje koliko računalnih resursa treba unajmiti u oblaku za izvođenje aplikacije, čije se značajke mogu definirati vrijednostima signifikantnih parametara, a uz organizacijski prihvatljivo vrijeme njezinog izvođenja.

Na kraju poglavlja je pokazano kako je provedenim istraživanjem i ostvarenim rezultatima dokazana druga hipoteza, postavljena na početku ovog rada, a odnosi se na postavljanje višedimenzionalne metrike potreba za računalnim servisima iz računalnog oblaka.

6 Ekonomika oblačnog računarstva

U ovome poglavlju bit će detaljno opisani ekonomski aspekti korištenja komercijalnih računalnih oblaka, uvodno spomenutih u poglavlju 4, s ciljem da se njihovom usporedbom s klasičnim računarstvom te korištenjem metrike za procjenu računalnih resursa (opisane u petom poglavlju) razvije metodika za razrješenje dvojbe da li računalne resurse kupiti ili najmiti u oblaku. Na taj način dokazat će se ekonomski aspekt druge hipoteze ovog rada, dok je razvojem metrike za procjenu računalnih resursa u petom poglavlju već dokazan tehnički aspekt te hipoteze.

U potpoglavlju 6.1, s tematikom pune cijene vlasništva (*total cost of ownership* – TCO) nad oblačnim centrom, prikazan je pogled na računalne oblake s investitorskoga gledišta, zanimljivog svim tvrtkama, ustanovama i pojedincima koji žele graditi vlastite računalne oblake ili ulagati u njihovu izgradnju. Pokazani su opći čimbenici koji određuju troškove rada i uvjete isplativosti računalnog oblaka, to jest koja je, primjerice, minimalna cijena CPU sata nužna da bi komercijalni oblačni centar, opisan u poglavlju 3, mogao biti profitabilan. Takvo je saznanje korisno i korisnicima računalnih resursa stoga što omogućuje dobru procjenu koliko oblačni poslužitelj zarađuje pružanjem usluga oblačnog računarstva, kakve su međusobne razlike ponuda poslužitelja i nije li isplativije graditi vlastiti podatkovni centar.

Izračunom pune cijene vlasništva oblačnog centra omogućit će se izračun cijene temeljnih računalnih resursa u oblaku – procesorskog vremena i podatkovne pohrane. Da bi se te cijene oblačnog najma mogle usporediti s cijenom računalnih resursa u vlasništvu, ovim poglavljem će se utvrditi i smjernice za općeniti izračun cijene računalnih resursa, kako u komercijalnom oblaku, tako i na vlastitim računalima koja se koriste u sklopu klasičnih serverskih farmi. Potom će se postaviti taksonomija oblačnih usluga komercijalnih oblačnih poslužitelja koji su već promatrani u poglavlju 4. Temeljem te taksonomije i saznanja iz poglavlja 5 o bitnim čimbenicima za procjenu računalnih resursa ovisno o aplikacijskim karakteristikama, konačno će se izložiti metodika za rješavanje dvojbe "kupnja ili najam".

Završni dio te metodike čini usporedba izračunatih troškova računalnih resursa sa stvarnom komercijalnom ponudom oblačnih poslužitelja. Komercijalne ponude ukratko opisane u četvrtom poglavlju predstavljaju poslužiteljev pogled na troškove oblačnog računarstva. To znači da njega zanima isključivo koliko pojedini korisnik troši oblačnih resursa i kakve vrste – podatkovnu pohranu, CPU-ova ili računala, mrežni promet, i tako dalje. Potencijalnog korisnika zanima, dakako, to isto, no njemu je daleko važnije odgovoriti na pitanje: **"Koliko računalnih resursa će uopće trošiti moja aplikacija?"** Tek odgovorom na to pitanje je moguće odabrati najpovoljniju ponudu usluga oblačnog računarstva, ako oblak uopće bude izabran kao platforma za pokretanje informacijskog sustava.

6.1 Puna cijena vlasništva oblačnog centra

Puna cijena vlasništva – TCO – bitna je svakom potencijalnom ulagaču u oblačne centre stoga što izravno utječe na cijene usluga koje komercijalni oblačni poslužitelji mogu ponuditi. Drugim riječima, naplata tih usluga mora pokriti barem TCO oblačnog centra. TCO ovisi o određenim parametrima [115] koji se mogu postaviti jako široko. Na primjer, može se uzimati u obzir tlak serverskih stalaka na podnicu prostorije jer to utječe na veličinu zgrade u kojoj je smješten oblak, što pak utječe na investiciju u nekretninu! Ili, godišnja količina padalina u nekom kraju gdje je smješten oblačni centar utječe na razinu akumulacijskih jezera hidroelektrana te tako i na cijenu struje. S druge strane, varijable tako velike složenosti mogu se pojednostavniti rezultatima izračuna prema odgovarajućim statističkim metodama, prosječnim vrijednostima ili saznanjima iz drugih struka, što investitori u podatkovne centre obično i rade. U ovom poglavlju razvijen je primjenljivi model izračuna ukupne cijene vlasništva te predstavlja razumni kompromis između nepotrebnog detaljiziranja i neprihvatljivog uopćavanja. Stoga se u modelu prikazuju i opisuju oni parametri koji neposredno utječu na punu cijenu vlasništva računalnog oblaka, bez obzira što neki od tih parametara mogu predstavljati sažetke detaljnijih istraživanja u drugim znanstvenim i stručnim područjima (kao što se cijena izgradnje adekvatnog poslovnog prostora na kraju može obračunati po m^2, a da se u računu investicija za podatkovni centar ne ulazi u detalje izračuna te cijene).

Prije pojedinačnog razmatranja pojedinih vrsta troškova treba definirati amortizacijski parametar AP koji predstavlja vrijednost opreme u nekom trenutku unutar razdoblja njezine amortizacije. Tako je uobičajeno da se zgrade amortiziraju kroz 20 godina, a računalna oprema kroz tri godine. Ako se cijeli račun troškova oblačnog centra izvede za jednu poslovnu godinu, onda će vrijednost amortizacijskog parametra biti (prema [115]):

$$AP = \frac{(1+p)^{\frac{P_A}{12}}}{30 \cdot 24 \cdot P_A} \cdot t \qquad (6.1)$$

Komponente izraza za vrijednost amortizacijskog parametra su:

- t – vrijeme u satima $[h]$
- p – cijena kapitala, odnosno godišnja kamatna stopa na kapital $[\%]$
- P_A – amortizacijski period u mjesecima

Iz izraza (6.1) može se (radi provjere ispravnosti) jednostavno izračunati da će vrijednost opreme, koja se amortizira za 3 godine ($P_A=36$), na kraju prve godine punog korištenja ($t=30 \cdot 24 \cdot 12$) iznositi trećinu nabavne vrijednosti, uvećano za godišnju kamatnu stopu.

Struktura troškova za investitora ili vlasnika oblačnog podatkovnog centra prikazana je modelom u tablici 6-I. Sve vremenski zavisne komponente vode se u ovom proračunu za razdoblje od jedne poslovne godine. Potrebna objašnjenja o značenju pojedinih komponenata te način izračunavanja su prikazani u drugom i trećem stupcu tablice.

Tablica 6-I: ***Struktura troškova oblačnog podatkovnog centra***

Komponenta troška za poslovnu godinu [$]	***Parametar***	***Opis parametra troška***	***Objašnjenje i izračun komponente troška iz parametara (prema [115])***
Ukupna vrijednost svih nabavljenih računala C_{RU}	N_R	Količina fizičkih računala	$C_{RU} = C_R \times N_R \times AP(t)$
	C_R [$]	Cijena jednog fizičkog računala	
Ukupna vrijednost nabavljenog softvera (po vrstama) C_{SWU}	C_{SW1} [$]	Vrijednost softvera tipa 1, čija se jedinična cijena računa po broju licenci za virtualna računala (npr. operacijski sustav).	$C_{SWU} = (C_{SW1} \cdot N_{SW1} \cdot PF_1 + C_{SW2} \times N_{SW2} \times PF_2 + C_{SW3} \times N_{SW3} \times PF_3) \times AP(t)$
	C_{SW2} [$]	Cijena softvera tipa 2 čija se cijena računa po broju procesora na kojemu se pokreće (npr. aplikacijski serveri, RDBMS).	
	C_{SW3} [$]	Cijena softvera tipa 3 koji se računa po broju procesora kojima se upravlja (npr. hypervisorski programi).	
	N_{SW1}	Broj licenci softvera tipa 1.	
	N_{SW2}	Broj licenci softvera tipa 2	
	N_{SW3}	Broj licenci softvera tipa 3.	
	PF_1	Faktor cijene softvera (dio od nabavne cijene koji se plaća kroz godišnju pretplatu), domena (0,1]	
	PF_2	Pretplatnički faktor softvera tipa 2 (udio od godišnje cijene)	
	PF_3	Pretplatnički faktor softvera tipa 3 (udio od godišnje cijene)	
Mrežni troškovi C_{NetU}	N_{SCH}	Broj preklopnika u podatkovnom centru	$C_{NetU} = \frac{C_{SCH} \times N_{M,K} \times N_{P,MK} \times N_R \times AP(t)}{N_{P,SCH}}$
	$N_{M,K}$	Broj mrežnih kartica po virtualnom računalu	$C_{NetU} = C_{SCH} \times N_{SCH} \times AP(t)$
	$N_{P,MK}$	Broj portova po mrežnoj kartici	$N_{SCH} = \frac{N_{M,K} \times N_{P,MK} \times N_R}{N_{P,SCH}}$
	C_{SCH} [$]	Cijena jednog preklopnika	
	$N_{P,SCH}$	Broj portova po preklopniku	
Troškovi potpore i održavanja C_{LJ}	N_S	Broj suradnika za pogon i održavanje programske i sklopovske opreme	S obzirom da u podatkovnim centrima rade stalno zaposleni suradnici, dalje će se računati samo s evidentiranim vremenom na radnom mjestu i udjelom koji oni
	T_G [h]	Godišnji broj sati rada	

	η	Koeficijent iskorištenosti radnog vremena za poslove podatkovnog centra.	potroše za poslove vezane uz održavanje oblačnih resursa: $C_{LJ} = N_S \cdot T_G \cdot PL/\eta$
	PL [$/h]	Brutto satnica	
Troškovi pogonske energije za sve računalne uređaje C_E	W_U [kWh]	Ukupna snaga svih servera u jednom stalku (*rack*)	$C_E = W_U \cdot N_{RACK} \cdot C_W \cdot H$
	C_W [$/$kWh$]	Jedinična cijena energije	
	N_{RACK}	Broj serverskih stalaka u podatkovnom centru	
	H [h]	Broj sati rada računala	
Troškovni pomoćne energije C_{PE}	K_{HP}	Energija utrošena za pomoćne uređaje podatkovnog centra (rasvjeta, baterije za UPS) ili takozvani hladni pogon	Prema [4] W_U je svega 36% ukupne potrošnje energije oblačnog centra, a pomoćni uređaji (rasvjeta, UPS) troše 14% energije. Iz toga približno vrijedi: $K_{HP} \approx 0{,}39$ $C_{PE} = C_E \cdot K_{HP}$
Troškovi hlađenja C_{HLAD}	C_E [$]	Prema [4] energija hlađenja je jednaka zbroju C_E i C_{PE}	$C_{HLAD} = C_E + C_{PE}$
	C_{PE} [$]		
Troškovi ostale opreme C_{OP}	N_{RACK}	Broj serverskih stalaka u podatkovnom centru	Ostala oprema podatkovnog centra (kablovi, priključci itd.) proporcionalna je broju serverskih stalaka: $C_{OP} = C_{OP,RACK} \times N_{RACK} \times AP(t)$
	$C_{OP,RACK}$	Cijena opreme za jedan serverski stalak.	
Cijena nekretnine C_{NEK}	C_{M2} [$/$m^2$]	Cijena kvadratnog metra zgrade.	Površina zgrade također je proporcionalna broju serverskih stalaka: $C_{NEK} = P_{RACK} \cdot N_{RACK} \cdot C_{M2} \cdot F_P \cdot AP(t)$
	P_{RACK} [m^2]	Brutto površina za postavljanje jednog serverskog stalka.	
	N_{RACK}	Broj serverskih stalaka u podatkovnom centru	
	F_P	Faktor uvećanja za pomoćni prostor	

Program za kompletni obračun troškova podatkovnog centra prema ovom modelu izveden je u Excelu te se nalazi u prilogu 6.1.

Radi provjere logičke konzistentnosti i dimenzijske ispravnosti formula iz tablice 6-I, izračunati će se cijena izgradnje te jednogodišnjeg korištenja i održavanja za poslovni slučaj virtualnog podatkovnog centra koji bi pružao usluge iz domene oblačnog računarstava. Podatkovni centar neka ima $N_R = 1000$ fizičkih računala, a svako računalo objedinjava 4 procesorske jezgre. Vrijednosti ostalih parametara vidljive su u proračunu koji je u cijelosti izveden programom iz priloga 6.1 i prikazan na sljedećoj stranici. Kriteriji za izračun ili izbor vrijednosti parametara kratko su obrazloženi u sljedećim odlomcima:

- Odabrana su računala više klase, čija je pojedinačna cijena $C_R = 1000$ $.
- Podatkovni centar koristi sve tri vrste softvera: operacijski sustav s pretpostavljenom cijenom 60 $ po računalnoj jezgri, RDBMS na polovici svih računala s jediničnom

cijenom 100 $ po jezgri i pomoćni programi instancirani na svakom fizičkom računalu čija se jedinična vrijednost procjenjuje na 10 $.

- Broj portova (utičnica) po mrežnoj kartici $N_{P,MK} = 1$.
- Broj portova (utičnica) po preklopniku (*switch*) $N_{P,SCH} = 48$.
- Broj mrežnih kartica po virtualnom računalu $N_{MK} = 2$.
- Ukupni broj preklopnika u podatkovnom centru koji se računa tako da jedan preklopnik dolazi na 24 računala pa je $N_{SCH} \approx 41$.
- Cijena jednog 48-portnog preklopnika je $C_{SCH} = 1200\ \$$. To je dosta niska cijena, no pretpostavimo da je ekonomija razmjera značajno utjecala na iznos.
- Prosječna bruto satnica visokostručnog radnika u oblačnom centru procjenjuje se (prema dostupnim podacima u USA) na $P_B = 42\ \$/h$.
- Prema poglavlju 4, jedan sistem-inženjer u oblačnom centru može održavati do 1.000 servera. Radi povremene odsutnosti (godišnji odmor, bolest), uzima se da će na 10.000 servera biti raspoređeno 12 sistem-inženjera. Tom broju treba dodati i 50% ostalog tehničkog osoblja visoke struke (strojarski inženjeri i tehničari za održavanje rashladnog sustava, *helpdesk*, uprava itd.). Ovih 18 suradnika mora raditi u tri smjene (jer je podatkovni centar uvijek u pogonu) što daje ukupan broj suradnika $N_S = 54$.
- Godišnji fond sati za svakog suradnika je otprilike $T_G \approx 1800\ h$.
- Ukupno potrebna snaga po jednom stalku (s 14 računala) iznosi $W_U = 24{,}5\ kWh$, ako se pretpostavi da jedno računalo zahtijeva snagu od 200 W.
- Cijena električne energije je $C_W = 0.2\ \$/kWh$ (što je iznos koji se, prema dostupnim izvorima plaća u USA, u Hrvatskoj je to zasad manje).
- Broj serverskih stalaka $N_{RACK} \approx 715$ (na svaki se stalak raspoređuje po 14 računala).
- Vrijednost opreme (okvir, napajanje, i drugo) za svaki serverski stalak procijenjena je na iznos $C_{OP,RACK} = 1000\ \$$.
- Pretpostavlja se da ukupna cijena izgradnje građevine za smještaj podatkovnog centra u USA iznosi $C_{M2} = 1000\ \$/m^2$.
- Efektivna površina koju zauzima jedan serverski stalak iznosi (uključivo površine za komunikaciju) približno 1,5 [m] ·1,5 [m] to jest $P_{RACK} = 2{,}25\ m^2$.
- Pomoćni prostori (radionice, sobe, administracija, sanitarije i slično) uvećavaju potrebne površine u podatkovnom centru oko 20% pa je $F_P = 20$.

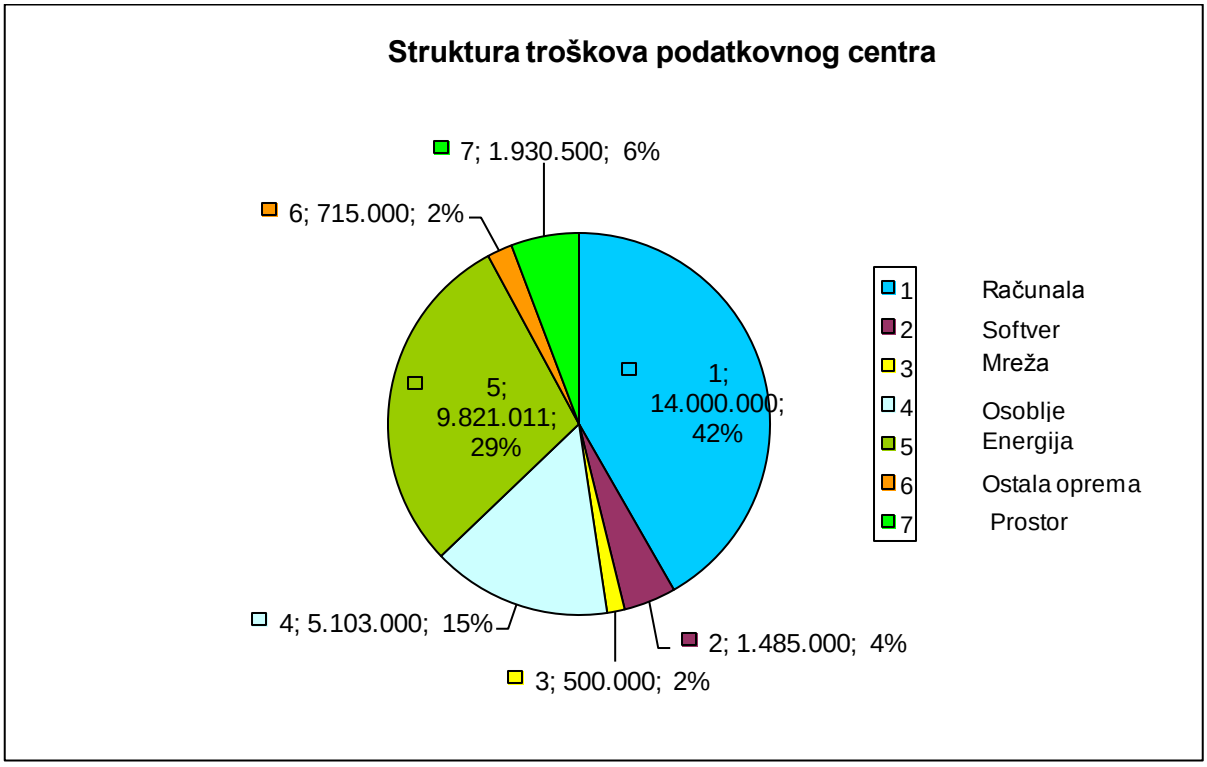
Na temelju navedenih podataka te uz primjenu kamatne stope na cijenu kapitala od $p=7\%$ proizlazi izračun, po programu iz priloga 6.1, koji je u cijelosti vidljiv na sljedećoj stranici.

Troškovna komponenta	*Vrijednost [US$]*	*Parametar*	*Oznaka*	*Jedinična vrijednost*	*Jedinica mjere*
Ukupna vrijednost svih nabavljenih računala C_{RU}	14.000.000,00	Broj računala	N_R	10.000,00	[kom]
		Cijena jednog računala	C_R	1.400,00	[US$/kom]
Ukupna vrijednost nabavljenog softvera C_{SWU}	1.485.000,00	Cijena softvera tipa 1	C_{SW1}	60,00	[US$/kom]
		Cijena softvera tipa 2	C_{SW2}	100,00	[US$/kom]
		Cijena softvera tipa 3	C_{SW3}	10,00	[US$/kom]
		Broj licenci softvera tipa 1	N_{SW1}	40.000,00	[kom]
		Broj licenci softvera tipa 2	N_{SW1}	20.000,00	[kom]
		Broj licenci softvera tipa 3	N_{SW1}	10.000,00	[kom]
		Dio koji se plaća godišnje za tip 1	P_{F1}	0,33	[]
		Dio koji se plaća godišnje za tip 2	P_{F1}	0,33	[]
		Dio koji se plaća godišnje za tip 3	P_{F1}	0,33	[]
Mrežni troškovi C_{NetU}	500.000,00	Broj preklopnika u podatkovnom centru	N_{SCH}	416,67	[kom]
		Broj mrežnih kartica po virtuelnom računalu	N_{MK}	2,00	[kom]
		Broj portova po mrežnoj kartici	$N_{P,MK}$	1,00	[kom]
		Cijena jednog preklopnika	C_{SCH}	1.200,00	[US$/kom]
		Broj portova po preklopniku	$N_{P,SCH}$	48,00	[kom]
Troškovi osoblja podatkovnog centra C_{LJ}	5.103.000,00	Broj suradnika u podatkovnom centru	N_S	54,00	[]
		Broj radnih sati godišnje	T_G	1.800,00	[h]
		Udjel radnog vremena za podatkovni centar	η	0,80	[]
		Brutto satnica	P_B	42,00	[US$/h]
Troškovi energije za računalne uređaje C_{ER}	3.507.504,00	Ukupna snaga svih servera u jednom stalku	W_U	2,80	[kW]
		Broj serverskih stalaka u podatkovnom centru	N_{RACK}	715,00	[kom]
		Jedinična cijena energije	C_W	0,20	[US$/kWh]
Troškovi pomoćne energije C_{PE}	1.403.001,60	% ukupne energije za pomoćne uređaje	K_{PE}	40,00	[%]
Troškovi hlađenja C_{HLAD}	4.910.505,60	Troškovi hlađenja	C_{HLAD}	4.910.505,60	[US$]
		Ukupni troškovi energije C_{UE}	C_{UE}	9.821.011,20	[US$]
Troškovi ostale opreme C_{OOP}	715.000,00	Cijena opreme za jedan serverski stalak	$C_{OOP/RACK}$	1.000,00	[US$/kom]
Cijena nekretnine C_{NEK}	1.930.500,00	Izgradbena cijena zgrade	C_{M2}	1.000,00	[US$/m2]
		Brutto površina za postavljanje jednog stalka	P_{RACK}	2,25	[m2]
		% uvećanja površine za pomoćni prostor	F_P	20,00	[%]
Ukupna cijena izgradnje i jednogodišnjeg rada	**33.554.511,20**				

Ukupna cijena postavljanja i rada ovakvog zamišljenog oblačnog podatkovnog centra iznosi **35.903.326,98 $** za prvu godinu rada. Ona uključuje sva ulaganja u izgradnju i opremanje, trošak kapitala i troškove operativnog rada tijekom prve godine funkcioniranja.

Ovaj okvirni proračun treba provjeriti da li je u skladu s komercijalnom ponudom (o kojoj će više riječi biti kasnije u ovome poglavlju) za iznajmljivanje računalni resursa. Dakle, godina ima ukupno 8760 sati, ali uz iskoristivost podatkovnog centra oko 95% proizlazi prosječna cijena sata rada oblačnog centra oko 4.315 $. Toliko, dakle, otprilike košta jedan sat pogona takvog postrojenja kao što je oblačni centar. Dalje, potrebno je taj iznos podijeliti s brojem fizičkih računala (10.000) iz čega proizlazi da rad pojedinačnog računala iznosi $0{,}4315\ \$/h$. Budući da prosječno računalo ima 4 procesorske jezgre, cijena jedne računalne instance (u poglavlju 4 nazvane ROJ) iznosi $c_h=0{,}1079\ \$/h$. To odgovara komercijalnim cijenama koje se kreću između 9 i 12 centi po satu. Ovim se može potvrditi utemeljenost ovdje izloženog načina računanja troškova izgradnje oblačnog podatkovnog centra.

Za cijeli ovaj proračun izrađen je program u Excelu koji se nalazi u prilogu 6.1 Za gornji (virtualni) poslovni slučaj izveden po tom programu prikazani su odnosi među troškovnim grupama kružnim grafom na slici 6.1. Podaci za usporedbu teško se nalaze u literaturi jer (što je razumljivo) investitori izbjegavaju javno objavljivati svoje investicijske proračune. Jedna od rijetkih referenci se nalazi u [4] s kojom se dobro slažu ovdje dobiveni podaci o strukturi ukupnih troškova oblačnog podatkovnog centra.



Slika 6.1: ***Struktura troškova podatkovnog centra*** *(u [$] za virtualni poslovni slučaj)*

Iz grafa na slici 6.1 se vidi da računala čine manje od polovice ukupnih troškova. Drugi po veličini su troškovi energije, koji su ovdje prikazani od tri komponente: troškovi pogonske energije za računalne uređaje, troškovi pomoćne uređaje i troškovi hlađenja. Ovakvo grupiranje je provedeno kako bi se mogla napraviti provjera rezultata dobivenih ovim modelom s podacima koji se mogu naći u literaturi, kao u [4].

Naravno, ovdje provedeni proračun cijene je pojednostavljen te daje nešto uvećan iznos za cijenu jedne računalne instance. Naime, ovdje se troškovi izračunavaju tako da se s cjelokupnom vrijednošću ulaganja u računalnu opremu tereti prva godina operativnog rada podatkovnog oblačnog centra, odnosno kao da se sva nabavljena računalna oprema amortizira tijekom jedne godine rada. U detaljnijem ekonomskom izračunu, koji bi vodio računa o postupnoj amortizaciji tijekom duljeg razdoblja rada (na primjer, kroz 3 godine, kako je uobičajeno za opremu i 10 godina kako se obično računa za zgradu) pokazalo bi se da je cijena sata rada virtualnog računala manja od ovdje izračunatih 10 centi, ali bi nas složenost takvog izračuna odvela daleko van okvira ovoga rada (premda su smjernice za takvo promišljanje obrađene u sljedećim potpoglavljima). To znači da ponuđači oblačnih računalnih resursa imaju (prema trenutno uobičajenim cijenama sata rada iznajmljenog virtualnog računala) relativno dosta prostora za profitabilni povrat investicije, odnosno da se tijekom nailazećih godina, zbog očekivanog povećanja konkurentskih ponuditelja, može računati sa smanjenjem cijene sata rada virtualnog računala uz diversifikaciju obračunskih modela. Tako se, recimo, već u trenutku završavanja ovog rada pojavila kod Amazona vrlo povoljna cijena najma takozvane mikroinstance, za poslove s vrlo niskom razinom zahtjeva na računala, uz cijenu najma od svega 3 centa po satu. Procjenjujemo da je svrha ovakve Amazonove ponude upoznavanje potencijalnih korisnika s mogućnostima rada u oblačnom okruženju, kako bi stekli potrebna znanja i vještine za izvođenje daleko složenijih aplikacija u budućnosti.

Međutim, u ovom je radu bilo važno razviti pouzdan model za proračun obujma potrebnih ulaganja u oblačni centar s motrišta investitora, polazeći pri tome od specifičnih parametara koji takvu investiciju razlikuju od investicija u druge tipove objekata. Vrijednosti pojedinih parametara, koje su korištene kod proračuna u zamišljenom poslovnom slučaju, mogu se mijenjati (također se mogu uključiti u račun i različite stope amortizacija za različite vrste opreme) ali smatramo da se matematički model za proračun troškova investicije u oblačni podatkovni centar ne će mijenjati sve dok vrijedi sadašnja paradigma oblačnog računarstva.

U skladu s onim što je rečeno u uvodu ovoga poglavlja, može se dalje nastaviti s usporedbenom analizom cijene CPU vremena za slučaj kada se računalni resursi pribavljaju iznajmljivanjem opreme u računalnom oblaku i za slučaj kada se isto takva oprema kupuje u vlasništvo korisnika.

6.2 Usporedbena analiza cijene CPU vremena

Današnje ustanove ili poduzeća mogu realizirati svoje potrebe za informatičkom infrastrukturom koju traži njihov informacijski sustav na dva načina, kako je to pokazano u prethodnim poglavljima: kupnjom servera ili najmom potrebnih računala.

Drugi način uključuje, naravno, najam procesorskog, CPU vremena (uz pripadnu memoriju, mrežnu protočnost, diskove i drugo što je opisano u poglavljima 2 i 3). Drugim riječima, najmom potrebnih računala uvijek se unajmljuje i neko CPU vrijeme, pri čemu je kod nekih oblačnih poslužitelja to vrijeme i jedini resurs koji se može eksplicitno unajmiti (kao u Google AppEngineu, što je opisano u poglavlju 4) dok se kod drugih, primjerice Amazona, CPU sat unajmljuje kroz najam virtualnog računala.

Zadatak ovog dijela rada je istražiti što zapravo znače te cijene. Da li je to za korisnika usluga oblačnog računarstva malo ili mnogo? Ima li za korisnika usluge taj iznos istu vrijednost za svaki obujam korištenja računalnih resursa? Kako izgleda račun rentabilnosti s motrišta investitora ako naplaćuje vrijednost CPU sata uz cijenu ponuđenu tržištu i o kojim parametrima zavisi taj račun rentabilnosti?

U svim dugoročnim financijskim proračunima valja voditi računa da se vrijednost uloženog ili potrošenog novca mijenja tijekom vremena. Svaki primitak (za prodavatelja roba ili usluga) ili izdatak (za kupca ili korisnika) vrijedi više ako je primljen ili isplaćen danas, nego ako je taj isti iznos primljen ili isplaćen u budućnosti. I obrnuto: svaki budući primitak ili izdatak (*future value – FV*) ima manju sadašnju vrijednost (*present value* – PV) nego da je primljen ili isplaćen danas. Izraz *(6.2)* ,predstavlja izračun sadašnje vrijednosti PV za iznos FV, koji će biti primljen u budućnosti nakon T godina, pri čemu je *k* vrijednost kapitala ili godišnja kamatna stopa.

$$PV = \frac{FV}{(1+k)^T} \tag{6.2}$$

Ako se u poslovanju tijekom svake od T $(T=1,2,\ldots Y)$ budućih godina može ostvariti neto primitak (to jest ukupni prihod umanjen za ukupne troškove) u iznosu C_T, onda se, polazeći od izraza *(6.2)*, takozvana neto sadašnja vrijednost svih primitaka (*net present value* - NPV) tijekom *Y* godina izračunava prema izrazu:

$$NPV = \sum_{T=0}^{Y-1} \frac{C_T}{(1+k)^T} \tag{6.3}$$

U radu [108] postavljeni su temelji kvantitativnog modela koji može omogućiti izvršnom direktoru (*chief executive officer* – CEO) ili voditelju informatike (*chief information officer* –

CIO) u nekoj tvrtki ili ustanovi objektivni izbor da li je za tu tvrtku ili ustanovu financijski povoljnije potrebni procesorski kapacitet kupiti ili unajmiti. Ostatak ovog potpoglavlja 6.2, sve do izučavanja prvog poslovnog slučaja, sadrži sažeti prikaz Walkerovog modela. Formule od *(6.2)* do *(6.12)* izvedene su iz tog modela opisanog u [108].

Za razliku od klasičnih financijskih modela, gdje se razmatraju opcije kupnje ili unajmljivanja bilo kojeg osnovnog sredstva, Walkerov model uključuje ne samo amortizaciju već i relativno umanjivanje CPU performansi tijekom vremena, izvedeno temeljem Mooreovog zakona[12] [67] prema kojem se performanse CPU-a udvostručuju svake dvije godine. Iz toga proizlazi da će performanse CPU-a koji je nabavljen danas biti dvostruko slabije od performansi CPU-a koji bi se mogao nabaviti za dvije godine[13]. To drugim riječima znači da se sadašnje performanse PC nekog T godina starog procesorskog kapaciteta mogu iskazati kao umanjene performanse budućeg procesorskog kapaciteta FC tako da je svake dvije godine upola manji ("tehnološki diskont"). Ta se zakonitost relativnog zastarijevanja može iskazati izrazom *(6.4)*.

$$PC = \frac{FC}{(\sqrt{2})^T} \tag{6.4}$$

Uzme li se da se ukupni iskoristivi procesorski kapacitet TC ostvaruje korištenjem skupine CPU-ova (*cluster*), onda se ukupno iskoristivi procesorski kapacitet TC može izračunati kao umnožak ukupnog broja korištenih CPU jedinica $TCPU$, očekivanog broja H radnih sati godišnje i koeficijenta iskoristivosti servera η prema izrazu *(6.5)*. Pod skupinom CPU-ova misli se na procesore u računalima unutar velike serverske farme kao što je oblačni ili poduzetnički podatkovni centar.

$$TC = TCPU \cdot H \cdot \eta \tag{6.5}$$

Ako se račun relativnog zastarijevanja procesorskog kapaciteta, definiran izrazom *(6.4)*, primijeni na ukupno iskoristivi procesorski kapacitet *TC* prema izrazu *(6.5)*, tada se može izračunati neto sadašnji raspoloživi procesorski kapacitet skupine CPU-ova NPC koji će se ostvariti tijekom *Y* godina operativnog korištenja te skupine:

$$NPC = TC \cdot \sum_{T=0}^{Y-1} (\frac{1}{\sqrt{2}})^T \quad \text{iz čega slijedi} \Rightarrow \quad NPC = TC \cdot \frac{1-(\frac{1}{\sqrt{2}})^Y}{1-(\frac{1}{\sqrt{2}})} \tag{6.6}$$

[12] Izvorni Mooreov zakon odnosi se na razinu integriranosti elektroničkih krugova, ali su kasnija istraživanja pokazala da se može primijeniti i na mikroprocesorske tehnologije koje se primjenjuju u proizvodnji CPU.

[13] Mooreov zakon vrijedi i danas i nastavit će vrijediti u dogledno vrijeme unatoč dosezanju krajnjih granica poluvodičke minijaturizacije stoga što sve više odnosi na količinu procesorskih jezgri jednog fizičkog procesora.

Stvarni trošak CPU sata R može se izračunati tako da se neto sadašnja vrijednost iz izraza *(6.3)* podijeli s neto sadašnjim raspoloživim procesorskim kapacitetom prema izrazu *(6.6)*, dakle:

$$R = NPV / NPC \tag{6.7}$$

Stvarni trošak CPU sata za slučaj **kupnje** R_K *i*znosi:

$$R_K = \frac{(1-\frac{1}{\sqrt{2}}) \cdot \sum_{T=0}^{Y-1} \frac{C_T}{(1+k)^T}}{(1-(\frac{1}{\sqrt{2}})^Y) \cdot TC} \tag{6.8}$$

Izračun stvarnog troška CPU sata u slučaju najma R_N može se također izračunati prema izrazu *(6.7)* ako se ne računa s procesorskim kapacitetom koji se smanjuje tijekom vremena, već s raspoloživim procesorskim kapacitetom TC iz formule *(6.5)*. Ta je pretpostavka opravdana jer se u najam uzimaju novi uređaji. Stoga će raspoloživi procesorski kapacitet tijekom Y godina biti:

$$NPC = Y \cdot TC \tag{6.9}$$

Uvrštavanjem izraza *(6.3)* i *(6.9)* u formulu *(6.7)* dobiva se stvarni trošak CPU sata R_N za slučaj **najma**:

$$R_N = \frac{\sum_{T=0}^{Y-1} \frac{C_T}{(1+k)^T}}{Y \cdot TC} \tag{6.10}$$

Na sličan način moguće je analizirati još jedan način pribavljanja potrebnih procesorskih kapaciteta: kupnja početne skupine CPU-ova i njihova redovita godišnja zamjena novim, kako bi se izbjeglo degradiranje kapaciteta. Ako se uzme da cijena novih CPU-ova ostaje približno ista (premda se, u skladu s Moorevim zakonom, performanse udvostručuju svake dvije godine), onda se izraz *(6.3)* može napisati tako da se početni učinak izdvoji ispred znaka sume, sumiranje počne s vrijednošću $T=1$, a u brojnik uvede vrijednost A koja označava troškove nabave serverske skupine. Izraz za neto sadašnju vrijednost onda glasi:

$$NPV = C_0 + \sum_{T=1}^{Y-1} \frac{C_T - A}{(1+k)^T} \tag{6.11}$$

S obzirom na redovito godišnje obnavljanje CPU-ova raspoloživi serverski kapacitet se ne umanjuje, pa konačno stvarni trošak CPU sata za slučaj kupnje uz dogradnju R_{Kd} glasi:

$$R_{Kd} = \frac{C_0 + \sum_{T=1}^{Y-1} \frac{C_T - A}{(1+k)^T}}{Y \cdot TC} \qquad (6.12)$$

Prema prethodnom teoretskom razmatranju u daljnjem tekstu ovoga potpoglavlja je provedena analiza za tri različita poslovna slučaja iz domene ekonomike oblačnog računarstva. Ti poslovni slučajevi čiji opis slijedi izvorni su doprinos autora ovog rada, namijenjen provjeri primjenjivosti Walkerovog modela za izračun troškova CPU vremena.

Poslovni slučaj 1: Kupnja ili najam računalnih kapaciteta

Općenito, ovaj se problem može postaviti ovako: tvrtka treba za svoj rad računalne kapacitete koji se mogu ostvariti u skupini od S servera, svaki s c jezgri. Potrebno je razmotriti dvije opcije za osiguranje takvog računalnog kapaciteta: kupnja vlastite serverske farme ili unajmljivanje potrebnih kapaciteta u komercijalnom računalnom oblaku. Parametri za rješavanje ovog problema navedeni su u tablici 6-II. Njihove vrijednosti određene su prema konkretnim podacima iz [108] gdje se razmatra način na koji je Texas Advanced Computing Center osigurao kapacitete za napredno računarstvo te prosječnim vrijednostima za cijene navedene u [19].

Tablica 6-II: ***Podaci za izračun odnosa kupnja/najam***

	Parametar	*Oznake*	*Vrijednost*	*Napomena*
1.	Broj servera	S	15.000	
2.	Broj jezgri po serveru	c	4	
3.	Cijena servera [$]	C_S	2.000	Cijena servera svojstava kao u 1. i 2.
4.	Sati rada servera kroz godinu [h]	H	7.488	Računato kao (365-53)·24 zbog održavanja jedan dan u tjednu.
5.	Početno ulaganje u servere i opremu [$]	C_0	30,000.000	$C_0=S \cdot C_s$
6.	Stupanj korištenja servera	η	0,95	
7.	Godišnji fiksni troškovi [$]	C_T	7,000.000	
8.	Godišnja kamatna stopa [%]	k	7	
9.	Cijena najma u oblaku [$/sat]	P	0,10	Sadašnja vrijednost, bez obzira na razinu zahtjeva prema CPU.

U tablici 6-III je izračunata cijena CPU [$/h] za slučaj da poduzeće koristi vlastite računalne kapacitete i za slučaj da su isti kapaciteti unajmljeni u komercijalnom računalnom oblaku. Kod izračuna je korišten izraz *(6.8)*, koji je nastao tako da su *(6.3)* i *(6.6)* uvršteni u izraz *(6.7)*. Račun se temelji na sljedećem promišljanju: ako tvrtka želi osigurati vlastite računalne

resurse, mora prije početka rada ($T=0$) nabaviti S servera od kojih svaki ima c procesorskih jezgri. Za to će jednokratno utrošiti iznos $S \cdot C_S$, odnosno 30.000.000 $, što je upisano u stupcu 2 za redak $T=0$ tablice 6-III. Pored toga, mora tijekom rada kroz T godina $(T=1,2,...Y)$ dodatno ulagati u tehničko održavanje, potrošnju energije, zakup prostora i plaće radnika. To znači da svake godine treba potrošiti svotu C_T, što je upisano u retke $T=1,2,..Y$. Svi se ovi ulozi diskontiraju na neto sadašnju vrijednost u stupcu 3 i kumuliraju tijekom promatranog razdoblja u tom stupcu. Takvim ulaganjem ostvaren je tehnički kapacitet koji u prvoj godini korištenja iznosi $TC=S \cdot c \cdot H \cdot \eta$ = 426.816.000 sati rada procesora godišnje, ali se tijekom godina umanjuje prema Mooreovom zakonu. Godišnji neto tehnički kapacitet (sveden na sadašnju vrijednost) je prikazan u stupcu 4, a kumulirani neto sadašnji kapacitet u stupcu 5 tablice 6-III.

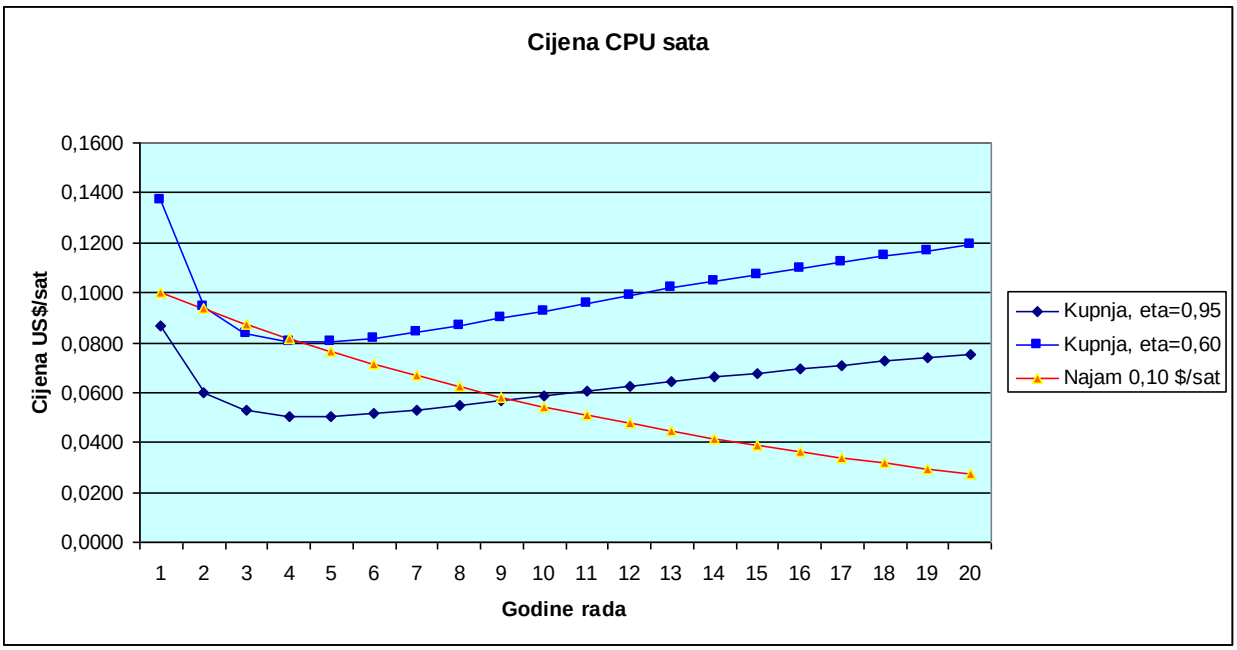
Ako se, u skladu s izrazom *(6.7)*, izračuna kvocijent $\Sigma NPV/\Sigma NPC$ za buduće razdoblje, dobije se vrijednost CPU sata za slučaj kupnje potrebnih računalnih kapaciteta, što je upisano u stupac 6. Da bi se mogla izabrati povoljnija opcija za tvrtku koja treba navedene računalne kapacitete, treba izraz za R_K usporediti s cijenom koju bi trebalo godišnje plaćati pružatelju usluga oblačnog računarstva za najam istog računalnog kapaciteta. Vrijednost najma treba također diskontirati na neto sadašnju vrijednost, što je prikazano u stupcu 8 tablice 6-III. Radi uočavanja trendova cijeli je izračun izveden za 20 godina, ali se time nikako ne implicira da će to biti vijek trajanja računalnih tehnologija koje danas poznajemo.

Svi detalji proračuna vidljivi su iz programa izrađenog u Excelu, koji se nalazi u prilogu 6.2. Proračun je izveden uz pretpostavku da će se kupljeni računalni kapacitet koristiti uz visoki stupanj iskoristivosti $\eta=0,95$. Radi analize odnosa za razne uvjete korištenja, kao i radi dopunske kontrole algoritma, proračun je ponovljen i za slučaj da će tehnički kapacitet biti manje iskorišten ($\eta=0,60$), što općenito daje višu cijenu CPU sata. Rezultati tog proračuna su upisani u stupac 7 tablice 6-II i općenito ukazuju na to da se cijena CPU sata za korisnika smanjuje s povećanjem stupnja iskorištenja računalnih resursa.

Tablica 6-III: ***Cijena CPU sata računalnih resursa u vlasništvu i najmu*** *(za podatke iz tablice 6-II)*

Godina rada T *(1)*	*Godišnji ulozi [$]* *(2)*	*Kumulirana ulaganja ΣNPV [$]* *(3)*	*Godišnji kapacitet [h]* *(4)*	*Kumulirani kapacitet ΣNPC [h]* *(5)*	R_K *(η=0,95) [$/h]* *(6)*	R_K *(η=0,60) [$/h]* *(7)*	P_T *[$/h]* *(8)*
0	30.000.000						
1	7.000.000	37.000.000	426.816.000	426.816.000	0,0867	0,1373	0,1000
2	7.000.000	43.542.056	301.804.488	728.620.488	0,0598	0,0946	0,0935
3	7.000.000	49.656.127	213.408.000	942.028.488	0,0527	0,0835	0,0873
4	7.000.000	55.370.212	150.902.244	1.092.930.732	0,0507	0,0802	0,0816
5	7.000.000	60.710.479	106.704.000	1.199.634.732	0,0506	0,0801	0,0763
6	7.000.000	65.701.382	75.451.122	1.275.085.854	0,0515	0,0816	0,0713
7	7.000.000	70.365.778	53.352.000	1.328.437.854	0,0530	0,0839	0,0666
8	7.000.000	74.725.026	37.725.561	1.366.163.415	0,0547	0,0866	0,0623
9	7.000.000	78.799.090	26.676.000	1.392.839.415	0,0566	0,0896	0,0582
10	7.000.000	82.606.626	18.862.780	1.411.702.195	0,0585	0,0926	0,0544
11	7.000.000	86.165.071	13.338.000	1.425.040.195	0,0605	0,0957	0,0508
12	7.000.000	89.490.720	9.431.390	1.434.471.586	0,0624	0,0988	0,0475
13	7.000.000	92.598.804	6.669.000	1.441.140.586	0,0643	0,1017	0,0444
14	7.000.000	95.503.555	4.715.695	1.445.856.281	0,0661	0,1046	0,0415
15	7.000.000	98.218.276	3.334.500	1.449.190.781	0,0678	0,1073	0,0388
16	7.000.000	100.755.398	2.357.848	1.451.548.628	0,0694	0,1099	0,0362
17	7.000.000	103.126.540	1.667.250	1.453.215.878	0,0710	0,1124	0,0339
18	7.000.000	105.342.561	1.178.924	1.454.394.802	0,0724	0,1147	0,0317
19	7.000.000	107.413.608	833.625	1.455.228.427	0,0738	0,1169	0,0296
20	7.000.000	109.349.167	589.462	1.455.817.889	0,0751	0,1189	0,0277

Prvi redak tablice 6-III (označen s 0 kao godinom rada) odnosi se na kraj razdoblja početnog ulaganja, dakle na trenutak kada još nisu korišteni kapaciteti ostvareni ulaganjem. Usporedba vrijednosti CPU sata kod rada na vlastitoj opremi, za slučaj visokog ($\eta=0,95$) i niskog ($\eta=0,60$) stupnja korištenja kupljenih računalnih kapaciteta, s cijenom CPU sata na resursima koji su unajmljeni u oblaku, najbolje se vidi na slici 6.2.



Slika 6.2: ***Odnos cijene CPU/sat kod korištenja vlastite opreme i najma u oblaku***

Analiza odnosa na slici 6.1 upućuje na vrlo interesantne zaključke. Uz pretpostavku visokog stupnja korištenja servera ($\eta=0,95$), cijena CPU sata u najmu je tijekom prvih osam godina rada uvijek niža od cijene po kojoj se isti računalni kapacitet može unajmiti u oblaku, u devetoj godini se te cijene izjednačavaju, a za razdoblje korištenja od 10 i više godina najam potrebnih CPU resursa u oblaku je uvijek povoljnija opcija. Međutim, za niži stupanj korištenja servera ($\eta=0,60$) opcija najma kapaciteta u računalnom oblaku je uvijek povoljnija od kupnje vlastitih resursa! Naravno, ovakav zaključak vrijedi za one pružatelje usluga oblačnog računarstva koji svoju cjenovnu politiku temelje samo na naplati broja rezerviranih računala u oblaku, ali ne i intenziteta njihovog korištenja. Iz prethodnih je razmatranja u poglavlju 4 vidljivo da takvu cjenovnu politiku vode Amazon i Microsoft.

Ovdje treba pobliže objasniti tok krivulje najma na slici 6.2. Cijena najma se smanjuje u budućnosti zato što se radi o budućim vrijednostima koje se sadašnje vrijednosti od 0,1 [$/h] diskontiraju na neto sadašnju vrijednost prema formuli 6.3.

Nadalje, graf obuhvaća razdoblje od 20 godina samo radi lakšeg uočavanja trendova pojedinih krivulja, što se nikako ne može tumačiti da one daju relevantne vrijednosti za razdoblje nakon 5 godina korištenja hardverske opreme.

Iz ovoga slijedi da vlastite računalne kapacitete valja kupovati onda ako su aplikacije tako dobro pripremljene da će nabavljeni računalni resursi biti iskorišteni u vrlo visokom stupnju i kroz dugo razdoblje. Nasuprot tome, ako organizacija nije spremna za intenzivno korištenje suvremenih ICT u svim svojim poslovnim procesima, onda je za nju povoljnije da za izabrani poslovni proces unajmi u računalnom oblaku točno toliko računalnih kapaciteta koliko treba

za pripremljene i uvedene aplikacije. Ova druga opcija je osobito povoljna za mala i srednja poduzeća koja uglavnom nemaju dovoljno stručnjaka informatike za organizaciju i pogon vlastitog računalnog centra, ali žele koristiti suvremene ICT za potporu nekog od svojih strateških poslovnih procesa.

U prilogu 6.2 nalazi se kompletan matematički model, izrađen u Excelu, u kojem je moguće varirati vrijednost svih nezavisnih varijabli korištenih u izrazima od *(6.2)* do *(6.12)* na kojima se cijeli model temelji. Tako se mogu istraživati različiti problemi te postavljati ili provjeravati različite strategije osiguranja potrebnih resursa s obzirom na cijene CPU sata, jednako s motrišta korisnika kao i davatelja usluga. Upravo ovaj analizirani poslovni slučaj 1 pomaže korisniku u izboru povoljnije opcije za pribavljanje potrebnih računalnih resursa.

Poslovni slučaj 2: Donja granična cijena za usluge oblačnog računarstva

Sljedeći poslovni slučaj odnosi se na problem važan za pružatelje komercijalnih usluga oblačnog računarstva koji se općenito može formulirati na sljedeći način: pružatelj usluga oblačnog računarstva želi investirati u izgradnju oblačnog podatkovnog centra te odrediti koja je to donja granična cijena po kojoj bi mogao iznajmljivati jedan sat rada virtualnog CPU u oblaku, a da pri tome ostvari prihvatljivu razinu povrata investicije. Za numeričko rješavanje tog problema koristiti će se ranije prikupljeni podaci, već navedeni u tablici 6-II.

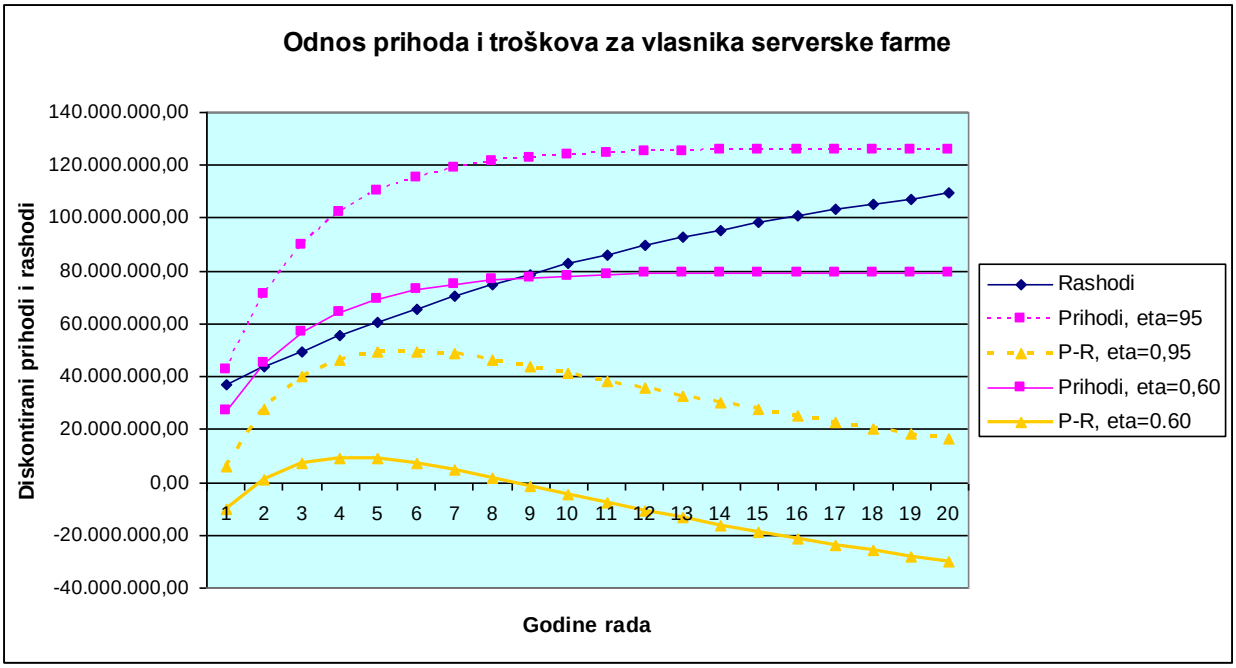
Za rješavanje tog problema promišlja se ovako: na početku razdoblja u kojem želi početi s pružanjem usluge investitor će utrošiti neki početni iznos za izgradnju oblačnog podatkovnog centra. Taj iznos zavisi o broju servera S, čiju snagu određena brojem procesorskih jezgri c. U izrazu *(6.9)* taj iznos je označen s C_T za $T=0$. Tijekom rada kroz T godina taj procesorski kapacitet treba održavati u radnom stanju, što znači da treba svake godine dodatno ulagati radi tehničkog održavanja, potrošnje energije, zakupa prostora i plaćanja suradnika. Drugim riječima, svake godine treba uložiti svotu C_T, gdje vrijedi $T=1,2,...Y$. Ovi iznosi, uloženi tijekom budućih godina $T=1,2,...Y$, diskontiraju se na sadašnju vrijednost što je prikazano stupcem *(2)* u tablici 6-IV koji, dakle, predstavlja kumulirane rashode po godinama. Početni tehnički kapaciteti, određeni izrazom *(6.5)* uz pretpostavku da je iskoristivost računala $\eta=0,95$ (te koji za numerički izveden primjer iznose godišnje 426.816.000 CPU radnih sati) relativno se umanjuju tijekom godina prema Mooreovom zakonu. Tako izračunat budući raspoloživi tehnički kapacitet, sveden na sadašnju vrijednost, prikazan je stupcem 3 u tablici 6-IV. Takav neto tehnički kapacitet će se u budućnosti prodavati na tržištu po nominalnoj cijeni P i kumulativno je prikazan u stupcu 4. Međutim, sve buduće primitke s tog osnova treba također diskontirati po kamatnoj stopi kapitala *k* [%] što je za $L=0,10\ \$/h$ prikazano stupcem 5. Kumulativna vrijednost svih **prihoda** do neke godine prikazana je u stupcu 6. Ako se sada za svaku godinu promatranog razdoblja prikaže **dobit** kao razlika prihoda (stupac 6) i dobiti

(stupac 5), dobiju se vrijednosti kumulirane zarade investitora u serversku farmu, što je prikazano u stupcu 7. Na isti način izveden je račun dobiti uz pretpostavku da se tehnički kapaciteti serverske farme koriste s $\eta=0,60$ što je upisano u stupac 8 tablice 6-IV.

Tablica 6-IV: ***Proračun prihoda i troškova serverske farme*** *uz L=0,10 $/h*

Godina (1)	*Rashodi ΣNPV [$] (2)*	*NPC [sati] (3)*	*ΣNPC [sati] (4)*	*NPV [$] (5)*	*Prihodi ΣNPV [$] (6)*	*Dobit (η=0,95) [$] (7)=(6)-(2)*	*Dobit (η=0,60) [$] (8)*
1	37.000.000	426.816.000	426.816.000	42.681.600	42.681.600	5.681.600	-10.043.200
2	43.542.056	301.804.488	728.620.488	28.206.027	70.887.627	27.345.571	1.229.077
3	49.656.127	213.408.000	942.028.488	18.639.881	89.527.508	39.871.381	6.887.562
4	55.370.212	150.902.244	1.092.930.732	12.318.118	101.845.626	46.475.414	8.953.341
5	60.710.479	106.704.000	1.199.634.732	8.140.397	109.986.023	49.275.545	8.754.378
6	65.701.382	75.451.122	1.275.085.854	5.379.561	115.365.584	49.664.202	7.161.092
7	70.365.778	53.352.000	1.328.437.854	3.555.069	118.920.653	48.554.875	4.742.003
8	74.725.026	37.725.561	1.366.163.415	2.349.358	121.270.011	46.544.986	1.866.560
9	78.799.090	26.676.000	1.392.839.415	1.552.567	122.822.579	44.023.489	-1.226.934
10	82.606.626	18.862.780	1.411.702.195	1.026.010	123.848.589	41.241.963	-4.386.464
11	86.165.071	13.338.000	1.425.040.195	678.036	124.526.625	38.361.555	-7.516.676
12	89.490.720	9.431.390	1.434.471.586	448.079	124.974.704	35.483.984	-10.559.328
13	92.598.804	6.669.000	1.441.140.586	296.112	125.270.816	32.672.012	-13.480.394
14	95.503.555	4.715.695	1.445.856.281	195.685	125.466.500	29.962.945	-16.261.555
15	98.218.276	3.334.500	1.449.190.782	129.318	125.595.818	27.377.542	-18.894.601
16	100.755.398	2.357.848	1.451.548.628	85.459	125.681.277	24.925.879	-21.377.749
17	103.126.540	1.667.250	1.453.215.878	56.476	125.737.753	22.611.212	-23.713.223
18	105.342.561	1.178.924	1.454.394.802	37.322	125.775.074	20.432.513	-25.905.672
19	107.413.608	833.625	1.455.228.427	24.664	125.799.738	18.386.130	-27.961.142
20	109.349.167	589.46	1.455.817.889	16.299	125.816.037	16.466.871	-29.886.406

Brojčani odnosi iz tablice 6-IV mogu se lakše analizirati i koristiti kod općenitih razmatranja ako se grafički prikažu u dijagramu kao na slici 6.3. Analizom tog grafa može izvesti nekoliko interesantnih zaključaka koji će pomoći kod provjere raznih strategija za određivanje cijene CPU sata koju pružatelj usluga oblačnog računarstva može ponuditi na tržištu :



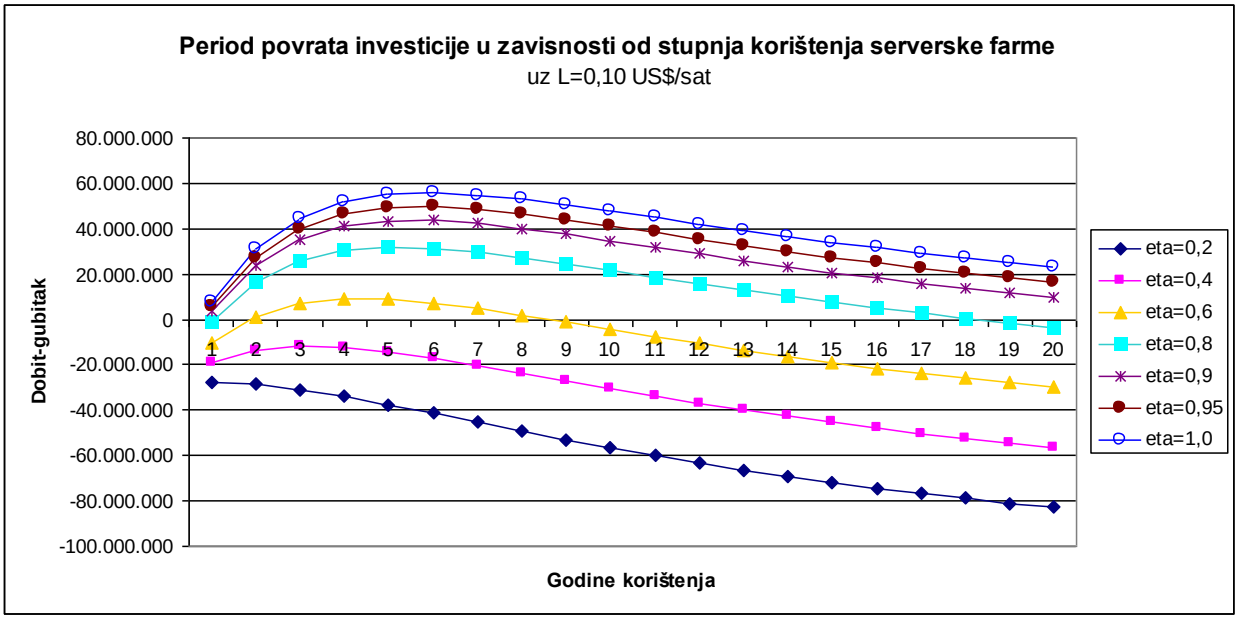
Slika 6.3: ***Prihodi, troškovi i dobit za dva različita stupnja korištenja kapaciteta*** η

- Ukupni rashodi serverske farme rastu tijekom vremena, zavise samo o vrijednosti početnog ulaganja i godišnjim troškovima održavanja, a nisu zavisni o stupnju iskorištenosti instaliranog računalnog kapaciteta.

- Ukupni prihodi serverske farme, uz konstantnu cijena najma CPU vremena (za izračun u tablici 6-III i na slici 6.2 to je $L=0,10\ \$/h$) snažno zavise o stupnju iskorištenosti instaliranih servera. Uz $\eta=0,60$ dobit u prvoj godini je negativna, pozitivna od druge do kraja osme godine te ponovno negativna od devete godine nadalje. Uz $\eta=0,95$ dobit je bitno veća i pozitivna za cijelo razdoblje promatranja, iako postupno opada nakon šeste godine rada serverske farme.

- Povrat investicije (*return of investment* – ROI) najviši je na kraju četvrte godine rada softverske farme uz $\eta=0,60$, a nakon šeste godine uz $\eta=0,95$. Ako se koeficijent ROI računa po izrazu: $ROI=(Ukupni\ prihod\ od\ ulaganja\text{-}Ukupni\ troškovi\ ulaganja)/Ukupni\ troškovi\ ulaganja$ onda najveće vrijednosti iznose: $ROI_{\max 0,6} = 0,162$ i $ROI_{\max 0,95} = 0,756$ ukupni prihod za račun u točkama 4 i 6 označen je u tablici 6-IV plavo, a vrijednosti nazivnika crveno). Prvo ulaganje je nerentabilno jer za 4 godine ostvaruje doprinos svega 8.953.341 $\$$ na dotad ukupno uložena sredstva od 55.370.212 $\$$ odnosno 16,2% ukupno ili oko 4% godišnje. Naprotiv, drugo ulaganje je sasvim rentabilno jer za 6 godina daje 75,6% na uložena sredstva, to jest oko 12,6% godišnje.

Cjeloviti pregled ostvarive dobiti (a to znači i podatke za optimalni period povrata investicije) u zavisnosti o stupnju korištenja servera η, uz pretpostavljenu cijenu najma CPU vremena od $L=0{,}10\ \$/h$, moguće je dobiti iz slike 6.4, za koju su vrijednosti također izračunate iz matematičkog modela u prilogu 6.2.

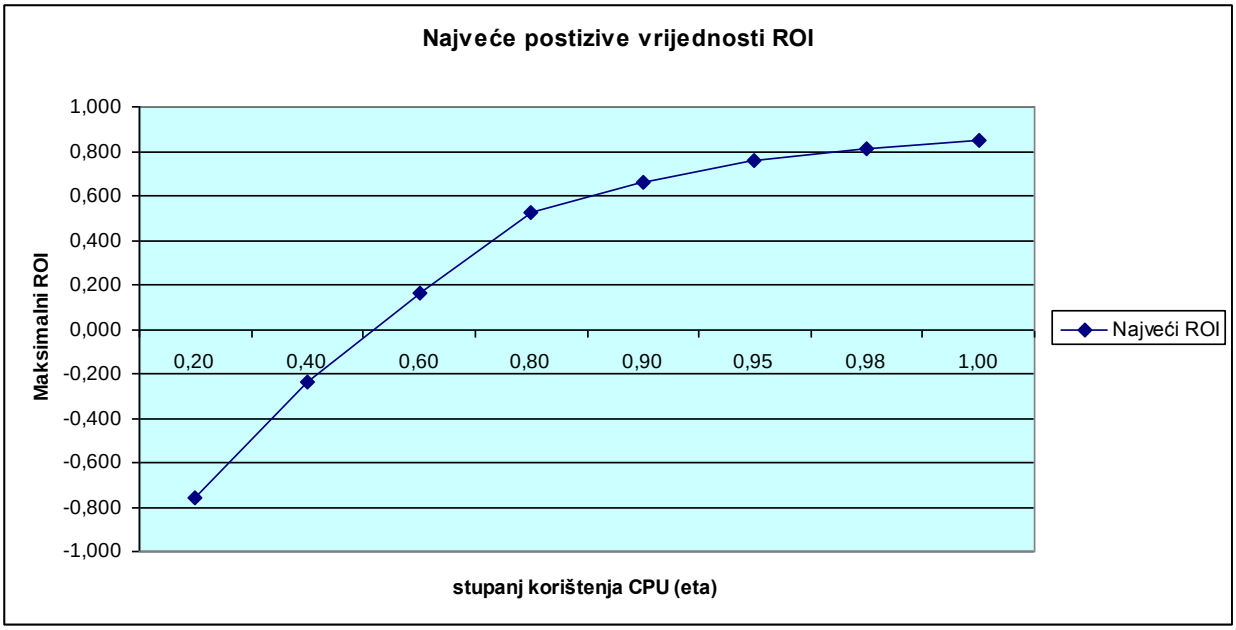


Slika 6.4: ***Optimalni period povrata investicije za razne η serverske farme***

Podatke iz slike 6.4 moguće je promatrati tako da se izračuna maksimalna vrijednost povrata investicije ROI za široki raspon vrijednosti stupnja korištenja CPU (to jest od $\eta=0{,}2$ do $\eta=1{,}0$) . Proračunom prema modelu iz priloga 6.2 dobiva se kretanje vrijednosti najvećeg ostvarivog ROI u zavisnosti od stupnja korištenja CPU, što je prikazano slikom 6.4.

U interpretaciji vrijednosti sa slike 6.4 treba imati u vidu da se maksimalna vrijednost za ROI kod raznih stupnjeva iskorištenja **ne postiže** nakon istog broja godina rada serverske farme. Ta godina kada se postiže maksimum se može očitati s grafa na slici 6.5. Tako se, na primjer, za $\eta=0{,}95$ maksimalni iznos ROI postiže nakon 6 godina rada serverske farme, a za $\eta=0{,}60$ nakon 4 godine. Naravno, ROI za $\eta=0{,}60$ je bitno manji u svakoj godini pojedinačno od onog koji se postiže uz viši stupanj iskorištenja.

Na temelju grafova na slici 6.4 i 6.5 (za koje je izveden proračun na temelju pretpostavke da je cijena najma CPU vremena $L=0{,}10\ \$/h$) može se zaključiti da bi iskorištenje CPU-a s manje od 40% donijelo gubitak za investitora, iskorištenje CPU između 40 i 70% daje pozitivan ali ekonomski neisplativ ROI (odnosno da bi se veći prinos na investirani kapital moga ostvariti ako se isti kapital oroči u komercijalnim bankama), dok iskorištenost CPU od 95% omogućava postizanje sasvim prihvatljiv povrat uloženog kapitala.



Slika 6.5: ***Najveće ostvarive vrijednosti za ROI u zavisnosti o*** η

Na osnovu prethodnih izračuna može se zaključivati da je za ekonomski uspješno djelovanje podatkovnog centra, koji želi pružati usluge oblačnog računarstva, bitno da njegovi CPU resursi budu što više iskorišteni, najmanje s $\eta>0,95$. Uz pretpostavku da su početna C_0 i godišnja ulaganja C_T (navedena u tablici 6-II) za ovaj numerički primjer dobro procijenjena, može se predvidjeti da će uspješno poslovati oni pružatelji usluga oblačnog računarstva koji će uspijevati u potpunosti iskoristiti instalirane serverske kapacitete. To je glavni razlog zašto se na serverskim farmama koriste takozvani *load balancing* programi. Sadašnja cijena najma CPU vremena od 0,10 američkih dolara po CPU satu omogućava dobru zaradu onima čiji su serveri u visokoj mjeri iskorišteni. Obrnuto, ako je stupanj iskorištenja servera nizak (ali je vjerojatno ipak $\eta>0,60$) onda je cijena najma od 0,10 američkih dolara po CPU satu još uvijek promotivna i usmjerena na to da privuče nove korisnike koji će povećati iskoristivost servera i donijeti zaradu u idućim godinama. Može se stoga predvidjeti da će u budućnosti uspješno poslovati samo oni pružatelji usluga koji će na vrijeme usklađivati svoja ulaganja u serversku opremu s povećanjem potražnje CPU vremena i pri tome, na tehničkoj razini, pružati svoje usluge uz maksimalni stupanj iskorištenja servera.

Na kraju ove analize treba naglasiti da cijeli postupak ima značaj modela te da su zaključci, izvedeni iz numeričkog izračuna, valjani tako dugo dok su točni podaci iz tablice 6-II. Ovdje izračunati iznosi cijene najma podudarni su s izračunima u prethodnom potpoglavlju. To dakle, znači da svaki korisnik i pružatelj usluga oblačnog računarstva može izračunati cijene CPU sata prema ovdje prikazanom modelu, ali da prije toga mora provjeriti numeričke vrijednosti iz tablice 6-II, predvidjeti eventualne dodatne specifične troškove i uzeti u račun onu amortizacijsku stopu za opremu i zgrade koja odgovara njegovoj poslovnoj politici.

U nastavku ovog poglavlja bit će prikazan slijed koraka kojeg treba provesti da bi se, u svakom pojedinačnom slučaju, objektivno izabrala ona od opcija kupnje ili najma, koja je povoljnija za pružatelja odnosno korisnika oblačnog računarstva.

6.3 Cijena podatkovne pohrane

Edward Walker s Teksaškog sveučilišta u Austinu je (uz cijenu CPU sata iz prethodnog poglavlja) u radu [109] također proučavao i problematiku troškova podatkovne pohrane. U ovom će potpoglavlju ukratko biti prikazani rezultati njegovog istraživanja, u što spadaju i formule od *(6.13)* do *(6.28)*. Ti rezultati će potom biti provjereni i primijenjeni na tri poslovna slučaja koji čine vlastiti doprinos ovog rada.

Dakle, za bilo koje kupljeno osnovno sredstvo može se, prema [109], njegova neto sadašnja vrijednost NPV na kraju svake godine izračunati prema izrazu:

$$NPV_P = \sum_{T=0}^{N} \frac{P_T - C_T^P}{(1+I_K)^T} + \frac{S}{(1+I_K)^N} - E \qquad (6.13)$$

Varijable korištene u gornjoj jednadžbi imaju sljedeća značenja:

- P_T – godišnji prihod koji se ostvaruje radom na kupljenom osnovnom sredstvu u godini T
- C_T^P – godišnji operativni troškovi rada na kupljenom osnovnom sredstvu u godini T
- I_K – troškovi kapitala, to jest kamata na novac koji je utrošen za kupnju osnovnog sredstva
- N – životni vijek osnovnog sredstva
- S – preostala vrijednost osnovnog sredstva nakon N godina
- E – kupovna vrijednost osnovnog sredstva

Slično tome, sve buduće isplate radi najma osnovnog sredstva potrebno je iskazati kroz NPV prema izrazu:

$$NPV_L = \sum_{T=0}^{N} \frac{P_T - C_T^L}{(1+I_K)^T} - \sum_{T=0}^{N} \frac{L_T}{(1+I_R)^T} \qquad (6.14)$$

Varijable u izrazu *(2)* imaju sljedeće značenje:

- C_T^L – godišnji operativni troškovi unajmljenog osnovnog sredstva u godini T
- L_T – najamnina isplaćena u godini T
- I_R – kamata na novac isplaćen za najam

Usporedbom vrijednosti iz izraza *(6.14)* i *(6.15)* svaki investitor može ocijeniti da li je za njega povoljnije ako neko osnovno sredstvo kupi u vlasništvo ili pak iste potrebne kapacitete osigura unajmljivanjem. Odluka će se temeljiti na sljedećem kriteriju: ako je razlika $\Delta NPV \geq 0$ onda treba kupiti, ako je $\Delta NPV<0$ unajmiti, pri čemu je:

$$\Delta NPV = NPV_P - NPV_L \tag{6.15}$$

Navedeni izraz zapravo predstavlja razradu nejednadžbe iz poglavlja 4, odlomka Optimalna nabava. Ako se primijeni na diskovne memorije kao osnovno sredstvo, onda se uvrštavanjem formula *(6.13)* i *(6.14)* u izraz *(6.15)* te sređivanjem (uz pretpostavku da se jednaka zarada može ostvariti bez obzira da li je diskovni prostor stečen kupnjom ili nabavom) dobije:

$$\Delta NPV = \sum_{T=0}^{N} \frac{C_T^L - C_T^P}{(1+I_K)^T} + \frac{S}{(1+I_K)^N} + \sum_{T=0}^{N} \frac{L_T}{(1+I_R)^T} - E \tag{6.16}$$

Način kupnje diskovnih memorija ima neke specifičnosti u odnosu na uobičajenu kupnju drugih osnovnih sredstava ili uređaja:

- Za više diskova, pojedinačnog kapaciteta Ω koristi se jedna upravljačka jedinica čiju nabavnu cijenu C treba posebno iskazati.
- Diskovi se ne moraju kupovati odjednom već postupno, u skladu s porastom potreba za memorijskim kapacitetom.
- Diskove treba povremeno zamjenjivati stoga što se kvare ili sustavno, nakon određenog broja sati rada, kako bi se spriječili pogonski kvarovi.

Ove specifičnosti dopunjuju klasični model određivanja neto sadašnje vrijednosti nekog ulaganja. Stoga se, prema [109], sadašnja neto vrijednost kapitala E uloženog u kupnju diskova izračunava na sljedeći način:

$$E = \frac{E_T}{(1+I_K)^T} + C \qquad \text{gdje je:} \qquad E_T = \left(\left(\lceil V_T \rceil_\Omega - \lceil V_{T-1} \rceil_\Omega\right) \cdot \Omega + R_T\right) \cdot G_T \tag{6.17}$$

Oznake uvedene u izraz *(6.17)* imaju sljedeće značenje:

- Ω – kapacitet diska [GB]
- V_T – potrebe za memorijskim kapacitetom u godini T [GB]
- $\lceil V_T \rceil_\Omega$ – operator koji računa najmanji broj diskova nominalnog kapaciteta Ω za pokriće potrebnog memorijskog kapaciteta V_T
- G_T – predviđena cijena diskovne memorije u godini T [$/GB$]
- R_T – diskovni memorijski kapacitet koji se mijenja u godini T [GB]

- C – cijena upravljačke jedinice za diskove

Nakon supstitucije vrijednosti za E_T, podrazumijevajući da godišnje ulaganje iskazano izrazom *(6.17)* treba također sumirati za svih N godina, izraz *(6.16)* dobiva oblik:

$$\Delta NPV = \sum_{T=0}^{N} \frac{C_T^L - C_T^P - E_T}{(1+I_K)^T} - C + \frac{S}{(1+I_K)^N} + \sum_{T=0}^{N} \frac{L_T}{(1+I_R)^T} \qquad (6.18)$$

Godišnje operativne troškove za **kupljene** diskovne memorije C_T^P iz izraza *(6.13)* čine troškovi električne energije potrebne za pogon diskova i njihovih upravljačkih jedinica te troškovi rada osoblja vezanih za pogon memorijskih kapaciteta. Ako se podrazumijeva da su memorijski kapaciteti stalno u pogonu, onda su operativni troškovi pogona određeni sljedećim izrazom:

$$C_T^P = (365 \cdot 24) \cdot \delta \cdot (P_C + P_D \cdot \lceil V_T \rceil_\Omega) + \alpha \cdot H_T \qquad (6.19)$$

Godišnji operativni troškovi za **unajmljene** diskovne memorije uključuju samo troškove osoblja pa se može računati da je:

$$C_T^L = \beta \cdot H_T \qquad (6.20)$$

Varijable u izrazima *(6.19)* i *(6.20)* imaju sljedeće značenje:

- δ – cijena električne energije [$\$/kWh$]
- P_C – snaga upravljačke jedinice za diskove [kW]
- P_D – snaga po diskovnoj jedinici [kW]
- H_T – troškovi osoblja podatkovnog centra u godini T [$\$$]
- α – udio troškova osoblja za pogon kupljenih diskovnih kapaciteta
- β – udio troškova osoblja za pogon unajmljenih diskovnih kapaciteta

Ako se u izraz *(6.18)* uvrste vrijednosti iz *(6.19)* i *(6.20)* te istovremeno uvede supstitucija $\rho = \alpha - \beta$ onda se ΔNPV može prikazati na sljedeći način:

$$\Delta NPV = \sum_{T=0}^{N} \frac{-\rho \cdot H_T - 8760 \cdot \delta \cdot (P_C + P_D) \cdot \lceil V_T \rceil_\Omega - E_T}{(1+I_K)^T} - C + \frac{S}{(1+I_K)^N} + \sum_{T=0}^{N} \frac{L_T}{(1+I_R)^T} \qquad (6.21)$$

U prethodnom izrazu razlikuju se kamata na financijska sredstva utrošena za kupnju diskovnih kapaciteta (I_K) od kamata na sredstva koja će se trošiti za najam (I_R). U radu [109] autori se pozivaju na ekonomski zakon poznat kao *Law of One Price*[14] [56] te predlažu da se obje vrijednosti zamijene jednom, proporcionalnom nekoj relativno čvrstoj kamatnoj stopi. Takva je, na primjer, eskontna stopa koju primjenjuju nacionalne banke pojedinih zemalja:

[14] Zakon jedinstvene cijene glasi: „*Na djelujućem tržištu istovrsne robe imaju jedinstvenu cijenu*“.

$I_K=I_R=I_F$. U tom se slučaju prethodni izraz pojednostavljuje (jer nazivnici dvaju članova izraza *(6.21)* nad kojima se provodi sumiranje postaju isti) i glasi:

$$\Delta NPV = \sum_{T=0}^{N} \frac{-\rho \cdot H_T - 8760 \cdot \delta \cdot (P_C + P_D) \cdot \lceil V_T \rceil_\Omega - E_T + L_T}{(1+I_F)^T} - C + \frac{S}{(1+I_F)^N} \qquad (6.22)$$

U prethodnim formulama još je ostala nerazjašnjena zakonitost po kojoj se očekuje promjena cijene diskovne memorije G_T tijekom budućih T godina. Trenutne cijene diskova mogu se naći na web stranici http://www.pricewatch.com, ali za vremensku analizu potrebno je predvidjeti promjenu te cijene tijekom vremena. U već spomenutom radu [109] autori su prikazali rezultate studije u kojoj su oni tjedno, tijekom više od 6 godina počevši od 20. travnja 2003. godine, prikupljali podatke o najnižim cijenama SATA (*Serial Advanced Technology Attachment*) diskova, neovisno o proizvođaču i modelu. Ako se ti podaci preračunaju na cijenu za 1 GB memorije, onda se vidi da ona zavisi samo o proteklom vremenu, što se može aproksimirati funkcijom:

$$G_T = K \cdot e^{-0,438T} \qquad (6.23)$$

U prethodnom izrazu je K najniža cijena za 1 GB diskovne memorije po kojoj se disk mogao kupiti u trenutku $T=0$. Autori rada [109] zaključuju da se cijena za 1 GB diskovne memorije u budućnosti može prognozirati ekstrapolacijom vrijednosti koje se dobiju izračunavanjem po izrazu *(6.23)*. Kod izračunavanja po tom G_T će dati buduću cijenu za 1 GB diskovne memorije, ako nezavisna varijabla T predstavlja broj godina koji je protekao od trenutka $T=0$ za koji je poznata cijena K. Pojednostavnjeno, kod prognoziranja buduće cijene uzima se da je $T=0$ kraj godine koja prethodi početku korištenja diskovne memorije, a K cijena diskovne memorije u tom trenutku.

U studiji koju su proveli Schroeder i Gibson [85] pokazano je da se u velikim podatkovnim centrima godišnje zbog kvara (ili prevencije kvara) mijenja od 0,5% do 13,5% ukupnog broja trenutno raspoloživih diskova, a da je najveća vjerojatnost da to bude oko 3%. Stoga se varijabla R_T u izrazu *(6.17)*, koja predstavlja broj novih diskova koje treba svake godine kupiti radi zamjene, može odrediti po formuli:

$$R_T = 0,03 \cdot \Omega \cdot \lceil V_T \rceil_\Omega \qquad (6.24)$$

Uvažavajući zakonitosti određene izrazima *(6.23)* i *(6.24)* konačno se može napisati vrijednost za E_T iz izraza *(6.17)* u obliku:

$$E_T = \left(1,03 \cdot \lceil V_T \rceil_\Omega - \lceil V_{T-1} \rceil_\Omega\right) \cdot \Omega \cdot K \cdot e^{-0,438T} \qquad (6.25)$$

Na kraju ostaje još utvrditi kakvu bi tržišnu vrijednost mogao imati diskovni pogon nakon N godina rada. Ta vrijednost se umanjuje zbog dva čimbenika – prvi je očekivano buduće smanjenje nabavne cijene zbog stalnog tehnološkog napretka u proizvodnji diskovnih

memorija, a prema zakonu koji je opisan izrazom *(6.23)*. Drugi razlog za smanjenje proizlazi iz činjenice da korišteni disk može postići samo manji dio one cijene koju će u tom trenutku imati novi disk istog kapaciteta. Ovaj drugi čimbenik predstavlja procjenu, a neka bude obuhvaćen faktorom γ koji ima vrijednost $[0...1]$. Uključujući oba ova čimbenika može se za vrijednost S iz izraza *(6.13)* napisati:

$$S = \gamma \cdot \Omega \cdot \lceil V_T \rceil_\Omega \cdot K \cdot e^{-0,438T} \tag{6.26}$$

Na taj način se izraz *(6.22)* konačno može napisati u obliku:

$$\Delta NPV = \sum_{T=0}^{N} \frac{C_T - E_T + L_T}{(1+k)^T} + \frac{S}{(1+k)^N} - C \tag{6.27}$$

U gornjem izrazu varijable S, C_T i E_T imaju sljedeća značenja:

- S – preostala tržišna vrijednost diska nakon T godina, što je definirano izrazom *(6.26)*
- C_T – godišnji operativni troškovi u godini T koji obuhvaćaju prva dva pribrojnika u nazivniku onog dijela izraza *(10)* koji se sumira, odnosno vrijedi:

$$C_T = -\rho \cdot H_T - 8760 \cdot \delta \cdot (P_C + P_D) \cdot \lceil V_T \rceil_\Omega) \tag{6.28}$$

- L_T – najamnina za diskove isplaćena u godini T
- E_T – troškovi kapitala uloženog u godini T, što se izračunava prema izrazu *(6.25)*
- C – cijena upravljačke jedinice za diskove

Na taj je način konačno formulama *(6.25)* do *(6.28)* definiran cijeli matematički model za procjenu koja je opcija – kupnja ili najam potrebnog diskovnog kapaciteta – povoljnija za korisnika.

Radi lakšeg računanja su u tablici 6-V još jednom sistematizirani svi parametri koji se pojavljuju u cjelovitom modelu, s tim da su posebno istaknute mjerne jedinice s kojima se vrijednosti tih parametara moraju uvrštavati u formule koje čine taj model. Cijeli ovaj model ugrađen je u Excel-program koji se nalazi u prilogu 6.3.

Za provjeru ispravnosti modela odlučivanja o kupnji ili najmu diskovnih kapaciteta provedeni su proračuni za tri poslovna slučaja. Ideje za primjere su preuzete iz rada [109], ali su podaci za proračun prilagođeni hrvatskim prilikama te upisani u stupce 4 do 6 tablice 6-V.

Tablica 6-V: ***Parametri modela za proračun vrijednosti*** **ΔNPV**

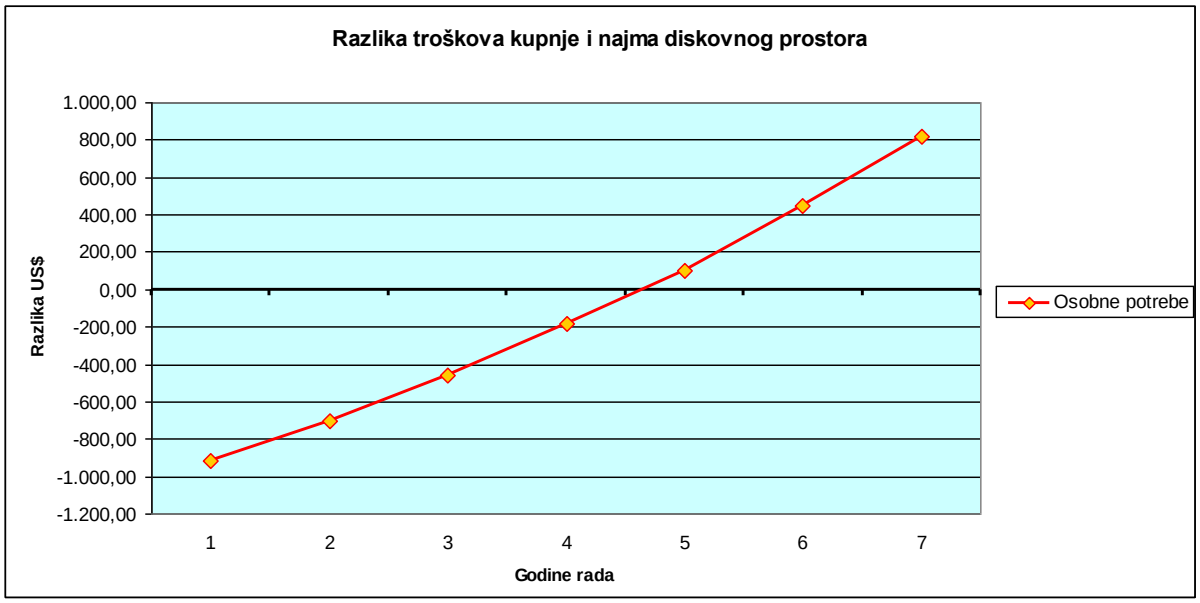
Oznaka (1)	*Opis parametra* (2)	*Jedinica mjere*[15] (3)	*Podaci za ogledne proračune*		
			Primjer 1 (4)	*Primjer 2* (5)	*Primjer 3* (6)
δ	Cijena električne energije	*[$/kWh]*	0,05	0,05	0,05
Ω	Kapacitet diska	*[GB]*	500	1.000	10.000
ρ	Razlika troškova rada osoblja (α-β)	*[relativni broj]*	0	0,5	0,5
γ	Umanjenje tržišne vrijednosti starog diska	*[relativni broj]*	0,2	0,2	0,2
C	Cijena upravljačke jedinice za diskove	*[$]*	500	1.000	2.000
H_T	Godišnji troškovi osoblja	*[$]*	-	50.000	75.000
I_F	Kamatna stopa na novčana sredstva	*[$/$]*	0,08	0,08	0,08
K	Cijena diskovnog kapaciteta u T=0	*[$/GB]*	0,30	0,30	0,30
L_T	Godišnji iznos najma diskovnog kapaciteta	*[$/GB]*	1,8	1,8	1,8
P_C	Snaga upravljačke jedinica za diskove	*[kW]*	0,50	0,70	1,00
P_D	Snaga koju treba diskovna jedinica	*[kW]*	0,01	0,01	0,01
V_T	Potrebni diskovni kapacitet po godinama T	*[GB]*	100	1.000	10.000

Poslovni slučaj 1

Analizira se rad pojedinačnog korisnika koji osobno koristi vlastito računalo pa zbog toga nema razlike u troškovima rada, bez obzira koristi li se kupljeni ili unajmljeni diskovni kapacitet (to jest vrijedi $\rho=0$). Procjenjuje se da se njegove potrebe za diskovnom memorijom svake godine povećavaju za 100 GB. Podrazumijeva se da će takav korisnik na početku rada kupiti disk od 500 GB čija je trenutna cijena oko 150 $\$$ (to jest oko 0,30 $\$/GB$) te upravljačku jedinicu za disk uz cijenu od oko 500 $\$$. U slučaju najma uzima se da je trenutna cijena (prema ponudama usluga oblačnog računarstva) 0,15 $\$/GB$ mjesečno (to jest 1,8 $\$/GB$ godišnje). Ostali podaci za proračun prikazani su u stupcu 4 tablice 6-V.

Rezultati proračuna prikazani su na slici 6.6 gdje se vidi da je $\Delta NPV<0$ za prve četiri godine. Prema polaznom kriteriju za odabir povoljnije politike investiranja određenom izrazom *(6.15)*, može se zaključiti da je opcija najma za ovog korisnika povoljnija tijekom prve četiri godine poslovanja. Ako bi se opisani posao radio na isti način više od četiri godine, kupnja bi bila bolja opcija. Obzirom da je pretpostavka o istom poslu na istoj informatičkoj tehnologiji malo vjerojatna za razdoblje dulje od 4 godine, najam diskovnog prostora u oblaku definitivno je preporučljivo rješenje za ovaj tip korisnika. Ovaj zaključak je očekivan jer bi troškove vlasništva povećala kupnja upravljačke jedinice za diskove na početku korištenja, dok opcija najma smanjuje početne troškove zbog postupnog povećavanja kapaciteta.

[15] Sve novčane vrijednosti u ovom proračunu su izražene u američkim dolarima [$] zato što još nema domaćih pružatelja usluga oblačnog računarstva, a također zbog mogućnosti usporedbe s podacima iz literature.



Slika 6.6: ***Postupni najam diskova za osobnog korisnika je bolji od kupnje***

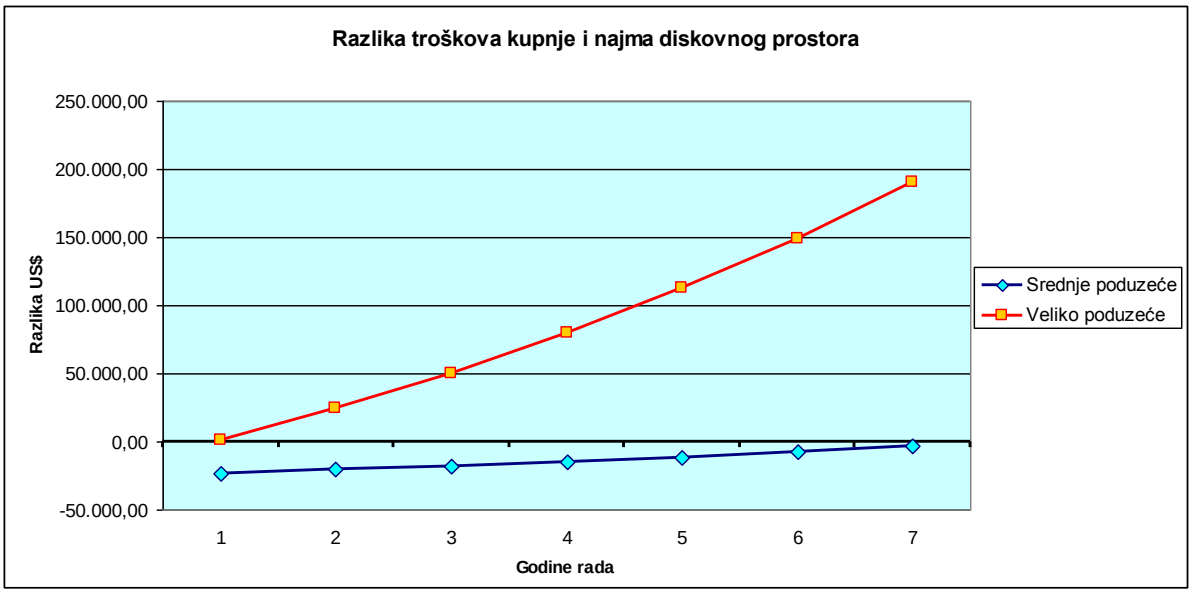
Poslovni slučaj 2

Razmatraju se opcije kupnje ili najma za srednje poduzeće, čije se potrebe za diskovnim prostorom postupno povećavaju za 1 TB godišnje. Prema [93] takva poduzeća koriste 10 do 100 servera u podatkovnom centru, a preko 50% podatkovnih centara u SAD su te veličine. U slučaju kupnje, takvo će poduzeće kupiti disk kapaciteta 3 TB i odgovarajuću upravljačku jedinicu (uz $C=1.000\ \$$) na početku rada, a zatim dokupljivati isti diskovni kapacitet na početku 3. i 6. godine rada. Potrebna snaga za rad takvih uređaja je $P_D = 0{,}02\ kW$ i $P_C = 0{,}7\ kW$, a cijena najma diskovnog prostora ista je kao u prethodnom slučaju (1,8 $\$$ za 1 GB godišnje). Nadalje, procjenjuje se da rad osoblja za nadzor i upravljanje podacima stoji 50.000 $\$$ godišnje, a da se za slučaj najma taj rad može smanjiti za 50% ($\rho=0{,}5$). Ostali podaci za proračun prikazani su u tablici 6-V, stupac 5, a rezultati su prikazani slikom 6.7, krivulja s oznakom "Srednje poduzeće". Očito je da je za takvo poduzeće najam diskovnog prostora najbolja opcija, jer je $\Delta NPV<0$ tijekom svih analiziranih 7 godina rada.

Poslovni slučaj 3

Ovaj se primjer odnosi na vrlo veliko poduzeće čije se potrebe za diskovnim kapacitetom povećavaju za 10 TB godišnje. Za takav kapacitet potrebna je RAID upravljačka jedinica koja može upravljati ukupnim diskovnim kapacitetom do 100 TB, čija je cijena oko 2.000 $\$$, a za pogon treba snagu oko 0,7 kW. Cijena rada osoblja za nadzor i upravljanje podacima procjenjuje se na 50.000 $\$$ godišnje, a u slučaju najma taj se rad može također smanjiti za 50% ($\rho=0{,}5$). Ostali podaci za proračun prikazani su stupcem 6 u tablici 6-V. Rezultati proračuna prikazani su slikom 6.7, krivulja s oznakom "Veliko poduzeće". Za poduzeće s

ovakvim potrebama diskovnog kapaciteta najam diskova nije dobra opcija, jer je $\Delta NPV > 0$ za sve godine rada. Dapače, pozitivni financijski učinak kupnja se povećava tijekom vremena.



Slika 6.7: ***Usporedba kupnje vlastitih diskova i najma za srednje i veliko poduzeće***

6.4 Komentar i zaključak o pribavljanju podatkovnih resursa

Izvedena analiza pokazuje da najam podatkovnih kapaciteta, prema uvjetima koje danas nude pružatelji usluga oblačnog računarstva, nije uvijek najpovoljnije rješenje, već da financijski efekt zavisi o opsegu posla i dinamici porasta potreba za diskovnim prostorom. Općenito bi se mogle dati sljedeće preporuke:

- Najam je pogodnija opcija za individualne korisnike ili vrlo mala poduzeća, ali ako ne traje dulje od tehnološkog vijeka opreme.
- Za mala i srednja poduzeća najam diskovnog prostora je uvijek financijski povoljniji od ulaganja u vlastite diskovne kapacitete.
- Za velika poduzeća i druge organizacije s vrlo velikim potrebama diskovnog kapaciteta kupnja opreme je daleko povoljnija opcija od najma.

Naravno, brojčane rezultate izračuna ne treba smatrati trajnom istinom. Izračunata vrijednost za ΔNPV zavisi od mnogo parametara pa rezultati vrijede tako dugo dok parametri imaju vrijednosti koje su prikazane u tablici 6-V. Međutim, provedena analiza potvrđuje da svaka organizacija, uz pomoć ovdje prikazane metodologije, uvijek može objektivno izabrati za nju povoljniji način osiguranja potrebnih diskovnih kapaciteta.

6.5 Taksonomija i ekonomika oblačnih usluga

Uvodna razmatranja o taksonomiji oblačnih usluga napisane su u poglavlju 3, u objašnjenju servisnih paradigmi. U poglavlju 4 uspoređene su komercijalne ponude tri najpoznatija pružatelja usluga oblačnog računarstva: Microsofta, Googlea i Amazona. Ovo poglavlje predstavlja sintezu tih prethodnih razmatranja s ciljem postavljanja temelja za konačno razrješavanje dvojbe "najam ili kupnja" računalnih resursa.

Kako je obrazloženo na početku ovog poglavlja, komercijalne ponude ukratko opisane u četvrtom poglavlju predstavljaju poslužiteljev pogled na troškove oblačnog računarstva koji je, naravno, zanimljiv i potencijalnom korisniku jer će on, prema cijenama koje proizlaze iz tog proračuna, na kraju donositi odluke o tome hoće li unajmiti ili kupiti potrebne računalne resurse. Međutim, s korisničke strane ipak je važnije dobro poznavati aplikacijska svojstva koja utječu na potrošnju oblačnih resursa. Za rješavanje dvojbe kupnja ili najam potrebno je napraviti taksonomiju oblačnih usluga (IaaS, PaaS i SaaS) na primjeru poslužitelja koji su već analizirani u poglavlju 3. Nakon toga bit će moguće definirati opće zakonitosti o načinu izračuna cijena za istovrsne tipove usluga, što će omogućiti potencijalnim korisnicima tih usluga donošenje odluka na temelju objektivnih kriterija. Stoga će se ovdje najprije detaljno analizirati komponente usluga koje nude tipični poslužitelji, što će dalje omogućiti formaliziranje strukture cijena. Potom će te usluge biti parametrizirane na način da im se pridruže one aplikacijske značajke koje su bitne za izračun potrebnih računalnih resursa, a koje su detaljno istražene već ranije u poglavlju 5: snaga računala $\boldsymbol{p}$, broj angažiranih računala $\boldsymbol{r}$, složenost aplikacijskog algoritma $\boldsymbol{a}$, broj CRUD akcija $\boldsymbol{q}$, veličina baze podataka $\boldsymbol{b}$ te količine podataka poslanih u oblak radi obrade $\boldsymbol{u}$. Naposljetku će se izvesti formule za izračun troškova korištenja oblačnih usluga svakog poslužitelja.

6.5.1 Oblačni poslužitelji i njihove usluge

Radi izgradnje cjelovite taksonomije usluga oblačnog računarstva u ovom su radu promatrane oblačne usluge Microsofta, Googlea i Amazona. Postoje i drugi oblačni poslužitelji navedeni u [74] koji pružaju većinu ili neke od oblačnih usluga te se uz već navedene prema [91] pojavljuju i drugi: Savvis, VMware, Rackspace, Verizon, GoGrid, Force.com (ogranak Salesforce.com) i AppNexus. Pregledom ponuda svakog od njih utvrđeno je da se svaka od nuđenih komponenata može naći u portfelju Microsofta, Googlea ili Amazona. Stoga bi uključivanje njihovih ponuda u ovu analizu bilo preopširan poduhvat za ovaj rad, koji ne bi dao neka šira saznanja od onih koja se mogu steći promatranjem ponuda tipskih pružatelja usluga. Izbor tipskih pružatelja usluga ograničen je na one koji:

- nude najveći raspon usluga,
- imaju karakteristične cijene i način obračuna tih usluga i

- koriste karakteristične tehničke platforme ili njihovu kombinaciju.

Amazon, Microsoft i Google po navedenim su kriterijima dobri za daljnju analizu jer:

- Svima se ROJ (opisan u poglavlju 4) temelji na količini procesora zakupljenih virtualnih računala pri čemu ta količina implicira i ostale hardverske karakteristike.
- Svi nude nerelacijsku pohranu (strukturiranu i nestrukturiranu) te pristup toj pohrani na REST načelima, izravno ili kroz softverske omotače.
- Svi obračunavaju količinu podataka koja ulazi u oblak i izlazi iz njega.
- Svi nude, u sklopu PaaS-a, razvojne alate za svoju platformu.

Microsoftov Azure oblak temelji se na Microsoftovim tehnologijama (Windows Server, SQL Server, IIS) i kao takav predstavlja cjelovitu tehničku platformu. Potrebno je još jednom napomenuti da to ne znači kako je u Azureu podržan softverski razvoj samo u Microsoftovim tehnologijama, jer Azure SDK postoji i za PHP, Javu i Ruby. Korištenje Googleovog oblaka temelji se pak na Pythonu i Javi, a virtualna računala u njemu pokretana su prilagođenom verzijom Linuxa. Računala u Amazonovom oblaku mogu biti na Linuxu i na Microsoftovim operacijskim sustavima, a razvojni alati – AWS SDK – postoje za mnoge programske jezike i razvoje platforme, uključivo .NET, Javu i PHP.

U prvi mah može se učiniti čudnim što je iz tih tipičnih oblačnih poslužitelja izostavljen poznati Salesforce.com (odnosno njegov PaaS ogranak Force.com), no više je razloga tome. Naime, Salesforce.com je uspješan SaaS poslužitelj, napose na području CRM aplikacija (*Customer Relationship Management*), no njihova je ponuda u ostalim područjima zamjetno slabija odnosno u začecima. Force.com oblak, u kojemu osim korištenja Salesforce.com aplikacija postoji i mogućnost razvoja vlastitih, je za sada tehnički prilično ograničen (primjerice, pohrana podataka nije mnogo više od obične *flat* liste, izvoz podataka je isključivo u CSV obliku), razvoj se temelji na vlasničkim jezicima, razvojni alati su nedovršeni (pa, na primjer, nema pravog korak-po-korak testiranja radi otklanjanja grešaka). Iako nesumnjivo interesantna za sve one kojima trebaju (uvjetno rečeno) gotova rješenja za jednostavnije poslove, Salesforce.com nije za sada namijenjen onima kojima treba složen i prilagođeni (*customized*) softver, obično razvijen iz početka (*from scratch*).

U poglavlju 3 je mnoštvo "…*as a Service*" paradigmi reducirano na tri osnovne grupe PaaS, IaaS i SaaS. Međutim, iz daljnjeg razmatranja ovdje je također ispuštena i paradigma SaaS, zato jer se odnosi na korištenje gotovih uslužnih programa. Nasuprot tome, glavna je svrha ovog rada usmjerena na korištenje oblačnih resursa za pogon vlastitih rješenja. Zbog toga su u tablici 6-VI sadržane samo one usluge koje se mogu razvrstati na paradigme PaaS (zelena polja) i SaaS (crvena polja). Tablica sadrži zbirne podatke za tri tipska pružatelja usluga

oblačnog računarstva. Za svaku njihovu uslugu u ovoj je sistematizaciji naveden originalni komercijalni naziv i kratko objašnjenje za temeljne funkcionalnosti u zagradi.

Tablica 6-VI: ***Pregled ponuditelja i njihovih usluga po komponentama***[16]

Usluga Poslužitelj	*PaaS*	*IaaS*
Microsoft-Azure (M)	**CDN** (isporuka sadržaja širokom krugu korisnika) **SQL Azure** (relacijska pohrana podataka) **.NET platforma** (izvršavanje korisničkih aplikacija razvijenih u .NET jezicima ili u Javi, PHP-u ili Rubyju) **AppFabric** (servisna sabirnica)	**Compute** (virtualna računala pod Win 2008 64-bitnim OS-om) **Table**, **Blob** i **Queue storage** (nerelacijska pohrana podataka: strukturiranih, blobova i redova poruka)
Amazon (A)	**MapReduce** (paralelna obrada podataka) **CloudFront** (CDN – isporuka sadržaja širokom krugu korisnika) **SimpleDB** (nerelacijski SUBP) **RDS** (relacijska pohrana podataka) **SQS** (komunikacija redovima poruka) **SNS** (obavještavanje o stanjima i događajima) **Flexible Payment Service** (sustav plaćanja) **DevPay** (Amazonov sustav naplate korisničkih e-business rješenja)	**EC2** (virtualna računala pod Windows ili Linux OS-om) **CloudWatch** (dijagnostika usluga i resursa) **AutoScaling** (automatsko skaliranje svih resursa u oblaku) **Elastic Load Balancing** (automatizirani balanser opterećenja) **VPC** (virtualni privatni oblak) **S3** (nerelacijska pohrana podataka) **EBS** (pohrana podataka na datotečnom sustavu)
Google (G)	**AppEngine** (izvršavanje korisničkih aplikacija razvijenih u jezicima Python i Java u razvojnom alatu Eclipse)	**CPU time** (računalno vrijeme) **Storage data** (nerelacijska pohrana podataka) **Recipients e-mailed** (elektronička pošta)

U sljedeće tri tablice (6-VII, 6-VIII i 6-IX) opisane su te usluge mnogo detaljnije po pojedinačnim ponuditeljima. Također su identificirani parametri, u skladu sa značenjima i oznakama koje su uvedene i detaljno obrađene u poglavlju 5, a prema kojima će se odrediti algoritam za izračun cijena njihovog korištenja. Također su, prema potrebi, dodani komentari o povezanosti između usluga istog ponuđača

[16] Valja napomenuti kako su u Amazonov cjenik uključene neke oblačne usluge izostavljene iz *tablice 6-VI* koje se dalje ne razmatraju u ovome radu (Fulfillment Web Service, AWS Import/Export, Amazon Mechanical Turk i Alexa servisi). One nisu vezane za oblačno računarstvo, nego se samo reklamiraju zajedno s drugim uslugama. Takav marketinški pristup odabran je radi uspješnije ponude korisnicima Amazonovog oblaka. FWS se odnosi na fizičko skladištenje, pakiranje, označavanje i prijevoz proizvoda. AWS Import/Export je usluga stvaranja, slanja i pohrane (na trajne medije poput diskova i USB stickova) pričuvnih kopija podataka i aplikacija. Amazon Mechanical Turk je burza složenih poslova koje ljudi rade daleko bolje od računala, primjerice prepoznavanje i razvrstavanje slika. Alexa servisi se odnose na analitiku internetskih stranica i pojavili su se prije računalnih oblaka, a Amazon je stekao vlasništvo nad njima kupnjom tvrtke Alexa.

Tablica 6-VII: ***Katalog Microsoftovih usluga***

Naziv	*Tip usluge*	*Opis funkcionalnosti*	*Parametri cijene*	*Veza s drugim uslugama*
Compute	IaaS	Virtualna računala određene hardverske konfiguracije	Broj računala (r) Snaga računala (p)	
Storage		Nerelacijska pohrana blobova, redova poruka i strukturiranih podataka	Količina podataka (b) Broj transakcija (q) Opseg poslanih podataka (u) Opseg primljenih podataka (d)	Uključeno u uslugu **Compute**, ali se naplaćuje zasebno.
CDN	PaaS	Isporuka sadržaja (*content delivery*)	Količina podataka (b) Broj transakcija (q) Opseg primljenih podataka (d)	
SQL Azure		Pohranjivanje podataka uz relacijski SUBP	Količina podataka (rb) Opseg poslanih podataka (u) Opseg primljenih podataka (d)	Temeljen na bazi podataka SQL Server.
AppFabric		Servisna sabirnica-unificiran način razmjene informacija među aplikacijama radi integracije heterogenih rješenja	Broj zakupljenih konekcija (nc) Broj transakcija (q) Opseg poslanih podataka (u) Opseg primljenih podataka (d)	

Struktura Microsoftovih usluga oblačnog računarstva u sklopu Azure platforme relativno je jednostavna, no istovremeno pokriva vrlo široki spektar funkcionalnosti. Primjetno je da im, u odnosu na konkurentski Amazon, nedostaje podrška za e-trgovinu, ali zato imaju servisnu sabirnicu kao vrlo kompleksnu uslugu koja omogućava korisničku integraciju aplikacijskih rješenja kroz oblak.

Struktura Amazonovih oblačnih usluga (za koje je uobičajen zbirni naziv *Amazon Web Services*-AWS), prikazana je tablicom 6-VIII. Amazon je prvi u svijetu razvio komercijalno oblačno računarstvo te zasad ima najkompleksniju ponudu usluga. Ogromna softverska i hardverska infrastruktura te tvrtke poslužila je kao temelj komercijalizacije prvotno razvijenih internih servisa te tvrtke, iz kojih su kasnije razvijeni AWS kao cjelovita Amazonova oblačna platforma. Za razliku od promatranih (i velike većine drugih) oblačnih poslužitelja, podrška za e-trgovinu (kroz DevPay i FPS) je potpuno funkcionalno rješenje koje je integralni dio Amazonovog računalnog oblaka, a ne konceptualno rješenje koje tek treba praktično razviti.

Također, valja primijetiti da Amazon već sada ima podršku za paralelizaciju obrade podataka kroz uslugu Elastic MapReduce, temeljenu na Hadoop softveru (korisnička izvedba softvera za realizaciju MapReduce algoritma), dok je slična usluga u Azureu tek najavljena. Uz takvu uslugu bi tipsko rješenje (korišteno kod istraživanja u poglavlju 5) bilo znatno jednostavnije stoga što se baza podataka slika za prepoznavanje ne bi morala ručno dijeliti na više disjunktnih podskupova (odvojenih blob-spremnika) nad kojima onda radne role mogu paralelno raditi.

Tablica 6-VIII: ***Katalog Amazonovih usluga***

Naziv	*Tip usluge*	*Opis funkcionalnosti*	*Parametri cijene*	*Veze s drugim uslugama*
EC2	IaaS	Iznajmljivanje virtualnih računala određene hardverske konfiguracije pod operacijskim sustavima Windows ili Linux	Broj angažiranih računala (***r***) Snaga računala (***p***) Operacijski sustav (***os***)	U zakup su uključeni dijagnostički i upravlja-čki alati: CloudWatch, AutoScaling i Elastic Load Balancing.
CloudWatch		Dijagnostika svih usluga zakupljenih na Amazonu	Broj računala (***r***)	Uključeno u EC2.
AutoScaling		Automatsko zaposjedanje i oslobađanje resursa prema opterećenju	Ne naplaćuje se.	Uključeno u EC2. Pretpostavlja korištenje CloudWatch.
Elastic Load Balancing		Automatsko ravnomjerno raspoređivanje zadaća na zaposjednute resurse.	Broj angažiranih računala (***r***) Opseg poslanih podataka (***u***)	Uključeno u EC2.
VPC		Povezivanje lokalnih i oblačnih resursa VPN-om u jedinstveni sustav	Trajanje VPN konekcije (*tc*) Opseg poslanih podataka (u) Opseg primljenih podataka (d)	
S3		Jednostavna, nerelacijska i skalabilna pohrana velikih količina podataka	Količina podataka (b) Opseg poslanih podataka (u) Opseg primljenih podataka (d)	Dostupnost HTTP-om korištenjem REST načela.
EBS		Pohrana podataka izravno na datotečni sustav	Količina podataka (***b***) Broj transakcija (q)	
Elastic MapReduce	PaaS	Paralelna obrada velikih količina podataka	Broj angažiranih računala (r) Snaga računala (p) Operacijski sustav (os)	Za paralelizaciju obrade podataka se koristi se Hadoop okolina.
CloudFront		Isporuka sadržaja (*content delivery*)	Opseg primljenih podataka (d) Broj transakcija (q)	Temeljen na S3 usluzi, a pohrana se naplaćuje po vlastitoj tarifi.
SimpleDB		SUPB za strukturirani pristup podacima pohranjenima u S3.	Količina podataka (b) Opseg poslanih podataka (u) Opseg primljenih podataka (d)	Temeljen na S3 usluzi, ali se naplaćuje neovisno.
RDS		Relacijski SUBP	Broj angažiranih računala (r) Snaga računala (p) Količina podataka (rb) Opseg poslanih podataka (u) Opseg primljenih podataka (d)	Temeljen na MySQL bazi podataka.
SQS		Usluga za razmjenu poruka (*message queues*)	Broj transakcija (q) Opseg poslanih podataka (u) Opseg primljenih podataka (d)	
SNS		Notifikacijska usluga temeljena na *push* mehanizmu za izvješta-vanje o događajima u oblaku.	Broj transakcija (q) Opseg poslanih podataka (u) Opseg primljenih podataka (d)	Mogući formati dojave su e-pošta, HTTP itd.
FPS		Sustav plaćanja	Broj transakcija (q)	
DevPay		Naplatni sustav za korištenje rješenja smještenih u Amazonovom oblaku.	PDV na cijenu naplaćenu krajnjemu korisniku, plus fiksna cijena po fakturi.	Za razvojne tvrtke koje žele vlastita rješenja u iznajmiti trećoj strani.

Googleova ponuda usluga oblačnog računarstva je uvjerljivo najjednostavnija po načinu naplate i raznovrsnosti usluga, a prikazana je tablicom 6-IX. Jedina je od promatranih koja ne naplaćuje računalne resurse po zakupljenoj količini, nego po stvarnom korištenju.

Tablica 6-IX: ***Katalog Googleovih usluga***

Naziv	*Tip usluge*	*Opis funkcionalnosti*	*Parametri cijene*	*Povezanost s drugim uslugama*
CPU time	IaaS	Iznajmljivanje procesorskog vremena uz automatsko zaposjedanje i otpuštanje računala	Broj angažiranih računala (***r***)	Procesorsko mirovanje (*idle*) se ne plaća. Postoji besplatna kvota korištenja.
Storage data		Nerelacijska pohrana podataka.	Količina podataka (***b***) Opseg primljenih podataka (***d***) Opseg poslanih podataka (***u***)	Za dohvat podataka koristi se GQL jezik. Postoji besplatna kvota pohrane slanja i prijema podataka.
Recipients e-mailed	PaaS	Slanje e-poruka.	Broj poslanih poruka (*e*)	
AppEngine		Radno okruženje za pokretanje aplikacija pisanih u Javi ili Pythonu	Besplatno.	

6.6 Struktura cijena usluga oblačnog računarstva

Nakon tipizacije vrsta usluga oblačnog računarstva mogu se postaviti formule za izračun cijene korištenja, koja će biti izražena po tipovima usluge PaaS i IasS. Formule se temelje na parametrima navedenim u tablicama 6-VII do 6-IX, koje primjenjuje svaki od tipskih ponuđača. U tablicama se pojavljuje ukupno 11 različitih parametara za izračun cijena, ali razni ponuđači uključuju u izračun različite parametre. Katalog parametara, s dodatnim obrazloženjima o domeni i naznakom koji ponuditelj ih koristi, prikazan je tablicom 6-X.

Tablica 6-X: ***Pregled parametara za izračun cijene***

	Značenje	*Domena i jedinica mjere*	*Microsoft (M)*	*Amazon (A)*	*Google (G)*
r	*Broj angažiranih računala*	{1,2, …} [kom]*	x	x	x
p	*Snaga angažiranih računala*	Tipske konfiguracije [ ]**	x	x	
u	*Opseg poslanih podataka*	{1,2, …} [kB]	x	x	x
d	*Opseg primljenih podataka*	{1,2, …} [kB]	x	x	x
b	*Količina podataka u bazi*	{1,2, …} [GB]	x	x	x
q	*Broj r/w transakcija*	{10,200} [broj akcija/upit]	x	x	
rb	*Količina podataka u relacijskoj bazi*	{1,2, …} [GB]	x	x	
tc	*Trajanje VPN konekcije*	[sek]		x	
os	*Operacijski sustav*	{Win, Linux, …} [ ]		x	
nc	*Broj zakupljenih konekcija*	{5,100} [ ]	x		
e	*Broj poslanih poruka*	{1,2, …} [ ]			x

Napomene: * Kod Googlea je to broj CPU
** Za bezdimenzionalne veličine i omjere koristi se [ ] kao oznaka za jedinicu mjere.

Utjecaj parametara $\{\boldsymbol{r},\boldsymbol{p},\boldsymbol{u},\boldsymbol{d},\boldsymbol{b},\boldsymbol{q}\}$ detaljnije je istražen u prethodnom poglavlju i oni su u prethodnoj tablici otisnuti podebljano[17].

Cilj ovih postupaka jest uvesti cjenovnu komponentu u metodiku izgradnje hibridnog IS-a, čije se neke komponente izvode u oblaku. Oznake u formulama za cijene bit će slijedeće:

- Cijena će se izražavati u obliku $C_{P,U}$ gdje P u indeksu znači kraticu za ponuditelja oblačne usluge tako da vrijedi $P \in \{A,M,G\}$ redom za Amazon, Microsoft i Google, dok se indeks U odnosi na tip oblačne usluge, odnosno $U \in \{P,I\}$ za PaaS ili IaaS;
- Dijelovi ukupne cijene izražavat će se u obliku $P_x \cdot y$. U toj formi je P_x jedinična cijena x-te komponente P-tog ponuđača, gdje se x odnosi na parametre cijene tako da je $x \in \{\boldsymbol{r},\boldsymbol{p},\boldsymbol{u},\boldsymbol{d},\boldsymbol{b},\boldsymbol{q},rb,tc,os,nc,e\}$ i $P \in \{A,M,G\}$, a y je broj jediničnih usluga koje se obračunavaju prema jednom od parametara cijene, tako da također (kao i za x) vrijedi: $y \in \{\boldsymbol{r},\boldsymbol{p},\boldsymbol{u},\boldsymbol{d},\boldsymbol{b},\boldsymbol{q},rb,tc,os,nc,e\}$.

Ovdje valja još jednom naglasiti da se x i y odnose na parametre koji ulaze u formule za cijene oblačnih usluga, pri čemu se sa x označavaju parametri, a y znači broj jediničnih usluga označenih tim parametrom, tako da vrijedi $x \cong y \in \{\boldsymbol{r},\boldsymbol{p},\boldsymbol{u},\boldsymbol{d},\boldsymbol{b},\boldsymbol{q},rb,tc,os,nc,e\}$.

Formule su za sva tri oblačna poslužitelja, izvedene iz prethodnih razmatranja, upisane su u odgovarajuća polja tablice 6-XI. Formule su izrađene na temelju vlastite analize i matematičkog uopćavanja podataka iz cjenika tih komercijalnih oblačnih poslužitelja.

Tablica 6-XI: ***Formule za izračun cijene oblačnog zakupa***

	IaaS	***PaaS***
Amazon	$C_{A,I} = A_r \cdot r \cdot p \cdot A_{os} + A_u \cdot u + A_d \cdot d + A_q \cdot q + A_{tc} \cdot tc + A_b \cdot b$	$C_{A,P} = A_r \cdot r \cdot p \cdot A_{os} + A_u \cdot u + A_d \cdot d + A_q \cdot q + A_{tc} \cdot tc + A_b \cdot b + A_{rb} \cdot rb$
Microsoft	$C_{M,I} = M_r \cdot r \cdot p + M_u \cdot u + M_d \cdot d + M_q \cdot q + M_b \cdot b$	$C_{M,P} = M_r \cdot r \cdot p + M_u \cdot u + M_d \cdot d + M_q \cdot q + M_b \cdot b + M_{rb} \cdot rb + M_{nc}.nc$
Google	$C_{G,I} = G_r \cdot r + G_u \cdot u + G_d \cdot d + G_b \cdot b$	$C_{G,I} = G_r \cdot r + G_u \cdot u + G_d \cdot d + G_b \cdot b + G_e \cdot e$

Za neke usluge, elementi platforme se ne zaračunavaju posebno jer ih ponuditelj tretira kao fiksne troškove pokrivene prihodom od mjerljivih usluga. To je slučaj za Google i Azure u, primjerice, pogledu čitanja podataka iz nerelacijske pohrane, koje može biti izvedeno elementarnim REST pristupom (*Representational State Transfer* - dohvat podataka putem web-servisa, ali se podaci ne vraćaju u SOAP-omotaču već u nekom drugom strukturiranom formatu, primjerice JSON), ali i elegantnije, kroz omotače besplatne softverske platforme.

[17] Činjenica da se u *tablici VI-X* pojavljuju još neki parametri nipošto nije u suprotnostima sa zaključcima u poglavlju 5 već to samo znači da razni ponuđači u izračun cijene uključuju još neke parametre koji nisu od značaja za vrijeme izvođenja aplikacije u oblaku.

6.7 Razrješavanje dvojbe kupnja ili najam

U ovom dijelu rada pokazat će se valjanost kompletnog postupka za izračunavanje cijene CPU vremena i cijene podatkovne pohrane, na temelju koraka razvijenih u potpoglavljima 6.2 i 6.3. Izračuni će biti primijenjeni na tri tipske aplikacije čija su svojstva opisana parametrima koji su detaljno istraženi u poglavlju 5. Tamo je u izrazu *(5.10)* utvrđeno da vrijeme odziva T aplikacije koja se izvodi u oblaku zavisi o četiri relevantna parametra: složenost algoritma $\boldsymbol{a}$, opseg poslanih podataka $\boldsymbol{u}$, broj CRUD akcija po upitu $\boldsymbol{q}$ i broj angažiranih računala $\boldsymbol{r}$. Također je na kraju poglavlja 5 pokazano da tri parametra, $\boldsymbol{a}$, $\boldsymbol{u}$ i $\boldsymbol{q}$ opisuju posao (to jest aplikaciju) koji će se izvoditi u oblaku, a četvrtim $\boldsymbol{r}$ se određuje koliko je temeljnih resursa angažirano u oblaku. Za preostala dva istraživana parametra, snaga angažiranih računala $\boldsymbol{p}$ i veličina baze podataka $\boldsymbol{b}$, pokazano je da njihovo variranje ne utječe značajno[18] na promjenu zavisne varijable T, ali naravno, oni se mogu pojaviti kao elementi za izračun troškova korištenja oblačnih resursa.

Za svrhu koja je planirana u ovom potpoglavlju provjeriti će se opcije kupnja ili najam potrebnih računalnih resursa za tri virtualne aplikacije označene kao ***Ap1***, ***Ap2*** i ***Ap3***. One su temeljene na tipskoj aplikaciji detaljno istraženoj u poglavlju 5 pa su i njihova svojstva opisana relevantnim parametrima $\boldsymbol{a}$, $\boldsymbol{u}$, $\boldsymbol{q}$ i $\boldsymbol{r}$ čije će vrijednosti biti unutar domene definirane tablicom 5-II. Svojstva ovih virtualnih aplikacija su:

- ***Ap1*** je niske složenosti algoritma, relativno male potrebe za slanjem radnih podataka u oblak i relativno jednostavnim CRUD akcijama. Kao primjer za ovaj tip aplikacije može se navesti jednostavno skladišno poslovanje. Stoga parametri koji je opisuju imaju sljedeće vrijednosti: $a=1$; $u=10$; $q=10$.
- ***Ap2*** ima srednju složenost algoritma i umjerene potrebe za slanjem radnih podataka u oblak, a obrada se temelji na prosječnom broju CRUD akcija. U ERP sustavu to odgovara planiranju potreba materijala raspuštanjem sastavnica na temelju zadanih primarnih potreba gotovih proizvoda, za manje poduzeće. Takva aplikacija se može opisati sljedećim vrijednostima parametara: $a=6$; $u=100$; $q=90$.
- ***Ap3*** ima algoritam visoke složenosti, šalje na obradu u oblak velike količine radnih podataka koji se obrađuju s velikom brojem CRUD akcija. U ERP sustavima to odgovara planiranju potreba raspuštanjem sastavnica, na temelju zadanih primarnih potreba velikog broja gotovih proizvoda kompleksne strukture. Takav se aplikacija može opisati parametrima: $a=10$; $u=500$; $q=200$.

[18] Ovdje valja još jednom naglasiti potrebu točne interpretacije rezultata istraživanja iz poglavlja 5: konstatacija da neki parametar, primjerice ($\boldsymbol{b}$), ne utječe na trajanje izvođenja aplikacije u oblaku T, nipošto ne implicira je da taj isti parametar nevažan kod izračuna cijene unajmljivanja oblačnih resursa.

Da bi se donijela objektivna odluka o tome gdje izvoditi svaku od ovih tipskih aplikacija, na vlastitoj opremi ili unajmljivanjem resursa u oblaku, za svaku od njih provest će se, temeljem saznanja iz prethodnih poglavlja, slijedeći **općeniti** postupak:

1. **Određivanje organizacijskog vremena:** Definiranje traženih organizacijskih vremena T_{ORG} i određivanje vremenskog perioda za kojeg se provodi proračun radi izbora povoljnije opcije iz tandema kupnja-najam.
2. **Procjena obujma podataka:** Procjena obujma podataka za trajnu pohranu te stope porasta ili smanjenja potreba za pohranom podataka.
3. **Izračun količine računalnih resursa:** Izračun količine potrebnih računalnih resursa za svaku radnu aplikaciju ($\boldsymbol{Ap1}$, $\boldsymbol{Ap2}$ i $\boldsymbol{Ap3}$). U ovome primjeru koristit će se polinom iz izraza *(5.10)*, već prilagođen izračunu za aplikacije temeljene na oglednoj, kao što su $\boldsymbol{Ap1}$, $\boldsymbol{Ap2}$ i $\boldsymbol{Ap3}$. Za taj izračun koristiti će se program iz priloga 5.1.
4. **Izračun troška računalnih resursa:** Prema prethodno izračunatoj količini potrebnih računala za radne aplikacije ($\boldsymbol{Ap1}$, $\boldsymbol{Ap2}$ i $\boldsymbol{Ap3}$), izračunava se, prema formulama u potpoglavlju 6.2, ukupni trošak kupnje vlastitih računala (TCO) te trošak najma istih računala u generičkom oblaku. Za to će se koristiti program iz priloga 6.2.
5. **Izračun troška pohrambenih resursa:** Prema zadanoj količini potrebnog prostora za pohranu podataka nad kojima radi svaka radna aplikacija ($\boldsymbol{Ap1}$, $\boldsymbol{Ap2}$ i $\boldsymbol{Ap3}$), izračunava se, prema formulama u poglavlju 6.3, trošak pohrane na vlastitim pohrambenim kapacitetima te trošak najma istih kapaciteta u generičkom oblaku. Za to će se koristiti program iz priloga 6.3.
6. **Izračun najma kod komercijalnih poslužitelja:** Za svaku radnu aplikaciju ($\boldsymbol{Ap1}$, $\boldsymbol{Ap2}$ i $\boldsymbol{Ap3}$), izračunava se, prema formulama u potpoglavlju 6.4, trošak najma u komercijalnim računalnim oblacima (ovdje: Amazonov, Microsoftov i Googleov).
7. **Usporedba cijene najma i kupnje:** Uspoređuju se cijene vlastitih računalnih i pohrambenih resursa izračunate u koracima 4 i 5 s cijenama najma izračunatima u koraku 6 te se temeljem toga izvlače zaključci o povoljnosti kupnje odnosno najma. Dodatno, može se izračunati cijena podizanja vlastitog podatkovnog centra (temeljem modela opisanog u poglavlju 6.1 i rezultata koraka 2 i 3) te okvirno procijeniti zarada oblačnog poslužitelja, ako je donijeta odluka o najmu (temeljem izračuna troškova računalnih i pohrambenih resursa za generički oblak u koracima 4 i 5 i usporedbe tih troškova sa stvarnom cijenom najma izračunatom u koraku 6).

Ovi koraci općeg postupka čine dio metodike procjene isplativosti kupnje ili najma nazvanoj **metodika KOR** (**k**orištenje **o**blačnih **r**esursa) i detaljnije razrađenoj u poglavlju 7. Mogućnost

korištenja predložene metodike provjerit će se na tri virtualne aplikacije. Polazno podaci i rezultati proračuna pregledno su prikazani u tablici 6-XII. Praktična primjena Osnovna svojstva triju virtualnih aplikacija upisana su u retke 1 do 5, zadana organizacijska vremena u redak 6, a izračunate vrijednosti u ostale retke tablice 6-XII. Za pokretanje ogledne aplikacije i na njoj temeljenih ***Ap1***, ***Ap2*** i ***Ap3*** potrebna je koristiti uslugu oblačnog računarstva tipa IaaS. Parametri za izračun troškova kupnje i najma, poput godišnjih fiksnih troškova C_T, preuzeti su iz odgovarajućih potpoglavlja 6.2 i 6.3.

U sljedećem tekstu, u nastavku tablice 6-XII, ukratko će biti prikazani načini praktične provedbe 7 prethodno definiranih koraka, koji su sastavni dijelovi metodike KOR, s ciljem da se za svaku aplikaciju izabere bolja između opcija kupnja ili najam.

Tablica 6-XII: ***Izračun podataka za odluku kupnja-najam za tri virtualne aplikacije***

			Aplikacija 1	***Aplikacija 2***	***Aplikacija 3***
1	Složenost algoritma (***a***) [ ]	Tražena svojstva virtualnih aplikacija	1	6	10
2	Opseg poslanih podataka mjesečno (***u***) [$^{GB}/_{mj}$]		10	100	500
3	Broj CRUD akcija po upitu (***q***) [ ]		10	90	200
4	Broj upita mjesečno (***x***) [$^{1}/_{mj}$]		10.000	100.000	500.000
5	Potrebe za pohranom podataka godišnje (***b***) [GB]		100+100+100	1000+1000+1000	5000+5000+5000
6	Organizacijsko vrijeme T_{ORG} [sek]		20	90	120
7	Broj angažiranih računala (***r***) [ ]	Izračunate vrijednosti za virtualne aplikacije	2	7	11
8	Snaga angažiranih računala (***p***) [ ]		1,7 GHz; 4 GB RAM-a		
9	Planski horizont Y [god]		3	3	3
10	Trošak kupnje računalnih resursa R_K [$^{\$}/_{h}$]		0,4614	0,4614	0,4614
11	Trošak najma računalnih resursa R_N [$^{\$}/_{h}$]		0,14	0,14	0,14
12	ΔNPV za diskovni kapacitet		-456,19	-456,19	-18.022,99
13	$C_{A,I}$ (***r***) [$^{\$}/_{h}$]		0,17 do 0,24	1,19 do 1,67	0,26 do 0,36
14	$C_{M,I}$ (***r***) [$^{\$}/_{h}$]		0,24	1,68	0,36
15	$C_{G,I}$ (***r***) [$^{\$}/_{h}$]		0,2	1,4	0,3

Korak 1: Određivanje organizacijskog vremena

Prihvatljiva organizacijska vremena za svaku aplikaciju procjenjuju korisnici i poslovni analitičari. U ovom primjeru neka su ona: $T_{ORG,1} = 20\ sekundi$, $T_{ORG,2} = 90\ sekundi$ i $T_{ORG,3} = 120\ sekundi$, redom za ***Ap1***, ***Ap2*** i ***Ap3***. Tako definirana organizacijska vremena upisana su u redak 6 tablice 6-XII.

Korak 2: Procjena obujma podataka

Obujam potrebnih podataka nad kojima radi neka aplikacija zavisi isključivo o podatkovnom fondu poduzeća. Stoga potrebe za diskovnim prostorom mogu zajedno odrediti poslovni analitičari i softverski inženjeri za dizajn i postavljanje baze podataka, na temelju opsega i vrste podatkovnog fonda, uzimajući i potrebe tijekom planskog horizonta. Ovdje se uzima da je za ***Ap1***, ***Ap2*** te ***Ap3*** potrebno odmah na početku rada pohraniti 100 GB, 1.000 GB odnosno 5.000 GB podataka te da se ta potreba povećava za 100 GB, 1.000 GB odnosno 5.000 GB godišnje. Ovi su podaci upisani u redak 5 tablice 6-XII.

Korak 3: Izračun broja potrebnih računala

Izračunavanje potrebnog broja računala u oblaku, zavisno o atributima $\boldsymbol{a}$, $\boldsymbol{u}$ i $\boldsymbol{q}$ (pomoću kojih su kvantificirani poslovni zahtjevi) i traženog organizacijskog vremena $T = T_{ORG}$, opširno je opisano u poglavlju 5. Na temelju rezultata mjerenja ogledne aplikacije definiran je izraz za $\boldsymbol{r}$ koje je moguće koristiti za sve tri virtualne aplikacije zbog njihove tipske podudarnosti s oglednom. Taj izraz glasi:

$$r = \frac{14{,}993 \cdot a + 0{,}135 \cdot u + 0{,}702 \cdot q - (T - 0{,}631)}{2{,}160 \cdot a + 0{,}84}$$

Uvrštavanjem vrijednosti parametara $\boldsymbol{a}$, $\boldsymbol{u}$, $\boldsymbol{q}$ i T_{ORG} za svaku od tri virtualne aplikacije dobiva se broj potrebnih računala $\boldsymbol{r}$, što je upisano u redak 7 tablice 6-XII. Za izračun se može koristiti program iz priloga 5.1. Dobiveni realni broj potrebno je zaokružiti na prvi veći cijeli broj. Ovdje će se pretpostaviti da su potrebna računala gornje klase (1,7 GHz i 4 GB RAM-a) te da imaju po četiri procesora.

Korak 4: Izračun troška računalnih resursa

Trošak računalnih CPU vremena na računalima u vlasništvu dobiva se iz izraza *(6.8)* u potpoglavlju 6.2, koji uzima u obzir diskontiranje budućih izdataka, ali i relativno smanjivanje tehničkog kapaciteta po Mooreovom zakonu. Ako se račun izvede za $Y=3\ godine$ uz godišnju kamatnu stopu od 7%, izraz *(6.8)* poprima oblik:

$$R_K = \frac{0{,}4531}{TC} \cdot \left(C_0 + \frac{C_1}{1{,}07} + \frac{C_2}{1{,}07^2}\right) \qquad (6.29)$$

U gornjem izrazu je TC početni tehnički kapacitet kupljenih računala, izražen u satima procesorskog vremena, a C_0, C_1 i C_2 su ulaganja u prvoj, drugoj i trećoj godini rada.

Tehnički kapacitet računa se prema izrazu *(6.5)* kao umnožak broja raspoloživih CPU, godišnjeg fonda radnih sati H i koeficijenta iskorištenja računala η. Ako se uzme da sva računala imaju po četiri procesora, da se godišnji fond od 7488 sati rada (prema tablici 6-II) koristi samo u jednoj smjeni te da je koeficijent iskorištenja računala $\eta=0{,}60$, onda se može

izračunati neto tehnički kapacitet TC svake konfiguracije računala kupljene za potrebe neke aplikacije. Ove vrijednosti za TC su upisane u prvi redak tablice 6-XIII.

Tablica 6-XIII: ***Podaci u slučaju kupnje računala za tri virtualne aplikacije***

			Ap1 (2 računala)		*Ap2 (7 računala)*		*Ap3 (11 računala)*	
1	Neto tehnički kapacitet [*h*]		11.980,8		41.932,8		65.894,4	
2	$C_0+C_1/1{,}07+C_2/1{,}07^2$ [*$*]	računala	12.201,0	4.000	42.704,0	14.000	67.107,0	22.000
		sistem-inženjer		8.201		28.704		45.107
3	R_K (nakon 3 godine) [*$/h*]		0,4614		0,4614		0,4614	

Vlastita ulaganja u kupnju računala i njihov operativni rad obuhvaćeni su onim dijelom izraza *(6.29)* koji se nalazi u zagradi. Sastoje se od troškova nabave računala u prvoj godini i troškova rada sistem-inženjera tijekom svake od tri godine. Cijena računala iznosi 2.000 *$* prema tablici 6-II. Ako se radi o podatkovnim centrima s vrlo malim brojem računala kao u ovom primjeru, onda potrebni rad sistem-inženjera nije moguće računati prema teoretskom modelu koji vrijedi u velikim podatkovnim centrima (jedan sistem-inženjer na 1.000 računala u smjeni, uvećano za 50% radi tjednog odmora, godišnjeg odmora i nepredviđenih izostanaka). Stoga je potrebni rad sistem-inženjera ovdje okvirno procijenjen na jedan sat tjedno po svakom instaliranom računalu, što za 2 računala potrebnih za ***Ap1*** iznosi 2 sata tjedno, to jest 7 i 11 sati rada tjedno za 7 odnosno 11 računala koja trebaju ***Ap2*** i ***Ap3***. To zapravo znači da će se poslovi sistem-inženjera dati u *outsourcing*, uz cijenu od 42 *$/h*. Uz te ulazne vrijednosti izračunat je dio izraza *(6.29)* koji se nalazi u zagradi i upisan u drugi redak tablice 6-XIII, ali tako da se odvojeno vide udjeli koji se odnose na trošak računala i trošak rada. Iz ovih podataka konačno je izračunata vrijednost za R_K u trećem retku. Ta ista vrijednost upisana je i u redak 10 zbirne tablice 6-XII radi kasnijih analiza. Iz ovih podataka se vidi da je stvarna cijena CPU-sata za slučaj kupnje računala u vlasništvo jednaka za sva tri poslovna slučaja te iznosi približno 0,46 *$* po satu.

Veliki udio u stvarnom trošku CPU-sata u slučaju kupnje vlastitih računala čine troškovi rada sistem-inženjera. Iako se ovdje pretpostavlja da će se takvi poslovi obavljati u *outsourcing*u, ipak u gornjim primjerima troškovi rada čine oko dvije trećine ukupnih troškova. Međutim, neke male tvrtke kojima informatika nije temeljni posao, ali imaju poslovanje jako zavisno o informatici, možda se neće odlučiti za *outsourcing*, već za zapošljavanje sistem-inženjera za izgradnju i podršku svojoj informatičkoj infrastrukturi. U tom bi slučaju udjel troškova rada stručnog osoblja još više porastao na više od ovdje izračunatih 0,46 $/sat. Takav trend zapravo povećava izglede za korištenje usluga oblačnog računarstva, što bi organizacijama omogućilo bolji nadzor nad ovom troškovnom komponentom.

Na sličan se način može izračunati neto sadašnja vrijednost CPU-sata za slučaj najma potrebnih kapaciteta u generičkom oblaku po jediničnoj cijeni P [$/h]. Ako se račun provede također za $Y=3\ godine$, tada se izraz *(6.10)* može pojednostavniti tako da glasi:

$$R_N = \frac{C_0 + \frac{C_1}{1,07} + \frac{C_2}{1,07^2}}{3 \cdot TC} \qquad (6.30)$$

S obzirom da je iznajmljeni tehnički kapacitet TC isti za sve tri godine, a godišnja vrijednost najma se računa kao umnožak tehničkog kapaciteta i jedinične cijene najma, to jest da vrijedi $C0=C1=C2=TC \cdot P$, lako se može vidjeti da izraz *(6.30)* prelazi u vrlo jednostavan oblik $R_N = 0{,}936 \cdot P$. Ako je jedinična cijena $P=0{,}15\$/h$, onda neto sadašnja vrijednost za slučaj najma iznosi $0{,}14\ \$/h$, što je i upisano u retku 11 tablice 6-XII.

Korak 5: Izračun troška pohrambenih resursa

Za izračunavanje troškova podatkovne pohrane u vlasništvu i generičkom oblaku koristi će se izraz *(6.27)*, koji je uređen tako da odmah daje ΔNPV, odnosno vrijednost razlike neto sadašnje vrijednosti za slučaj kada se diskovni prostor kupuje ili unajmljuje. Pri tome vrijedi kriterij: ako je razlika $\Delta NPV \geq 0$ onda treba kupiti, ako je $\Delta NPV<0$ diskove treba unajmiti. Ova razmatranja detaljno su izvedena u poglavlju 6.3.

Ako se detaljnije analiziraju pretpostavljeni zahtjevi za diskovnim kapacitetima koje imaju ogledne aplikacije, onda se vidi da ***Ap1*** i ***Ap2*** odgovaraju poslovnom slučaju 1 iz poglavlja 6.3, a zahtjevi koje ima ***Ap3*** odgovaraju poslovnom slučaju 2 iz istog poglavlja. Stoga se vrijednosti za ΔNPV mogu odmah izračunati korištenjem programa iz priloga 6.3 pa su, za $Y=3\ godine$, upisane u redak 12 tablice 6-XII. Valja upozoriti da su kod izračuna korištene vrijednosti parametara za izraz *(6.27)* navedene u tablici 6-V. Vidi se odmah da su sve tri vrijednosti za ΔNPV negativne, to jest da bi unajmljivanje diskovnih kapaciteta tijekom 3 godine bilo jeftinije za 456 $ u slučaju rada s ***Ap1*** i ***Ap2***, a 18.023 $ u slučaju rada s ***Ap3***.

Korak 6: Izračun najma kod komercijalnih poslužitelja

Troškovi oblačnog najma izračunati u četvrtom koraku odnose se, zapravo, na generički oblak, odnosno hipotetski i vjerojatno neprofitabilni oblačni podatkovni centar. U ovom će se koraku, temeljem formula iz tablice 6-XI, za svaku od tri virtualne aplikacije izračunati konkretne cijene u tri komercijalna oblaka. U svim izračunima uzimat će se u obzir **godišnji** troškovi najma. Izračuni će biti napravljeni na postepen način kako bi se precizno mogao pratiti njihov tijek. Krajnji rezultat predočen je tablicom 6-XIV.

Prije izvođenja proračuna, važno je napomenuti slijedeće o veličini baze podataka ***b***: radi jednostavnijeg izračuna, smatrat će se da je ukupna godišnja količina pohranjenih podataka u svakom mjesecu konstantna. Znači, prve godine iznosi tijekom svih 12 mjeseci 100 GB za

Ap1, za ***Ap2*** tijekom svih 12 mjeseci 1.000 GB, te za ***Ap3*** tijekom svih 12 mjeseci 5.000 GB. Što se tiče preuzetih podataka koji bi se inače opisivali varijablom $\boldsymbol{d}$ koja u tablici 6-XII nije navedena, radi preglednosti proračuna smatrat će se da je $\boldsymbol{u}$=$\boldsymbol{d}$, odnosno da je količina podataka koja se preuzima iz oblaka nakon obrade približno jednaka količini koja se šalje u oblak. Iz rezultata proračuna da se ovo izjednačavanje odnosi na relativno male troškove, pa to ne bi trebalo utjecati na konačno zaključivanje.

Troškovi najma za slučaj da je pružatelj usluga Amazon

Prvi izračun se odnosi na slučaj kada bi pružatelj usluga bio Amazon. Uz sve do sada navedeno u opisu ovoga koraka, a prema podacima za svaku aplikaciju iz tablice 6-XII, za Amazon će vrijediti slijedeće:

- Podatkovna pohrana temelji se na SimpleDB nerelacijskoj bazi podataka (a ne na S3 podatkovnoj pohrani).
- U cijenu korištene SimpleDB pohrane obračunava se slanje podataka u oblak $\boldsymbol{u}$, preuzimanje podataka iz oblaka $\boldsymbol{d}$ i veličina baze podataka $\boldsymbol{b}$
- U cijenu korištene SimpleDB pohrane **ne** obračunava se količina $\boldsymbol{x}$ CRUD akcija $\boldsymbol{q}$. Te se veličine obračunavaju za S3, ali se u tom slučaju ne obračunavaju $\boldsymbol{u}$ i $\boldsymbol{d}$.
- Obračunat će se cijene najma Linux i Windows virtualnih računala. Budući da su njihove satnice 0,085 $\$/h$ odnosno 0,12 $\$/h$, slijedi da je A_{OS} = 1 za Linux (kao temeljni, jedinični operacijski sustav) odnosno A_{OS} = 1,41 za Windows.

Izraz za IaaS najam u Amazonovom oblaku, prema tablici 6-XI, glasi:

$$C_{A,I} = A_r \cdot r \cdot p \cdot A_{os} + A_u \cdot u + A_d \cdot d + A_q \cdot q + A_{tc} \cdot tc + A_b \cdot b$$

Ako za zadani vremenski period (od jedne godine) treba izračunati apsolutnu cijenu, onda desnu stranu izraza treba pomnožiti s vremenskim parametrom – brojem mjeseci u godini za $\boldsymbol{u}$, $\boldsymbol{d}$ i $\boldsymbol{b}$, a brojem sati u godini za $\boldsymbol{r}$ – kako bi dimenzijski bio u skladu s aplikacijskim parametrima iz tablice 6-XII. Osim toga, ne koristi se virtualni privatni oblak ($tc = 0$), a podatkovna pohrana se temelji na SimpleDB pa je $q = 0$. Iz svega toga proizlazi izraz apsolutne cijene IaaS najma za ***Ap1***, ***Ap2*** i ***Ap3***:

$$C_{A,I}[\$] = A_r \left[{}^{\$}/_{h}\right] \cdot r \cdot p \cdot A_{os} \cdot 7488\,[h] + A_u \left[{}^{\$}/_{GB}\right] \cdot u \left[{}^{GB}/_{mj}\right] \cdot 12\,[mj] + A_{d\left[{}^{\$}/_{GB}\right]} \cdot d \left[{}^{GB}/_{mj}\right] \cdot 12\,[mj]$$
$$+ A_b \left[{}^{\$}/_{GB\,\cdot\,mj}\right] \cdot b[GB] \cdot 12[mj]$$

Rezultati za ***Ap1*** su slijedeći:

- Za Linux računala $A_r \cdot r \cdot p \cdot A_{os} \cdot 7488\,[h] = 0{,}085\,\left[{}^{\$}/_{h}\right] \cdot 2 \cdot 1 \cdot 1 \cdot 7488\,[h] = 1272{,}96\,[\$]$, a za Windows računala se, uz drukčiji A_{os}, dobije cijena od $1794{,}87\,[\$]$

- $A_u \cdot u \cdot 12\ [mj] = 0{,}1\ \left[{}^{\$}/_{GB}\right] \cdot 10\ \left[{}^{GB}/_{mj}\right] \cdot 12\ [mj] = 12\ [\$]$
- $A_d \cdot d \cdot 12\ [mj] = 0{,}15\ \left[{}^{\$}/_{GB}\right] \cdot 10\ \left[{}^{GB}/_{mj}\right] \cdot 12 [mj] = 18\ [\$]$
- $A_b \cdot b \cdot 12\ [mj] = 0{,}25\ \left[{}^{\$}/_{GB\ \cdot mj}\right] \cdot 100 [GB] \cdot 12 [mj] = 300\ [\$]$

Ukupna cijena je zaokružena na cijeli broj te iznosi 1.603 $\$$ za Linux računala te 2.125 $\$$ za Windows računala.

Za $\boldsymbol{Ap2}$:

- Za Linux računala $A_r \cdot r \cdot p \cdot A_{os} \cdot 7488\,[h] = 0{,}085\ \left[{}^{\$}/_{h}\right] \cdot 7 \cdot 1 \cdot 1 \cdot 7488\ [h] = 4455{,}36\ [\$]$, za Windows računala se uz drukčiji A_{os} dobije cijena od $6282{,}06\ [\$]$
- $A_u \cdot u \cdot 12\ [mj] = 0{,}1\ \left[{}^{\$}/_{GB}\right] \cdot 100\ \left[{}^{GB}/_{mj}\right] \cdot 12\ [mj] = 120\ [\$]$
- $A_d \cdot d \cdot 12\ [mj] = 0{,}15\ \left[{}^{\$}/_{GB}\right] \cdot 100\ \left[{}^{GB}/_{mj}\right] \cdot 12\ [mj] = 180\ [\$]$
- $A_b \cdot b \cdot 12\ [mj] = 0{,}25\ \left[{}^{\$}/_{GB\ \cdot mj}\right] \cdot 1000\ [GB] \cdot 12 [mj] = 3.000\ [\$]$

Ukupna cijena je zaokružena na cijeli broj i iznosi 7.755 $\$$ za Linux računala te 9.582 $\$$ za Windows računala.

Za $\boldsymbol{Ap3}$:

- Za Linux računala $A_r \cdot r \cdot p \cdot A_{os} \cdot 7488\,[h] = 0{,}085\ \left[{}^{\$}/_{h}\right] \cdot 11 \cdot 1 \cdot 1 \cdot 7488\ [h] = 7001{,}28\ [\$]$, za Windows računala se uz drukčiji A_{os} dobije cijena od $9871{,}8\ [\$]$
- $A_u \cdot u \cdot 12\ [mj] = 0{,}1\ \left[{}^{\$}/_{GB}\right] \cdot 500\ \left[{}^{GB}/_{mj}\right] \cdot 12\ [mj] = 600\ [\$]$
- $A_d \cdot d \cdot 12\ [mj] = 0{,}15\ \left[{}^{\$}/_{GB}\right] \cdot 500\ \left[{}^{GB}/_{mj}\right] \cdot 12\ [mj] = 900\ [\$]$
- $A_b \cdot b \cdot 12\ [mj] = 0{,}25\ \left[{}^{\$}/_{GB\ \cdot mj}\right] \cdot 5000\ [GB] \cdot 12 [mj] = 15.000\ [\$]$

Ukupna cijena je zaokružena na cijeli broj i iznosi 23.501 $\$$ za Linux računala te 26.372 $\$$ za Windows računala. Sve izračunate vrijednosti upisane su u stupac Amazon tablice 6-XIV.

Troškovi najma za slučaj da je pružatelj usluga Microsoft

Izraz za Microsoftov IaaS najam, prema tablici 6-XI, glasi:

$$C_{M,I} = M_r \cdot r \cdot p + M_u \cdot u + M_d \cdot d + M_q \cdot q + M_b \cdot b$$

Ako za zadani vremenski period (od jedne godine) treba izračunati apsolutnu cijenu, onda desnu stranu izraza treba pomnožiti s vremenskim parametrom – brojem mjeseci u godini za $\boldsymbol{u}$, $\boldsymbol{d}$, $\boldsymbol{b}$ i $\boldsymbol{q}$, a brojem sati u godini za $\boldsymbol{r}$ – kako bi dimenzijski bio u skladu s aplikacijskim

parametrima iz tablice 6-XII. Iz svega toga proizlazi izraz za izračun apsolutne cijene najma tipa IaaS za ***Ap1***, ***Ap2*** i ***Ap3***:

$$C_{M,I}[\$] = M_r \left[{}^{\$}/_{h}\right] \cdot r \cdot p \cdot 7488\,[h] + M_u \left[{}^{\$}/_{GB}\right] \cdot u \left[{}^{GB}/_{mj}\right] \cdot 12\,[mj] + M_d \left[{}^{\$}/_{GB}\right] \cdot d \left[{}^{GB}/_{mj}\right] \cdot 12\,[mj]$$
$$+ M_q \left[{}^{\$}/_{10000}\right] \cdot q \cdot x \left[{}^{1}/_{mj}\right] \cdot 12\,[mj] + M_b \left[{}^{\$}/_{GB\,\cdot\,mj}\right] \cdot b\,[GB] \cdot 12[mj]$$

Rezultati za ***Ap1*** su slijedeći:

- $M_r \cdot r \cdot p \cdot 7488\,[h] = 0{,}12 \left[{}^{\$}/_{h}\right] \cdot 2 \cdot 1 \cdot 1 \cdot 7488\,[h] = 1797{,}12\,[\$]$
- $M_u \cdot u \cdot 12\,[mj] = 0{,}1 \left[{}^{\$}/_{GB}\right] \cdot 10 \left[{}^{GB}/_{mj}\right] \cdot 12\,[mj] = 12\,[\$]$
- $M_d \cdot d \cdot 12\,[mj] = 0{,}15 \left[{}^{\$}/_{GB}\right] \cdot 10 \left[{}^{GB}/_{mj}\right] \cdot 12[mj] = 18\,[\$]$
- $M_q \cdot q \cdot x \cdot 12\,[mj] = 0{,}01 \left[{}^{\$}/_{10000}\right] \cdot 10 \cdot 10000 \left[{}^{1}/_{mj}\right] \cdot 12[mj] = 1{,}2\,[\$]$
- $M_b \cdot b \cdot 12\,[mj] = 0{,}15 \left[{}^{\$}/_{GB\,\cdot\,mj}\right] \cdot 100\,[GB] \cdot 12[mj] = 180\,[\$]$

Ukupna cijena je zaokružena na cijeli broj i iznosi 2.008 $\$$.

Za ***Ap2***:

- $M_r \cdot r \cdot p \cdot 7488\,[h] = 0{,}12 \left[{}^{\$}/_{h}\right] \cdot 7 \cdot 1 \cdot 1 \cdot 7488\,[h] = 6289{,}92[\$]$
- $M_u \cdot u \cdot 12\,[mj] = 0{,}1 \left[{}^{\$}/_{GB}\right] \cdot 100 \left[{}^{GB}/_{mj}\right] \cdot 12\,[mj] = 120\,[\$]$
- $M_d \cdot d \cdot 12\,[mj] = 0{,}15 \left[{}^{\$}/_{GB}\right] \cdot 100 \left[{}^{GB}/_{mj}\right] \cdot 12[mj] = 180\,[\$]$
- $M_q \cdot q \cdot x \cdot 12\,[mj] = 0{,}01 \left[{}^{\$}/_{10000}\right] \cdot 10 \cdot 100000 \left[{}^{1}/_{mj}\right] \cdot 12[mj] = 12\,[\$]$
- $M_b \cdot b \cdot 12\,[mj] = 0{,}15 \left[{}^{\$}/_{GB\,\cdot\,mj}\right] \cdot 1000[GB] \cdot 12[mj] = 1800\,[\$]$

Ukupna cijena je zaokružena na cijeli broj i iznosi 8.402 $\$$.

Za ***Ap3***:

- $M_r \cdot r \cdot p \cdot 7488\,[h] = 0{,}12 \left[{}^{\$}/_{h}\right] \cdot 11 \cdot 1 \cdot 1 \cdot 7488\,[h] = 9884{,}16\,[\$]$
- $M_u \cdot u \cdot 12\,[mj] = 0{,}1 \left[{}^{\$}/_{GB}\right] \cdot 500 \left[{}^{GB}/_{mj}\right] \cdot 12\,[mj] = 600\,[\$]$
- $M_d \cdot d \cdot 12\,[mj] = 0{,}15 \left[{}^{\$}/_{GB}\right] \cdot 500 \left[{}^{GB}/_{mj}\right] \cdot 12[mj] = 900\,[\$]$
- $M_q \cdot q \cdot x \cdot 12\,[mj] = 0{,}01 \left[{}^{\$}/_{10000}\right] \cdot 10 \cdot 500000 \left[{}^{1}/_{mj}\right] \cdot 12[mj] = 60\,[\$]$
- $M_b \cdot b \cdot 12\,[mj] = 0{,}15 \left[{}^{\$}/_{GB\,\cdot\,mj}\right] \cdot 5000[GB] \cdot 12[mj] = 9000\,[\$]$

Ukupna cijena je zaokružena na cijeli broj i iznosi 20.444 $\$$. Sve tri vrijednosti upisane su u stupac Microsoft tablice 6-XIV.

Troškovi najma za slučaj da je pružatelj usluga Google

Izraz za Googleov IaaS najam, prema tablici 6-XI, glasi:

$$C_{G,I} = G_r \cdot r + G_u \cdot u + G_d \cdot d + G_b \cdot b$$

Ako za zadani vremenski period (od jedne godine) treba izračunati apsolutnu cijenu, onda desnu stranu izraza treba pomnožiti s vremenskim parametrom – brojem mjeseci u godini za $\boldsymbol{u}$, $\boldsymbol{d}$ i $\boldsymbol{b}$ a brojem sati u godini za $\boldsymbol{r}$ – kako bi dimenzijski bio u skladu s aplikacijskim parametrima iz tablice 6-XII. Iz svega toga proizlazi izraz za izračun apsolutne cijene IaaS najma za ***Ap1***, ***Ap2*** i ***Ap3***:

$$C_{G,I}[\$] = G_r \left[{}^{\$}/_{h}\right] \cdot r \cdot 7488\ [h] + G_u \left[{}^{\$}/_{GB}\right] \cdot u \left[{}^{GB}/_{mj}\right] \cdot 12[mj] + G_d \left[{}^{\$}/_{GB}\right] \cdot d \left[{}^{GB}/_{mj}\right] \cdot 12[mj]$$
$$+ G_b \left[{}^{\$}/_{GB\ \cdot\ mj}\right] \cdot b\ [GB] \cdot 12[mj]$$

Rezultati za ***Ap1*** su slijedeći:

- $G_r \cdot r \cdot 7488[h] = 0{,}1 \left[{}^{\$}/_{h}\right] \cdot 2 \cdot 1 \cdot 1 \cdot 7488\ [h] = 1497{,}6\ [\$]$
- $G_u \cdot u \cdot 12\ [mj] = 0{,}1 \left[{}^{\$}/_{GB}\right] \cdot 10 \left[{}^{GB}/_{mj}\right] \cdot 12[mj] = 12\ [\$]$
- $G_d \cdot d \cdot 12\ [mj] = 0{,}12\ \left[{}^{\$}/_{GB\ \cdot\ mj}\right] \cdot 10 \left[{}^{GB}/_{mj}\right] \cdot 12[mj] = 14{,}4\ [\$]$
- $G_b \cdot b \cdot 12\ [mj] = 0{,}15\ \left[{}^{\$}/_{GB\ \cdot\ mj}\right] \cdot 100[GB] \cdot 12\ [mj] = 180\ [\$]$

Ukupna cijena je zaokružena na cijeli broj i iznosi 1.704 $\$$.

Za ***Ap2***:

- $G_r \cdot r \cdot 7488[h] = 0{,}1\ \left[{}^{\$}/_{h}\right] \cdot 7 \cdot 1 \cdot 1 \cdot 7488\ [h] = 5241{,}6\ [\$]$
- $G_u \cdot u \cdot 12\ [mj] = 0{,}1\ \left[{}^{\$}/_{GB}\right] \cdot 100 \left[{}^{GB}/_{mj}\right] \cdot 12[mj] = 120\ [\$]$
- $G_d \cdot d \cdot 12\ [mj] = 0{,}12\ \left[{}^{\$}/_{GB\ \cdot\ mj}\right] \cdot 100 \left[{}^{GB}/_{mj}\right] \cdot 12[mj] = 144\ [\$]$
- $G_b \cdot b \cdot 12\ [mj] = 0{,}15\ \left[{}^{\$}/_{GB\ \cdot\ mj}\right] \cdot 1000[GB] \cdot 12\ [mj] = 1800\ [\$]$

Ukupna cijena je zaokružena na cijeli broj i iznosi 7.306 $\$$.

Za ***Ap3***:

- $G_r \cdot r \cdot 7488[h] = 0{,}1\ \left[{}^{\$}/_{h}\right] \cdot 11 \cdot 1 \cdot 1 \cdot 7488\ [h] = 8236{,}8\ [\$]$
- $G_u \cdot u \cdot 12\ [mj] = 0{,}1\ \left[{}^{\$}/_{GB}\right] \cdot 500 \left[{}^{GB}/_{mj}\right] \cdot 12[mj] = 600\ [\$]$
- $G_d \cdot d \cdot 12\ [mj] = 0{,}12\ \left[{}^{\$}/_{GB\ \cdot\ mj}\right] \cdot 500 \left[{}^{GB}/_{mj}\right] \cdot 12[mj] = 720\ [\$]$
- $G_b \cdot b \cdot 12\ [mj] = 0{,}15\ \left[{}^{\$}/_{GB\ \cdot\ mj}\right] \cdot 5000[GB] \cdot 12\ [mj] = 9000\ [\$]$

Ukupna cijena je zaokružena na cijeli broj i iznosi 18.557 $\$$.

Cjenovne komponente za svaku od analiziranih virtualnih aplikacija (***Ap1***, ***Ap2*** i ***Ap3***) i ukupna cijena, kao zbroj tih komponenti, grupirani su prema pružateljima usluga oblačnog računarstva i pregledno prikazani tablicom 6-XIV. Za svaku analiziranu aplikaciju posebno je naznačeno koja komponenta se odnosi neki od parametara detaljno istraženih u poglavlju 5.

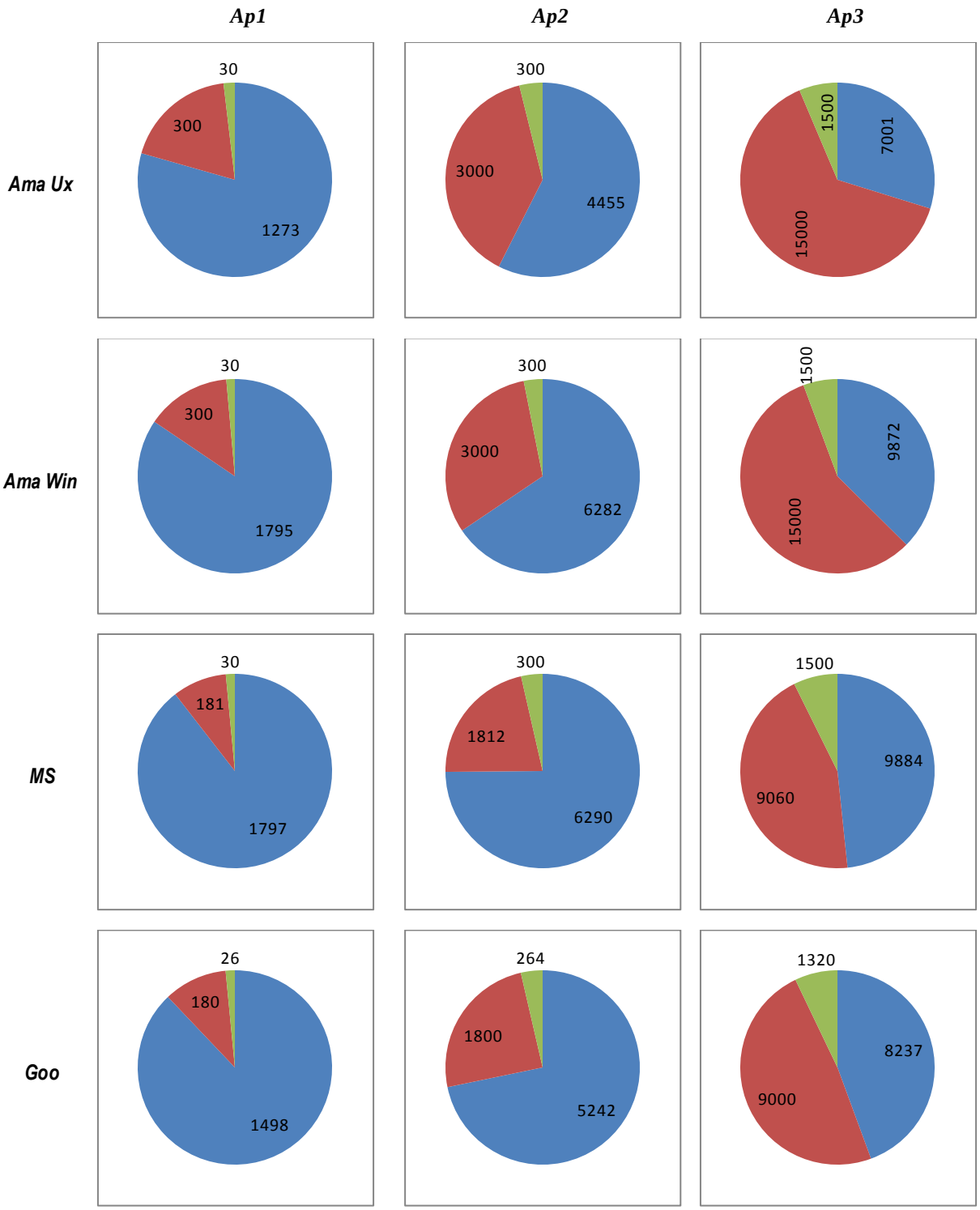
Tablica 6-XIV: ***Komponente cijene najma za*** **Ap1, Ap2** ***i*** **Ap3** ***po poslužiteljima***

	Komponente cijene	*Iznos za* ***Amazon*** *[$]*	*Iznos za* ***Microsoft*** *[$]*	*Iznos za* ***Google*** *[$]*
Ap1	Računalni resursi (***r***)	1.273 do 1.795	1.797	1.498
	Pohrana podataka (***b*** + ***q***)	300	181	180
	Promet podataka (***u*** + ***d***)	30	30	26
	Ukupno	**1.603 do 2.125**	**2.008**	**1.704**
Ap2	Računalni resursi (***r***)	4.455 do 6.282	6.290	5.242
	Pohrana podataka (***b*** + ***q***)	3.000	1.812	1.800
	Promet podataka (***u*** + ***d***)	300	300	264
	Ukupno	**7.755 do 9.582**	**8.402**	**7.306**
Ap3	Računalni resursi (***r***)	7.001 do 9.872	9.884	8.237
	Pohrana podataka (***b*** + ***q***)	15.000	9.060	9.000
	Promet podataka (***u*** + ***d***)	1.500	1.500	1.320
	Ukupno	**23.501 do 26.372**	**20.444**	**18.557**

Detaljna analiza podataka iz tablice 6-XIV upućuje na interesantne zaključke:

- Troškovi najma različitih pružatelja usluga se razlikuju do 30%, pri čemu je Google najjeftiniji. Međutim, iznajmljivanjem oblačnih resursa tijekom kraćeg probnog razdoblja moglo bi se utvrditi da li se takva razlika u cijeni može opravdati razlikom u kvaliteti usluge, sigurnosti, pouzdanosti ili nekim drugim parametrima koji se ne mogu obuhvatiti kvantitativnom analizom. Stoga se preporuča strategija probnog najma prije konačnog izbora pružatelja usluga.
- Relativne vrijednosti pojedinih komponenata troškova za svakog pružatelja usluge se razlikuju, ali kod svih se daleko najveći trošak odnosi na najam računala, a najmanji su troškovi podatkovnog prometa.

Struktura cijene svake aplikacije po određenom poslužitelju prikazana je grafovima na slici 6.7. Troškovi računalnih resursa su prikazani plavom bojom, pohrane podataka crvenom, a troškovi prometa podataka zelenom. Struktura cijene za Amazonova Linux računala označena je s Amazon Ux, za Amazonova Windows računala kao Ama Win, dok je struktura cijene Microsoftovog oblaka označena s MS, a Googleovog kao Goo.



Slika 6.7: ***Struktura troškova aplikacija*** **Ap1, Ap2** ***i*** **Ap3** ***po poslužiteljima***

Korak 7: Usporedba cijene najma i kupnje

Temeljem provedenih izračuna vidljivo je da je za sve tri virtualne aplikacije, u razdoblju od 3 godine, najam povoljniji od kupnje. Isplativost najma pada s porastom potrebne količine računalnih i pohrambenih resursa. Drugim riječima, isplativost najma je obratno proporcionalna s količinom najmljenih resursa. Naime, **prosječna** cijena najma resursa za ***Ap1*** je oko 1.860 *$* za prvu godinu to jest 5.580 *$* u tri godine. Isti ti resursi u vlasništvu, kroz isti vremenski period, su oko 2,19 puta skuplji (11.981 *$*)! Međutim, troškovi najma resursa za najzahtjevniju aplikacija ***Ap3*** kroz period od 3 godine približavaju se (ili čak izjednačuju, ovisno o poslužitelju) troškovima kupnje. "Zlatna sredina," ***Ap2*** je kroz 3 godine u prosjeku skuplja oko 1,72 puta ako se pokreće na vlastitim resursima, a ne onima najmljenim u oblaku.

Zanimljivo je primijetiti da navedena opažanja vrijede unatoč tome što kod resursa u vlasništvu ne postoji (pa se niti ne zaračunava dodatno) komponenta komunikacijskih resursa (slanje podataka u oblak ***u*** i preuzimanje podataka iz oblaka ***d***). Ta komponenta jest, doduše, jeftina, ali iz grafova strukture cijene na slici 6.7 jasno je da kod složenih aplikacija nije zanemariva.

Podatkovna pohrana zapravo je vrlo bitan čimbenik računalnih oblaka što se vidi i po njenom utjecaju na cijenu kada se povećava obujam podataka. Troškovi pohrane velikih količina podataka mogu nadmašiti troškove najma računalnih resursa! To se može objasniti jer, unatoč svojoj relativnoj jednostavnosti, već je i organizacija nerelacijske pohrane (s kojom je izveden proračun) opsežan posao. Drugim riječima, BASE zahtjevi na baze podataka opisani u poglavlju 4.8 nisu jednostavniji za podržati računalima, diskovima i softverom od klasičnih ACID zahtjeva, iako se često percipiraju takvima. To samo potvrđuje pretpostavku da su relacijske ACID baze podataka namijenjene za drugu svrhu nego KVS BASE baza podataka, pa stoga nije opravdano smatrati da je jedan koncept ispravan a drugi koncept pogrešan.

Na kraju, jasno je da su tehničke mogućnosti platforme proporcionalne njenoj prosječnoj cijeni. Općenito, Googleova platforma je najjednostavnija pa je stoga i najjeftinija. Računalna komponenta Amazonovog oblaka za Linux sustave je jeftinija od Microsoftove, dok su cijene Amazonove Windows ponude otprilike jednake Microsoftovoj. Istovremeno, Amazonova podatkovna pohrana je najskuplja, ali s nešto boljim osobinama od konkurencije (poput već spomenute podrške za Hadoop).

6.8 Osvrt na drugi dio hipoteze 2

U prethodnom je poglavlju pokazano kako je moguće, objektiviziranom kvantifikacijom resursnih (r i $\boldsymbol{p}$), aplikacijskih (a i $\boldsymbol{q}$), podatkovnih ($\boldsymbol{u}$ i $\boldsymbol{b}$) te organizacijskih (T_{org}) parametara, postaviti metriku za utvrđivanje potreba koje klasične aplikacije ili već razvijeni SOA servisi imaju prema računalnim resursima u oblaku. Na taj je način dokazan prvi dio hipoteze 2.

U ovom je poglavlju postavljen općeniti postupak (temeljen na parametrima koji čine metriku uspostavljenu u prethodnom poglavlju) koji omogućava da se za neku aplikaciju objektivno procijeni da li je, s motrišta korisnika, bolje takvu aplikaciju pokretati u računalnom oblaku ili pak na vlastitim računalnim resursima. Pri toj procjeni u račun su uključeni svi oni resursni, aplikacijski, podatkovni i organizacijski parametri koji su korišteni u oblikovanju metrike. U skladu s tim se može utvrditi da je na taj način dokazan drugi dio hipoteze 2 ovog rada, koji glasi: „*Korištenje takve metrike će povećati sigurnost pri odlučivanju o tome treba li neki servis stalno ili povremeno pokretati u računalnom oblaku*.“

Drugi dio ove hipoteze također je i praktično provjeren na primjeru tri virtualne aplikacije koje tipski odgovaraju često korištenim aplikacijama u ERP sustavima. Može se dakle zaključiti da su rezultati provedenih teoretskih istraživanja također omogućili izvedbu praktičnih izračuna od kojih će imati koristi svi koji su zainteresirani za pokretanje vlastitih rješenja u oblaku, jer im predloženi postupak daje jasne smjernice kada je isplativo koristiti računalni oblak, a kada je bolje kupiti vlastita računala.

6.9 Sažetak

U ovom je poglavlju najprije obrađena problematika pune cijene vlasništva nad oblačnim centrom, zanimljiva potencijalnim investitorima u takva postrojenja i korisnicima računalnih oblaka. Razvijen je matematički model za izračun TCO te izrađen odgovarajući program u Excelu. Računska provjera, provedena za tipičan podatkovni centar, pokazala je ispravnost modela. Također je pokazano da u TCO hardver sudjeluje s oko 40% a energija s oko 30%.

Potom su, polazeći od radova E. Walkera, razvijena dva matematičko-ekonomska modela: prvi za izračun troška CPU vremena (računalnih resursa), a drugi za izračun troška diskovnih resursa za pohranu podataka. Kod prvog modela uzima se povećavanje računalne snage po Moore-ovom zakonu, a kod drugog empirijski potvrđeno smanjivanje cijene memorijskih resursa, izraženo u $\$/GB$ mjesečno. Oba modela omogućavaju investitoru uvid u rentabilnost ulaganja, a korisniku usporedbu troškova najma u odnosu na troškove vlasništva. Za oba modela razvijena je potrebna programska podrška.

Troškovi računalnih resursa su iskazani cijenom rada računala po satu, pa je stoga moguće uspoređivati troškove za slučaj najma u hipotetskom i neprofitnom računalnom oblaku, ali i

za slučaj kada se računala nabavljaju. Na temelju provedenih proračuna za dva tipična poslovna slučaja pokazano je da cijena rada CPU po satu snažno zavisi o stupnju iskorištenosti računala. Ako je iskorištenost računala visoka, npr. $\eta \geq 0{,}95$ investitor u serversku farmu može, uz cijenu iznajmljivanja računala od $L=0{,}10$ *[$/sat]* ostvariti godišnju zaradu od 16,2% na uloženi kapital, s tim da je ROI najveći u šestoj godini rada serverske farme. Uz istu cijenu iznajmljivanja CPU vremena donja granica profitabilnosti ulaganja je oko $\eta = 0{,}60$ što znači da uz niži stupanj iskorištenosti investitor ne može računati na dobit. Pokazano je također da bi korisnik u slučaju kupnje računala mogao sniziti troškove na približno *0,10 [$/sat]* ako bi postigao stupanj iskorištenja svojih računala od 95% i to samo za prve tri godine rada, dok bi nakon tog razdoblja za opcija najma računalnih resursa za korisnika bila uvijek povoljnija od kupnje.

Model za troškove diskovnog kapaciteta, uz pretpostavku određenog rasta potreba godišnje, postavljen je tako da se računa neto sadašnja vrijednost za slučaj kupnje ili najma, što također omogućava izbor povoljnije opcije. Analiza za tri poslovna slučaja, provedena prema postavljenom modelu, pokazala je nešto drugačije odnose kupnje ili najma nego za računalne kapacitete. Za individualne korisnike, čije su potrebe oko 500 GB, s povećanjem po 100 GB godišnje, opcija najma je povoljnija od kupnje za prve četiri godine rada. Za malo poduzeće, čije se potrebe stalno povećavaju za 1 TB diskovnog kapaciteta godišnje, najam je trajno povoljnija opcija. Za vrlo veliko poduzeće, čije potrebe kontinuirano rastu za 10 TB godišnje, opcija kupnje je uvijek povoljnija od najma diskovnih kapaciteta u oblaku.

U četvrtom potpoglavlju izvedena je taksonomija komercijalnih usluga oblačnog računarstva, uzimajući u obzir potrebe za CPU resursima i diskovnim kapacitetima, na primjeru tri značajna pružatelja usluga (Amazon, Microsoft i Google), čije se usluge tipa IaaS i PaaS mogu smatrati reprezentativnim za sadašnje stanje na tržištu. Za svaki tip usluga i za svakog od tih ponuđača postavljeni su matematičke formule za izračun ukupnih troškova, koje bi imao korisnik u slučaju smještaja neke aplikacije u oblak i njezinog redovitog korištenja. U formule su uključene aplikacijske, podatkovne i resursne varijable istražene u prethodnom poglavlju, ali i druge varijable koje su specifične za ponudu svakog ponuđača.

Na ovaj su način ispunjeni preduvjeti za postavljanje metodike odlučivanja o kupnji ili najmu svih resursa, potrebnih za rad informacijskog sustava (odnosno jest njegovih SOA servisa) na platformi oblačnog računarstva. Naime, metrika aplikacijskih potreba za računalnim resursima postavljena je u poglavlju 5, a ona je bila tehnički preduvjet za ekonomiku oblačnog računarstva i za postavljanje metodike čijom primjenom korisnik može donijeti argumentirane odluke o kupnji ili najmu potrebnih resursa.

U predzadnjoj točki ovog poglavlja izložena je metodika odlučivanja o kupnji ili najmu, kao postupak od 7 koraka koji se temelje na integraciji saznanja iz prethodnog i ovog poglavlja. U te je korake uključeno: kvantifikacija aplikacijskih zahtjeva, određivanje organizacijskog vremena, procjena obujma podataka s kojima rade aplikacije ili SOA servisi, izračun količine računalnih resursa potrebnih za pokretanje aplikacija te izračun troška računalnih i pohrambenih resursa za slučaj njihove vlastite nabave i za slučaj najma u komercijalnom oblaku. Cijeli je ovaj postupak provjeren analizom očekivanih troškova za tri zamišljene aplikacije, čija se svojstva temelje na specijalizaciji općih svojstva ogledne aplikacije iz petog poglavlja. Za svaku od njih izračunati su očekivani troškovi, koje bi imao korisnik ako bi ih izvodio na računalnim resursima nekog od tipskih pružatelja usluga oblačnog računarstva.

Na kraju poglavlja argumentirano je kako je kroz provedene postavke u petom i šestom poglavlju potvrđena druga hipoteza ove disertacije.

7 Metodika razvoja oblačnih rješenja

Problematika razvoja rješenja za oblak trenutno je iznimno dinamično područje istraživanja i praktičnog rada zbog relativno nedavne konsolidacije oblačnih platformi. Obećanje ušteda i pogodnosti pri uporabi komercijalnih računalnih oblaka kao hardverske infrastrukture za informacijske sustave i servise postavlja određena pitanja pred uprave tvrtki i javnih ustanova te projektante i softverske inženjere:

1. Kada i koje poslovne procese je korisno infrastrukturno podržati privatnim računalima odnosno komercijalnim računalnim oblakom? Na to pitanje, koje je i temelj Hipoteze 2 ovoga rada, uspješno je odgovoreno u prethodnim poglavljima ovoga rada, zaključno sa šestim.
2. Kako razraditi strategiju računalne potpore zahtjevnih SOA servisa potpuno ili djelomično smještenih u oblaku? Općenito, kako sistematizirati postupke za projektiranje IS-a koji ima barem neke komponente smještene u komercijalnom oblaku?

Ovim poglavljem odgovara se na pitanja iz točke 2.

Za projektiranje klasičnih (i mobilnih) informacijskih sustava koristi se danas veći broj metodika, kratko i pregledno prikazanih u [12]. Projektiranje IS-a potpuno ili djelomično smještenog u oblaku suštinski se ne razlikuje od projektiranja klasičnog IS-a. Stoga će ovim poglavljem biti predložena metodika za projektiranje "oblačnih" informacijskih sustava temeljena na postojećim metodama i tehnikama namijenjenim toj djelatnosti, nazvana metodika **k**orištenja **o**blačnog **r**ačunarstva – **metodika KOR**.

Valja naglasiti kako će (u skladu s činjenicom da je projektiranje IS-a smještenog u oblaku slično projektiranju klasičnog IS-a) određena pažnja biti poklonjena uklapanju metodike KOR u klasične modele razvoja informacijskih sustava. Posebice će biti obrađeno kako se koraci metodike KOR uklapaju u faze životnog ciklusa razvoja informacijskih sustava (*Systems Development Life Cycle* - SDLC). Pri tome će se istražiti i definirati na koje faze V-modela se odnose pojedini koraci metodike KOR.

Prije detaljnijeg izlaganja metodike KOR valja također definirati neke terminološke pojedinosti:

- Riječ *metodika* koristiti će se u njezinom izvornom značenju – *način svrhovitog rješavanja praktičnih problema*. S obzirom da je projektiranje IS-a složen posao, u ovom će se radu smatrati da se metodika primjenjuje izvođenjem više koraka po točno određenom redoslijedu. Za izvođenje svakog od tih koraka može se koristiti

jedna ili više različitih metoda. Svojstvo konzistentne metodike je da izlazni parametri iz jednog koraka predstavljaju ulazne veličine za sljedeći korak.

- Pod pojmom *metoda* podrazumijevat će se *postupak koji pomaže ostvarenju željenog rezultata u nekom praktičnom poslu ili znanstvenom istraživanju*.

S obzirom da je projektiranje i razvoj IS-a samo po sebi složeni poslovni proces, to je i metodika projektiranja i razvoja informacijskog sustava, potpuno ili djelomičnog smještenog u oblaku, prikazana metodom modeliranja poslovnih procesa (*Business Process Modeling* – BPM). Metoda BPM je korištena za prikaz metodike KOR zato jer omogućava strogi formalni opis pojedinih koraka, a logička struktura i simbolički jezik BPM-a su općenito prihvaćeni industrijski i profesionalni standard za modeliranje poslovnih procesa, pod nazivom BPMN (trenutno je važeća verzija BPMN 2).

Metodika KOR ima ukupno 23 koraka te se može smatrati dopunom opće metodike za projektiranje i razvoj klasičnog IS-a. U ovom je poglavlju ta metodika KOR objašnjena grafičkim modelom procesa i tabličnim opisom.

7.1 V-model

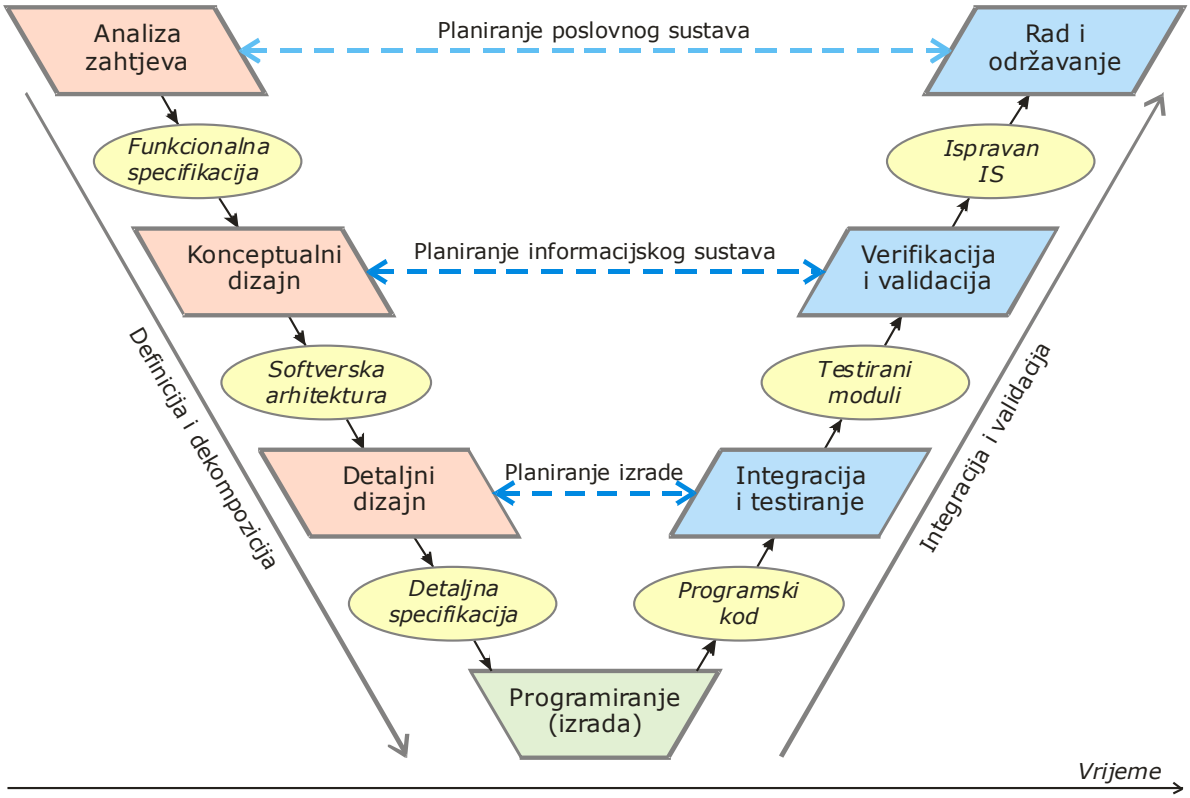
V-model je standardni, široko prihvaćeni model za prikaz životnog ciklusa razvoja informacijskih sustava [37, 63]. Povezuje faze projektiranja, razvoja i korištenja IS-a, od konceptualnog osmišljavanja do održavanja softverskog proizvoda. Predočava se dijagramom u obliku slova V, koji je prikazan slikom 7.1.

Lijevi krak V-dijagrama obuhvaća dizajn i projektiranje IS-a sa slijedećim fazama:

- ***Analiza zahtjeva***. Rezultat je *funkcionalna specifikacija* softvera.
- **Konceptualni dizajn**. Rezultat je *softverska arhitektura* budućeg IS-a.
- **Detaljni dizajn**. Rezultat je *detaljna specifikacija* pojedinih modula budućeg IS-a.

Desni krak V-dijagrama obuhvaća testiranje IS-a i njegovo puštanje u pogon:

- **Integracija i testiranje**. Rezultat su *testirani moduli* informacijskog sustava, potvrđeno interoperabilni, kako među sobom, tako i s drugim sustavima.
- **Verifikacija i validacija**. Rezultat ove faze je *verificirani informacijski sustav* za kojeg je potvrđeno da doista podržava izvođenje traženih poslovnih procesa.
- **Rad i održavanje**. Ovo je iterativna aktivnost čiji je rezultat *ispravan informacijski sustav* prilagođen novim poslovnim uvjetima.



Slika 7.1: ***Životni ciklus razvoja informacijskog sustava prikazan V-dijagramom*** *(prema [63])*

Dno V-dijagrama predstavlja kompleksna faza **izrade IS-a** to jest programiranja pojedinih modula te postavljanja i konfiguriranja potrebne hardverske i softverske opreme. Rezultat ove faze je *programski kod* pojedinih modula novog informacijskog sustava

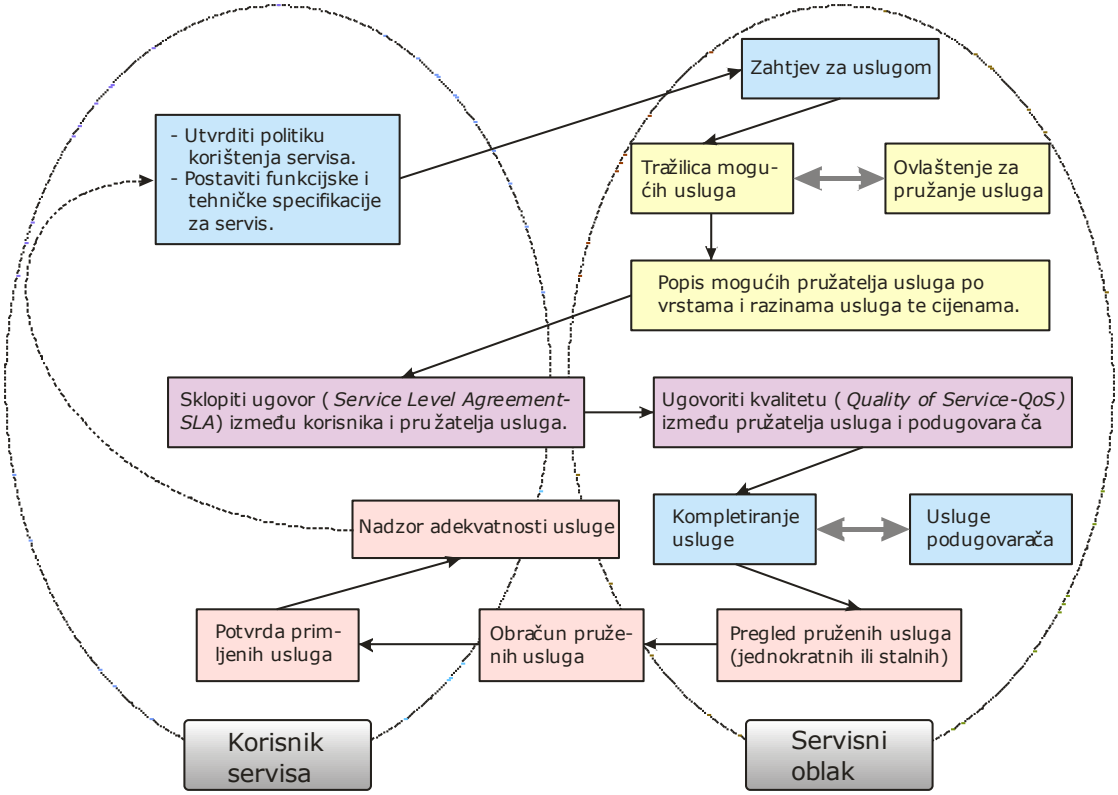
Najbitnije svojstvo V-modela jest stalna povratna veza između faza na oba kraka V-dijagrama: funkcionalna specifikacija (u lijevom kraku) dopunjava se tijekom rada i održavanja, softverska arhitektura sustava provjerava se u fazi verifikacije i validacije te prema potrebi opet šalje u fazu konceptualnog dizajna, a ispravnost detaljne specifikacije obavlja se u fazi integracije i testiranja gdje se otkrivaju eventualne logičke greške i nekompatibilnosti koje se potom vraćaju u fazu detaljnog dizajna na popravak. U poglavlju 7.4 bit će detaljnije opisano kako se pojedini koraci metodike KOR uklapaju u faze opisane V-dijagramom.

7.2 Životni ciklus IT proizvoda u oblaku

Pojam životnog ciklusa informacijskih tehnologija i proizvoda odnosi se na radne aktivnosti, procese i metode za stvaranje, dopunu i održavanje softverskih ili hardverskih rješenja. U kontekstu komercijalnog oblačnog računarstva, istraživače, korisnike i ulagače zanimaju načini upravljanja programskim rješenjima smještenim u računalnom oblaku, pogotovo u usporedbi s takvim rješenjima smještenima na klasičnoj infrastrukturi.

Opći cilj metodike KOR je dakle podrška životnom ciklusu razvoja i korištenja informacijskog sustava na informatičkim resursima koji se koriste prema konceptu oblačnog računarstva. U temeljnim koracima metodika je usklađena s rezultatima istraživanja drugih znanstvenika, primjerice [51], prema kojima se životni ciklus IT usluga u oblaku može predočiti *slikom 7.2*. Ta slika, kao i ovo potpoglavlje, predstavljaju najopćenitiji pogled na ovu problematiku (*thirty thousand feet overview*), dok će u idućem potpoglavlju metodika KOR biti opisana dovoljno detaljno za praktičnu upotrebu te prikazana slikom 7.3.

U procesu **Identificirati funkcionalna i tehnička svojstva servisa i odrediti servisnu politiku** (*Identify functional and technical specifications. Determine service policies*) korisnik usluga oblačnog računarstva će specificirati poslovne zahtjeve koje mora podržati aplikacija izvedena kao SOA servis u oblaku. Osim poslovnih, na aplikaciju se postavljaju i zahtjevi tehničke prirode kao što je maksimalno dopušteno organizacijsko vrijeme T_{ORG}, opširno opisano u poglavlju 5. Rezultat svih aktivnosti ovoga procesa jest *Zahtjev za servisom* (*Request for Service – RFS*). Taj zahtjev može se odnositi podjednako na najam već gotovih servisa u oblaku, kao i na izradu novih servisa.



Slika 7.2: ***Životni ciklus IT usluga u oblaku*** *prema [51]*

Nakon definiranja servisa, potencijalni oblačni korisnik može pokušati **Pronaći servis** koji zadovoljava tražene zahtjeve, primjerice korištenjem namjenskih tražilica (*service discovery engine*), sukladno poslovnim i tehničkim zahtjevima te financijskim ograničenjima. Jasno, oni koji su se odlučili za izradu vlastitih servisa preskaču ovu fazu te se bave razvojem. Proces

otkrivanja servisa daje *popis servisa s karakteristikama* kao što su opis funkcionalnosti, pružatelj usluge oblačnog računarstva, cjenik usluga i druge.

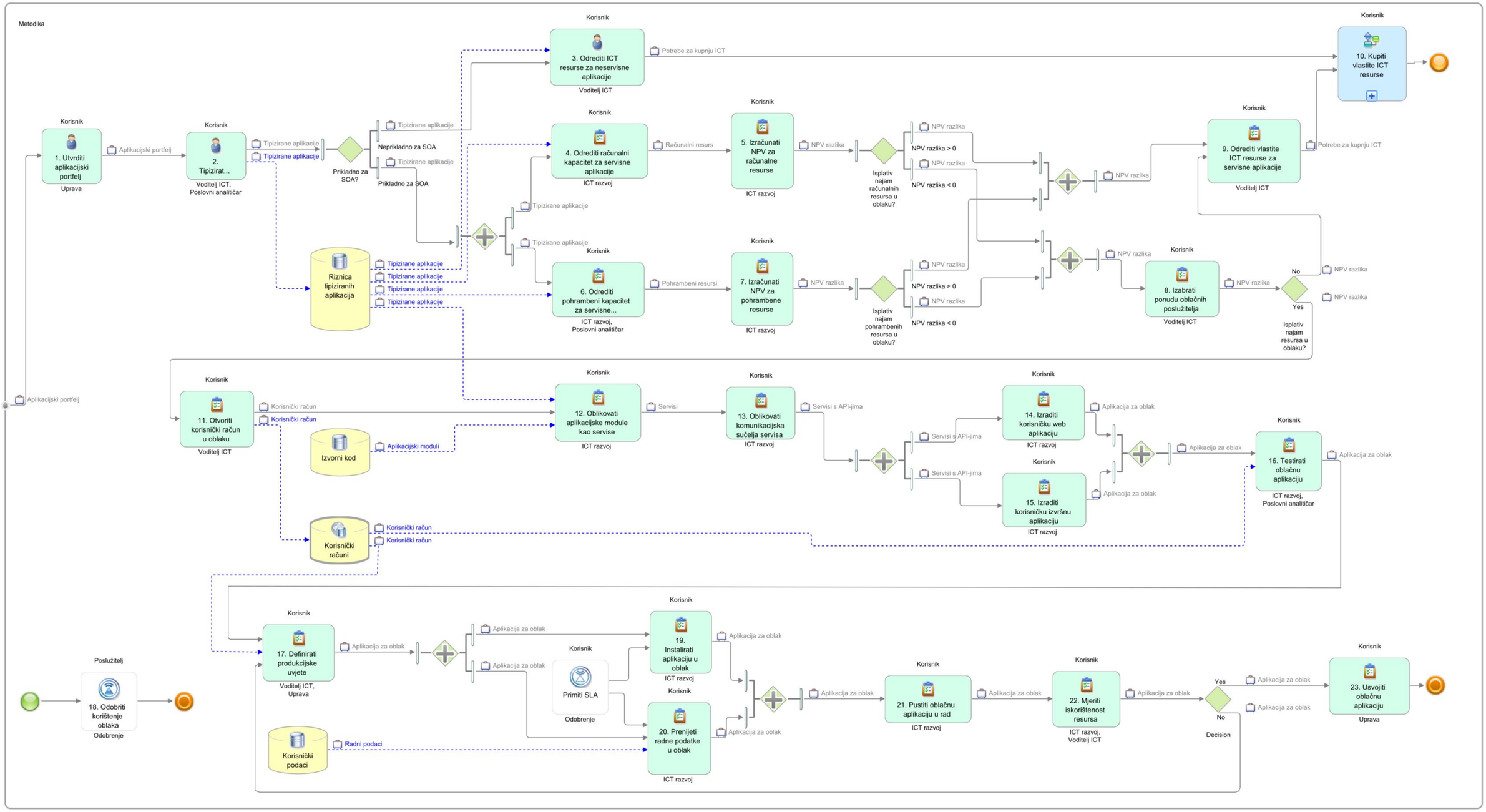
Ako popis servisa s karakteristikama sadrži prihvatljivu ponudu za najam servisa u oblaku, u slijedećoj fazi potrebno je **Ugovoriti korištenje servisa** u oblaku. To je također potrebno i ako je prethodni korak preskočen zbog izgradnje vlastitih servisa. Tada će se, nakon njihovog programiranja (na simulatorima oblačnih poslužitelja u okviru razvojnih alata koje obično stavlja na raspolaganje pružatelj usluge oblačnog računarstva), treba pronaći oblačnog poslužitelja kod kojeg će ti servisi biti smješteni. Sukladno svojstvu samoposlužnosti, ugrađenoj u definiciju oblačnog računarstva, ovaj korak može biti automatiziran, to jest proveden bez neposredne interakcije s osobljem poslužitelja. Rezultat ove faze je *ugovor o korištenju* (*Service Level Agreement* – SLA) oblačnih resursa. Preporuča se da ugovor bude u skladu s ITIL (*Information Technology Infrastructure Library*) specifikacijama.

Nakon izgradnje servisa i njihovog smještaja u produkcijsku okolinu ili pak najma potrebnih računalnih resursa, servise treba povezati u funkcionalni informacijski sustav. To se čini metodama **orkestracije** i **kompozicije**, koje se najčešće provode izradom korisničkih programa za povezivanje servisa u smislen i cjelovit poslovni sustav. Drugim riječima, potrebno je izraditi *korisnička sučelja* (u obliku klasičnih izvršnih programa ili web aplikacija) preko kojih će se servisi usklađeno pozivati i pomoću kojih će se dobivati rezultati njihovoga rada.

Zadnje faze životnog ciklusa IT usluga u oblaku su iterativna faza korištenja (*consumption*) i nadzora (*monitoring*) servisa u oblaku. U skladu sa svojstvom elastičnosti računalnih oblaka, mjeri se potrošnja računalnih i pohrambenih resursa servisa pokretanih u oblaku te se zauzimaju ili otpuštaju računalni resursi u skladu s poslovnim potrebama. Postupak je automatiziran i traje do kraja životnog ciklusa oblačnog servisa.

7.3 Grafički i opisni prikaz metodike KOR

Na slici 7.3 koriste se standardni simboli i konstrukti BPMN-a. Pojedini koraci metodike su prikazani kao poslovni procesi, koji na slici imaju naveden naziv koraka (unutar simbola za proces) te nositelje posla iznad a odgovorne organizacijske jedinice ispod simbola za proces. Slijed koraka prikazan je strelicama. Natpisi nad strelicama predstavljaju ulaze i ishode pojedinih koraka metodike. Metode prikladne za neki korak upisane su na slici ispod simbola za radne korake. Podatkovni objekti predstavljaju informacijske sadržaje koji su potrebni za metodu korištenu u nekom koraku, ali se korištenjem ne troše.



Slika 7.3: ***Grafički prikaz metodike KOR***

U sljedećim potpoglavljima ovog poglavlja opisana su 23 pojedinačna koraka predložene metodike. Tekstualni opis metodike KOR najbolje je čitati uz istovremeno praćenje slike 7.3 na kojoj je redni broj koraka usklađen s rednim brojem u tekstu koji slijedi. Sinteza predložene metodike po koracima i korištenim metodama prikazana je u tablicama 7-I i 7-II: prva tablica sadrži opis poslovnih aktivnost, a druga opis poslovnih sadržaja (*business item*). Uz svaki korak metodike KOR bit će kratak komentar o njegovom uklapanju u V-dijagram, odnosno u životni ciklus IT usluga u oblaku prema [51] i kratko opisan u potpoglavlju 7.2.

Korak 1: Utvrditi aplikacijski portfelj

Uprava tvrtke ili ustanove zainteresirane za korištenje računalnih oblaka treba **utvrditi aplikacijski portfelj** tako da ***metodom analize*** pobroji i opiše sve aplikacije postojećeg informacijskog sustava i analizira temeljne osobine svake od njih:

- Funkcionalnost: Što aplikacija radi?
- Ulazi i izlazi: Koji su ulazni podaci potrebni za rad aplikacije? Što je izlazni rezultat obrade tih podataka?
- Koliki je opseg baze podataka nad kojom aplikacija radi? Kakvo je očekivano povećanje (ili smanjenje) postojećeg opsega baze podataka?
- Kakvi su hardverski i softverski preduvjeti (*requirements*) za izvođenje aplikacije?
- Učestalost izvođenja: Kada, tko i u kojoj mjeri koristi aplikaciju?

Rezultat ovoga koraka je precizan *aplikacijski portfelj* u kojem je svaka aplikacija opisana navedenim temeljnim karakteristikama. Prema V-modelu, ovaj korak spada u analizu zahtjeva.

Korak 2: Tipizirati aplikacije

U ovom koraku treba **tipizirati aplikacije** iz popisa izrađenom u prethodnom koraku. Voditelj ICT-a i poslovni analitičari utvrđuju ***metodom analize resursa*** tehnička i sigurnosna svojstva svake od tih aplikacija, s ciljem određivanja udovoljava li ona preduvjetima za uključivanje u servisno orijentiranu arhitekturu, u skladu s razmatranjima u [34]). Preduvjeti koje treba posebno pažljivo analizirati su:

- povjerljivost i sigurnost podataka
- specifične hardverske potrebe i klijentske platforme
- zakonska ograničenja (primjerice, dozvoljeno zemljopisno područje za pohranu podataka) i ograničenja korporativne politike (poput ovlasti zaposlenika, broja i

organizacijske razine osoblja koje sudjeluje u donošenju odluka, politike nabavke softvera i hardvera, i slično)

- procijenjeni broj korisnika kod prosječnog i vršnog opterećenja
- modularnost aplikacije ili mogućnost da se ona restrukturira po modulima i
- prihvatljivo organizacijsko vrijeme izvođenja aplikacije $\mathrm{T_{ORG}}$.

Za svaku aplikaciju koja odgovara uvjetima za uključivanje u SOA treba razmotriti osnovnu arhitekturu i prepoznati one module koje mogu predstavljati servise u SOA smislu [59]. Također, za svaku aplikaciju, bilo da je izrađena kao web servis ili je izvedena na prethodnim tehnologijama (*legacy application*), u ovom koraku se određuje prihvatljivo organizacijsko vrijeme njenog izvođenja T_{ORG} koje je detaljnije opisano u poglavlju 5, a potrebno je za uspješno obavljanje posla.

Rezultat ovoga koraka je riznica *tipiziranih aplikacija*. Prema V-modelu, ovaj korak također spada u poslove analize zahtjeva.

Korak 3: Odrediti ICT resurse za neservisne aplikacije

Iz riznice tipiziranih aplikacija voditelj ICT treba **odrediti ICT resurse za neservisne aplikacije**, a to su one koje nisu pogodne za uklapanje u servisno orijentiranu arhitekturu. Na primjer, mobilne aplikacije za skladišnu evidenciju obično nisu pogodne za uklapanje u SOA, a trebaju i specifičan hardver (dlanovnike s barkod čitačima) i softver. Određivanje količine i vrste tih resursa provodi se također ***metodom analize resursa***.

Ovaj korak rezultira popisom (*bill of material*) *potreba za kupnju ICT*, koji može obuhvaćati računala i pripadni softver poput operacijskog sustava, diskovne kapacitete, specijalizirani hardver poput barkod čitača, RFID opreme, industrijskih kontrolera, plotera i slično. Ovim korakom se, dakle, utvrđuje vrsta i količina potrebnog hardvera za neservisne aplikacije u postojećem organizacijskom sustavu. Ovaj korak odgovara u V-modelu poslovima analize zahtjeva.

Korak 4: Odrediti računalni kapacitet za servisne aplikacije

Za aplikacije iz riznice tipiziranih aplikacija koje jesu pogodne za uklapanje u SOA model, u ovom se koraku mora **odrediti potrebni računalni kapacitet** za njihovo pokretanje. Da bi se taj kapacitet mogao odrediti, osoblje iz ICT razvoja koristi metodu ***mjerenja aplikacijskih performansi*** opisanu u poglavlju 5 ovoga rada. Korištenjem te metode, temeljene na teoriji višefaktorskog parcijalnog plana pokusa, analiziraju se bitni aplikacijski čimbenici koji utječu na brzinu rada i zaposjedanje resursa. Ti čimbenici su nezavisne varijable (na primjer, količina i snaga računala, složenost aplikacija, broj CRUD akcija nad bazom podataka i

slično) u matematičkom modelu koji se koristi za izračunavanje očekivanog vremena izvođenja aplikacije (prema postupku koji je razvijen u poglavlju 5). Te nezavisne varijable moraju biti prikladno podešene što znači da moraju biti takve da vrijeme obavljanja aplikacijskog posla bude u okvirima prihvatljivog organizacijskog vremena T_{ORG}. Ukratko, treba rezervirati dovoljan računalni kapacitet da bi se izvođenje pojedinih servisa odvijalo unutar zadanog T_{ORG}.

Rezultat ovoga koraka jest saznanje o količini *računalnih resursa* potrebnih za uspješno pokretanje SOA aplikacija. Pri tome računalne resurse predstavljaju virtualna ili fizička računala sa svojstvima koja zahtijeva aplikacija (poput brzine i arhitekture procesora, količine RAM-a, propusnosti mrežne kartice, tipa operacijskog sustava i drugih). Ovim korakom se, dakle, mjeri količina potrebnih računala za servisne aplikacije IS-a što u V-modelu spada u poslove analize zahtjeva.

Korak 5: Izračunati NPV za računalne resurse

Temeljem saznanja o potrebnoj količini računalnih resursa za pokretanje SOA aplikacija, izračunatih u prethodnom koraku, služba ICT razvoja treba **izračunati neto sadašnju vrijednost za računalne resurse**. Ta vrijednost treba se izračunati za dva pretpostavljena slučaja:

- kada su računalni resursi u vlasništvu korisnika i
- kada su računalni resursi u smješteni generičkom oblaku.

Proračuni se provode prema modelu za trošak računalnih resursa, koji je detaljno opisan u potpoglavlju 6.2.

Rezultat tih proračuna je *račun sadašnjih vrijednosti računalnih resursa* za vlastita računala i za računala najmljena u oblaku. Budući da je razlika ta dva računa bitan čimbenik za odluku o kupnji ili najmu, zajednička oznaka za njih je *NPV razlika*. Ako je ta razlika veća od nule, to jest ako je neto sadašnja vrijednost najma računalnih resursa manja od neto sadašnje vrijednosti njihove kupnje, onda valja razmišljati o najmu računalnih resursa u oblaku. Konačna provjera isplativosti takvog najma u komercijalnim uvjetima obavlja se u koraku 8. Budući da se ovim korakom izračunavaju parametri nužni za proračun ukupne cijene računalnih resursa, riječ je i dalje o poslovima analize zahtjeva prema V-modelu.

Korak 6: Odrediti pohrambeni kapacitet za servisne aplikacije

Za aplikacije iz riznice tipiziranih aplikacija, koje su pogodne za uklapanje u SOA model, u ovom koraku treba **odrediti potreban diskovni kapacitet** za smještaj podataka s kojima aplikacija radi. Da bi se taj kapacitet mogao odrediti, osoblje iz ICT razvoja, uz pomoć poslovnih analitičara, koristi ***metodu analize resursa***. Naime, predviđanje kretanja opsega

podatkovne pohrane ovisi o vrsti, opsegu i obliku tih podataka (što spada u nadležnost ICT osoblja), ali i o broju i interesima korisnika tvrtke i drugim svojstvima poslovnog sustava (što bolje poznaje poslovni analitičar). Na primjer, predviđanje kretanja opsega podatkovne pohrane uvelike će se razlikovati za knjigovodstveni servis, proizvodnu tvrtku ili kabelsku televiziju. Ono što je zajedničko u pokušajima tog predviđanja jest da spadaju u domenu poslovne logike pa je stoga za uspjeh predviđanja nužna konzultacija između razvojnih stručnjaka i poslovnih analitičara.

Rezultat ovoga koraka jest saznanje o količini *pohrambenih resursa* potrebnih za smještaj podataka s kojima rade SOA aplikacije informacijskog sustava. Pohrambeni resursi koriste se za spremanje strukturiranih podataka (u relacijskim ili nerelacijskim bazama podataka), blobova ili redova poruka. Fizički, ti pohrambeni resursi su realizirani kao klasični diskovi ili SSD. Ovim korakom se, dakle, definira količina potrebnih diskovnih kapaciteta za informacijski sustav, što u V-modelu spada u poslove analize zahtjeva.

Korak 7: Izračunati NPV za pohrambene resurse

Temeljem saznanja o potrebnoj količini diskovnih resursa potrebnih za smještaj podataka s kojima rade SOA aplikacije izračunatog u prethodnom koraku, ICT razvoj treba **izračunati neto sadašnju vrijednost za pohrambene resurse**. Tu vrijednost treba također izračunati za dva slučaja: kada su pohrambeni resursi u vlasništvu i kada su smješteni u generičkom oblaku. ***Proračuni se provode prema Walkerovom modelu za izračun pohrambenih resursa*** iz potpoglavlja 6.3.

Rezultat tih proračuna je *račun sadašnjih vrijednosti pohrambenih resursa* za vlastitu podatkovnu pohranu i podatkovnu pohranu najmljenu u oblaku. Budući da je razlika ta dva računa bitan čimbenik za odluku o kupnji ili najmu, zajednička oznaka za njih je *NPV razlika*.

Ako je ta razlika već od nule, to jest ako je neto sadašnja vrijednost najma podatkovne pohrane u oblaku manja od neto sadašnje vrijednosti kupnje pohrambene infrastrukture, onda valja razmišljati o najmu pohrambenih resursa u oblaku. Konačna provjera isplativosti takvog najma u komercijalnim uvjetima obavlja se u koraku 8. Budući da se ovim korakom izračunavaju parametri nužni za proračun ukupne cijene pohrambenih resursa, riječ je i dalje o poslovima analize zahtjeva prema V-modelu.

Korak 8: Izabrati ponudu oblačnih poslužitelja

U koracima 5 i 7 dobiveni su računi za pohrambene i računalne resurse za slučaj najma u generičkom, neprofitnom oblaka i za slučaj kupnje. Na temelju tih poznatih neto sadašnjih vrijednosti računalnih i pohrambenih resursa treba **provjeriti ponudu oblačnih poslužitelja**. To znači da voditelj ICT, za količine računalnih i pohrambenih kapaciteta izračunate u

koracima 4 i 6, a za koje se neto sadašnja vrijednost najma izračunata u koracima 5 i 7 pokazala manjom od neto sadašnje vrijednosti kupnje, treba ***metodom analize resursa*** izračunati trošak najma u konkretnim, komercijalnim računalnim oblacima.

Valja naglasiti da taj trošak, čak i ako je dosta povoljniji u odnosu na nabavku računalnih i pohrambenih resursa u vlasništvu, nije jedini čimbenik odluke "kupnja ili najam." Drugi utjecajni faktori mogu biti kvaliteta usluge (to jest SLA uvjeti) te tehnička svojstva oblačne platforme (na primjer, temelji li se na .NET-u ili Javi, omogućuje li i relacijsku pohranu podataka i tako dalje)

Izračun tog stvarnog troška (u čijoj je jezgri *NPV razlika*) opisan je, zajedno s čitavim algoritmom odluke "kupnja ili najam", u poglavlju 6.5 i temelji se na taksonomiji komercijalnih oblačnih usluga iz potpoglavlja 6.4. Ovim korakom prelazi se, prema V-modelu, na osmišljavanje konceptualnog dizajna oblačnog informacijskog sustava.

Korak 9: Odrediti vlastite ICT resurse za servisne aplikacije

U ovom koraku treba odrediti vlastite ICT resurse za servisne aplikacije iz riznice tipiziranih koje jesu prikladne za SOA, ali ne i za pokretanje u oblaku. Potrebe za tim resursima, zapravo već ***izračunate po Walkerovom modelu*** za vlastite računalne i pohrambene resurse u koracima 5 i 7, voditelj ICT zbraja potom s potrebama za neservisne aplikacije koje su izračunate u koraku 3.

Ovaj korak rezultira popisom (*bill of material*) *potreba za kupnju ICT* koji može obuhvaćati računala i pripadni softver poput operacijskog sustava, diskove i druge pohrambene resurse, i slično. Tu se, dakle, računa količina potrebnog hardvera za servisne aplikacije informacijskog sustava smještene na vlastitom hardveru, što u V-modelu također spada u poslove analize zahtjeva.

Korak 10: Kupiti vlastite ICT resurse

Do ovoga koraka već je određeno koji će se servisi i podaci informacijskog sustava pokretati odnosno držati u oblaku, a koji na vlastitim serverima odnosno diskovima. Mjerodavni donositelji odluka u tvrtki ili ustanovi stoga mogu pokrenuti postupak **kupnje vlastitih ICT resursa**, softverskih, hardverskih i komunikacijskih, namijenjenih pokretanju dvije klase aplikacija iz riznice tipiziranih i to:

- neservisne aplikacije i
- servisno orijentirane aplikacije za koje je, zbog raznih razloga, odlučeno da će biti smještene i pokretane na ICT infrastrukturi u vlasništvu.

Postupak nabave složeni je proces koji ovisi o mnogo čimbenika (na primjer, u Hrvatskoj, da li se radi o privatnoj tvrtki ili tvrtki u većinskom državnom vlasništvu odnosno javnoj ustanovi za koje je propisan postupak javne nabave) pa ovdje neće biti detaljnije razmatran jer ne spada u doseg ovog rada.

Korak 11: Otvoriti korisnički račun u oblaku

Da bi mogle započeti tehničke pripreme za korištenje oblaka, potrebno je da ovlaštena osoba (primjerice, voditelj ICT-a) **otvori korisnički račun** kod odabranog oblačnog poslužitelja. Ograničeni razvoj i rudimentarno testiranje može, doduše, započeti i bez tog koraka stoga što namjenski razvojni alati u pravilu sadrže i odgovarajuće simulatore računalnih oblaka, no ozbiljan razvoj i pravo testiranje oblačnog rješenja može se provoditi samo uz stalnu provjeru funkcionalnosti u realnim uvjetima te jasan uvid u raspoložive oblačne resurse i njihove postavke. To je moguće samo ako je otvoren *korisnički račun* u oblaku kod odabranog oblačnog poslužitelja.

Budući da je ovo prvi korak realizacije softverskog rješenja u oblaku te da se njime odabire i tehnologija izrade tog rješenja, radi se o fazi konceptualnog dizajna prema V-modelu.

Korak 12: Oblikovati aplikacijske module kao servise

Ključni podaci za provođenje ovog koraka se nalaze u riznici tipiziranih aplikacija, koja je izrađena u koraku 2. **Aplikacijske module** onih tipiziranih aplikacija iz te riznice koje su prilagodljive za SOA (i za koje je odlučeno da će ubuduće biti smještene u oblaku) **treba oblikovati kao servis**, radi prilagodbe servisno orijentiranoj arhitekturi i izvođenja u računalnom oblaku. Prilagodbu provodi osoblje ICT razvoja ***metodama objektno orijentiranog dizajna i programiranja (OOD i OOP)***, uz korištenje izvornog koda dosadašnjih aplikacija. Pokazni primjer za ovaj korak je izvedba ogledne aplikacije, koja je prikazana u poglavlju 5.1 ovog rada i korištena za mjerenje aplikacijskih performansi za različite vrijednosti nezavisnih varijabli.

Rezultat ovog koraka su moduli tipiziranih aplikacija koji su ocijenjeni kao prikladni za uklapanje u servisno orijentiranu arhitekturu te preuređeni kao *servisi*. Svaki modul sadrži poslovnu logiku za obavljanje jednog organizacijskog poslovnog procesa.

Ovaj korak spada u izvedbeni (*implementation*) dio V-modela. Ovaj prijelaz od konceptualnog dizajna na programiranje moguć je zbog pretpostavke da metodika KOR implicira postojanje dosadašnjeg informacijskog sustava i izvornog koda njegovih sastavnih aplikacija. Drugim riječima, dizajn informacijskog sustava, konceptualni i detaljni, je proveden već ranije, tijekom razvoja. Doduše, bit će potrebno ponoviti neke poslove koji spadaju u te

faze V-modela, ali isključivo radi prilagodbe postojećih rješenja servisno orijentiranoj arhitekturi i oblačnoj infrastrukturi.

Korak 13: Oblikovati komunikacijska sučelja servisa

Pri oblikovanju aplikacijskih modula kao servisa (u prethodnom koraku) koriste se standardni protokoli i tehnike pristupa tim servisima (HTTP i HTTPS, odnosno SOAP i REST) koje, uglavnom, nije potrebno posebice programirati. Zbog potencijalne asinkronosti izvođenja servisa (uzrokovane dugotrajnošću procesa koje servis predstavlja ili zbog velikog opsega podataka s kojima servis radi) može se pojaviti potreba za dodatnim i specifičnim programskim sučeljima servisa smještenih u oblaku i namijenjenih podršci toj asinkronosti. Najčešće se radi o programskim sučeljima koja spremaju razna stanja procesa u red poruka, strukturiranu tablicu ili blob. Sve takve specifičnosti rješavaju se u ovom koraku, korištenjem ***metoda objektno orijentiranog dizajna i programiranja (OOD i OOP)*** te ***metodama za modeliranje baza podataka (KVS ili relacijskih)***.

Po uspješnom provođenju ovih koraka dobivaju se *servisi s API-jima* (*application program interface*) pomoću kojih se obavlja poziv tih servisa i čitanje rezultata njihovog rada. U ovom koraku izvode se aktivnosti koje ulaze u dvije faze V-modela: detaljni dizajn (ako postoje doista specifični komunikacijski, sigurnosni i drugi zahtjevi glede pokretanja servisa u oblaku) i softverska izvedba.

Korak 14: Izraditi korisničku web aplikaciju

Ako novi informacijski sustav, temeljen na SOA načelima i korištenju oblačnog računarstva, ima korisnička sučelja predviđena za izvođenje unutar web preglednika (*browser*), onda treba **izraditi korisničku web aplikaciju**. Drugim riječima, osoblje iz ICT razvoja treba izraditi web aplikaciju, također smještenu u računalnom oblaku i namijenjenu za posredovanje između čovjeka kao korisnika i poslovne logike sadržane u servisima koji su oblikovani u koraku 12. Razvoj takve aplikacije provodi se korištenjem ***metoda objektno orijentiranog dizajna i programiranja (OOD i OOP)***.

Rezultat ovoga koraka jest *aplikacija za oblak*, modularna i sastavljena od servisa (oblikovanih u koraku 12) koji sadrže poslovnu logiku te korisničkog sučelja (oblikovanog u idućem koraku) izvedenog kao web aplikacija ili izvršni program. Servisi su elementi servisno orijentirane arhitekture, smješteni su u oblaku i dostupni standardnim protokolima i tehnikama (HTTP i HTTPS odnosno SOAP i REST). Svaki servis je izveden iz jednog modula postojeće aplikacije koja je prikladna za SOA, ali se dosad nije tako izvodila. Ovaj korak odnosi se na tri faze V-modela: konceptualni dizajn, detaljni dizajn i softversku izvedbu.

Korak 15: Izraditi korisničku izvršnu aplikaciju

Ako novi informacijski sustav temeljen na SOA načelima i korištenju oblačnog računarstva ima korisnička sučelja u obliku klasičnog izvršnog programa predviđenog za izvođenje na radnim stanicama (*workstation*), onda ICT razvoj treba **izraditi korisničku izvršnu aplikaciju**. Razvoj se obavlja korištenjem ***metoda objektno orijentiranog dizajna i programiranja (OOD i OOP)***.

Rezultat ovog koraka jest *aplikacija za oblak*, modularna i sastavljena od servisa koji sadrže poslovnu logiku (oblikovanih u koraku 12) te korisničkog sučelja, izvedenog kao web aplikacija ili izvršni program. Servisi su elementi servisno orijentirane arhitekture, smješteni su u oblaku i dostupni standardnim protokolima i tehnikama (HTTP i HTTPS odnosno SOAP i REST). Svaki servis je izveden iz jednog modula postojeće aplikacije koja je prikladna za SOA, ali se dosad nije tako izvodila. Ovaj korak također se odnosi na tri faze V-modela: konceptualni dizajn, detaljni dizajn i softversku izvedbu.

Korak 16: Testirati oblačnu aplikaciju

Do ovoga koraka završene su aplikacije za izvođenje u oblaku te sastavljene od SOA servisa i korisničkih aplikacija pa osoblje ICT razvoja, u suradnji s poslovnim analitičarima, treba **testirati oblačnu aplikaciju**. Takvo testiranje treba provesti u radnim uvjetima što znači da testirano rješenje mora biti smješteno u oblaku, odnosno njezino pokretanje na razvojnom simulatoru oblaka više nije adekvatno za testne svrhe. Smještaj aplikacije za oblak u radnu okolinu radi testiranja se provodi na temelju podataka o korisničkom računu koji je otvoren u koraku 11.

Rezultat ovog koraka je testirana *aplikacija za oblak*. Testiranje je iterativno, no radi jasnoće prikaza cijelog postupka metodike KOR nisu uvedeni takvi ponavljajući koraci (razumljivi svakom razvojnom inženjeru), koji su po BPMN-u mogući. Ovaj korak spada, prema V-modelu, u fazu integracije i testiranja.

Korak 17: Definirati produkcijske uvjete

Unatoč elastičnosti i optimalnoj nabavi, pri korištenju oblačne platforme potrebno je definirati **početne** parametre poput inicijalne količine angažiranih računala i njihove snage, početnog kapaciteta podatkovne pohrane, zemljopisne lokacije oblačnog centra (ako poslužitelj dozvoljava taj izbor), minimalne mrežne propusnosti i slično.

Također je važno definirati svojstva i programe na klijentskim računalima s kojih se pokreće korisnička web ili izvršna aplikacija. Na primjer, ako je korisnička aplikacija izvedena kao bogata (*rich*) web aplikacija, temeljena na pregledničkim dodacima poput Flasha ili Silverlighta, onda ti preglednički dodaci trebaju biti instalirani na klijentskim računalima.

Definiranje produkcijskih uvjeta provodi se ***uvidom u aplikacijska svojstva***. Rezultat te aktivnosti je *aplikacija za oblak*, dopunjena dokumentacijom o produkcijskim uvjetima. Aplikacija se šalje oblačnom poslužitelju na odobrenje radi postavljanja u produkcijsku okolinu, obično na automatiziran način u skladu sa svojstvom samoposlužnosti oblaka. Takve aktivnosti odnose se na pripremu za redovito korištenje te stoga po V-modelu spadaju u verifikaciju i validaciju.

Korak 18: Odobriti korištenje oblaka

Oblačni poslužitelj zaprima korisničko rješenje (s dokumentacijom o početnim produkcijskim uvjetima) i **odobrava korištenje oblaka**, to jest inicijalno zauzimanje resursa i pokretanje oblačne aplikacije u produkcijskim uvjetima na način definiran ugovorom o usluzi (*service-level agreement* – SLA). Prema samoj definiciji oblačnog računarstva kao samposlužnog sustava (poglavlje 3), ovaj je proces automatiziran, iako se ovlaštene osobe oblačnih poslužitelja može izravno kontaktirati radi dogovaranja o eventualnim posebnim uvjetima korištenja usluga oblačnog računarstva.

Odobrenje korištenja oblaka najčešće je formalizirano *ugovorom o usluzi – SLA* – između oblačnog poslužitelja i korisnika oblačnog računarstva. Takve aktivnosti također se odnose na pripremu za rad te stoga po V-modelu spadaju u verifikaciju i validaciju.

Korak 19: Instalirati aplikaciju u oblak

Po dobivanju odobrenja za produkcijsko korištenje oblaka u SLA obliku, osoblje iz ICT razvoja treba **instalirati aplikaciju u oblak**. Postupak je najčešće poluautomatiziran jer se posljednja, ispravna testna verzija oblačne aplikacije može proglasiti za produkcijsku tako da se vrijednosti testnih konfiguracijskih postavki (na primjer, URL-ovi za pristup servisima i web aplikacijama u oblaku) promijene u produkcijske.

Nakon provedene instalacije, *aplikacija za oblak* je djelomično spremna za produkcijsku uporabu. Potpuna spremnost postiže se nakon prijenosa radnih podataka u oblak. Instalacijom aplikacije u oblak definitivno se potvrđuje mogućnost korištenja oblačne platforme pa ovaj korak spada u fazu verifikacije i validacije po V-modelu.

Korak 20: Prenijeti radne podatke u oblak

Po dobivanju odobrenja za produkcijsko korištenje oblaka u obliku usuglašenog SLA, osoblje iz ICT razvoja treba **prenijeti radne podatke u oblak**. Podaci se raznim ***metodama upravljanja bazom podataka*** prenose iz trenutnih baza podataka, a odnose se na poslovnu domenu te korisničke račune radnika koji će koristiti aplikaciju za oblak. Postupak je najčešće poluautomatiziran jer se posljednja, ispravna testna verzija baze podataka može

proglasiti za produkcijsku, tako da se vrijednosti testnih konfiguracijskih postavki (na primjer, URL-ovi za pristup servisima i web aplikacijama u oblaku) promijene u produkcijske.

Nakon prijenosa radnih podataka, *aplikacija za oblak* je djelomično spremna za produkcijsku uporabu. Potpuna spremnost postiže se nakon instalacije oblačne aplikacije u prethodnom koraku. Prijenosom radnih podataka u oblak definitivno se potvrđuje mogućnost korištenja oblačne platforme pa ovaj korak spada u fazu verifikacije i validacije po V-modelu.

Korak 21: Pustiti oblačnu aplikaciju u rad

Puštanje u rad je aktivnost koja obuhvaća korisničku obuku, ovjeru ispravne instalacije i integriteta podataka, početak periodičke izrade sigurnosnih kopija programa i podataka te ispunjavanje specifičnih uvjeta na klijentskim računalima, ako su takvi definirani u koraku 17.

Za pokretanje cjelovitog informacijskog sustava, uključivo i dijelova smještenih u oblaku, potrebno je da budu završene sve njegove programske komponente: servisi koji se pokreću u oblaku i korisničke aplikacije (koje mogu biti izvedene kao web aplikacije također smještene u oblaku i dostupne preko web preglednika, ili pak kao klasični izvršni programi za radne stanice).

Nakon puštanja u rad kojeg provodi osoblje ICT razvoja, *aplikacija za oblak* podvrgava se stalnoj dijagnostici opisanoj u slijedećem koraku. Puštanjem u rad ostvaruje se faza rada i održavanja po V-modelu.

Korak 22: Mjeriti iskorištenje resursa

Ovaj korak se ponavlja tijekom čitavog radnog vijeka oblačne aplikacije radi maksimalnog iskorištenja oblačnih svojstava elastičnosti i optimalne nabave. Ta su svojstva obrazložena u poglavlju 4. Mjeri se iskorištenost unajmljenih računala, kapacitet zaposjednute podatkovne pohrane, učestalost pristupa pojedinim vrstama podataka te iskorištenost drugih oblačnih resursa. Također se provjerava da li je postignuto zadovoljavajuće organizacijsko vrijeme T_{ORG} izvođenja aplikacije koje je definirano u koraku 2. Ako za neki servis oblačne aplikacije to vrijeme nije zadovoljavajuće, u ovom se koraku također angažiraju dodatni resursi u oblaku, kako bi se postigao zadovoljavajući T_{ORG}. I obratno, ako se ustanovi da su računala angažirana za obavljanje određenih servisa premalo opterećena, u ovom se koraku smanjuje broj računala angažiranih za te servise.

Mjerenja obavlja osoblje ICT razvoja u sklopu faze održavanja životnog ciklusa programskog proizvoda, koristeći se pri tome ***metodama računalne dijagnostike***. U tu svrhu koriste se alati koje daju na raspolaganje pružatelji usluga oblačnog računarstva ili specifični alati razvijeni u ICT službi korisnika.

Osiguranje elastičnosti i optimalne nabave, odnosno skaliranje unajmljenih resursa u skladu s trenutnim opterećenjem, može se obavljati automatizirano ili ručno. Na primjer, u Amazonovom oblaku za automatsko skaliranje postoji usluga AutoScaling koja se naplaćuje (tablica 6-VI). Međutim, korisnik može i samostalno pratiti opterećenje oblačne aplikacije i njenih komponenti te osobno, preko upravljačke konzole Amazonovog oblaka (koja je izvedena kao web aplikacija) upravljati skaliranjem. Na slične načine upravlja se skaliranjem i pri ostalim pružateljima usluga oblačnog računarstva.

Ako se *aplikacija za oblak* pokaže dovoljno elastičnom za prilagodbu promjenjivim opterećenjima te pokaže i ostale prednosti oblačne platforme na koju je postavljena, onda se šalje upravi tvrtke na konačno usvajanje. Mjerenje iskorištenosti resursa jedna je od temeljnih stavki u fazi rada i održavanja po V-modelu.

Korak 23: Usvojiti oblačnu aplikaciju

Završni korak metodike KOR je usvajanje oblačnog rješenja od strane naručitelja. Valja primijetiti da se i nakon ove aktivnosti i dalje iterativno mjeri iskorištenost unajmljenih resursa, kako je to opisano u prethodnom koraku. Također, aplikacija smještena u oblak, prelazi nakon ovog koraka iz faze razvoja u fazu održavanja. Poslovni uvjeti mogu zahtijevati povremene izmjene i dorade radi usklađivanja s novim okolnostima u kojima djeluje organizacija. Ta iterativnost posljednjeg koraka jest jedna od temeljnih svojstava faze rada i održavanja po V-modelu.

Dosljednom primjenom ovih koraka metodike KOR mogu se osigurati uvjeta za ekonomično i učinkovito korištenje servisno orijentiranih aplikacija u komercijalnom računalnom oblaku.

Tablica 7-I: **Tablični prikaz metodike KOR**

Korak	*Naziv koraka*	*Ulazi*	*Metode i tehnike*	*Izlazi*
1.	Utvrditi aplikacijski portfelj	Aplikacijski portfelj	Dekompozicija	Aplikacijski portfelj
2.	Tipizirati aplikacije	Aplikacijski portfelj	Analiza resursa	Tipizirane aplikacije
3.	Odrediti ICT resurse za neservisne aplikacije	Tipizirane aplikacije	Analiza resursa	Potrebe za kupnju ICT
4.	Odrediti računalni kapacitet za servisne aplikacije	Tipizirane aplikacije	Mjerenje aplikacijskih performansi	Računalni resursi
5.	Izračunati NPV za računalne resurse	Računalni resursi	Walkerov model za računalne resurse	NPV razlika
6.	Odrediti pohrambeni kapacitet za servisne aplikacije	Tipizirane aplikacije	Analiza resursa	Pohrambeni resursi
7.	Izračunati NPV za pohrambene resurse	Pohrambeni resursi	Walkerov model za pohrambene resurse	NPV razlika
8.	Provjeriti ponudu oblačnih poslužitelja	NPV razlika	Analiza resursa	NPV razlika
9.	Odrediti vlastite ICT resurse za servisne aplikacije	NPV razlika	Walkerov model za računalne resurse Walkerov model za pohrambene resurse	Potrebe za kupnju ICT
10.	Kupiti vlastite ICT resurse	Potrebe za kupnju ICT	Standardni načini i zakonske procedure za nabavku opreme	Kraj
11.	Otvoriti korisnički račun u oblaku	NPV razlika	Registracija korisničkog računa preko weba	Korisnički račun
12.	Oblikovati aplikacijske module kao servise	Tipizirana aplikacija Aplikacijski moduli Korisnički račun	OOD OOP	Servisi
13.	Oblikovati komunikacijska sučelja servisa	Servisi	OOD OOP Modeliranje baza podataka	Servisi s API-jima
14.	Izraditi korisničku web aplikaciju	Servisi s API-jima	OOD OOP	Aplikacija za oblak
15.	Izraditi korisničku izvršnu aplikaciju	Servisi s API-jima	OOD OOP	Aplikacija za oblak
16.	Testirati oblačnu aplikaciju	Korisnički račun Aplikacija za oblak	Testiranje	Aplikacija za oblak
17.	Definirati produkcijske uvjete	Korisnički račun Aplikacija za oblak	Analiza resursa	Aplikacija za oblak
18.	Odobriti korištenje oblaka	Aplikacija za oblak		Aplikacija za oblak
19.	Instalirati aplikaciju u oblak	Aplikacija za oblak SLA	Deployment	Aplikacija za oblak
20.	Prenijeti radne podatke u oblak	Aplikacija za oblak Radni podaci SLA	Upravljanje bazom podataka	Aplikacija za oblak
21.	Pustiti oblačnu aplikaciju u rad	Aplikacija za oblak	Testiranje Probni rad	Aplikacija za oblak
22.	Mjeriti iskorištenost resursa	Aplikacija za oblak	Računalna dijagnostika	Aplikacija za oblak
23.	Usvojiti oblačnu aplikaciju	Aplikacija za oblak		Kraj

Tablica 7-II: ***Poslovni sadržaji metodike KOR***

Naziv poslovnog sadržaja	*Opis poslovnog sadržaja*
Aplikacija za oblak	Servisno orijentirana aplikacija namijenjena pokretanju u računalnom oblaku. Sastoji se od SOA servisa i korisničkih sučelja. Korisnička sučelja mogu biti izvedena u web tehnologiji ili kao običan izvršni program.
Aplikacijski moduli	Izvorni kod trenutnih aplikacija informacijskog sustava to jest njihovih funkcionalnih cjelina – modula.
Aplikacijski portfelj	Popis trenutnih aplikacija informacijskog sustava. Neke je aplikacije moguće preurediti u servisno orijentiranu arhitekturu – SOA.
Korisnički račun	Korisnički račun tvrtke ili ustanove pri komercijalnom oblačnom poslužitelju. Preduvjet za korištenje računalnog oblaka.
NPV razlika	Razlika iznosa računa sadašnje vrijednosti računalnih, pohrambenih i drugih resursa u najmu spram vlasništva.
Podaci o oblačnom poslužitelju	Podaci o oblačnom poslužitelju nužni za razvoj i pokretanje aplikacija u njegovome oblaku: tehnologija, URL-ovi, cjenik oblačnih usluga, uvjeti korištenja, i drugo.
Pohrambeni resursi	Diskovni kapaciteti za pohranu podataka.
Potrebe za kupnju ICT	Račun materijala (hardvera i softvera) kojeg treba kupiti radi korištenja aplikacija.
Potrebe za najam ICT	Račun materijala (hardvera i softvera) kojeg treba najmiti radi korištenja aplikacija.
Radni podaci	Radne baze podataka informacijskog sustava.
Računalni resursi	Računala, virtualna ili fizička, koja su potrebna za funkcioniranje informacijskog sustava. Karakterizira ili procesorski kapacitet, veličina RAM-a, propusnost mrežne kartice, operacijski sustav i druga, izvedena svojstva, poput cijene CPU vremena.
SLA	Ugovor o usluzi između oblačnog poslužitelja i korisnika oblačnog računarstva.
Servisi	Aplikacijski moduli izvedeni kao servisi unutar servisno orijentirane arhitekture. Svaki takav servis (modul) obavlja neki dio poslovne logike (aktivnost ili čitav proces).
Servisi s API-jima	SOA servisi s eventualnim specifičnim programskim sučeljima nužnim za ispravnu primjenu poslovne logike ili asinkrono izvođenje.
Tipizirane aplikacije	Popis trenutnih aplikacija informacijskog sustava klasificiranih prema određenim karakteristikama kao što su povjerljivost i sigurnost podataka, specifične hardverske potrebe i klijentske platforme, zakonska ograničenja i ograničenja korporativne politike, procijenjeni broj korisnika, modularnost u smislu prikladnosti za SOA, prihvatljivo organizacijsko vrijeme izvođenja, i drugo.
Odobrenje	Odobrenje oblačnog poslužitelja korisniku za uporabu i korištenje oblačnih resursa.

7.4 Oblačno računarstvo i SOA

Zaključno s ovim poglavljem u radu je pokazano sljedeće:

1. S nabavnog gledišta, postoje dva glavna načina korištenja računalnih resursa – kupnja ili komercijalni najam odgovarajuće ICT opreme. Geneza ideja o najmu ICT resursa i njihovom korištenju po utrošku (kao kod drugih javno dostupnih resursa) opisana je u poglavljima 1 i 2.
2. U današnje vrijeme, komercijalni računalni oblaci su način na koji tvrtke i ustanove mogu riješiti svoje potrebe za promjenljivim ICT resursima, tako da ih unajme i troše prema korištenju. Geneza i svojstva računalnih oblaka, kao suvremenog koncepta korištenja ICT resursa najmom, umjesto kupnje vlastitih, opisana je u poglavlju 3.
3. Najam i trošenje računalnih resursa prema korištenju može za korisnika biti povoljnije od kupnje. Potencijalne ekonomske prednosti takvog pristupa postavljene su u poglavljima 3.7 i 3.8, a opširnije obrazložene u četvrtom poglavlju.
4. Postoje egzaktni načini izračuna moguće uštede korištenja računalnih oblaka spram nabave vlastitih ICT resursa. Ako je priroda posla koji se želi računalno i informatički podržati takva, da je sa zakonske i sigurnosne strane moguće koristiti računalne oblake (pri čemu su potencijalni problemi opisani u poglavlju 3.9) onda je za izračun isplativosti najma ICT resursa spram kupnje potrebno ustanoviti koji aplikacijski parametri značajno utječu na zauzeće i trošenje ICT resursa.
5. Metode za mjerenje relevantnih aplikacijskih parametara, koji utječu na zauzeće ICT resursa, razvijene su i opširno opisane u poglavlju 5. Te su metode nazvane *Mjerenje aplikacijskih performansi*. Temelje se na utvrđivanju funkcijske ovisnosti koja se dobiva izvedbom optimalnog broja mjerenja prema višefaktorskom planu pokusa. Ovaj postupak je primjenjiv za sve vrste aplikacija, ali su moguće prilagodbe matematičkog modela, zavisno o rezultatima mjerenja.
6. U poglavlju 6 je pokazano kako odabrati aplikacije za smještaj i izvođenje u oblaku radi uštede na troškovima za ICT. Postupak se temelji na povezivanju mjerenja aplikacijskih performansi, ekonomskim proračunima za izračun troškova računalnih i pohrambenih resursa u vlasništvu i najmu (koje je postavio Edward Walker [108, 109], a detaljnije su razrađeni i provjereni u potpoglavljima 6.2 i 6.3 ovog rada) te taksonomiji komercijalnih oblačnih usluga značajnih poslužitelja.

U ovom poglavlju, metodikom KOR je sistematiziran način izgradnje aplikacija za oblak, razvijen u poglavlju 6. **Namjena metodike KOR jest planirati korištenje oblačnog**

računarstva u procesu izgradnje rješenja temeljenih na servisno orijentiranoj arhitekturi. U tom smislu posebice su značajni sljedeći koraci te metodike:

- U koraku 2 predviđeno je tipiziranje aplikacija po raznim kriterijima među kojima je najbitniji prikladnost za redizajn aplikacija radi njihovog uključivanja u SOA. Potom je u trećem koraku predviđeno planiranje nabave potrebnih ICT resursa za one aplikacije iz portfelja koje nisu prikladne za SOA.
- Korak 8, provjera ponude oblačnih poslužitelja, namijenjen je za razlučivanje onih SOA servisa informacijskog sustava koji jesu prikladni za pokretanje u oblaku od onih za koje oblak nije dobra platforma (zbog ekonomskih ili drugih razloga, primjerice sigurnosne ili zakonske naravi). Zbog toga je u devetom koraku predviđeno planiranje nabave ICT resursa za SOA aplikacije neprikladne za oblak.

Čitatelju ovoga rada može se, međutim, činiti čudnim što servisno orijentirana arhitektura nije na neki dodatni, specifični način povezana s oblačnim računarstvom. Drugim riječima, možda će se zapitati: *"Zar je najbitnije poglavlje ovoga rada, barem sudeći po naslovu, zaista ovako kratko?"*

Kako je već detaljno analizirano i utvrđeno u poglavljima 2, 3 i 4, s mnogo referenci na adekvatne i recentne radove, **računalni oblaci su evolucija dosadašnjih oblika naprednog računarstva**. Oblačno računarstvo neće izbaciti iz uporabe niti "debele klijente", niti klasične serverske farme u vlasništvu, a vjerojatno niti starije oblike *utility* računarstva poput gridova. Oblačno računarstvo je naprosto novi koncept korištenja i dopuna lepezi tehnologija koje razvojni inženjeri i poslovni stručnjaci mogu upotrijebiti kao ICT resurs.

Istovremeno, uporaba računalnih oblaka temelji se na servisnoj paradigmi opisanoj u poglavlju 3.2. To znači da su pojedine funkcionalnosti oblaka enkapsulirane te se mogu koristiti odvojeno od ostalih, po načelima plaćanja po utrošku. Na primjer, moguće je koristiti samo računalne resurse u smislu procesorske snage (s pripadnim značajkama poput količine RAM-a) odvojeno od pohrane u oblaku ili drugih funkcionalnosti (primjerice, servisne sabirnice). Korisnici Google AppEnginea ne moraju koristiti Googleovu e-poštu (spada u PaaS ponud, vidljivu u tablici 6-VI) nego mogu na neki drugi način izvesti slanje e-poruka i obavijesti. U tom smislu, kao i zbog činjenice naglašene u prethodnom odlomku da računalni oblaci neće zamijeniti ostale oblike računarstva, možemo zaključiti: **većina informacijskih sustava koji koriste oblačne resurse će biti hibridni informacijski sustavi**. Svojstvo hibridnosti znači da će neki dijelovi informacijskog sustava biti smješteni u komercijalnom oblaku, a neki na računalima u vlasništvu tvrtke ili organizacije. Informacijski sustavi nekih manjih organizacija će možda u cijelosti biti smješteni u oblaku i dostupni korisnicima preko tankog klijentskog računala ili putem web preglednika, no smatramo da će to biti rjeđi slučaj.

Komercijalni oblačni poslužitelji zapravo računaju s hibridnošću budućih informacijskih sustav, iako takva poslovnu strategiju za sada nitko nije izričito iskazao. Razlozi tome su jasni – svaki oblačni poslužitelj htio bi vidjeti svoje korisnike kako u najvećoj mogućoj mjeri koriste (i plaćaju) baš njegovu što opsežniju oblačnu ponudu. Ipak, tehnološki predvodnici oblačnog računarstva ne mogu više ostvariti takve odnose koji bi rezultirali nametanjem ovisnosti o samo jednom oblačnom poslužitelju. Otvoreni industrijski standardi usvojeni su u najširoj profesionalnoj zajednici te ih se moraju pridržavati i tehnološki predvodnici. Dapače, neki od njih su izrazito aktivni u oblikovanju općeprihvaćenih normi. Stoga su sve oblačne platforme inherentno interoperabilne s okruženjem jer koriste do sada razvijene i općenito prihvaćene standarde kao što su TCP/IP, HTTP, HTTPS, SOAP i XML te REST i JSON. Standardizacija oblačnog računarstva tu ne staje – u poglavlju 3.9 je spomenuto da se radi i na standardizaciji pohranjivanja slika virtualnih računala, kako bi se olakšao prelazak od jednog oblačnog poslužitelja drugom. Takav je razvoj usmjeren na ostvarivanje barem djelomične interne kompatibilnosti oblačnih platformi raznih poslužitelja.

Detaljnije razmatranje ovih odnosa prelazi okvire ovog rada, u čijem kontekstu je najvažnija vanjska interoperabilnost komercijalnih oblačnih platformi koja se ostvaruje kroz prihvaćanje i korištenje standardnih internetskih protokola i tehnika. **Koncept servisno orijentirane arhitekture temelji se na istim komunikacijskim metodama i tehnikama kao i oblačno računarstvo**. Iz toga slijedi zaključak da računalni oblaci mogu biti, i jesu, vrlo pogodna platforma za učinkovito korištenje servisno orijentirane arhitekture. Metodika KOR daje smjernice o tome kako i pod kojim uvjetima uklopiti servisno orijentirane aplikacije u računalni oblak.

7.5 Osvrt na Hipotezu 1

Ogledna aplikacija, koja je pripremljena za izvođenje u oblaku radi mjerenja performansi, izvedena je u konkretnim alatima – Visual Studio 2008, tehnologijama – .NET Framework i Azure SDK i na konkretnoj oblačnoj platformi – Windows Azure. Moguće ju je također izvesti i na drugim tehnologijama te na drugim oblačnim platformama. Naime, već su i u Azure oblaku podržane ne-Microsoftove tehnologije (što je navedeno u poglavlju 5) te domaći (*native*) programski jezici kao što je C++. Drugi pak oblačni poslužitelji (kao Amazon) imaju u svojoj ponudi usluga oblačnog računarstva i opciju najma virtualnih računala pod Microsoftovim serverskim operacijskim sustavima. Nadalje, Python je, primjerice, jedan od dva podržana programska jezika za razvoj na Google AppEngineu, no bitno je primijetiti da je Python višeplatformni jezik koji ima i .NET inkarnaciju zvanu IronPython. I sam .NET framework ima svoju Linux varijantu zvanu Mono Framework. Općenito se može zaključiti

da je, uz dobro konfigurirana virtualna računala u nekom oblaku, u načelu sasvim moguće pokrenuti većinu .NET aplikacija izravno na oblačnoj platformi temeljenoj na Linuxu.

Metodika KOR primijenjena je u izgradnji ogledne aplikacije te se pokazalo da može opisati postupke za razvoj aplikacija u oblaku, odnosno hibridnih informacijskih sustava čiji su dijelovi smješteni u oblaku. Metodika KOR usmjerava proces razvoja hibridnih informacijskih sustava pa je stoga i sama formalizirana prema BPM standardu. Stoga smatramo da je dokazana Hipoteza 1 ovoga rada koja glasi: *Metodika razvoja hibridnih SOA rješenja može se izgraditi temeljem normiranih metoda, tehnika i notacija za modeliranje, usklađivanje i opisivanje poslovnih procesa.*

7.6 Sažetak

U ovom poglavlju definirana je metodika KOR, kao sinteza istraživanja provedenih u prethodnim poglavljima. Ona usmjerava postupke razvoja i uklapanja servisno orijentiranih aplikacija u novu tehničku platformu računalnih oblaka. Metodika KOR je izvedena kao proces te je stoga i formalizirana po notaciji koja je definirana BPM standardom. Provodi se u 23 koraka čiji je krajnji cilj objektivizirati odluke o tome koje aplikacije informacijskog sustava tvrtke ili ustanove je isplativo smjestiti u oblak i tamo ih izvoditi, za koje treba nabaviti vlastite ICT resurse i kako sve to provesti na sistematičan način. Aplikacije koje se pokreću u oblaku treba preoblikovati kao servise unutar servisno orijentirane arhitekture.

Pokazano je također da su računalni oblaci dobra tehnička platforma za smještaj i pokretanje SOA rješenja iz dva razloga: (1) stoga što su protokoli i tehnologije korištene za njihovu interoperabilnost s vanjskim svijetom jednaki standardnim protokolima i tehnologijama za interoperabilnost SOA rješenja te (2) zato što su filozofije korištenja slične, temeljene na servisnoj paradigmi. S obzirom da se u metodici KOR koriste normirane metode, tehnike i notacije za modeliranje, usklađivanje i opisivanje poslovnih procesa koje su obuhvaćene industrijskim standardom BPM, te da je uporabivost metodike provjerena kroz njezinu uspješnu primjenu u izgradnji ogledne aplikacije, može se smatrati da je ovim poglavljem dokazana prva hipoteza ovog rada.

8 Zaključak

U ovom radu objašnjen je pojam oblačnog računarstva (*cloud computing*) i njegovi odnosi i veze s kronološki ranijim tehnologijama kao što je distribuirano računarstvo (*distributed computing*), paralelno računarstvo (*parallel computing*), uslužno računarstvo (*utility computing*) te mrežno računarstvo (*grid computing*). Pokazano je kako su sve te ranije tehnologija preteča, ali i svojstvo oblačnog računarstva u smislu da su računalni oblaci izgrađeni kao distribuirani sustavi sposobni uposliti masovni računalni paralelizam u poslovne i znanstvene svrhe te široko dostupni na zahtjev poput klasičnih komunalnih usluga. Ranija tehnologija poznatija kao gridovi također ima neka svojstva oblačnog računarstva, no znatno je složenija za uporabu i sastavljena od heterogenih sustava te stoga prikladna za znanstvenu zajednicu.

Oblačno računarstvo je jasno definirano kao skup mrežnih servisa namijenjenih pružanju raznih računalnih usluga, od digitalne pohrane podataka do kompletnih softverskih rješenja, kojeg odlikuju slijedeće osobine: dostupnost temeljem standardnih informacijskih i komunikacijskih tehnologija, utemeljenost na računalnoj virtualizaciji, proširivost, naplata prema korištenju (*pay as you go* ili *pay per use*) te korištenje i skaliranje na zahtjev. Objašnjene su servisne paradigme IaaS, PaaS i SaaS i razlozi zašto se najbolje koriste upravo kroz računalne oblake – stoga što oba koncepta povezuje servisna paradigma korištenja i dostupnost korištenjem standardnih mrežnih tehnologija i protokola.

Pokazane su snažne strane uporabe računalnih oblaka, ekonomske i tehničke. Ekonomske prednosti svode se na isplativost uzrokovanu elastičnošću i optimalnom nabavom. Elastičnost znači da se računalni resursi mogu angažirati kada su potrebni i otpustiti kada te potrebe nestane, dok optimalna nabava znači da ne treba unaprijed nabavljati računalne resurse (koji u redovnim situacijama nisu potrebni) samo radi pokrivanja vršnih opterećenja. Za ilustraciju tehničkih prednosti oblačnog računarstva predstavljene su razne klase aplikacija koje posebice dobro mogu koristiti prednosti računalnih oblaka. Zajedničko tim aplikacijama jest mogućnost paralelizacije izvođenja algoritama ili podrška djelatnostima gdje se javljaju vršna opterećenja.

Opširno su obrađene specifičnosti uporabe i korištenja računalnih oblaka u odnosu na klasične serverske farme, kako s ekonomskog, tako i s korisničkog i razvojnog gledišta. Poseban je naglašeno kako svatko tko razmišlja o uporabi oblaka, bilo kao korisnik, bilo kao poslužitelj, mora moći izračunati uvjete isplativosti zakupa oblačnih resursa odnosno postavljanja oblaka i davanja tih resursa u najam.

Naime, korisnik mora imati način usporedbe komercijalne cijene i resursne ponude oblačnih poslužitelja s troškovima nabave i održavanja vlastite opreme istih mogućnosti. U tim

procjenama važnu ulogu ima, osim klasičnih ekonomskih stavki poput sadašnje i buduće vrijednosti (*net present value* i *net future value*) te amortizacije, još i Mooreov zakon.

Potencijalni poslužitelj pak mora znati strukturu cijene oblačnih usluga kako bi mogao formirati ponudu koja ga neće odvesti u poslovne gubitke, ali će istovremeno biti dovoljno konkurentna da privuče određen broj oblačnih korisnika. U usporedbi s klasičnom, omanjom serverskom farmom, oblačnom poslužitelju ide na ruku ekonomija razmjera koja dolazi do izražaja u velikim računalnim centrima poput oblačnih, a opisuje pojavu smanjenja prosječne cijene proizvoda ili usluge što se više povećava količina tog isporučenog proizvoda ili usluge.

Postavljanje modela za navedene izračune je istraživački fokus ovoga rada. Da bi se ti proračuni mogli provesti, pokazano je da treba:

1. odrediti varijable koje opisuju svojstva složene aplikacije koja se može smjestiti u oblak ili na lokalna računala
2. izmjeriti kako te varijable utječu na komercijalne parametre pod kojima se unajmljuju oblačni resursi, a kako na parametre kojima se određuje nabava vlastitih računala.

Svrha tih mjerenja je utvrditi oblik zavisnosti vremena izvođenja aplikacije T o nezavisnim varijablama koje opisuju izvođenje te aplikacije na računalnim resursima u oblaku i vlasništvu. Pokazano je da značajnih varijabli ima 6 te se mogu grupirati u aplikacijske, podatkovne i resursne. Utvrđeno je da aplikacijske varijable čine složenost algoritma $\boldsymbol{a}$ i broj CRUD akcija po upitu $\boldsymbol{q}$. Podatkovni aspekt opisan je opsegom podataka koji se šalje u oblak radi obrade $\boldsymbol{u}$ i veličinom baze podataka nad kojom se izvodi obrada $\boldsymbol{b}$. Resursne varijable su broj angažiranih računala u oblaku $\boldsymbol{r}$ i njihova snaga $\boldsymbol{q}$.

Kao objekt mjerenja izabrana je ogledna aplikacija koja dobro predstavlja veliki skup aplikacija tipičnih za poslovne informacijske sustave. Ogledna se aplikacija odnosi na prepoznavanje zadanih oblika u odnosu prema referentnim oblicima iz baze podataka. Ona se pokazala prikladnom za variranje aplikacijskih i podatkovnih varijabli u dovoljno širokom rasponu za istraživanje te je pripremljena za izvođenje na računalnim resursima u oblaku na Microsoftovoj platformi Azure.

Tijekom probnog istraživanja provedeno je ukupno 11 mjerenja vremena T, pri čemu su svi parametri varirali između njihove donje i gornje očekivane vrijednosti. Pokazalo se da je srednja vrijednost za T, postignuta kad su svi parametri postavljeni na srednju vrijednost iznosi 156 sekundi te da varira od 11 do 367 sekundi za razne vrijednosti parametara. Na temelju saznanja iz predistraživanja postavljen je višefaktorski parcijalni plan mjerenja tipa 2^{6-2} prema kojem je izvedeno 16 mjerenja. Rezultati mjerenja obrađeni su statističkim

metodama multivarijantne analize te je izračunat matematički model – polinom – kojim se opisuje ponašanje ogledne aplikacije u oblaku. Taj polinom ima oblik:

$$T = 0{,}631 + 14{,}993 \cdot a + 0{,}135 \cdot u + 0{,}702 \cdot q - 0{,}840 \cdot r - 2{,}160 \cdot a \cdot r$$

To je linearni model, s četiri signifikantna parametra ($\boldsymbol{a}$, $\boldsymbol{u}$, $\boldsymbol{q}$ i $\boldsymbol{r}$) te jednom dvofaktornom interakcijom $a \cdot r$, čija je adekvatnost potvrđena korištenjem metode ANOVA. Ovaj matematički model ponašanja ogledne aplikacije u oblaku je interpretiran grafički i analitički s različitih aspekata. Na temelju toga su date smjernice i izrađen nomogram za praktično određivanje koliko računalnih resursa treba unajmiti u oblaku za izvođenje aplikacije, čije se značajke mogu definirati vrijednostima signifikantnih parametara uz organizacijski prihvatljivo vrijeme izvođenja te aplikacije.

Bitno je, međutim, primijetiti slijedeće: u slučaju da se neka druga aplikacija značajno razlikuje od ogledne ne bi se neposredno mogli primijeniti zaključci izvedeni na oglednoj aplikaciji, ali bi metodološki postupak ostao isti. Dakle, iako je ogledna aplikacija izvedena u .NET tehnologiji i pokrenuta u Microsoftovom Azure oblaku, metodološki postupak provjeren obradom rezultata mjerenja ogledne aplikacije je univerzalan i neovisan o platformi i tehnologiji. Za neku drugu aplikaciju bilo bi potrebno izvesti mjerenja po shemi 2^{n-k} i ponoviti izračun radi određivanja adekvatnog polinoma. Time je pokazano kako je provedenim istraživanjem i ostvarenim rezultatima izvedivo postavljanje višedimenzionalne metrike potreba za SOA servisima smještenim u računalnom oblaku to jest formaliziran je tehnički postupak za postavljanje takve metrike.

Potom je u poglavlju 6 obrađena problematika pune cijene vlasništva nad oblačnim centrom, zanimljiva prije svega potencijalnim investitorima u takva postrojenja, ali i svim korisnicima računalnih oblaka i analitičarima koje zanima procjena zarade oblačnih poslužitelja. Nakon toga su predstavljena dva matematičko-ekonomska modela koja je utemeljio Edward Walker s Teksaškog sveučilišta, jedan za izračun troška CPU vremena (to jest računalnih resursa), a drugi za izračun troška pohrambenih resursa (to jest diskovnih kapaciteta). Značajka oba ta modela je da se njihovim korištenjem mogu izračunati troškovi, jednako za slučaj najma potrebnih računalnih resursa (u hipotetskom, generičkom, neprofitnom računalnom oblaku), kao i za slučaj kada se računala i diskovi nabavljaju. Zatim je u poglavlju 6.4 provedena taksonomija komercijalnih oblačnih usluga na primjeru tri značajna poslužitelja, Amazon, Microsoft i Google. Pomoću te taksonomije provjerena je valjanost troškovnih modela.

Potpoglavljem 6.6 je predstavljena i metrika odlučivanja o kupnji ili najmu kao algoritam od 7 koraka koji obuhvaćaju saznanja iz poglavlja 5 te prethodnih dijelova šestog poglavlja:

1. **Određivanje organizacijskog vremena:** Definiranje traženog organizacijskog vremena prihvatljivog za izvođenje aplikacije.

2. **Procjena obujma podataka:** Procjena obujma podataka za trajnu pohranu te stope rasta i smanjenja pohrane.
3. **Izračun količine računalnih resursa:** Izračun količine potrebnih računalnih resursa za svaku radnu aplikaciju temeljem metrike iz poglavlja 5.
4. **Izračun troška računalnih resursa:** Prema prethodno izračunatoj količini potrebnih računala za radne aplikacije, izračunava se, korištenjem Walkerovog modela za trošak CPU vremena predstavljenog u potpoglavlju 6.2, ukupni trošak kupnje vlastitih računala (TCO) te trošak najma istih računala u generičkom oblaku.
5. **Izračun troška pohrambenih resursa:** Prema zadanoj količini potrebnog prostora za pohranu podataka nad kojima radi svaka radna aplikacija izračunava se, korištenjem Walkerovog modela za trošak pohrambenih resursa predstavljenog u potpoglavlju 6.3, trošak pohrane na vlastitim pohrambenim kapacitetima te trošak najma istih kapaciteta u generičkom oblaku.
6. **Izračun najma kod komercijalnih poslužitelja:** Za svaku radnu aplikaciju izračunava se, prema taksonomiji predstavljenoj u poglavlju 6.4, trošak najma u komercijalnim računalnim oblacima.
7. **Usporedba cijene najma i kupnje:** Uspoređuju se cijene vlastitih računalnih i pohrambenih resursa izračunate u koracima 4 i 5 s cijenama najma izračunatima u koraku 6 te se temeljem toga izvlače zaključci o povoljnosti kupnje odnosno najma. Dodatno, može se izračunati cijena podizanja vlastitog podatkovnog centra (temeljem modela opisanog u potpoglavlju 6.1 i rezultata koraka 2 i 3) te okvirno procijeniti zarada oblačnog poslužitelja, ako je svoje resurse namijenio najmu. Taj postupak je proveden temeljem izračuna troškova računalnih i pohrambenih resursa za generički oblak u koracima 4 i 5 i usporedbe tih troškova sa stvarnom cijenom najma izračunatom u koraku 6.

Potom je, na kraju šestog poglavlja, provedena provjera postupka odlučivanja na slučaju zamišljene aplikacije s karakteristikama poput opisanih u poglavlju 5. Uspješnim uspostavljanjem takve metrike za odlučivanje o kupnji ili najmu **dokazana je hipoteza 2** ovoga rada.

U zadnjem, sedmom poglavlju definirana je, kao sinteza dosadašnjih istraživanja, **metodika KOR** (*korištenje oblačnog računarstva*) namijenjena uklapanju razvoja servisno orijentirane arhitekture u novu tehničku platformu računalnih oblaka. Metodika KOR, prikazana kao proces po BPM standardu, sastoji se od 23 koraka s krajnjim ciljem pomaganja u odluci koje aplikacije informacijskog sustava tvrtke ili ustanove je isplativo smjestiti u komercijalni oblak i tamo ih izvoditi, a za koje treba nabaviti vlastite ICT resurse. One aplikacije koje se pokreću

u oblaku trebaju biti preoblikovane kao servisi prema načelima servisno orijentirane arhitekture. **Pokazano je da su računalni oblaci dobra tehnička platforma za smještaj i pokretanje SOA rješenja** iz dva razloga: stoga što su protokoli i tehnologije korištene za njihovu interoperabilnost s drugim računalima jednaki standardnim protokolima i tehnologijama za interoperabilnost SOA rješenja te zato što su i filozofije korištenja slične, temeljene na servisnoj paradigmi. Budući da se u metodici KOR koriste normirane metode, tehnike i notacije za modeliranje usklađivanje i opisivanje poslovnih procesa obuhvaćene industrijskim standardom BPM, te budući da je njenim korištenjem izgrađena ogledna aplikacija, **dokazana je i prva hipoteza** ovog rada.

Za daljnja istraživanja mogu se preporučiti slijedeće smjernice:

- Izučiti mogućnost primjene metodike KOR u razvoju servisno orijentiranih aplikacija na tehnologijama drugih oblačnih poslužitelja. Iako su zaključci ovoga rada općeniti, interesantno bi bilo dodatno ih potvrditi na izgradnji rješenja u nekom drugom oblaku umjesto Microsoftovog.
- Istražiti granice validnosti metodike KOR kod razvoja aplikacija vrlo velikih resursnih potreba, reda veličine stotina računala na više.
- Ispitati da li je moguće razviti uzorke SOA servisa specifičnih za računalne oblake. Ovime bi se s jedne strane dopunio skup uzoraka prikupljenih u knjizi Thomasa Erla [26], a s druge strane detaljnije razradili neki koraci metodike KOR koji se odnose na oblikovanje SOA servisa.

9 Literatura

1 Amazon (2010). *AWS Case Study:TC3 Health*. http://aws.amazon.com/solutions/case-studies/tc3-health/, učitano 13.2.2011

2 Amazon (2011). *Amazon EC2 Instance Types*. http://aws.amazon.com/ec2/instance-types/, učitano 13.2.2011.

3 Amazon (2011). *Amazon RDS FAQs*. http://aws.amazon.com/rds/faqs/, učitano 13.2.2011.

4 APC (2004). Sawyer, V. *Calculating Total Power Requirements for Data Centers. White Paper 3, 1-16. http://dcmms.com/whitepapers/2.pdf.* učitano 13.2.2011.

5 Armbrust, M; Fox, A; Griffith, R; Joseph, A.D; Katz, R; Konwinski, A; Lee, G; Patterson, D; Rabkin, A; Stoica, I; Zaharia, M. *Above the Clouds: A Berkeley View of Cloud Computing.* Technical Report No UCB/EECS-2009-28, University of California at Berkeley, Berkeley, 2009.

6 Armbrust, M; Fox, A; Griffith, R; Joseph, A.D; Katz, R; Konwinski, A; Lee, G., Patterson, D; Rabkin, A; Stoica, I; Zaharia, M. *A View of Cloud Computing. Communications of the ACM*, 53(4):50-58, 2010.

7 Babcock, C. *Management Strategies for the Cloud Revolution*: *How Cloud Computing Is Transforming Business and Why You Can't Afford to Be Left Behind.* McGraw-Hill, Columbus, Ohio, 2010.

8 Bahree, A; Mulder, D; Cicoria, S; Peiris, C; Pathak, N. Pro WCF: Practical Microsoft SOA Implementation. Springer-Verlag, New York, 2007.

9 Bell M. *Service-Oriented Modeling (SOA): Service Analysis, Design, and Architecture*. John Willey&Sons Inc, Hoboken, New Yersey, 2008.

10 Berger P; Maurer R. *Experimental Design with Applications in Management, Engineering and the Science*. Thompson Learning, London, 2002.

11 Brocke vom, J; Thomas, O: Sonnenberg, C. *Towards an Economic Justification of Service Oriented Architectures - Measuring the Financial Impact*. In *Proceedings of the Fourteenth Americas Conference on Information Systems,* pages 1-10, Toronto, Canada, 2008.

12 Brumec, S. *Razvoj mobilnih informacijskih sustava. M*agistarski rad, Svečilište u Zagrebu, Fakultet organizacije i informatike, Varaždin, 2008.

13 Buyya, R; Venugopal, S. *A Gentle Introduction to Grid Computing and Technologies*. *Computer Society of India (CSI) Communications*, 1-11, 2005.

14 Cáceres, J; Vaquero, L; Rodero-Merino, L; Polo, A; Hierro, J. Service Scalability Over the Cloud. In *Handbook of Cloud Computing*, pp 357-377, Editors: Borko Furht, Armando Escalante, Springer, New York, 2010.

15 Carolan, J; Gaede, S. *Introduction to Cloud Computing Architecture*. White paper, Sun Microsystems, 2009.

16 Chappell, D. *Introducing Windows Azure*. David Chappell & Associates, San Francisco, California, 2009.

17 Chee, B; Franklin, C. *Cloud Computing: Technologies and Strategiesof the Ubiquitous Dana Center*. CRC Press, Boca Raton, Florida, 2010.

18 Close, K; Dean, D; Iyer, S; Saleh, T. *Cloud Computing in Large Enterprises: Questions for the C-Suite*, The Boston Consulting Group Inc., Boston, September 2010.

19 Cloud Computing Journal (2010), Banerjee, U. *Architectural Consideration for Building Cloud Applications*. http://cloudcomputing.sys-con.com/node/1316025, učitano 13.2.2011.

20 *Creative Commons (2009). Open Cloud Manifesto*, http://www.opencloudmanifesto.org/, učitano 13.2.2011.

21 Dan, A; Johnson, R; Arsanjani, A. *Information as a Service: Modeling and Realization. In Proceedings of the International Workshop on Systems Development in SOA Environments (SDSOA'07, ICSE Workshops 2007)*, pages 1-6,Mineapolis, 2007.

22 Dana Center Knowledge (2009). *Ensuring High Service Levels in Cloud Computing*. http://www.datacenterknowledge.com/archives/2010/06/21/ensuring-high-service-levels-in-cloud-computing/. učitano 13.2.2011.

23 Dean, D; Saleh, T. *Capturing the Value of Cloud Computing: How Enterprises Can Chart Their Course to the Next Level*. The Boston Consulting Group Inc, Boston, November 2009.

24 Dreher, P; Vouk, M; Sills, E; Averitt, S. Evidence for a Cost Effective Cloud Computing Implementation Based Upon the NC State Virtual Computing Laboratory Model. In *Advances in Parallel Computing, High Speed and Large Scale Scientific Computing*, pp 236-250, Editors: Wolfgang Gentzsch, Lucio Grandinetti, Gerhard Joubert, Volume 18, IOS Press, Elsevier Science, Amsterdam, 2009.

25 Emgu CV. http://www.emgu.com/wiki/index.php/Main_Page, učitano 13.2.2011.

26 Erl, T; Booch, G. *SOA Design Patterns*, Person Education, Boston, Massachusetts, 2009.

27 Fielding, R.T; Taylor, R.N. Principled Design of the Modern Web Architecture. *ACM Transactions on Internet Technology*, 2(2):115-150, 2002.

28 Fingar, P. *Cloud-Oriented Business Architecture. A BPTrends Column*, June:1-4, 2009.

29 Fingar, P. *Dot.Cloud: The 21st Century Business Platform Build on Cloud Computing.* Meghan-Kiffer Press, Tampa, Florida, 2009.

30 Forbes, D.(2010). *Getting Real about NoSQL and the SQL-Isn't-Scalable Lie*. http://www.yafla.com/dforbes/Getting_Real_about_NoSQL_and_the_SQL_Isnt_Scalable_Lie/, učitano 13.2.2011.

31 Fortier, P; Michel, H. *Computer Systems Performance Evaluation and Prediction*. Elsevier, Boston, 2003.

32 Foster, I; Yong, Z; Raicu, I; Shiyong, L. *Cloud Computing and Grid Computing 360-Degree Compared*. White Paper, Department of Computer Sccience, University of Chicago, 2008.

33 Furht, B. Cloud Computing Fundamentals. In *Handbook of Cloud Computing*, pp 3-19, Editors: Borko Furht, Armando Escalante, Springer, New York, 2010.

34 Gerić, S; *Razvoj metrika za utvrđivanje razine primjenjivosti servisno-orijentiranih arhitektura pomoću modela organizacijske zrelosti*. Doktorska disertacija, Sveučilište u Zagrebu, Fakultet organizacije i informatike Varaždin, 2009.

35 Ghosh, S.: *Distributed Systems: An Algorithmic Approach*. Chapman & Hall/CRC Computer & Information Science Series, London, 2006.

36 Gilbert, S; Lynch, N. *Brewer's conjecture and the feasibility of consistent, available, partition-tolerant Web services. ACM SIGACT News*, 33(2):51-59, 2002.

37 Green, S. *Information Systems Design*, International Thomsom Computer Press, London, 1996.

38 Greenberg, A; Hamilton, J; Maltz, D. A; Patel, P. The *Cost of Cloud: Research Problems in Data Center Networks*. Microsoft Research, Redmont, Washington, 2009.

39 Harms, R; Yamartino, M. *The Economics of the Cloud*. White Paper, Microsoft Research, Redmont, Washington, November 2010.

40 Havey M. *Essential Business Process Modeling*, O'Reilly Media Inc., Sebastopol, California, 2005.

41 High Scalability (2009). Atwood, J. *Stack Overflow Architecture*. http://highscalability.com/. 13.2.2011.

42 Hitney, B(2010). *Thoughts on Windows Azure Pricing*. http://blog.structuretoobig.com/post/2010/02/09/Thoughts-on-Windows-Azure-Pricinge280a6.aspx, učitano 13.2.2011.

43 Hubbard, D. *How to Measure Anything-Finding the Value of Intangibles in Business*, John Wiley&Sons, Hoboken, New Jersey, 2010.

44 Iamnitchi, A; Foster, I; Nurmi, D.C. *A Peer-to-Peer Approach to Resource Discovery in Grid Environments*. Technical Report TR-2002-06, Department of Computer Sccience, University of Chicago, 2002.

45 IBM. *Seeding the Clouds: Key Infrastructure Elements for Cloud Computing*. IBM Corporation, Systems and Technology Group, Somers, New York, November 2009.

46 IEEE Computer Society (2009). Leavitt, N. *Is Cloud Computing Ready for Prime-Time.* http://www.computer.org/portal/web/computingnow/0209/theme/computer/, učitano 12.02.2011.

47 InfoQ (2009). Pallman, D. *Grid Computing on the Azure Cloud Computing Platform*; *Part1, Part2: Developing a Grid Aplication, Part3: Running a Grid Aplication*. http://www.infoq.com/author/David-Pallmann, učitano 13.2.2011.

48 Infoworld. *Cloud Computing–Deep Dive*. Special Report 1-21, Infoworld Inc, 2009.

49 ISACA. *Cloud Computing: Business Benefits With Security, Governance and Assurance Perspectives*. ISACA, Rolling Meadows, Illinois, 2009.

50 Jennings, R. *Cloud Computing with the Windows Azure Platform*, Wiley Publishing, Indianapolis, Indiana, 2009.

51 Joshi, K; Finin, T; Yesha, Y. *Integrated Lifecycle of IT Services in a Cloud Environment. In Proceedings of The Third International Conference on the Virtual Computing Initiative (ICVCI 2009)*, Research Triangle Park, North Caroline, 2010.

52 Kitta, T. *Proffesional Windows Workflow Foundation*. Wiley Publishing Inc, Indianapolis, Indiana, 2007.

53 Klein, S. *Professional WCF Programming: .NET Development with the Windows Communication Foundation*, Wiley Publishing Inc, Indianapolis, Indiana 2007.

54 Kontogiannis, K; Lewis, G.A; Smith, D.B. *A Research Agenda for Service-Oriented Architecture.* 30th International Conference on Software Engineering, Leipzig, 2008.

55 Korri, T. *Cloud computing: utility computing over the Internet. Seminar on Internetworking*, Helsinki University of Technology, Helsinki, 2009.

56 Lamont, O.A; Thaler, R.H. *The Law of One Price in Financial Markets. Journal of Economic Perspective*, 17(4): 191-202, 2003.

57 Leusse de, P; Periorellis, P; Watson, P; Maierhofer, A. *Secure & Rapid composition of infrastructure services in the Cloud.* In *Proceedings of The Second International Conference on Sensor Technologies and Applications (SENSORCOMM '08)*, pages 770-775, Cap Esterel, France, 2008.

58 Lindner, M.A; Vaquero, L.M; L. Rodero-Merino, L; Caceres, J. *Cloud economics: dynamic business models for business on demand. International Journal of Business Information Systems*, 5(4):373-392, 2010.

59 Linthicum, D.S. *Cloud Computing and SOA Convergence in Your Business*. Addison-Wesley Professional, Reading, Massachusetts, 2009.

60 Lynch, N. A. *Distributed Algorithms*. Morgan Kaufmann Publishers, Burlington, Massachusetts, 1996

61 Marks E.A; Bell, M. *Service-Oriented Architecture (SOA): A Planning and Implementation Guide for Business and Technology*. John Willey & Sons Inc., Hoboken, New Yersey, 2006.

62 Mc Evoy, G. V; Schulze, B. *Using Clouds to address Grid Limitations*. *In Proceedings of the 6th international workshop on Middleware for grid computing (MGC'08)*, pages 1-6, Leuven, Belgium, 2008.

63 McDermid, J; Rook, P. Software development process models. *In Software Engineer's Reference Book*, pp 15/1-15/36, Editor: John A. McDermid, Butterworth Heinemann, Oxford, 1991.

64 Microsoft (2009). *Virtualization and Consolidation Mainframe and x86-x64–Based Solutions in Today's Environment*. http://download.microsoft.com/download/3/8/a/38aba79c-9a99-4a42-9f30-4fe8e4ed22d7/Virtualization%20and%20Consolidation%20white%20paper.pdf, učitano 13.2.2011.

65 Microsoft (2010), Berry, W. W. *SQL Azure Horizontal Partitioning: Part 2*. http://blogs.msdn.com/b/sqlazure/archive/2010/06/24/10029719.aspx, učitano 13.2.2011.

66 Miller, R. Data Center Knowledge (2009). *Google Unveils Its Container Data Center*, http://www.datacenterknowledge.com/archives/2009/04/01/google-unveils-its-container-data-center/, učitano 13.2.2011

67 Moore, G.E. *Cramming more components onto integrated circuits. Electronics*, 38(8):1-4, 1965.

68 MSDN (2010). *How to Configure Virtual Machine Sizes*. http://msdn.microsoft.com/en-us/library/ee814754.aspx, učitano 13.2.2011.

69 Naghshineh, M; Ratnaparkhi, R; Dillenberger, D; Doran, J. R; Dorai, C; Anderson, L; Pacifici, G; Snowdon, J. L; Azagury, A; VanderWiele, M; Wolfsthal, Y. IBM Research Division cloud computing initiative. *IBM Journal of Research and Development*, 53(4):1.1-1.10, 2009.

70 Natis, Y. *Applied SOA: Transforming Fundamental Principles Into Best Practices*. Gartner, 2007.

71 NIST (2009). Mell, P; Grance, T. *Effectively and Securely Using the Cloud Computing Paradigm*. http://www.nist.gov/itl/cloud/, učitano 13.2.2011.

72 NIST (2009). Mell, P; Grance, T. *The NIST Definition of Cloud Computing*. http://www.nist.gov/itl/cloud/upload/cloud-def-v15.pdf, učitano 13.2.2011.

73 Nurmi, D; Wolski, R; Grzegorczyk, C; Obertelli, G; Soman, S; Youseff, L; Zagorodnov D. The Eucalyptus Open-source Cloud-computing System. In *Cloud Computing and Its Applications workshop (CCA'08)*, Pages 1-5, Chicago, Illinois, 2008.

74 onCloudComputing (2009). *List of Top "Cloud Computing Solution Providers to Watch in 2009"*. onCloudComputing, http://www.oncloudcomputing.com/en/2009/07/list-of-top-cloud-computing-solution-providers-to-watch-in-2009/, učitano 13.2.2011.

75 OpenCVWiki. http://opencv.willowgarage.com/wiki/, učitano 13.2.2011

76 Pantelić, I. *Uvod u teoriju inženjerskog eksperimenta*. Prosveta, Novi Sad, 1975.

77 Peleg, D. *Distributed Computing: A Locality-Sensitive Approach*. Society for Industrial and Applied Mathematics (SIAM), University City Science Center, Philadelphia, 1987.

78 Reese, G.: *Cloud Application Architectures: Building Applications and Infrastructure in the Cloud.* O'Reilly Media, Sebastopol, California, 2009.

79 Rhoton, J. *Cloud Computing Explained*. Recursive Press, 2010.

80 Richardson, L; Ruby, S. *RESTful Web Services*. O'Reilly, Sebastopol, California, 2007.

81 Rittinghouse, J.W; Ransome, J.F. *Cloud Computing-Implementation, Magement and Security*. CRC Press, Taylor&Francis Group, New York, 2010.

82 Roseblum, M. The Reincarnation of Virtual Machines. *ACM Queue*, 2(5):34–40, 2004.

83 Salesforce (2009). *5 Reasons Why CIOs Are Adopting Cloud Computing*. https://www.salesforce.com/form/pdf/force_top5_reasons.jsp. učitano 13.2.2011.

84 Schaffer, H; Averitt, S; Hoit, M; Peeler, A; Sills, E; Vouk, M. NCSUs Virtual Computing Lab: A Cloud Computing Solution. *IEEE Computer*, 7:94-97, 2009.

85 Schroeder, B; Gibson, G. *Disk Failures in the Real Word: What does an MTTF of 1,000.000 Hours Mean to You?.* In *Proceedings 5th Usenix Conference "File and Storage Technologies" (FAST 07)*, Usenix Association, pp 1-16, 2007.

86 Spacek, L. (2008). *Description of the Collection of Facial Images*. http://cswww.essex.ac.uk/mv/allfaces/index.html, 2008, učitano 13.2.2011

87 Spinola, M. *An Essential Guide to Possibilities and Risks of Cloud Computing*. http://www.mariaspinola.com/, 2009.

88 Stanić, J. *Metod inženjerskih merenja*. Mašinski fakultet, Beograd, 1975.

89 Strassman, P.A. *A Model for the Systems Architecture of the Future (GMU Lecture)*. George Mason University, Fairfax, Virginia, 2005.

90 T10 Media (2011). *Azure Pricing vs Amazon AWS Pricing Comparison*. http://www.azuresupport.com/2010/03/azure-pricing-comparison/, učitano 13.2.2011.

91 The Gaea Times (2010). *Top 10 Cloud Computing Service Providers in 2010*. http://tech.gaeatimes.com/index.php/archive/top-10-cloud-computing-service-providers-in-2010/, 2010, učitano 13.2.2011

92 Thingstrategies (2009). *Enterprise Cloud Computing: The CIO Perspective (An Independent Analysis Published on Behalf of salesforce.com)*. http://doblegroup.com/pdf/Whitepapers/WP_THINKstrategies-Salesforce-Enterprise_Cloud_Computing_WP_for_CIOs_v082709final.pdf. učitano 13.2.2011.

93 U.S. Environmental Protection Agency. *Report to Congress on Server and Data Center Energy Efficiency Public Law 109-431*, 2007.

94 Vaquero L.M; Rodero-Merino, L; Buyya, R. Dynamically Scaling Applications in the Cloud. *ACM SIGCOMM Computer Communication Review*, 41(1):45-52, 2011.

95 Varia, J. *Cloud Architectures*. Amazon Web Services, 2008.

96 Velte, A.T; Velte, T.J; Elsenpeter, R. *Cloud Computing: A Practical Approach*. McGraw-Hill, New York, 2010.

97 Vitharana, P; Bhaskaran, K; Jain, H; Wang, H; Zhao, L. Service-Oriented Enterprises and Architectures: State of the Art and Research Opportunities. In *AMCIS 2007 Proceedings,*Paper 343, Toronto, Canada, 2007.

98 VMWare (2010). Understanding Full Virtualization, Paravirtualization, and Hardware Assist. http://www.vmware.com/files/pdf/VMware_paravirtualization.pdf, učitano 13.2.2011.

99 Vogels, W. *Eventually Consistent*. ACM Queue, http://queue.acm.org/detail.cfm?id=1466448, 2008, učitano 13.2.2011

100 Vouk, M. Cloud Computing-Issues, Research and Implementations. *Journal of Computing and Information Technology*, 16(4):235-246, 2008.

101 Vouk, M. Economic of Cloud Computing. Invited Lecture, *Cental European Conference on Information and Intelligent Systems (CECIIS 2010)*, University of Zagreb, Faculty of Organization and Informatics, Varaždin, 2010.

102 Vouk, M; Rindos, A; Averitt, S; Bass, J; Bugaev, M; Peeler, A; Schaffer, H; Sills, E; Stein, S; Thompson, J; Valenzisi, M. Using VCL Technology to Implement Distributed Reconfigurable Data Centers and Computational Services for Educational Institutions. *IBM Journal of Research and Development*, 53(4):2.1-2.18, 2009.

103 Vuk, D. *Non-SQL baze podataka – jesu li relacijske baze ugrožene?*. Zbornik radova CASE 22:112-123, CASE, Rijeka, 2010.

104 W3C (2002). *Web Service Choreography Interface*. http://www.w3.org/TR/2002/NOTE-wsci-20020808/, učitano 13.2.2011.

105 W3C (2002). *Web Service Conversation Language (WSCL)*. http://www.w3.org/TR/2002/NOTE-wscl10-20020314/, učitano 13.2.2011.

106 W3C (2005). *Web Services Choreography Description Language Version 1.0*. http://www.w3.org/TR/2005/CR-ws-cdl-10-20051109/, učitano 13.2.2011.

107 W3C (2007). *SOAP Version 1.2 Part 1: Messaging Framework (Second Edition)*. http://www.w3.org/TR/2007/REC-soap12-part1-20070427/, učitano 13.2.2011.

108 Walker, E.: *The Real Cost of a CPU Hour*, IEEE Computer, 35-41, April 2009.

109 Walker, E; Brisken, W; Romney, J. *To Lease or Not to Lease from Storage Clouds*. IEEE Computer 43(4):44-50, 2010.

110 Wang, L; von Laszewski, G. *Cloud Computing – A Perspective Study*, Rochester Institute of Technology, Rochester, New York, 2008

111 Wang, L; von Laszewski, G. *Scientific Cloud Computing: Early Definition and Experience*. Rochester Institute of Technology, Rochester, New York, 2008.

112 White, S; Miers, D. *BPMN Modeling and Reference Guide*. Future Strategies Inc, Lighthouse Point (FL) USA, 2008.

113 Wikipedia: *Google Platform*, Wikipedia, http://en.wikipedia.org/wiki/Google_platform, učitano 13.2.2011

114 Wikipedia: *Intermodal container*, http://en.wikipedia.org/wiki/Intermodal_container, učitano 13.2.2011

115 Xinhui Li, Ying Li, Tiancheng Liu, Jie Qiu, Fengchun Wang. *The Method and Tool of Cost Analysis for Cloud Computing*. In *2009 IEEE International Conference on Cloud Computing (CLOUD)*, pages 93-100, Bangalore, India, 2009.

Bazična dokumentacijska kartica na hrvatskom jeziku

MRIZ (FOI – Sveučilište u Zagrebu) UDK 004.75:004.42:338.5(043.3)

Doktorski rad

Računalni oblaci kao dio servisno orijentirane arhitekture

Slaven Brumec

Fakultet organizacije i informatike

Varaždin, Hrvatska

Glavna tema ovog rada je ekonomika oblačnog računarstva. Postavljen je postupak za usporedbu isplativosti kupnje određene količine ICT resursa (i njihovog korištenja na klasičan način) s najmom adekvatne količine ICT resursa u komercijalnom računalnom oblaku. Bitan dio tog postupka jest mjerenje performansi aplikacija za koje treba ispitati da li je isplativije smjestiti ih na vlastitim ili oblačnim ICT resursima. Aplikacijske performanse izražavaju se prihvatljivim vremenom *T* obavljanja aplikacije, a ovise o raznim aplikacijskim svojstvima (npr. složenost algoritma, broj CRUD akcija, opseg podataka koje treba obraditi i dr.) te o količini i snazi računala na kojima se aplikacija pokreće. Ta svojstva su nezavisne varijable $(x_1, x_2, \dots x_n)$ o kojima ovisi vrijeme izvođenja aplikacije *T*, prema:

$$T = f(x_1, x_2, \dots\ x_n)$$

Za veći broj nezavisnih varijabli, mjerenje se provodi prema višefaktorskom parcijalnom planu pokusa radi smanjenja broja potrebnih mjerenja. Iz rezultata mjerenja očitavaju se vrijednosti nezavisnih varijabli (npr. količine računala i njihove snage). Te vrijednosti se potom:

1. uvrštavaju u kalkulatore troškova komercijalnih oblačnih poslužitelja radi izračuna cijene komercijalnog oblačnog najma i
2. uvrštavaju u Walkerove modele radi određivanja troška kupnje ICT resursa te procjene da li je najam adekvatne količine tih resursa u oblaku povoljan ili ne.

Postavljena je metodika razvoja servisno orijentiranih rješenja uz korištenje oblačnog računarstva, nazvana metodika KOR. Izvedena je kao proces formaliziran po BPMN standardu. Njome se objektivizira odluka o tome koje aplikacije informacijskog sustava je isplativo smjestiti u oblak, a za koje treba nabaviti vlastite ICT resurse te kako sve to sistematično provesti. Aplikacije smještene u oblaku treba preoblikovati kao servise u servisno orijentiranoj arhitekturi. Pokazano je da su računalni oblaci dobra platforma za smještaj i pokretanje SOA rješenja iz dva razloga:

1. stoga što su protokoli i tehnologije za interoperabilnost računalnih oblaka s vanjskim svijetom jednaki protokolima i tehnologijama za interoperabilnost SOA rješenja te
2. zato što su filozofije korištenja slične, temeljene na servisnoj paradigmi.

Uz ostalo, ovim radom je razjašnjen odnos između računalnih oblaka i drugih oblika naprednog računarstva kao što je *grid computing*, distribuirano računarstvo, virtualizacija, itd.

Voditelj rada: prof.dr.sc. Neven Vrček

Povjerenstvo za ocjenu i obranu: prof.dr.sc. Vjeran Strahonja, prof.dr.sc. Neven Vrček, prof.dr.sc. Mladen Vouk

Obrana: 14. srpnja 2011.

Rad je pohranjen u Biblioteci Fakulteta organizacije i informatike u Varaždinu

(XVI + 191 stranica, 27 tabela, 37 slika, 5 priloga i 115 bibliografskih referenci)

MRIZ-2

I. Računalni oblaci kao dio servisno orijentirane arhitekture
II. Brumec, S.
III. Fakultet organizacije i informatike, Varaždin, Hrvatska

UDK 004.75:004.42:338.5(043.3)

računalni oblaci
computing clouds
oblačno računarstvo
cloud computing
servisno orijentirana arhitektura
SOA
ekonomika
metodika KOR
mjerenje aplikacijskih performansi
višefaktorski parcijalni plan pokusa
ICT resursi
isplativost
kupnja
najam

Bazična dokumentacijska kartica na engleskom jeziku

MRIZ (FOI – University of Zagreb) UDK 004.75:004.42:338.5(043.3)

Dissertation

Computing Clouds as a Part of Service-Oriented Architecture

Slaven Brumec

Faculty of Organization and Informatics

Varaždin, Croatia

This dissertation's main subject is economics of cloud computing. We have set a procedure for comparison of cost effectiveness between buying one's own ICT resources (and using them in a classical way) and leasing of adequate ICT resources in a commercial computing cloud. Essential part of this procedure is application performance measurement of those applications for which decision should be made whether is more cost-effective to deploy them on private ICT resources or in commercial cloud. Application performances are expressed in acceptable elapsed time T and depend on miscellaneous application properties (such as algorithm complexity, number of CRUD actions, amount of data processing, etc.) and on number and power of computers on which application is run. These properties are independent variables $(x_1, x_2, \dots x_n)$ that influence elapsed time T according to:

$$T = f(x_1, x_2, \dots\ x_n)$$

For greater amount of independent variables, measurements should be made according to fractional factorial design in order to reduce the number of necessary measurements. From measurement results one can find out independent variables' values (such as number of needed computers and their power). These values are further used in following way:

1. They are substituted in commercial cloud providers' cost calculators in order to calculate cost of commercial leasing.
2. They are substituted in Walker's models to determine private ICT resources procurement cost and to evaluate is leasing of adequate resources in a commercial cloud beneficial.

Methodology for SOA solutions development using cloud computing has been made (and named methodology KOR). It is visualized as a business process according to BPMN. Using this methodology one can objectively decide which applications can be cost-effectively deployed and run in a cloud, which should be deployed on private ICT resources and how to do all that systematically. Application for the cloud should be re-engineered as SOA services. It has been shown that computing clouds are good platform for SOA solutions' deployment and running because of the following:

1. Protocol and technologies for computing clouds' interoperability with external systems are also used for ensuring SOA solutions' interoperability.
2. Usage philosophies are similar and rooted in service paradigm.

In this dissertation we also clarified relationships between cloud computing and other advanced computing technologies, such as grid computing, distributed computing, virtualization, etc.

Supervisor: prof.dr.sc. Neven Vrček

Examiners: prof.dr.sc. Vjeran Strahonja, prof.dr.sc. Neven Vrček, prof.dr.sc. Mladen Vouk

Oral Examination: 14. srpnja 2011.

The dissertation is deposited at the Library of the Faculty of Organization and Informatics, Varaždin.

(XVI + 191 pages, 27 tables, 37 figures, 5 attachments and 115 references, original in Croatian)

MRIZ-2

UDK 004.75:004.42:338.5(043.3)

I. Computing Clouds as a Part of Service-Oriented Architecture

II. Brumec, S.

III. Faculty of Organization and Informatics, Varaždin, Croatia

computing clouds

cloud computing

service-oriented architecture

SOA

economics

methodology KOR

application performance measurement

fractional factorial design

ICT resources

cost-effectiveness

buying

leasing