UNIVERZITET U NIŠU

ELEKTRONSKI FAKULTET

Martin T. Štufi

PREDLOG ARHITEKTURE SISTEMA VISOKIH PERFORMANSI ZA GENERALNU OBRADU PODATAKA NA KLASTERIMA ZA PODATKE VELIKOG OBIMA

DOKTORSKA DISERTACIJA

Niš, 2022.

UNIVERSITY OF NIS

FACULTY OF ELECTRONIC ENGINEERING

Martin T. Stufi

An Architecture Proposal for High-Performance and General Data Processing System on Big Data Clusters

DOCTORAL DISSERTATION

Niš, 2022.

Podaci o doktorskoj disertaciji

Mentor: **Dr Leonid Stoimenov, redovni profesor,**

Univerzitet u Nišu, Elektronski fakultet

Naslov: **PREDLOG ARHITEKTURE SISTEMA VISOKIH PERFORMANSI ZA GENERALNU OBRADU PODATAKA NA KLASTERIMA ZA PODATKE VELIKOG OBIMA**

Rezime: Poslednjih godina, primena kao i široko usvajanje novih tehnologija zasnovanih na velikoj količini podataka (Big Data), Internet stvari (IoT), oblaka (Cloud) povećali su upotrebu sistema za obradu velike količine podataka. Time je došlo do osetljivog i eksponencijalnog povećanja količine generisanih heterogenih podataka. Takvi podaci su u suštini struktuirani, nestruktuirani i polu strukturni. Obrada i analiza velike količine podataka je glomazna i postepeno se kreće od klasične „serijske“ obrade, preko tehnike izdvajanja, transformacije podataka, kao i njihovog učitavanja (ETL), obrade podataka. Ova obrada se odvija u realnom vremenu i kao takvu je možemo prepoznati u raznim domenima od industrije, zdravstva, energetike, pa do finansijskog, bankarskog ali i drugih sektora. Praćenje toka podataka, obrada podataka, njihovo upravljanje, praćenje podataka vremenskih serija ili istorijski skup podataka su ključni za modele predviđanja ne samo u pomenutim domenima ali i šire.

Ova doktorska disertacija se odnosi na projektovanje uopštene arhitekture za obradu velike količine podataka. Arhitektura kao takva omogućava efikasnu akviziciju podataka, njihovo optimalno smeštanje, obradu velike količine podataka, upotrebu raznih algoritama za donošenje zaključaka kao i za prikazivanje podataka. Doktorska disertacija prikazuje kompletan proces modeliranja i projektovanja arhitekture, izbora odgovarajućih softverskih komponenti za njenu realizaciju. U doktorskoj disertaciji je predstavljena platforma koja je ispunila vrlo zahtevne parametre performansi sistema, uključujući standard za podršku o odlučivanju Saveta za obradu transakcija (TPC-H), a u skladu sa zakonodavstvom Evropske unije (EU) i Češke Republike. Predstavljeni koncept (PoC), koji je kasnije prošao nadogradnju na proizvodno okruženje, ujedinio je do tada izolovane delove zdravstvene zaštite Češke Republike. Ova platforma, artefakti i koncept kao takav se može preneti na zdravstvene sisteme drugih zemalja koje imaju interesovanja za razvoj ili nadogradnju sopstvene, nacionalne zdravstvene infrastrukture na isplativ, bezbedan, skalabilan način uz zadovoljavanje visokih performansi.

Naučna oblast:	Elektrotehnika i računarsko inženjerstvo
Naučna disciplina:	Distribuirani (klaster) sistemi za obradu podataka
Ključne reči:	Klaster, Big Data, Stream, Vertica, NoSQL, obrada podataka u realnom vremenu, stream podaci
UDK:	004.6/65:004.275
CERIF klasifikacija:	**T 120 Sistemski inženjering, računarska tehnologija**
Tip licence kreativne zajednice:	**CC BY-NC-ND**

Data on Doctoral Dissertation

Doctoral Supervisor: PhD Leonid Stoimenov, full professor,

University of Nis, Faculty of Electronic Engineering

Title: **A Universal Architecture for High-Performance and General Data Processing on Big Data Clusters**

Abstract:

In recent years, the application and widespread adoption of Big Data, Internet of Things (IoT), Cloud technologies have increased the use of large-scale data processing systems. These technologies increased significantly and exponentially with the heterogeneous data generated (structured, unstructured, and semi-structured). The processing and analysis of a tremendous amount of data is cumbersome and is gradually moving from the classic "batch" processing - extraction, transformation, loading (ETL) techniques to real-time processing. For example, in the domain of the automobile industry, healthcare, but also in other disciplines. Tracking, data processing, environmental management, time-series data, and historical data set are crucial to forecasting models not only in these domains.

This doctoral dissertation is about the design of a general architecture for processing a large amount of data. The architecture as such enables efficient acquisition of data, their optimal placement, processing of large amounts of data, use of various algorithms for drawing conclusions as well as for displaying data. The doctoral dissertation shows the complete process of modeling and designing architecture, the selection of appropriate software components for its realization. The presented platform met very demanding parameters for meeting the system's performance, including the standard for decision support of the Transaction Processing Council (TPC-H) by following the European Union (EU) legislation and the Czech Republic. Currently, the presented proof of concept (PoC) that has been upgraded to the production environment has united isolated parts of the Czech Republic's healthcare. The reported PoC Big Data Analytics platform, artefacts and concepts can be transferred to health systems in other countries interested in developing or upgrading their national health infrastructure in a cost-effective, secure, scalable, and high-performance way.

Scientific Field: Electrotechnics and Computer Engineering

Scientific Discipline: Distributed (cluster) data processing systems

Keywords: Big Data, Big Data Analytics, TPC-H, NoSQL Database cluster, Real time BDA

UDC: 004.6/65:004.275

CERIF classification: **T 120 System engineering, computer technology**

Zahvalnica

Iskreno se zahvaljujem svom mentoru Prof. dr Leonidu Stoimenovu za svu pruženu podršku tokom mojih doktorskih studija a posebno u trenucima kada nisam ni sam verovao da se cilj može postići jer je završni deo mojih doktorskih studija izgledao kao put u nedogled, bez cilja i kraja. Ova zahvalnost, između ostalog, odnosi se na nesebičnu podršku, otvorenost i razumevanje kao i pružanju slobode kao autoru doktorske disertacije.

Veliku zahvalnost izražavam Prof. dr Borisu Bačiću, sa Fakulteta inženjerskih, računarskih i matematičkih nauka na Tehnološkom univerzitetu u Oklandu, Novi Zeland. Njegovo iskustvo, pružena podrška i motivacija značajno mi je pomoglo pri izradi naučnih radova i doktorske disertacije. Bez tebe Borise bi puno toga bilo nemoguće.

Iskreno se nadam, da ćemo u narednom periodu našeg života nastaviti profesionalnu saradnju, koja će nas zajednički unaprediti.

Zahvalnost takođe upućujem svojim kolegama iz firme Solutia s.r.o. sa sedištem u Pragu, za pruženu podrški i saradnju, na čijem čelu sam već 18 godina svog profesionalnog života. Nadam se da nam tek predstoji interesantna budućnost koja će dalje obogatiti našu profesionalnu karijeru i profesionalni razvoj.

Posebnu i neizmernu zahvalnost dugujem mojoj porodici na beskrajnom razumevanju, neograničenoj podršci i vremenu koje su mi omogućili da se posvetim svom usavršavanju, a koje nažalost nisam proveo sa njima. Iskreno se nadam da će mojoj deci ovo biti motivacija da ispune svoje snove i životne ciljeve na putu da postanu i budu bolji ljudi nego što sam ja.

Zahvaljujem se svojoj majci Dušanki i ocu Tomi koji su mi omogućili da postignem puno toga u životu, a nažalost nisu mogli biti svedoci svega na mom životnom putu.

U znak sećanja na mog brata dr. Marjana Štufi, do trenutka dok se opet ne pronađemo, negde gde ovaj događaj neće biti značajan, a svet će biti drugačiji …

„Kada nešto istinski želiš, ceo
univerzum se ujedini u želji da ti pomogne da
to i ostvariš.

“**Paulo Coelho**

“Rad će vam zauzeti veliki deo života, i jedini način da budete iskreno zadovoljni jeste da radite nešto u što verujete da je dobro. A jedini način da dobro radite posao jeste da volite to što radite. Ako ga još uvek niste pronašli, nastavite tražiti. Nemojte se smiriti. Svim svojim srcem ćete znati kada ste ga pronašli.”

Steve Jobs

“Your time is limited, so don't waste it living someone else's life. Don't be trapped by dogma - which is living with the results of other people's thinking. Don't let the noise of others' opinions drown out your own inner voice. And most important, have the courage to follow your heart and intuition.”

Steve Jobs

Posvećeno mojoj deci

To my children

Mario, Mia, Matteo

Svojoj porodici

To my family

1. Sadržaj

Spisak korišćenih skraćenica i pojmova

Skraćenica	Popis
AAP	Ad-hoc Analysis Processing
ACID	Atomicity, Consistency, Isolation & Durability
ATC	Anaplastic thyroid cancer
BDA	Big Data Analytics
BI	Business Intelligence
CPU	Central Processor Unit
CSV	Comma Separated Value
DaaS	Data Analytics as a Service
DB	Sistem baze podataka
DDL	Data Definition Language
DI	Data Integration
DML	Data Manipulation Language
DQM	Data Quality Management
DS	Data Storage
DV	Data Visualization
DWH	Data Warehouse
EDW	Enterprise Data Warehouse
ENISA	European Network and Information Security Agency
ERD	Entity Relationship Diagram
ETL	Extract Transform and Load
EU	European Union
GB	Giga bajt
Gb/s	Gigabit per second
GPU	Graphical Processor Unit

GUI	Graphical user Interface
HDD	Hard disk drajv računara
HDFS	Hadoop Distributed File System
HW	Hardware
IHIS	Institute of Health Information and Statistics of the Czech Republic
IHIS	Institut za zdravstvene informacione sisteme i statistiku
IoT	Internet of Things
MDM	Meta Data Management
MHDA	Massive High Data Analytics
ML	Machine Learning
MPP	Massive Parallel Processing
MQTT	MQ Telemetry Transport, protokol komunikacije u IoT oblasti
NoSQL	Not Only SQL
OEL	Oracle Enterprise Linux
OLTP	Online Transaction Processing
OS	Operacioni sistem
PB	Peta bajt
PoC	Proof of Concept
PPT	Power Point Template
RAM	Operaciona memorija računara
RDD	Resilient Distributed Data
RHEL	Red Hat Enterprise Linux
SF	Scale Factor
SQL	Structured Query Language
SRS	Software Requirement Specification
SuSE	Ime distribucije Linux
SW	Software

TB	Terabyte
TOSDI	Talend Open Studio Data Integration
TPC-H	Decision Support Benchmark - Performans test za sisteme za odlučivanje
ZB	Zettabyte – 10^{21} Bytes

Spisak slika

Spisak tabela

1. Uvod

Živimo u eri velikog obima podataka koji nas u svakom trenutku okružuju i koji su generisani informacionim sistemima [1-3], socijalnim mrežama [4, 5] kao i raznim drugim uređajima. Veoma intenzivni tehnološki razvoj u oblasti senzora [6], senzorskih mreža [7], računarskih mreža kao i njihova sve obimnija dostupnost uz prihvatljive ili pristupačne cene ima značajan uticaj na kreiranje velike količine podataka ili podataka velikog obima (engl. Big Data) [8-11]. Ovakvo dinamično i ubrzano generisanje podataka ima kao posledicu povećanje zahteva za njihovu obradu[12]. Kao primer možemo navesti: Google ima preko 3.5 milijardi upita dnevno [13], WhatsApp omogućava razmenu preko 65 milijardi poruka dnevno sa preko 2 milijardi korisnika. U 2021. godini svaka osoba je generisala 1.7 MB podataka svake sekunde.

Na osnovu Forbes istraživanja [11], 95% preduzeća pokazuju potrebu za upravljanjem nestrukturiranim podacima i tu potrebu predstavlja kao problem za svoje poslovanje. Procenjuje se da će rastuća potražnja za globalnim tržištem analitike podataka i usluga poslovnog obaveštavanja povećati prihode više od 200 milijardi dolara u 2022. godini. Više od 150 ZB, kao i 150 milijardi GB podataka, zahtevaće analizu do 2025. godine. Da bi se ostvario pun potencijal koji imaju u sebi velike količine podataka, istraživači i inženjeri sa praktičnim iskustvom moraju razrešiti više izazova, kao i razviti određena konceptualna i tehnološka rešenja[14, 15] za rešavanje ovih problema.

Generalno, pojam Big Data ili „velika količina podataka“ se može definisati na više načina. Ne postoji jednoznačni konsenzus o jedinstvenoj definiciji ovog pojma. U oblasti analize podataka, vrlo često se pojam Big Data vezuje za proces ekstrakcije, transformacije i učitavanja podataka (engl. Extract-Transform-Load) tj. ETL. Definicija Big Data se takođe vezuje za veliku količinu podataka koji se ne mogu obraditi konvencionalnim informacionim sistemima. Ova popularna definicija se oslanja na tri glavna atributa, obim podataka (Volume), brzina (Velocity) i različitost (Variety) – poznata kao 3V. Atribut Volume (obim podataka), se odnosi na veliku količinu podataka, atribut Velocity se odnosi na brzinu podataka (npr. stream podaci) kojom se oni mogu generisati i smeštati. Variety atribut se odnosi na raznorodnost u podacima, koji mogu biti strukturno organizovani (npr. baze podataka, xls fajlovi i sl.). koji se ne mogu strukturno organizirati (dokumenti, fotografije, video materijal). Ovakve osobine velike količine podataka zahtevaju arhitekture visokih performansi za njihovo čuvanje i

obradu. Definicija 3V je osnova na kojoj je u ovoj doktorskoj disertaciji obrađena i predložena arhitektura sistema visokih performansi za generalnu obradu podataka na Big Data klasterima.

Noviji trend [16, 17] je da su informacioni sistemi po svojoj arhitekturi koncipirani na osnovu većeg ili velikog broja pojedinačnih, tradicionalnih ili konvencionalnih servera spojenih u jednu celinu tzv. klaster. Sistemi ili platforme za obradu velike količine podataka se nazivaju Big Data klasteri. Takvi sistemi imaju između ostalog osobinu horizontalne skalabilnosti (Horizontal Scalability), koja omogućava povećanje njihovog kapaciteta uzajamnim povezivanjem u jednu sistemsku celinu. Na osnovu toga može doći do značajnog povećanja njihovog kapaciteta i performansi uz upotrebu skalabilnosti sistema.

1.1 Predmet istraživanja

Informacioni sistemi kao takvi, koji se koriste za obradu podataka, vrlo često su u upotrebi kao tzv. tradicionalni sistemi, koji su bili izgrađeni po principu „silos“ arhitekture. Ova arhitektura je karakteristična za sisteme koji imaju nekoliko slojeva. Vrlo često se radi o tri sloja, kao što je (1) sloj aplikacionog dela (Application Layer), (2) srednji sloj (Business Layer) i (3) sloj gde su baze podataka (Database Layer). Podešavanje takvih sistema u smislu performansi ima niz nedostataka. Prvenstveno su nedostaci koji se tiču podešavanja performansi, a koji se javljaju na osnovu limita i kapaciteta njihovog hardvera tj. njihove skalabilnosti (Scalability). Ovakva ograničenja koja postoje, prvenstveno se odnose na maksimalni broj računarskih procesora, operativne memorije, diskova, mrežnih kartica i ostalih komponenti koje možemo ugraditi u veće računarske sisteme.

Jedan od dodatnih zahteva modernog informatičkog društva i moderne civilizacije prema informacionim sistemima je zahtev za njihovu sposobnost tokom smeštanja velike količine podataka usled njihovog značajnog porasta [18]. U poslednjih 10 godina, informaciono društvo doživljava veliki preokret u smislu eksponencijalnog povećanja količine i tipova podataka. Posledica toga je da se računarski sistemi susreću sa izazovom akvizicije ili čitanja podataka velikog obima, njihovog skladištenja, kao i njihove obrade i vizualizacije [19]. Pored problema vezanih za njihovo smeštanje ili transformaciju, često je potrebno obezbediti obradu i analizu takvih podataka u što kraćem vremenskom periodu i istovremeno nad njima obezbediti dalje operacije u cilju donošenja određenih zaključaka. Primer takvih operacija može biti na nivou čitanja podataka, njihovog čišćenja, kao i transformacija u specijalizovani deo sistema za smeštanje podataka na poseban fajl sistem (Hadoop), operativnu memoriju ili bazu podataka.

Operacije koje mogu biti primenjene se mogu odnositi na razne kompleksne i zahtevnije obrade podataka u smislu primene u okviru analitičkih i agregacionih operacija. Nad takvim podacima dalje je moguće primeniti razne algoritme za mašinsko učenje. Vrlo često je potrebno prikazivanje odnosno vizualizacija podataka velikog obima radi njihove vizualne percepcije.

Podaci velikog obima, proističu između ostalog i kao posledica digitalizacije društva, kao i sve veće upotrebe informacionih sistema u različitim domenima savremenog društva. Obrada velike količine podataka je aktuelna oblast istraživanja u različitim oblastima kao npr. industrijska proizvodnja, automobilska industrija i transport, energetika, zdravstvo, bankarstvo i sl.

Jedan od najznačajnijih domena se odnosi na zdravstvo. Podaci velikog obima mogu imati na primer veliki značaj u nagoveštavanju mogućnosti da se njihovom značajnijom upotrebom može doći do ključnog povećanja kvaliteta medicinskih usluga prema pacijentima. U isto vreme se može postići smanjenje troškova na osnovu značajno veće optimizacije rada prilikom njihovog korišćenja. Cilj uvođenja takvih sistema se odnosi na podizanje kvaliteta pruženih zdravstvenih usluga. Podaci velikog obima se mogu obrađivati u okviru specijalizovanih državnih institucija na osnovu principa centralizacije obrade velikog obima podataka ili distribuiranim principom u okviru pojedinačnih ministarstava kao što su na primer: Ministarstvo zdravlja, Ministarstvo unutrašnjih poslova, Ministarstvo odbrane i Evropska unija.

Sama analiza podataka velikog obima u tradicionalnim sistemima postaje usko grlo, prvenstveno zbog njihove količine, tj. masivnosti, složenosti, različitosti, preciznosti, brzine i načina generisanja.

Ovakvo dramatično povećanje podataka vodi ka novim zahtevima za informacionim sistemima u smislu njihovog smeštanja, dostupnosti, efikasne obrade i njihove vizualizacije. Sistemi sa horizontalnim prilagođavanjem, zasnovani na klasterima, se nameću kao rešenje kako u smislu performansi i resursa, tako i u smislu cene njihove izgradnje, implementacije i efikasnosti u obradi raznih tipova i vrsta podataka.

Predmet naučnog istraživanja u ovoj doktorskoj disertaciji se odnosi na projektovanje uopštene arhitekture koje omogućava efikasnu akviziciju podatka, njihovo optimalno smeštanje, obradu velikog obima podataka, upotrebu raznih algoritama za donošenje zaključaka, kao i prikazivanje odnosno vizualizaciju podataka. Dodatno, predmet istraživanja

doktorske disertacije, omogućiće i generalizaciju arhitekture za smeštanje i obradu velike količine podataka. Osnovna ideja je da se takva arhitektura može upotrebiti u raznim domenima, uz adekvatna prilagođenja. Poseban deo predmeta naučnog istraživanja je vezan za evaluaciju predložene arhitekture u domenu zdravstva i njena praktična upotreba konkretno u zdravstvu Češke Republike.

Doktorska disertacija u svom završnom delu predlaže arhitekturu za obradu velike količine podataka, podešavanje performansi (TPC-H) [20, 21] u okviru definisanih zahteva od strane Instituta za zdravstvene informacione sisteme i statistiku Češke Republike.

1.2 Cilj istraživanja

Cilj naučnog istraživanja ove doktorske disertacije se odnosi na istraživanje i predlog rešenja koje daje odgovor na pitanje: da li je i kako je moguće izgraditi modernu i generalizovanu, uopštenu arhitekturu koja se odnosi na skalabilnu arhitekturu informacionog sistema, zasnovanu na klasterima, za obradu velike količine podataka, a koja će omogućiti akviziciju, transformaciju i smeštanje podataka, obezbediti njihovu visoku dostupnost kao i obradu na vrlo efikasan način? Dodatni ciljevi se vezuju za probleme koji se odnose na performanse klaster arhitekture - kako efikasno i optimalno odrediti kapacitet klastera informacionog sistema, kako da performanse klaster arhitekture rastu prilikom povećanja pojedinačnih servera (nodova ili čvorova), kao i kako postići efikasna merenja performansi takvih rešenja.

Predložena arhitektura treba da obezbedi efikasnost planiranog sistema, njegove primene, kao i mogućnost da se na osnovu dodatne potrebe, tokom upotrebe sistema može lako proširiti, promeniti na osnovu zahteva za povećanje performansi, kapaciteta za smeštaj podataka kao i njihove dostupnosti. Ova arhitektura treba takođe da obezbedi pouzdanost usled prekida rada ili prilikom pojavljivanja bilo koje vrste kvarova. Predlog ovakve nove arhitekture sa horizontalnom skalabilnošću mora suštinski da se razlikuje od arhitekture tzv. tradicionalnih sistema (poglavlje 1.1), kako bi se rešili problemi koji su posledica njihovih osnovnih osobina, a odnose se na limite za povećanja njihovih kapaciteta. Ograničenja se odnose na hardverske limite koje tradicionalna arhitektura kao takva ima, a tiču se npr. maksimalnog broj procesora, veličine operativne memorije, kapaciteta diskova, mrežnih kartica i sl.

Dodatni zahtevi se odnose na primenu predložene arhitekture u različitim domenima na osnovu specifičnosti datih domena. U okviru doktorske disertacije biće izvršena evaluacija predložene arhitekture i predstavljen primer projektovanja i implementacije takvog sistema u domenu zdravstva na realnom primeru zdravstvenog sistema Češke Republike.

Obavljena istraživanja, koja su prikazana u okviru doktorske disertacije, obuhvataju analizu načina postizanja zahtevanih performansi. Prilikom dizajniranja arhitekture kao analitičke platforme za obradu velike količine podataka, trebaju biti ispunjeni uslovi na osnovu TPC-H benchmarka (poglavlje 3) tj. TPC-H testa performansi za podršku u odlučivanju. U doktorskoj disertaciji je dalje prikazano kako je moguće izgraditi sistem na osnovu unapred definisanih parametara performansi, koje novi sistem mora dostići, kako bi ispunio zahtevane uslove. Dodatno, jedan od ciljeva ovog istraživanja je da se u okviru predložene nove arhitekture obezbedi mogućnost njenog prilagođenja i podešavanja u smislu performansi, kako bi se postigli traženi rezultati, saglasni sa TPC-H testom performansi. Dalje, mogućnost njenog proširenja usled očekivanog eksponencijalnog porasta podataka u narednim godinama na šta ukazuju svi trendovi u oblasti obrade podataka velikog obima.

1.3 Osnovne pretpostavke istraživanja

Fokus u naučnim i drugim istraživanjima se odnosi na upotrebu naprednih tehnologija arhitekture kao i njene praktične implementacije koji je obrazložen i ovoj doktorskoj disertaciji. Pretpostavke koje su vezane za ovo istraživanje odnose se na sledeće aspekate:

1. Učitavanje velike količine podataka (Data Acquisition),
2. Kontrola njihovog kvaliteta (Data Quality Management),
3. Smeštanje i obrada velike količine podataka (Data Storage, Data Processing).

Upotreba naprednih metoda kao što su:

1. Upravljanje meta podacima (Metadata Management),
2. Upravljanje kvalitetom podataka (Data Quality Management),
3. Prikazivanje rezultata obrađenih podataka (Data Visualization).

Prilikom izrade predloga arhitekture sistema visokih performansi i sistema za generalnu obradu velike količine podataka na Big Data klasterima korišćena je metoda eksperimenta. Predložena arhitektura je implementirana u realnim uslovima, kako „On-premise“, tako i u „Cloud“ sredini. Predložena arhitektura proizilazi iz detaljnog istraživanja i praktičnih iskustava kako na nivou izgradnje arhitekture, tako i na nivou testiranja performansi ali i rešenja koja su implementirana ne samo u sredini korisnika ali i sredini za računarstvo u oblaku [22, 23].

1.4 Materijali i metode istraživanja

U cilju izrade predložene doktorske disertacije korišćene su različite istraživačke metode koje treba da omoguće ispunjenje zahtevanih ciljeva, kao što su karakterizacija, hipoteza, pretpostavka na osnovu hipoteze, eksperimentalna metoda, evaluacija i potvrđivanje zaključaka.

Na samom početku korišćena je **karakterizaciona metoda** za unapred definisane parametre performansi koje je potrebno postići ispravnim dimenzionisanjem. Sledeća metoda je **hipoteza arhitekture**, rešenja koje bi moglo ispunjavati očekivane zahteve na osnovu kojih metodom pretpostavke bi se moglo doći do utvrđivanja parametara performansi na predloženu arhitekturu.

Korišćena je i **komparativna metoda**, koja podrazumeva upoređivanje dobijenih rezultata prilikom raznih merenja na osnovu različite količine podataka 1TB, 3TB. Završna metoda je **metoda sinteze** kojom su pojedini rezultati, pojedinačno dobijeni, spojeni u odgovarajuću metodologiju za kreiranje odgovarajuće veličine klastera za obradu izuzetno velikog obima podataka.

U okviru doktorske disertacije primenjene metode za predlog arhitekture sistema visokih performansi i sistema za generalnu obradu podataka na Big Data klasterima se mogu prikazati u sledećim koracima:

1. **Pretpostavka** – implicitni testovi u skladu sa referentnim modelom.
2. **Karakterizacija i analitičke tehnike** - koriste se za identifikaciju, kvantifikaciju i karakterizaciju arhitekture sistema visokih performansi.

3. **Hipoteza, ono što se eksplicitno testira eksperimentom** - hipoteza mora uvek proći kroz proces verifikacije i istrage. Pretpostavka može ili ne mora biti verifikovana ili istražena. U istraživanju, pretpostavka označava postojanje odnosa između promenljivih, dok hipoteza uspostavlja odnos utvrđen pretpostavkom.

4. **Komparativni metoda** - koristiće se za poređenje rezultata dobijenih tokom različitih merenja.

5. **Eksperimentalna metoda** - koristi se u kombinaciji sa komparativnom metodom na osnovu različitih konfiguracija predložene arhitekture sistema visokih performansi i sistema za generalnu obradu podataka.

6. **Procena** – momenat kada je arhitektura sistema visokih performansi za generalnu obradu velikog obima podataka spremna za realnu upotrebu.

7. **Sinteza** - pojedinačni rezultati biće kombinovani u odgovarajuću metodologiju kako bi se stvorila klaster platforma usklađena sa motivacijom doktorske disertacije.

1.5 Rezultati istraživanja

Prikazani rezultati naučnog istraživanja u okviru doktorske disertacije fokusirani su na predlog uopštenog modela arhitekture informacionog sistema. Predloženi rezultati će omogućiti izgradnju modela arhitekture uz obezbeđenje optimalnog i generalizovanog načina preuzimanja podataka. Dalje, omogućiće njegovu integraciju, kao efikasno i optimalno smeštanja podataka. Efikasna obrada podataka uključuje upotrebu naprednih principa i algoritama za obradu podataka. Omogućuje njihovu analizu kao i vizualizaciju podataka. Pored toga, ovakva platforma je u saglasnosti sa određenim standardima. Za takve standarde za sisteme za obradu podataka velikog obima u smislu performansi, uzima se TPC-H test performansi, a to na osnovu unapred definisanih zahteva (više od 100 zahteva).

Očekivani rezultati doktorske disertacije vezani su za evaluaciju i analizu informacija. Ovi rezultati se odnose na predloženo rešenje klaster arhitekture na osnovu unapred definisanih merenja, TPC - H testa performansi, i to na sledeće analize:

1. Rezultata merenja performanse klastera prilikom upotrebe podataka obima 1TB, 3TB.

2. Uticaja performansi klastera sa 3, 4, 5 ili više čvorova u odnosu na definisane zahteve.

3. Upoređivanja rezultata na osnovu upita Q1 - Q22 [20] (Tabela 5) na osnovu TPC-H testa performansi.

4. Rezultata na osnovu kojih se može doći do zaključka kako doći do povećanja performansi klastera prilikom povećanja broja čvorova u klasteru.

U okviru ove doktorske disertacije predstavljeni su rezultati koji se vezuju za teorijski i praktični doprinos u oblasti arhitekture sistema visokih performansi. Oni se mogu koristiti za obradu podataka na klasterima za obradu velike količine podataka.

1.6 Organizacija disertacije

Doktorska disertacija je organizovana u devet poglavlja. U uvodnom poglavlju, opisuje se predmet i cilj istraživanja, istovremeno se daju osnovne pretpostavke istraživanja uz sagledavanje različitih aspekata. U ovom poglavlju su takođe predstavljeni materijali i metode istraživanja. Materijali predstavljaju hardverske komponente koje su upotrebljene. Metode se odnose na postupak njihove upotrebe.

Drugo poglavlje uvodi definiciju i razvoj Big Data kao discipline za obradu velike količine podataka. Opisuje različite tipove podataka, kao i način njihove obrade. Dalje, u ovom poglavlju se opisuju i različiti izazovi klaster sistema, koji se koriste za obradu velike količine podataka. Opisuju su tradicionalni silos sistemi, koji se upoređuju sa modernim sistemima za obradu velike količine podataka, kao i njihovim karakteristikama. Značaj takvih klaster sistema je takođe opisan kroz načine njihove upotrebe za rešavanje problema iz različitih domena. Na kraju, moderna industrija i njeni trendovi imaju revolucionarni razvoj između ostalog upravo i zbog upotrebe ovakvih klaster sistema.

U trećem poglavlju je opisan eksperiment kao takav u smislu praktičnog izvršenja, a koji se dalje povezuje sa testom performansi (engl. Benchmark), kao i dizajn optimalnog testa performansi za njihovo merenje na osnovu postavljenog cilja. Ovim se pokazuje sposobnost dizajna i izgradnje sistema u uslovima koji su definisani krajnjim korisnikom, u realnom okruženju i u realnim uslovima rada. Dalje, opisane su pojedinačne funkcije delova sistema, zahtevi za upotrebu alata, kao i sama priprema eksperimenta. Na kraju, pored opisanog testa performansi, predstavljeni su rezultati, svrha njihove upotrebe za izgradnju finalnog rešenja

sistema za obradu velike količine podataka. Pored ovoga, predstavljeni rezultati uključuju i alate bez kojih ovaj sistem ne bi mogao funkcionisati u saglasnosti sa zahtevima krajnjeg korisnika.

Četvrto poglavlje predstavlja opis analize informacionih sistema kao što su tzv. silos sistemi organizovani za smeštanje i obradu podataka. Predstavljaju se njihovi limiti, nedostaci, ograničenja kao i karakteristike i upoređuju se sa Big Data sistemima za obradu velike količine podataka. Dalje, u delu koji se odnosi na sisteme za obradu velike količine podataka, predstavljena je uopštena arhitektura koja se može koristiti u različitim domenima modernog društva. Navode se alati kao i njihova upotreba, opisuju se trendovi i zahtevi za moderne informacione sisteme.

Peto poglavlje opisuje rezultate merenja benchmark testa u smislu projektovanja arhitekture sistema visokih performansi i sistema za generalnu obradu podataka. Kao suštinski cilj ove doktorske disertacije, navedena je metodologija za kreiranje klastera, model univerzalne arhitekture sistema kao i način njegove implementacije. Pored ostalog, u ovom poglavlju se opisuje realna arhitektura koja je praktično korišćena za izbor sistema, sa najboljim performansama na Institutu zdravstvenih informacionih sistema i statistike Češke Republike.

Šesto poglavlje opisuje praktičnu upotrebu sistema za obradu podataka velikog obima izgrađenog na klaster arhitekturi.

Sedmo poglavlje navodi primere praktične upotrebe platforme tj. njene arhitekture sistema visokih performansi za generalnu obradu velike količine podataka na klasterima. Ono prikazuje pojedinačne komponente koje su upotrebljene za izgradnju predloženog sistema za obradu velike količine podataka.

Osmo poglavlje predstavlja diskusiju nad upotrebljenim tehnologijama, kao i njihovo mogućnosti vezane za unapređenje i upotrebu u oblasti zdravstva.

Deveto poglavlje navodi glavne zaključke doktorske disertacije primenjene prilikom izgradnje klaster sistema a za podršku nacionalne strategije za usvajanje analitičkih sistema u oblasti zdravstva Češke Republike za obradu velike količine podataka. Ona opisuje zaključke dizajnirane arhitekture sistema za obradu ovakvih podataka. Uz to, navode se i aspekti upotrebljenih tehnologija kao i sumiraju se odgovori na pitanja koja se vezuju za povećanje

performansi, kako postići sisteme visokih performansi i pri tom obezbediti optimalne troškove za upotrebu sistema kao takvog.

Poslednje poglavlje navodi upotrebljenu literaturu.

Dodaci A, B i C navode tehničke uslove i pretpostavke koje se vezuju za evaluaciju predložene platforme i proveru ispunjenosti uslova. Dalje, navode popis beneficijalnih funkcija, definicija tehničkih kriterijuma kao i ekonomskih parametara.

Dodatak A opisuje zahteve za ispunjene tehničkih uslova.

U delu označenom kao Dodatak B, navedeni su zahtevi minimalnog ispunjenja.

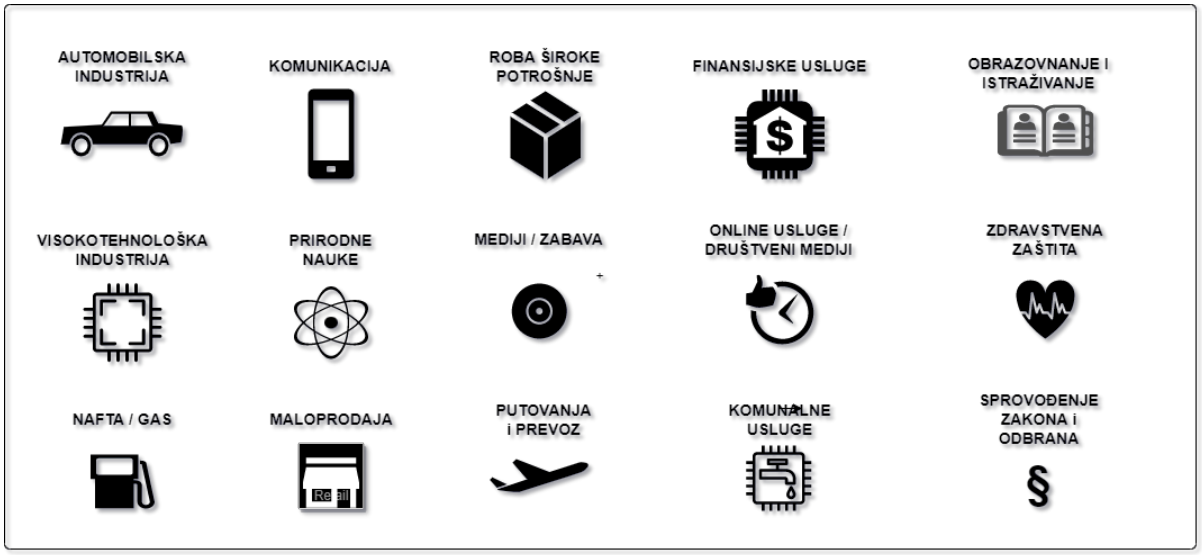
Dodatak C opisuje test performansi u smislu tendera za proveru brzine platforme, tj. klastera koji bi trebao biti označen kao pobedničko rešenje za izgradnju arhitekture klaster sistema za obradu velikog obima podataka.

2. Definicija i razvoj velike količina podataka

Tokom poslednjih godina mogli smo videti značajne promene u razvoju računarskih sistema [24-28], na osnovu uticaja značajnog povećanja količine podataka [29-32]. Takođe, ne mali broj softverskih aplikacija je stvoren baš na osnovu ovakvog fenomena[33]. Bilo da se radi o komercijalnoj, javnoj ili naučnoj oblasti, njihove primene možemo uočiti takođe na osnovu različitih izvora podataka. Takvi izvori podataka mogu biti bilo kakvi informacioni sistemi koji generišu veliku količinu podataka [34] u realnom vremenu, kao što su IoT [35] uređaji, RFID [36] uređaji i sl. Obrada ovakvih podataka koji neprekidno rastu, zahteva veliku efektivnost informacionih sistema[37, 38] ukoliko se ta obrada izvršava na jednom serveru [39], što predstavlja „usko grlo“ i nedostatak sistema. Kao rezultat ovakve vrste nedostatka sve veći broj organizacija uvodi obradu podataka na sistemima koji moraju biti skalabilni i implementirani tako da omoguće paralelnu obradu podataka. Takvi sistemi su implementirani na principu klaster. Kao rezultat upotrebe više navedenih principa, prilikom obrade velike količine podataka, uvodi obrada podataka na klaster sistemima [40].

2.1 Oblasti primene klastera za obradu velike količine podataka

U modernom društvu, princip obrade velike količine podataka je prisutan pre svega u okviru velikih organizacija, kao i u različitim industrijskim oblasti (Slika 1) [19, 41-43] koje se sreću sa velikom količinom podataka ali i sa raznim tipovima (strukturni, nestrukturni i polu-strukturni podaci). U novije vreme, ovaj princip obrade podataka se postepeno uvodi u svim segmentima ljudske delatnosti [27, 44-46]. Vrlo često se vode debate na temu da li je čovečanstvo na pragu revolucije, uz mnoge promene u trenutnom funkcionisanju sveta kroz ovaj i ovakav pristup. Međutim, opisom i identifikacijom oblasti u kojima se koristi ovakav način obrade [47] velike količine podataka, moguće je ukazati da on zauzima važno mesto u mnogim oblastima ljudske delatnosti.



Slika 1 - Oblasti primene Big Data (Domeni upotrebe)

Industrija

U poslednje vreme u ovoj oblasti se najčešće pominje takozvana „Industrija 4.0“, što je naziv za četvrtu industrijsku revoluciju, koja se postavljena uglavnom na razvoju takozvanih „pametnih fabrika“.

Ovde, nekoliko tehnologija treba da budu međusobno povezane. Različita saznanja iz ove oblasti podataka velikog obima su sastavni deo povezivanja ovih tehnoloških dostignuća. Radi se uglavnom o automatizacije, koja sve više i više prodire u trend digitalizacije. To znači da će fabrike generisati više podataka ili ih prikupljati iz posrednog ili neposrednog okruženja. Na osnovu njih će pojedinačni proizvodni subjekti automatski donositi odluke bez ili minimalne ljudske intervencije.

Nauka i istraživanje

Evropska organizacija za nuklearna istraživanja (CERN) je najveći svetski istraživački centar u fizici čestica. Ona je dobar primer praktične primene tehnologija velikih podataka u nauci. Protok podataka iz sva četiri eksperimenta (Alice, Atlas, CMS, LHCb) za takozvani „Run 2“, koji je bio planiran od 2015. do 2018. godine, procenjen je na 25 GB/s, dok se čuva samo 0,01% podataka. Na ovim eksperimentima radi oko 8.000 svetskih stručnjaka - analitičara. Svako može daljinski pristupiti i analizirati neke od dobijenih podataka u skoro realnom vremenu. Za analitičke aktivnosti koriste se neke od tehnologija, kao što su na primer: Vertica, Apache Hadoop [48, 49], R, Apache Pig, Apache Hive, Apache Spark i mnoge druge.

Zdravstvena zaštita

U zdravstvu se očekuje veliki broj različitih analiza podataka, kao i njihove raznorodne kompleksnosti na osnovu postavljenih zahteva. To je iz razloga što zdravstvene ustanove sadrže veliki broj zapisa podataka koji se nedovoljno dostupan. Jedan od ključnih razloga je osetljivost zdravstvenih podataka. U ovim zapisima, pored ostalog, mogu se naći i vrlo kompleksni i međusobno zavisni podaci, što dodatno stvara zahteve za računarske resurse i kapacitete klastera. Istovremeno, ključni, pametni uređaji koji su neposredno u vezi sa ovako koncipiranim rešenjima, a koji na primer mere osnovne vitalne funkcije sistema, mogu predvideti određene događaje opasne po život. Na primer, to može biti srčani zastoj ili moždani udar kod ljudi tj. kasnije pacijenta. Generalno, na osnovu ovoga možemo pretpostaviti da obradom velike količine podataka možemo poboljšati prevenciju ili ranu dijagnozu, kao lečenje bolesti kod pacijenata. Takođe može se predvideti pojavu raznih epidemija, kao što je prikazano „Google Flu Trends“ [50]. Izgrađena rešenja na principu klasteru za obradu velike količine podataka koriste se između ostalog i u slučaju širenja epidemija, pandemija [51]. Na primer, praćenjem prenošenja virusa kod stanovništva, broj zaraženih stanovnika, vrste populacije, starosna kategorija i sl. može se na vrlo efikasan način smanjiti prenošenje virusa i njegovog uticaja na sve aspekte civilizovanog društva.

Razvoj tehničke opreme

Primer može biti razvoj trkačkog automobila Formule F1. Ovi automobili se sastoje od oko 25.000 komponenti, od čega 11.000 ima karoserija, 6.000 motor, a 8.500 elektronika. Svaka od ovih komponenti nosi sa sobom određeni rizik od oštećenja. Zbog toga su automobili različito testirani i opremljeni senzorima koji mere sve vrste voznih karakteristika i karakteristika samog automobila. Nije ni čudo što su tokom Velike nagrade SAD još 2014 godine, trkački timovi prikupili 243 TB podataka . Od posebnog interesa za ove automobile je da se oko 5.000 komponenti menja svake 2 nedelje tokom perioda razvoja. One su napravljeni po meri na osnovu analize prikupljenih podataka.

Društvene mreže

Zbog svoje ogromne popularnosti, društvene mreže su predmet velikog interesovanja mnogih analitičkih subjekata. Ranije je bilo moguće dobiti podatke samo putem upitnika ili postupkom zapažanja, a sada je moguće masovno realizovati detaljne analize zahvaljujući

uspešnom i sofisticiranim načinu prikupljanja podataka. Zahvaljujući društvenim mrežama mogu se raditi razne psihometrijske analize. Na primer, korisnici društvenih mreža često objavljuju puno informacija i ličnih podataka na osnovu kojih mogu biti napravljene vrlo komplekse analize takvih osoba. Ovde treba napomenuti, da publiciranje ličnih podataka ima za posledicu povećanja bezbednosnog rizika u smislu zloupotrebe podataka. Ovo nije predmet ove doktorske disertacije i predstavlja samostalnu temu u smislu analize podataka u smislu bezbednosti, osetljivosti i njihovog uticaja na savremeno civilizacijsko društvo..

Maloprodaja

Mnogo kompanija u ovom sektoru radi sa različitim vrstama podataka. To mogu biti podaci od kupaca proizvoda, kartica lojalnosti, ekonomski i demografski podaci, pa preko podataka društvenih mreža i veba uopšte. Zahvaljujući ovim informacijama, oni mogu, na primer, da kreiraju poslovne strategije, prognoziraju potražnju i prodaju proizvoda, optimizuju cene i identifikuju kupce i prate njihove osobine u potrošačkom smislu reči.

Finansije i bankarstvo

Ovaj sektor je pod velikim nadzorom bankarskih regulatorskih organa. Glavni razlog je bezbednost podataka, njihova osetljivost i kontrolisana upotreba. Na drugu stranu, kupcima je takođe potrebno sve više individualnih usluga, pa se najčešće razvijaju i uvode u praktičnu primenu za korisničku upotrebu, na osnovu raznih tehnika analize velike količine podataka. Takvi su na primer segmentacija kupaca, upravljanje rizicima, praćenje poslovanja, otkrivanje prevara i sl.

2.2 Osnovne karakteristike velike količine podataka

U informacionim tehnologijama podaci velikog obima [10] zahtevaju analizu, obradu i smeštanje podataka koji potiču iz različitih izvora podataka [52]. Ovi podaci se u praksi pojavljuju onda kada tradicionalni analitički sistemi nisu u stanju da obezbede analizu podataka, njihovu obradu i smeštanje ili onda kada tradicionalne tehnike iz ove oblasti nisu dovoljne [10]. Od 1997. godine počinje se sa upotrebom više atributa za pojam podataka velikog obima. Razlikujemo tri atributa koja postaju vrlo popularna. Oni su stalno u upotrebi i široko citiranja u naučnom istraživanju. Radi se o tzv. Gartner interpretaciji [53] ili 3V atributima. Osnovu ovog termina postavio je Douglas Laney [54]. On je primetio da su zbog

naglog povećanja aktivnosti e-poslovanja podaci porasli u tri dimenzije, i to: (1) **Obim - Volume**, što znači dolazni tok podataka i kumulativni obim podataka; (2) **Brzina - Velocity**, koja predstavlja tempo podataka koji se koriste za podršku interakciji i generišu se interakcijama; (3) **Raznolikost - Variety**, koja označava raznolikost nekompatibilnih i nedoslednih formata podataka i struktura podataka. Definicija 3V Douglas Laney-a, koja se odnosi na atribute velikih podataka [17], je dugo smatrana „uobičajenom". Kako se ova oblast razvijala, nastavilo se sa dodeljivanjem dodatnih „V" atributa koje se odnose na pojam „podataka velikog obima".

Slika 2 - Karakteristike velike količine podataka – atributi

Osnovna karakteristika velike količine podataka je tzv. osobina 7V koja opisuje njihovu različitost u odnosu na tzv. tradicionalne podatke (Slika 2). Osobina 7V se odnosi na: **(1) Volume**, odnosno količina podataka, **(2) Velocity**, brzinu generisanja podataka i **(3) Variety**, podaci mogu biti različiti, struktuirani, ne-struktuirani i polu-struktuirani podaci. U poslednje vreme srećemo se sa dodatnim osobinama podataka velike količine kao što su: **(4) Value**, vrednost podataka koja se postiže nakon njihove interpretacije, **(5) Variability**, različitost podataka **(6) Veracity**, odnosi se na verodostojnost podataka u smislu njihove preciznosti tj. kvaliteta podataka **(7) Visualization**, koja se odnosi na metodu konzumacija informacija od strane korisnika.

2.3 Definicija i razvoj klastera za obradu velike količine podataka

Klaster predstavlja platformu sastavljenu od hardvera i softvera [55], koja je sastavljena od više čvorova (engl. node), a koja je podeljena i u upotrebi od strane većeg ili velikog broja korisnika. Klaster također zahteva runtime, koji istovremeno može biti skalabilan, a koji povećava mogućnost smetnje ili ispada bilo kod čvora samog klastera. Kao rezultat svega toga, iskorišćeni su različiti programski modeli koji služe za izradu, tj. dizajn klastera. Jedan od prvih modela bio je Google MapReduce [40, 56] predstavljen kao jednostavan i generalizovani model klastera za batch obradu podataka. On je bio u stanju da sam ispravi greške (hardvera, softvera) koje su se mogle desiti prilikom rada sistema.

Prvi izazov bio je pisanje programa za njegovo paralelno izvršavanje prilikom obrade podataka, kao i stvaranje programskog modela, koji je trebao zadovoljiti ovako postavljene zahteve. Dalji izazov ovakvih sistema se odnosi na greške, koje mogu nastati na klasterima kao takvim. Greške na nivou čvorova su rezultat njihovog ispada. Ovde se javlja problem performansi (sporo izvršavanje aplikacije na određenom čvoru ili njegovo neočekivano usporenje). Klaster je obično podeljen između više korisnika, koji izvršavaju više različitih programa. Pri tom oni imaju dinamičnu skalabilnost sistema ali i istovremeno realni rizik da dođe do ispada, kvara, tj. prekida funkcionisanja pojedinih komponenti klastera ili njegovih delova.

Budući da obrada velike količine podataka zahteva specijalizovane sisteme, tj. računare visokih performansi, u te svrhe se koristi klaster kao računarsko okruženje [57], koje nam daje prednosti u smislu paralelne obradu podataka (paralelno računarstvo).

Ovakvi specijalizovani sistemi iako na prvi pogleda izgledaju savršeni, nose sa sobom nedostatke, rizike i izazove prilikom rada [57], koje možemo ovako predstaviti:

1. **Obavljanje različitih operacija prilikom paralelne obrade podataka** – ovaj izazov se odnosi se na rešavanje osnovog problema kod klastera u smislu distribuirane obrade podataka i obavljanja različitih operacija nad podacima koji su smešteni u klasteru. Kao primer možemo navesti da SQL sistemi baza podataka [58] koriste algoritme za mašinsko učenje kako bi postigli optimalnu agregaciju na nivou paralelne obrade SQL upita nad podacima.

2. **Kompozicija klaster sistema** – odnosi se kao izazov na izvršavanje kompleksnih operacija nad podacima koje su glomazne s obzirom da treba obezbediti njihovo izvršenje nad velikom količinom podataka koji su smešteni u klasteru kao takvom, tj. u njegovim čvorovima. Ukoliko imamo zahteve koji postoje za operaciju eksporta, replikacije podataka ili njihovo smeštanje, ti zahtevi mogu dodatno uticati na cenu celokupnog rešenja u smislu njenog povećanja. Kompleksnije operacije nad podacima zahtevaju već performanse klastera koje je moguće postići adekvatnim povećanjem broja čvorova u klasteru uz odgovarajući izbor softverskih komponenti.

3. **Ograničena količina** – ukoliko radimo obradu velike količine podataka, aplikacija koja se izvršava na samom klasteru mora biti napisana u saglasnostima sa zahtevima klaster sistema na kome se izvršava. Radi se o specijalnom načinu programiranja za izvršavanje obrade velike količine podataka u što kraćem vremenskom periodu. Ograničenje količine obrade podataka je u direktnoj zavisnosti za kapacitetima projektovanog klastera, koji je u upotrebi. U slučaju da se pojavi potreba za proširenjem klastera, usred povećanja podataka, već je bilo napomenuti da je ova osobina klastera u direktnoj vezi sa njegovom osobinom skalabilnosti.

4. **Deljenje resursa** – na nivou klastera je prirodno da se pojedini ili svi čvorovi klastera dele prilikom obrade podataka. To je osnovni način za postizanje paralelne obrade velike količine podataka. Ovakva osobina klaster sistema nosi sa sobom i to, da je potrebno dodatno voditi računa o tome da pojedini čvorovi klastera mogu imati ispade u smislu funkcionisanja te ih je potreba blagovremeno popraviti ili kompletno zameniti novim.

5. **Upravljanje i administracija** – kako se radi o vrlo kompleksnom računarskom sistemu, pojedini njegovi delovi zahtevaju značajniju administraciju, ali i uzajamnu integraciju pomoću API interfejsa, kao načina uzajamnog povezivanja sistema u informacionim tehnologijama.

2.4 Pitanja vezana za izgradnju klastera za obradu velike količine podataka

Prilikom predlaganja arhitekture sistema visokih performansi za generalnu obradu velike količine podataka na klasterima možemo postaviti nekoliko ključnih pitanja koje su osnovne postavke za ovo istraživanje.

(1) Kako dizajnirati platformu za analizu performansi visoke propusnosti, tj. platformu otpornu na greške i kako obezbediti njenu efikasnu i isplativu upotrebu prilikom obrade velike količine podataka?

Platforma za obradu velike količine podataka mora biti izgrađena na principima klaster arhitekture, uz upotrebu tzv. tradicionalnog hardvera. Radi se o. ekonomski dostupnijim serverima tipa X86 od poznatih proizvođača kao što su DELL, HP, Lenovo, Fujitsu ili nekih drugih proizvođača ove platforme. Kako se radi o klaster arhitekturi, svaki od elemenata klaster arhitekture ima svoj rezervni režim na nivou mrežne komunikacije, hard diskova, operativne memorije, procesora, napajanja servera, servera, kao takvih pa do klaster menadžment sistema i replikacije podataka smeštenih na nivou klastera. Podaci su podeljeni na blokove podataka određene veličine (Hadoop verzija 2.x, 3.x, default veličina bloka podataka je 128 MB), koji su replikovani na klasteru. Podaci su replikovani kako na nivou jednog računarskog ormara, tako i na nivou više njih, ukoliko postoje (Rack Awareness). Uz pomoć TPC-H testa za testiranje performansi, pažljivim izborom komponenti možemo doći do traženog rešenja. Testiranjem celog klastera na performanse, doći ćemo do optimalnog broja servera koji čine klaster celokupnog rešenja.

(2) Kako možemo realizovati predloženu platformu uz potpuno poštovanje sofisticiranih zahteva koji se tiču performansi, bezbednosti?

Ciljno rešenje, tj. zahtevanu platformu možemo izgraditi na osnovu unapred dizajnirane arhitekture rešenja. Ukoliko pođemo od toga, da nam klaster arhitektura između ostalog omogućava horizontalno skaliranje sistema, onda možemo postupnim dopunjavanjem čvorova klastera povećavati ne samo performanse sistema, ali i njegove kapacitete. U našem slučaju, koristili smo rezultate izvršenih testova da bismo izračunali snagu obrade TPC-H test upita za svaku bazu podataka. Koristili smo sledeću jednačinu [20]:

$$\mathrm{TPC-H\_Power@Size} = \frac{3600 * \mathrm{SF}}{\sqrt[24]{\prod_{i=1}^{22} \mathrm{QI}(i,0) * \prod_{j=1}^{2} \mathrm{RI}(j,0)}} \qquad (1)$$

gde je QI (i,0) vremenski interval, u sekundama, upita Qi unutar jednog toka upita u TPC-H test performansi; RI(j,0) je vremenski interval u sekundama funkcije osvežavanja RIj unutar jednog toka upita testa. SF predstavlja odgovarajući faktor razmere veličine baze podataka.

(3) Kako efikasno razviti savremena softverska rešenja koja rade na arhitekturi koja omogućava visoke performanse, njenu horizontalnu skalabilnost uz održavanje kontinuiranog rada i obezbeđivanje visoke dostupnosti?

Savremena softverska rešenja je moguće razviti uz praćenje tehnologija kao i njihovih osobina. Specijalizovane namene u okviru date problematike u domenu u kojem se ta tehnologija najbolje može upotrebiti. Zbog osobina arhitekture kao takve, mogućnosti njenog horizontalnog skaliranja potrebno je dobro proceniti na osnovu npr. testa performansi koje performanse želimo postići. U sledećem periodu testiranja, usled povećanja podataka, doći će do potencijalnog povećanja zahteva na platformu kao takvu, kako bi smo omogućili njen dalji razvoj. Ovaj razvoj je moguće postići na osnovu dodavanja dodatnih, novih čvorova klastera, kako bi se povećali parametri performansi i kapaciteta klaster rešenja.

3. Test performansi klastera za obradu velike količine podataka

Test performansi [59] (engl. benchmark) je postupak poređenja dva ili više proizvoda, procesa, rezultata ili metoda. On predstavlja uporedni test performansi gde tokom njegovog izvršenja se testiraju na primer dva ili više proizvoda, rešenja. Pitanje koje se često postavlja u praksi je šta čini kvalitetan test performansi? U proteklih desetak godina, prilikom obrade informacija došlo je do stvaranja desetina različitih „standarda" [60] koji se vezuju za testiranje performansi sistema. Neki od tih standarda su veoma efikasni i vrlo često u praksi zastupljeni, a neki od njih manje pošto ne sadrže sve detalje koje želimo između ostalog testirati.

Ukoliko želimo izgraditi platformu koja ima visoke performanse, potrebno je unapred postaviti referentne vrednosti testa koje želimo da dostignemo. Definisanje referentnih vrednosti počinje sa izgradnjom sistema, njegovom praktičnom upotrebom i proverom na osnovu izabranog testa performansi. Zasnivanje referentne vrednosti je isključivo izgrađeno na scenarijima upravljanja podacima iz „stvarnog života". Primeri iz stvarnog života karakterišu šta određena tehnologija trenutno može da uradi, radi postizanja očekivanih i zahtevanih performansi. Praktičnom proverom izabrane tehnologije, dolazimo do konkretnih rezultata, koje upoređujemo sa referentnim vrednostima i zahtevanim rezultatima, koje predloženi sistem mora ili očekujemo da će ispuniti. Na osnovu toga, dolazimo do zaključka koji nam pomaže, da steknemo predstavu o upotrebi sistema u realnim uslovima i realnoj sredini sa realnim podacima. Izmerene rezultate upoređujemo sa referentnim vrednostima ili vrednošću. Dobijene rezultate možemo iskoristiti za redizajn sistema ili promenu njegovih ključnih komponenti sa ciljem postizanja boljih performansi ili njihovog unapređenja.

Test performansi predstavlja neku vrstu merila za podršku u ispravnom i odgovornom odlučivanju, radi ispunjenja određenih ciljeva i parametara kvaliteta koji su neophodni za njihovo ispunjenje. U slučaju kada se radi o sistemima za odlučivanje, u okviru kompleksnih informacionih sistema, oni se obično sastoje od skupa poslovno orijentisanih ad-hoc upita i istovremenih modifikacija podataka, koji se smeštaju u bazu podataka. Na primer, puno proizvođača baza podataka tvrdi da su baš njihove baze podataka najbrže. Na osnovu testiranja performansi, možemo dobiti pouzdani odgovor na osnovu koga možemo napraviti procenu i doneti ispravnu odluku o odgovarajućoj bazi podataka, koja je predmet testiranja. Do takve procene dolazimo unapred pripremom baze podataka sa odgovarajućim podacima, koje u nju učitavamo. Za proveru funkcije i performanse baze podataka koristimo odgovarajuće upite u

bazu podataka uz pomoć specijalizovanog jezika (SQL), koji nam omogućava rad sa njom. U tom slučaju podaci koji se nalaze u bazi podataka kao i SQL upiti koje koristimo za interakciju sa njom, odabrani su tako da imaju široku upotrebu. Između ostalog, jedna od vrste upotreba tih upita je kód testiranja performansi baze podataka ili celokupnog sistema, koji sa tom bazom interaguje. Ovi SQL upiti imaju bar jednu ključnu ulogu. Upotrebom ovih upita u različitim merenjima, dolazimo do određenih zaključaka vezanih za performanse baze podataka ili celokupnog sistema na koji sigurno baza podataka ima uticaj.

Merenje performansi po unapred podešenim principima merenja dolazimo do zaključka da imaju važnu ulogu u unapređenju dizajnerskih i inženjerskih rešenja kod različitih sistema kao i kod sistema baza podataka. Na primer, Savet za performanse obrade transakcija (Transaction Process Performance Council - TPC) [20, 21] ima važnu ulogu u podsticanju usvajanja industrijskih merenja u računarstvu, koje danas naširoko koriste mnogi vodeći proizvođači kako bi demonstrirali performanse svojih proizvoda. Slično tome, veliki korisnici ovakvih sistema često koriste rezultate TPC-H test performansi [61] kao merljivu tačku poređenja, između novih računarskih sistema i upotrebljenih tehnologija za njihovu izgradnju. Jedan od razloga je i potreba za osiguranjem visokog nivo performansi u svom radnom okruženju i time postigli svoje poslovne ciljeve.

Ova doktorska disertacija ima ambiciju da istraži i predložiti opšte zahteve za sprovođenje testa performansi, koristeći postojeće industrijske standarde sa posebnim osvrtom na TPC-H test performansi [20].

3.1 Test performansi baze podataka

Da bi smo razumeli kako se performanse baza podataka mogu meriti, pomoći će nam ukoliko budemo razlikovali njihovu klasifikaciju i pogodnost za izgradnju specijalizovanih sistema. U smislu njihove strukture, može se smatrati da baze podataka mogu da pokreću dva široka tipa upita:

1. **Upiti za obradu transakcija na mreži (OLTP)** – Ovo su obično upiti koji se pokreću na tzv. dnevnoj bazi u preduzećima. Na primer, to može biti baza podataka za upravljanje odnosima sa klijentima (CRM) ili sistemi za planiranje resursa preduzeća (ERP), gde se na primer, vrednosti faktura ubacuju, ažuriraju ili brišu u bazama podataka na osnovu interakcije sa krajnjim korisnikom. Ovakve baze podataka

dizajnirane su za upisivanje velike količine podataka u kratkom vremenskom intervalu.

2. **Upiti za onlajn analitičku obradu (OLAP)** – Ovo su upiti koji se pokreću u svrhu generisanja izveštaja, obično od strane tima za poslovnu inteligenciju kompanije. Ovakvi ili slični upiti mogu biti generisani od strane drugih timova koju su zaduženi za obezbeđenje analitičkih uloga, kao npr. procenu KPI-ja (Key Performance Indicator), tj. praćenje metrika od npr. strane menadžmenta, srednjeg i višeg nivoa kompanije. Još jedna karakteristika koju treba primetiti je ta, da je baza podataka obično strukturirana kroz tabele činjenica i dimenzija koristeći nešto što je poznato kao dizajn zvezdaste šeme, kako bi se obezbedila brža obrada podataka. Ovakve baze podataka vrlo često se koriste za generisanje ad-hoc ili serijski pripremanih izveštaja.

TPC Benchmark™ H (TPC-H) je merilo za podršku u odlučivanju. Sastoji se od skupa poslovno orijentisanih ad-hoc upita i istovremenih modifikacija podataka. Upiti i podaci koji popunjavaju bazu podataka odabrani su tako da imaju široku relevantnost za čitavu industriju uz održavanje dovoljnog stepena lakoće implementacije. Ovo merilo ilustruje sisteme za podršku u odlučivanju koji:

1. Ispituju velike količine podataka
2. Izvršavaju upite visokog stepena kompleksnosti
3. Daju odgovore na kritična poslovna pitanja

TPC-H procenjuje performanse različitih sistema za podršku u odlučivanju izvršavanjem skupova upita prema standardnoj bazi podataka pod kontrolisanim uslovima. TPC-H postavlja upite, koji:

1. Daju odgovore na stvarna poslovna pitanja
2. Simuliraju generisane ad-hoc upita (npr. preko grafičkog korisničkog interfejsa)
3. Daleko su kompleksniji od većine OLTP transakcija
4. Uključuju širok spektar elemenata podataka na kojim se izvršava operacija i ograničenja selektivnosti

5. Generišu intenzivne aktivnosti na delu serverskih komponenti baze podataka sistema koji je predmet testiranja

6. Izvršavaju se prema bazi podataka koja je u skladu sa specifičnim zahtevima za populaciju i skaliranje

7. Implementiraju se uz ograničenja koja proizilaze iz bliskog usklađivanja sa on-line proizvodnom bazom podataka

TPC-H operacije su modelirane na sledeći način:

1. Baza podataka je kontinuirano dostupna 24 sata dnevno, 7 dana u nedelji, za ad-hoc upite od više krajnjih korisnika i modifikacije podataka u odnosu na sve tabele, osim eventualno retkih sesija održavanja (npr. jednom mesečno).

2. TPC-H baza podataka prati, verovatno sa izvesnim zakašnjenjem, stanje OLTP baze podataka kroz tekuće funkcije osvežavanja koje skupljaju brojne promene koje utiču na neki deo baze podataka za podršku u odlučivanju.

3. Zbog prirode poslovnih podataka koji se čuvaju u bazi podataka TPC-H, upiti i funkcije osvežavanja mogu se izvršavati prema bazi podataka u bilo kom trenutku, posebno u međusobnom odnosu. Pored toga, ova kombinacija upita i funkcija osvežavanja podleže specifičnim ACID (Atomicity, Consistency, Isolation, Durability) zahtevima, pošto se upiti i funkcije osvežavanja mogu izvršavati istovremeno.

4. Da bi se postigao optimalan kompromis između performansi i operativnih zahteva, administrator baze podataka može trajno postaviti nivoe zaključavanja i pravila istovremenog planiranja za upite i funkcije osvežavanja.

3.2 Cilj testa performansi kod klastera

Ciljem testa performansi kod klastera je iskazati sposobnost ponuđenog rešenja da izvrši unapred definisane zadatke na osnovu kojih će sistem biti isporučen, tj. kupljen od strane korisnika i pušten u rad. Predloženo rešenje mora ispuniti u očekivanom vremenu određene performanse koje se vezuje za njega, kao i njegov kvalitet, njegovu upotrebljivost pod uslovima uporedivim sa planiranom implementacijom u realnom okruženju. Za potrebe PoC-a, ovi

zadaci i strukture podataka su značajno pojednostavljeni da oponašaju, a ne direktno implementiraju budući sistem. Korišćeni podaci sadrže približno isti broj redova očekivanih podataka, ali sa smanjenim brojem kolona sa značenjem i veštačkim dodavanjem suvišnih atributa kako bi se održala približna potrebna količina podataka. Podaci za potrebe PoC-a su generisani i ne sadrže nikakve realno upotrebljive podatke (ne sadrže lične podatke ili poslovne tajne, već su tzv. anonimni).

Projektovanje sistema visokih performansi sa statističkim i analitičkim funkcijama, koji će se koristiti za pripremu i razvoj ad-hoc statističkih i analitičkih izlaza detaljno su predstavljeni u poglavlju 5.

U sledećim poglavljima, opisani su generalizovani zahtevi koji čine okosnicu platforme za obradu velike količine podataka. Platforme za obradu velike količine podataka nude sledeće funkcionalnosti i možemo ih podeliti u tri kategorije: osnovne, dodatne i proširene.

Osnovni zahtevi platforme za obradu velike količine podataka su:

1. Učitavanje podataka iz datoteka i baza podataka, spajanje, transformisanje i modifikovanje podataka, uključujući agregacije, merenje, deduplikaciju, eksport podataka, generisanje izlaza u obliku tabela i grafikona, kao i uključenje mogućnost njihovog uređivanja.

2. Osnovne deskriptivne statističke funkcije – srednje vrednosti, percentili, tabele učestalosti i histogrami, tabele kontingencije, parametarski i ne parametarski testovi, standardne devijacije i greške, analiza varijanse, intervali poverenja itd.

3. Modeliranje i njihova primena za evaluaciju problema u odnosu na predikativne analitičke modele – modeli korelacije i regresije uključujući multidimenzionalnu i nelinearnu ili multidimenzionalnu logističku regresiju, diskriminantnu analizu, grupisanje, metode najmanjih kvadrata uključujući preračunavanje (odmeravanje, ravnoteža) i dvostepene, višedimenzionalno skaliranje, modele odnosa između kategoričkih varijabli, višedimenzionalni opšti linearni model i sl.

4. Interaktivno kreiranje kompleksnih korisničkih tabela.

Dodatni zahtevi platforme za obradu velike količine podataka su:

1. Potrebna je godišnja korisnička podrška, uključujući uputstvo za ispravljanje grešaka i ažuriranje novih verzija alata.

2. Čak i nakon godinu dana podrške celokupnog rešenja od strane isporučioca, licence moraju biti i dalje važeće. Upotrebljeni alati se i dalje mogu koristiti bez korisničke podrške, pošto se radi o trajnim tj. vremenski neograničenim licencama.

3. Serverska licenca je potrebna za paralelnu izvršenje zadataka računanja na serveru sa najmanje 7 jezgara (14 niti) dvoprocesorskog sistema sa procesorima koji su u skladu sa arhitekturom Intel Xeon E7 (fizički ili virtuelni procesori).

4. Potrebna je licenca za najmanje 7 korisnika klijentske aplikacije na krajnjim uređajima koji pokreću gore navedene analize na serveru (istovremeno) ili u klijentskoj aplikaciji (obavezne su obe opcije pokretanja).

5. Potrebna je godišnja korisnička podrška, uključujući metodološki postupak za ispravljanje grešaka i ažuriranje novih verzija alata.

6. Licenca je potrebna za najmanje 3 korisnika klijentskih aplikacija na krajnjim uređajima (pored licenci prema tački 1), koji ovde mogu da izvrše sve analize osnovne i napredne opreme alata.

7. Ova 3 korisnika takođe mogu da pokrenu (istovremeno) analize barem osnovne varijante na serverskom delu alata.

Prošireni zahtevi platforme za obradu velike količine podataka podržavaju iste funkcije kao u osnovnoj varijanti, koji su dodatno dopunjeni analitičkim metodama i oni su:

1. Analiza višedimenzionalnih kategoričkih podataka (optimalno skaliranje, kategorička regresivna analiza, mape percepcije i sl.).

2. Različite vrste stabala odlučivanja i regresije.

3. Metode za analizu vremenskih serija.

4. Modeli zasnovani na neuronskim mrežama.

5. Metode za definisanje i testiranje složenih uzoraka podataka.

6. Priprema podataka za analizu, uključujući profilisanje parametara, identifikaciju sumnjivih i ekstremnih vrednosti itd.

7. Analiza malih fajlova ili nedovoljno zastupljenih grupa.

8. Analiza nedostajućih vrednosti, njihove strukture, procene vrednosti koje nedostaju.

9. Metode validacije kompleksnog modela (bootstrapping).

3.3 Priprema testa performansi klastera

Priprema testa performansi je vrlo zahtevan i poseban deo celokupnog procesa. Naime, radi se o vrlo preciznom izboru ali i kompromisu prilikom izbora hardvera i softvera. Sa strane hardvera, radi se o tzv. tradicionalnim x86 serverima, koji imaju standardizovani jedan ili dva procesora. Dalji parametri su parametri operativne memorije, tip i veličina hard diskova. Među poslednjim parametrima spada i brzina mrežne kartice, koja mora biti minimalno 10Gb/s.

Sledeće karakteristike će se koristiti da bi se procenilo da li je određena implementacija klaster rešenja ima posebnu referentnu vrednost koju možemo uzeti u obzir radi stvaranje finalnog predloga rešenja:

1. Da li je implementacija klastera opšte dostupna, eksterno dokumentovana i podržana?

2. Da li njegova implementacija ima značajna ograničenja u pogledu upotrebe ili primenljivosti koja ograničavaju njenu upotrebu izvan TPC merila?

3. Da li je implementacija ili deo implementacije klastera i na koji način kvalitetno integrisana u veći proizvod?

4. Da li implementacija koristi posebne prednosti ograničene prirode TPC referentnih vrednosti (npr. profili upita, mešavina upita, konkurentnost i/ili konflikt, zahtevi za izolaciju, itd.) na način koji ne bi bio generalno primenljiv na okruženje i koje merilo predstavlja?

5. Da li dobavljač ohrabruje implementaciju od strane korisnika? (Ova mogućnost uključuje da se promoviše implementacija na način sličan drugim proizvodima i tehnologijama)?

6. Da li implementacija zahteva neuobičajenu sofisticiranost od strane krajnjeg korisnika, programera ili administratora sistema?

7. Da li se implementacija (uključujući beta verziju) kupuje ili koristi za aplikacije u oblasti tržišta koju predstavlja referentna vrednost? Koliko veb sajtova je to implementiralo? Koliko krajnjih korisnika ima koristi od toga? Ako se implementacija trenutno ne kupuje ili ne koristi, da li postoje dokazi koji ukazuju da će je kupiti ili koristiti značajan broj sajtova krajnjih korisnika, ukoliko se radi o novom proizvodu?

3.4 Rezultati testova performansi klastera

Rezultati testiranja će se koristiti u sledeće svrhe:

1. Prilikom predstavljanja rešenja gde je potrebno proveriti osnovne, nužne i potrebne osobine sistema,

2. Tokom testiranja, ceo PoC mora biti ponovljiv, uključujući sve faze inicijalnog učitavanja primarnih podataka. Rezultati na osnovu evaluacije se koriste radi provere ispunjenosti svih uslova testa i mogu biti verifikovani bilo u kome trenutku ili fazi PoC.

4. Tradicionalni i sistemi za obradu velike količine podataka

U informacionih tehnologija, sve više organizacija donosi odluku na osnovu podataka. Ti podaci su obično smešteni u informacione sisteme za poslovno skladištenje podataka (EDW - Enterprise Data Warehouse). Takav sistem predstavlja skladište za smeštaj velike količine podataka, koje postaje centralni deo organizacije u smislu smeštanja i infrastrukture podataka. Dalji pojam koji se vezuje za sisteme za poslovno skladištenje podataka je ETL sistem (Extract Transform Load). Podaci (Word, Excel, Power Point, PDF dokumenti, email, fotografije ili skraćeno alfa-numerički i grafički podaci) su smešteni u jednu ili više baza podataka (Data Storage). Otuda se ekstrahuju, transformišu i prenose u deo sistema koji je odgovoran za analizu, analitički softver. U slučaju sistema za obradu velike količine podataka obično podaci bivaju smešteni u delu za smeštanje podataka, vrlo često izgrađenom na Hadoop, fajl sistemu. Ovaj fajl sistem može smestiti podatke različitih formata. Hadoop predstavlja distribuirani fajl sistem koji je instaliran na klaster sistemu i koji služi za smeštanje velike količine podataka. Ovi podaci su istovremeno različitih tipova bez ikakvog ograničenja i sa vrlim niskim i efikasnim troškovima vezanim za njihovo smeštanje. U ovom slučaju, zahtevi za posebnu kontrolu su vrlo jednostavni i nisu bazirani i fokusirani u delu koji se tiče smeštanja velike količine podataka.

U sledećoj tabeli (Tabela 1) je prikazano poređenje tradicionalnih sistema sa sistemima za obradu velike količine podataka. Suštinska razlika je u tome da tradicionalni sistemi generišu podatke unutar svojih granica. Imaju jedan, dva i više reda veličine manje količine podataka. Kod podataka sa kojima rade možemo prepoznati ključnu razliku u odnosu na sisteme za obradu velike količine podataka. Oni isključivo rade sa struktuiranim oblikom podataka, gde se obično radi o relacionim bazama podataka. Ovi drugi imaju prednost u tome da rade sa svim vrstama podataka, tj. strukturnim, nestrukturiranim ili polu strukturnim. Sistemi za obradu velike količine podataka su izgrađeni na principima distribuirane obrade podataka. Konfiguracija ovakvih sistema je daleko komplikovanija od tradicionalnih kao i njihova integracija, iako oni pružaju mogućnosti integracije na svim nivoima ovakvih sistema, kao što je nivo za smeštanje podataka, srednji nivo, aplikacioni i prezentacioni deo sistema. I na kraju, kada je reč o performansama sistema za obradu velike količine podataka, možemo govoriti o izražajno snažnim, otpornim i vrlo efikasnim sistemima. Oni daju mogućnost korisnicima, kao ni jedan do sada sistem, koji nije ovakvog tipa, arhitekture i performansi.

#	Tradicionalni sistemi	Sistemi za obradu velike količine podataka
1.	Tradicionalni podaci se generišu u sistemima preduzeća	Generišu se izvan sistema preduzeća, tj. na svetskom nivou
2.	Rade sa bazama podataka	Rade sa strukturnim, polu strukturnim i nestrukturnim podacima
3.	Podaci se generišu na nivou sata, dana ili većeg vremenskog perioda	Podaci se generišu sa većom frekvencijom koja je većini slučajeva na nivou sekundi ili čak milisekundi
4.	Podaci se centralizovani i oni su upravljani centralnim načinom upravljanja	Podaci su distribuirani kao i forma upravljanja
5.	Integracija podataka je vrlo jednostavna	Integracija podataka je vrlo komplikovana i zahtevna
6.	Standardizovana konfiguracija sistema omogućava obradu podataka	Vrlo zahtevan način konfiguracije sistema radi obrade podataka, koji zahteva i visoko specijalizovani IT personal
7.	Veličina podataka je mala, red veličine TB	Velika količina podataka, koji počinje na nivou PB podataka
8.	Tradicionalni, poznati sistemi baza podataka su dovoljni za obradu ovih podataka	Posebna grupa sistema baza podataka ali i daljih alata je potrebna za obradu podataka
9.	Standardizovane funkcije za manipulaciju podacima u okviru sistema	Specijalizovane funkcije za manipulaciju podatak u okviru sistema
10.	Model baze podataka je striktno predstavljen	Model baze podataka je ravan i dinamički
11.	Tradicionalni podaci imaju između sebe relacije	Velika količina podataka je podložna promenama i nemaju međusobne relacije
12.	Količina ovih podataka je relativno lak za upravljanje	Velika količina podataka je veoma komplikovana i zahtevna za upravljanje
13.	Izvor podataka uključuje ERP, CRM, finansijski sistem, organizacioni, HR, veb sistem	Izvori podataka uključuju socijalne mreže, podatke uređaja, senzorskih podataka, video, slike, audio podatke i sl.

Tabela 1 - Upoređenje tradicionalnih sistema i sistema za obradu velike količine podataka

4.1 Tradicionalni (silos) sistemi za obradu podataka

Tradicionalni sistemi imaju osnovnu osobinu arhitekture koja je obično organizovana u tri sloja: sloj baza podataka, aplikacioni i prezentacioni sloj. Ovi sistemi imaju vrlo prepoznatljive limite prilikom njihove izgradnje i upotrebe u smislu performansi. Kao osnova potreba koja se uobičajeno pojavljuje kod informacionih sistema je povećanje njihovog kapaciteta. Tradicionalni sistemi pružaju takvu mogućnost ali su vrlo izraženi tehnološki limiti. Kao primer možemo da navedemo server baze podataka ili aplikacioni server. Ukoliko želimo povećati njegovu procesorsku snagu, a pretpostavimo da imamo server sa dva soketa za procesor od kojih je prvi već zauzet, možemo dodati još jedan odgovarajući procesor i time smo dostigli hardverski limit platforme. Slična ili ista situacija je sa operacionom memorijom, diskovima.

4.1.1 Nedostaci tradicionalnih sistema za obradu podataka

Tradicionalni sistemi kao takvi imaju ključnu osobinu da se obično radi o sistemima koji imaju bazu ili baze podataka, aplikacioni i prezentacioni deo. Svi ovi delovi aplikacije su povezani u jednu celinu kako bi bio obezbeđen osnovni aspekt rada aplikacije. Prilikom rada, tj. upotrebe tradicionalnih sistema, može doći do pojave daljih potreba za razvojem ovakvog sistema, pogotovo u oblasti performansi. Usled razvoja ili npr. zbog kontinualnog razvoja tehnologija, može doći do daljih zahteva za poboljšanje parametara ovakvog sistema. Jedan od zahteva može biti i zahtev za poboljšanje performansi izabrane CPU ili RAM memorije, kapaciteta diskova i sl. U ovom slučaju možemo se sresti sa nekakvim ograničenjima ili limitima, koja realno mogu uticati na otežani razvoj našeg eSistema, ukoliko nemamo mogućnosti za obezbeđenje ovakvih promena.

Generalno kod ovakvih tradicionalnih sistema možemo prepoznati sledeće nedostatke:

1. Ograničeni kapacitet za smeštanje podataka.
2. Ograničenje kapaciteta procesora za obradu podataka.
3. Nemaju mogućnost skalabilnosti.
4. Osetljivi su na kvarove sistema usled čega može doći do izuzeća cele platforme.

5. Sekvencijalni način obrade podataka, koji vodi ka sniženju performansi cele platforme.

6. Baze podataka su isključivo relacione (RDBMS) i mogu smeštati isključivo strukturne podatke.

4.1.2 Tehnološka ograničenja servera

Upotrebljena tehnologija za izradu klastera, ima svoje limite i od početka u toku proizvodnje ima podešene parametre koji se tiču razvoja. U praksi se srećemo sa tim, da jedan server ima maksimalni broj procesora koje možemo instalirati na osnovnoj (matičnoj) ploči. Dalji primer je broj slotova kod RAM memorije ili broj portova kod hard diskova. Istovremeno, propusnosti mogu biti ograničene kako u smislu mrežne kartice ali npr. i magistrale servera. Sve ovo gore navedeno vodi ka jednom zaključku, da ovakva tehnologija ima svoje limite ali istovremeno ima i svoje maksimume u smislu performansi.

4.1.3 Silos arhitektura informacionih sistema za obradu podataka

Informacioni sistemi se obično sastoje od sistema ili servera za baze podataka, aplikacionih servera na kojima radi aplikacije, kao i prezentacionog dela, koji omogućava prenos podataka ka korisniku informacionog sistema preko računarske mreže.

Na nivou baze podataka u praksi imamo jedan ili više servera na kome imamo jednu ili više instanci baza podataka. Kao i svaki drugi server, serveri baza podataka se sastoje od CPU, RAM, HDD, mrežne karte, matične ploče i sl. Prilikom razvoja sistema baza podataka, jedna od uloga dizajnera sistema, je njegovo odgovarajuće dimenzioniranje u smislu njegovog odgovarajućeg kapaciteta i performansi tj. veličine. U slučaju kada je potrebno proširiti njegov kapacitet radi povećanja performansi, možemo doći u situaciju da to nije moguće. Razlog je taj, što maksimalno iskorišćenje ovakvog sistema, koje ne omogućava dalji razvoj, tj. ograničenost sistema.

Ovakvu situaciju možemo zapaziti i na ostalim nivoima kao što su aplikacioni i prezentacioni deo informacionog sistema. Ovaj problem je moguće rešiti kod informacionih sistema koji si organizovani na klaster principu.

4.2 Analitički sistemi za obradu velike količine podataka

Analitički sistemi za obradu velike količine podataka predstavljaju konfiguraciju umreženih servera tzv. čvorova (engl. node) koji između sebe imaju interakciju radi izvršenja određenog zadatka (npr. Map Reduce). Poznato je da je čvor predstavlja računar. Svima je dostupan i ima uobičajene karakteristike kao što su CPU, RAM, HDD, I/O i sl. Svaka od ovih komponenti može biti virtualizirana. Ovakav sistem pruža veoma visoke performanse prilikom obrade podataka u zavisnosti od broja čvorova. U zavisnosti od njegove veličine, broj čvorova može biti od nekoliko pa do nekoliko hiljada, kao npr. Facebook, Uber, Google ili drugi primeri [30, 62-64]. U novije vreme, ovakvi sistemi mogu biti hibridnog tipa. Hibridni sistemi imaju deo svoje infrastrukture u radnom okruženju korisnika ostatak je implementiran u sredini za računarstvo u oblaku.

Sledeća slika, Slika 3 prikazuje uopštenu arhitektura sistema za obradu velike količine podataka. U okviru nje razlikujemo:

1. Nivo podataka, gde možemo razlikovati strukturne podatke, ali i ne strukturne ili polustrukturne podatke. Ove podatke možemo pronaći u različitim informacionim sistemima i u različitim oblastima.

2. Integracioni deo platforme, koji služi za povezivanje nivoa podataka i njihovog sakupljanja iz različitih izvora podataka, radi transporta u sledeći nivo. Ovaj nivo služi za integraciju podataka, koji takođe može obezbediti transformaciju podataka u realnom ili odloženom vremenu, čišćenje podataka kao i rad sa meta podacima.

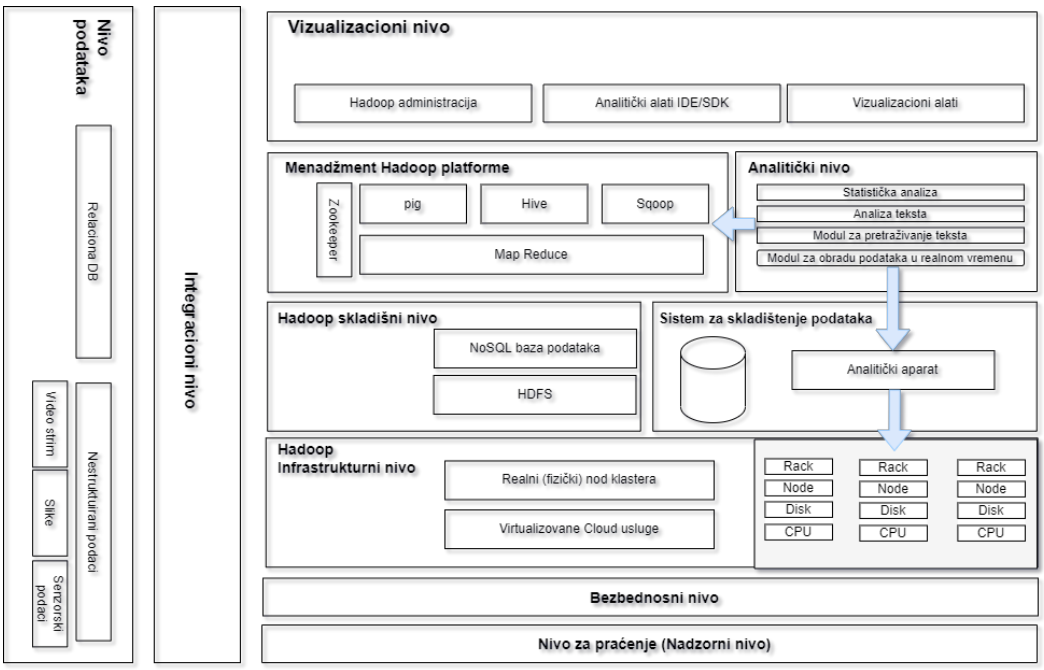
3. Infrastrukturni nivo se sastoji od fizičkih komponenti, koje se odnose na fizičku arhitekturu sistema za obradu velike količine podataka. Ovde se radi o fizičkoj organizaciji ali i povezivanju pojedinačnih komponenti ovog sistema. Ove komponente moraju ispunjavati striktne uslove, kao npr. minimalna brzina mreže, minimalni kapacitet hard diskova, broj procesora, njihova brzina i sl.

4. Sledeći nivo predstavlja logičku prezentaciju podataka smeštenih na nivou fajl sistema kod sistema za veliku obradu podataka. Pored specijalizovanog fajl sistema, obično se pojavljuju i baze podataka, koje su NoSQL tipa [65] ili kolonske baze podataka. Zbog svoje specifičnosti u smislu smeštanja podataka i njihovog indeksiranja, imaju dramatično bolje performanse od tradicionalnih relacionih baza

podataka. U suštini, ovde se radi o pretraživanju i radu sa podacima po kolonskoj organizaciji rada sa bazom podataka. Time se može postići znatno povećanje performansi, koje može biti veće red ili dve veličine u odnosu na tradicionalne RDBMS.

5. Menadžment nivo omogućava krajnjem korisniku upotrebu različitih programskih jezika (npr. Pig, Hive, Scala, Java, Kotlin, Python i sl.), radi upravljanja obradom velikom količinom podataka.

6. Poslednji nivo generalizovane arhitekture za obradu velike količine podataka, može se predstaviti slično kao i na prethodnim nivoima, kao skup različitih alata, koji su specijalizovani za tu svrhu obrade. Ovde možemo razlikovati deo koji se odnosi na pružanje usluga vezanih za poslove administracije sistema ali i deo koji je daleko važniji i odnosi se na krajnje korisnike. Ovaj deo omogućava krajnjim korisnicima vizualno predstavljanje podataka. Na taj način omogućava poboljšanje percepcije krajnjih korisnika u smislu obrađenih podataka koji su u okviru opisne platforme. Na ovaj način, krajnji korisnik može doneti na jednostavniji način svoje odluke zavisno od pozicije i vrste posla koju obavlja.



Slika 3 - Arhitektura sistema za obradu velike količine podataka

Na osnovu prethodno opisane platforme, koja predstavlja uopšteni ili generalizovani model, u okviru sledeće tabele (Tabela 2) su predstavljeni softverski proizvodi, tj. alati. Oni

se uobičajeno koriste u današnje vreme i vrlo su popularni u okviru savremenog doba obrade velike količine podataka. Prilikom realizacije projekta, potrebno je voditi računa i pažljivo izabrati odgovarajuće alate na osnovu potrebe i zadatka kojeg želimo ispuniti. U poslednje vreme, moguće je zapaziti da razna rešenja sadrže kombinaciju Open Source alata i komercijalnih proizvoda, što zbog cene, što zbog funkcionalnosti i njihovog kapaciteta. Istovremeno ove softverske produkte možemo naći u implementaciji kako u lokalnom okruženju krajnjeg korisnika, tako i kod subjekata koji pružaju razne usluge u sredini za računarstvo u oblaku. Na kraju, potrebno je napomenuti da na osnovu svojih osobina ovih alata, kao i mogućeg načina njihove upotrebe, možemo ih naći u tzv. hibridnoj ili mešovitoj implementaciji, te svako od takvih rešenja možemo na određeni način uzimati kao unikat.

Upotreba	Proizvodi/Alati
Nivo za prihvatanje podataka	Apache Flume, Apache Storm, Apache Kafka, Apache Sqoop
Hadoop smeštaj podataka	HDFS
NoSQL baza podataka	HBase, Cassandra
Automat pravila	MapReduce
NoSQL Data Warehouse	Hive
Alat za upite u bazu podataka	MapReduce, Pig, Hive
Alat za pretraživanje	Solr
Koordinacioni alat klastera	ZooKeeper, Oozie
Analitička obrada	R, Pentaho
Alat za vizualizaciju podataka	Tableau, Qlickview, PowerBI
Big Data analitički uređaj	Oracle Exalytics, IBM Pure System, EMC Greenplum
Monitoring	Ganglia, Nagios
Analitički IDE Big Data	Talend, Pentaho
Administracija Hadoop-a	Cloudera, DataStax, Hortonworks, IBM Big Insights
Cloud virtualni uređaji	Amazon AWS & S3, Rackspace, Azure, Oracle Cloud, Google Cloud

Tabela 2 - BDA uobičajeni softverski alati

4.3 Trendovi u IT na osnovu podataka, servisa i tehnike učenja

Nije tako teško objasniti i razumeti kako se sadašnji svet i civilizacija pomera ka tzv. digitalnom svetu, okruženom naprednim tehnologijama. Stoga, implementacija trendova velikih podataka za naše poslovanje može biti i definitivno će biti trend, koji je potrebno i moramo pratiti ukoliko želimo biti konkurentno sposobni na tržištu. Što pre prepoznamo ovaj trend, biće nam lakše da izaberemo put koji ćemo pratiti sledeći razvoj. Očekuje se da će tržište velikih podataka preći granicu od 245 milijardi USD do 2025. godine [66]. U nastavku se možemo osvrnuti na 10 aktualnih trendova za 2022. godinu u oblasti Big Data.

4.3.1 TinyML i Auto ML tehnika mašinskog učenja

TinyML je tehnika mašinskog učenja koji se povezuje sa uređajima male snage kao što su mikro kontroleri. Najbolji deo TinyML-a je taj koji radi sa malom latencijom. Troši mikrovate ili milivate što je milion puta manje od standardnog GPU (engl. Graphical Processing Unit). Na ovakav način TinyML-a omogućava uređajima da rade duži vremenski period, a u nekim slučajevima ovaj period može trajati čak i nekoliko godinama. Pošto imaju malu potrošnju energije, ne dozvoljavaju skladištenje podataka i to je najbolja prednost kada su u pitanju bezbednosni problemi.

Tehnika mašinskog učenja (AutoML) se smatra modernim mašinskim učenjem današnjeg vremena u kome živimo. Ona se koristi za smanjenje ljudske interakcije i automatsku obradu svih zadataka kako bi se rešili problemi iz stvarnog života. Ova funkcionalnost uključuje ceo proces od neobrađenih podataka do konačnog modela mašinskog učenja. Motiv AutoML-a je da ponudi opsežne tehnike učenja i modele za obične korisnike mašinskog učenja. Iako AutoML ne zahteva ljudsku interakciju, to ne znači da će je u potpunosti prevazići. Ovako to bar izgleda danas sa stanovišta saznanja i modernih aspekata u nauci.

4.3.2 Data Fabric koncept

Ovaj koncept se koristi na integracionom nivou podataka i njihovog procesa koji ih povezuju radi efektivnog odlučivanja. Data Fabric je već neko vreme u svetskom trendu i nastaviće svoju dominaciju i u narednom periodu sa velikom verovatnoćom još intenzivnije. Data Fabric predstavlja arhitekturu i grupu servisa podataka u okruženju za računarstvo u

oblaku. Na ovom mestu je potrebno napomenuti, da je struktura podataka navedena kao najbolji analitički alat od strane Gartner institucije (https://www.gartner.com). Sastoji se od ključnih tehnologija za upravljanje podacima koje uključuju tokove podataka, njihovu integraciju, upravljanje podacima i slično. Otvoreno je prihvaćen od strane preduzeća jer troši manje vremena za dobijanje poslovnih uvida koji mogu biti od pomoći za donošenje uspešnih poslovnih odluka.

4.3.3 Obrada prirodnog jezika

Obrada prirodnog jezika (engl. NLP - Natural Language Processing) je vrsta veštačke inteligencije koja pomaže u proceni unosa teksta ili glasa od strane ljudi. To je dostignuće sledećeg nivoa u tehnologiji, koje će sve više biti prisutnije u savremenom društvu. Vrlo lako možemo pronaći neke od primera u kojima možemo preneti zadatak na mašinu ili na robota da čita naglas umesto nas. NLP koristi metodologiju da izvuče nejasnoće u govoru i da mu pruži prirodan dodir. Naš najbolji primer može biti Apple-ov Siri, Amazon Alexa ili Google Asistent, gde razgovaramo sa veštačkom inteligencijom i ona nam pruža korisne informacije prema našim potrebama.

4.3.4 Migracija u okruženje za računarstvo u oblaku

U današnjem svetu tehnologije, moderna preduzeća se orijentišu ka tehnologijama alociranim u okruženjima za računarstvo u oblaku. Migracija u okruženje za računarstvo u oblaku je već neko vreme u trendu i ovo je budućnost celokupne tehnologije koja je već prisutna, ali će imati još veći intenzitet upotrebe. Kretanje ka računarstvu u oblaku ima nekoliko prednosti koje nisu vezane samo za preduzeća. Mi, kao pojedinci se takođe u potpunosti oslanjamo na tehnologiju rešenja za računarstvo u oblaku (email, skladište raznih vrsta i tipova podataka od dokumenata, fotografija, nacrta i sl.). Migracija na računarstvo u oblaku je od velike pomoći u pogledu performansi, brzine prilikom izvršenja raznih računarskih operacija i skalabilnost platforme bilo za koje operacije. Posebno je važna tokom učestalog saobraćaja u našem okruženju. Prelazak na računarstvo u oblaku i njegova prisutnost danas u velikoj meri utiče na funkcionisanje svih subjekata, kako komercijalnih tako i javih i državnih institucija. Sve većom prisutnosti ovih trendova, koji pored ostalih, imaju i ekonomski uticaj na sniženje troškova, sve više su prisutni rizici i izazovi koji se tiču bezbednosti podataka. Zbog toga, usled razvoja ovog aspekta ka računarstvu u oblaku imamo i rešenja koji su hibridna. Komercijalni korisnici, javne i državne institucije se ne retko odlučuju za hibridni režim upotrebe računarstva

u oblaku. Kod ovog principa deo informatičkog okruženja je u sredini korisnika a deo je smešten u oblaku.

4.3.5 Regulacija velike količine podataka

Pošto su industrije počele da menjaju svoje radne obrasce i mere poslovne odluke u smislu efikasnosti i efektivnosti, to im sada olakšava upravljanje svojim podacima i operacijama na efikasniji način. Međutim, veliki podaci tek treba da imaju i imaće sve veći uticaj na industriju. U stvari, neki industrijski segmenti i njihovi subjekti su počeli usvajati velike strukture podataka, ali je dug put do toga. Ovo se odnosi veliku odgovornost za predaju velike količine. Takve slučajeve možemo naći u nekim specifičnim segmentima industrije kao što su zdravstvo, saobraćaj, pravna oblast i sl. Bolji propisi o podacima će igrati glavnu ulogu tokom ove ali još više tokom narednih godina.

4.3.6 Kvalitet podataka kao IT trend

Kvalitet podataka je jedna od najvažnijih briga današnjih, modernih kompanija, koje svoju budućnost grade na obradi podataka, idealno, ukoliko je moguće u realnom vremenu. Ova briga je postala još veća kada su kompanije shvatile da im kvalitet podataka postaje sve veći problem. One kompanije koje se nisu fokusirale na kvalitet podataka proizilazeći iz različitih alata, suočavaju se sa rezultatom lošeg upravljanja na osnovu tih podataka. Razlog je taj što su „podaci“ zapravo glavna činjenica kod donošenja odluka. Zato se na osnovu podataka, uz pomoć raznih sistema, algoritama, tehnika i tehnologija donose vrlo važne odluke. Ove odluke se mogu praviti skoro u realnom vremenu. To je jedna od ključnih činjenica zašto podaci igraju ključnu ulogu u savremenom i modernom društvu i civilizaciji.

4.3.7 Predikativna analitika kao IT trend

Pomaže u identifikovanju budućih trendova i predviđanja uz pomoć određenih skupova podataka i statističkih alata. Predikativna analitika analizira obrazac na unapred osmišljen način. Koristi se na primer za vremensku prognozu. Međutim, njene sposobnosti i tehnike nisu ograničene samo na ovo, već se mogu koristiti u sortiranju bilo kojih podataka, a na osnovu obrasca analizira statistiku.

Neki od primera su istraživanja proizvoda (engl. Product Research) ili njegov udeo na tržištu (Market share). Na osnovu dostavljenih podataka, informacioni sistem unapred meri i

daje pun izveštaj ako bilo koji tržišni udeo opada ili ako želimo lansirati bilo koji proizvod koga pripremamo za tržište. Prikupljanjem podatke iz različitih regiona i na osnovu njihovih interesovanja, analitički sistem će nam pomoći da analiziramo svoju poslovnu odluku. U svetu rastuće konkurencije ovaj trend postaje sve zahtevniji i biće sve prisutniji narednih godina.

4.3.8 Internet stvari (IoT)

Sa rastućim tempom razvoja i prisutnosti moderne tehnologije, sve više se oslanjamo na ovu vrstu tehnologije. IoT (engl. Internet of Things) igra veliku ulogu u poslednjih nekoliko godina i verujemo da će igrati zanimljiviju ulogu u bliskoj budućnosti. Danas napredne tehnologije i arhitekture podataka imaju posebnu vrednost zahvaljujući IoT uređajima. Pretpostavlja se da bi IoT uređaje trebalo koristiti u većem obimu za skladištenje i obradu podataka u realnom vremenu. Njihova upotreba bi trebala biti radi rešavanja neobičnih problema kao što su upravljanje saobraćajem, proizvodnja, zdravstvo, kao i daljim segmentima društva i našeg života. U svakodnevnom životu, vrlo često se srećemo sa pametnim domaćinstvima (smart home, home automation), pametnom proizvodnjom (production automation), pametnim uređajima (smart devices) i sl.

4.3.9 Bezbednost podataka

Sa porastom pandemije (COVID-19), gde je svet bio primoran da se zatvori, određene aktivnosti u smislu rada, organizacije firmi, njihovog poslovanja su se počele suštinski menjati. Čak i nakon toliko meseci i godina, tokom 2020 i 2021 godine, ljudi su se fokusiraju na primer na dobijanje posla na daljinu (engl. remote job). Sve ima prednosti i mane na svoj način ali u ovom slučaju došlo je do povećanja zahteva direktnog uticaja na bezbednost informacionih sistema u smislu njihove dostupnosti ali i otvorenosti iz Internet sredine. Ova situacija donosi mnogo izazova koji uključuju sajber napade. U stvari, rad na daljinu podrazumeva implementaciju mnogo većih bezbednosnih mera i njihove odgovornosti. Pošto je zaposleni van dometa bezbednosti podataka, samim tim postaje sve veća briga za kompanije. Rad na daljinu rezultirao je sajber napadači postaju sve češći u pokušajima da napadači (engl. hackers) dođu do podataka, pronalazeći različite načine i vrste napada. Ovi napadi se sastoje u tendencijama da napadači dođu kako do podataka, tako i do privilegija upravljanja sistemima.

Uzimajući ovo u obzir, uvedeni su XDR (engl. Extended Detection and Response) tehnike, koje pomažu u otkrivanju bilo kakvog sajber napada primenom napredne bezbednosne

analitike na njihovu mrežu. Stoga je i biće jedan od glavnih trendova u 2022, kao i u narednim godinama obrada ovakve velike količine podataka, prvenstveno u analitičkom smislu radi otkrivanja i prevencije od sajber napada.

5. Rezultati naučnog istraživanja predloga arhitekture visokih performansi

Očekivani rezultati naučnog istraživanja u okviru doktorske disertacije fokusirani su na predlog uopštenog modela arhitekture sistema visokih performansi i sistema za generalnu obradu velike količine podataka na klasterima. Oni će omogućiti izgradnju klastera za obradu velike količine podataka uz obezbeđenje optimalnog i generalizovanog načina preuzimanja podataka, tj. njihove integracije, njihovog efikasnog i optimalnog smeštanja, efikasnu obradu podataka uključujući upotrebu naprednih principa i algoritama za obradu podataka, analizu i vizualizaciju. Pored toga, platforma treba da poštuje određene standarde za takve sisteme za obradu velike količine podataka u smislu performansi klaster rešenja, TPC-H testa performansi na osnovu unapred definisanih zahteva (više od 100).

Očekivani rezultati naučnog istraživanja uključuju ispunjenje sledećih ciljeva:

1. Predlog metodologije za kreiranje klaster arhitekture sistema za obradu velike količine podataka.

2. Model univerzalne arhitekture koja obuhvata od akvizicije podataka do njihovog smeštaja i prikazivanja.

3. Predlog konkretne arhitekture rešenja klaster sistema, za postizanje visokih performansi za obradu podataka.

4. Način implementacija arhitekture novog sistema, kao i njegova organizacija u smislu njegove prilagodljivosti.

5. Evaluaciju predloženog rešenja u smislu efikasnosti i performansi.

6. Predlog postupaka na osnovu koga može doći do kreiranja takvog klaster sistema, koji omogućava efikasno postizanje postavljenih ciljeva uz što je moguće nižu cenu njegove izgradnje.

Dalje, očekivani rezultati će se odnositi i na evaluaciji i analizu informacija koje se odnose na predloženo rešenje klaster arhitekture na osnovu unapred definisanih merenja, TPC- H testa performansi, i to na sledeće analize:

1. Rezultata merenja performansi klastera prilikom upotrebe podataka veličine 1TB, 3TB

2. Uticaja performansi klastera sa 3, 4, 5, … čvorova u odnosu na definisane zahteve

3. Upoređivanja rezultata na osnovu SQL upita Q1, Q2, … Q22 na osnovu TPC-H testa performansi

4. Rezultata na osnovu kojih se može doći do zaključka kako dolazi do povećanja performansi klastera prilikom povećanja broja nodova u klasteru.

U daljem postupku ove rezultate evidentiramo na poseban način kako bi smo mogli kroz iteracioni postupak u okviru daljih merenja sprovesti komparativnu analizu.

Predloženo rešenje ima između ostalog za cilj da pokaže sledeće rezultate:

1. Predložiti metodologiju za kreiranje klaster arhitekture sistema za obradu velike količine podataka.

2. Dati odgovor na pitanje kako izgraditi model univerzalne arhitekture koji obuhvata oblasti od akvizicije podataka pa sve do nivoa njihovog skladištenja, obrade i vizualne prezentacije.

3. Predlog konkretne arhitekture za klaster sistemska rešenja za postizanje visokih performansi za obradu podataka.

4. Način implementacije nove arhitekture sistema kao i njena organizacija u pogledu njene skalabilnosti.

5. Ocena predloženog rešenja u pogledu efikasnosti i performansi.

6. Predloženi postupak na osnovu kojeg je kreiran ovakav klaster sistem koji omogućava efikasno postizanje postavljenih ciljeva uz što niže troškove njegove izgradnje.

5.1 Predlog metodologije za izgradnju klastera visokih performansi

Za stvaranje finalnog rešenja, korišćene su niže navedene metodologije. Svaka od njih je ponaosob opisana u vezi sa izgradnjom klaster rešenja koje je bilo predloženo.

1. **Metodologija karakterizacije** – Ova metoda se koristi za unapred definisane parametre performansi koje treba postići. U našem slučaju, imali smo više različitih koraka koji se vezuju za promenu količine podataka, razne tipove podataka, njihovu poverljivost i bezbednost, kao i definisane zahteve performansi.

2. **Metodologija pretpostavke** - implicitno se testira usaglašenost zahteva na osnovu pretpostavki sa testiranjem performansi na osnovu postavljenih zahteva a u saglasnosti sa rezultatima koji si izmereni tokom testa performansi kao i rezultate, koje predstavlja Slika 12 i Slika 13. Merenje testa performansi predstavlja iteracioni postupak, koji obuhvata ponavljanje postupka uključujući sve pretpostavke za testiranje performansi, koje nalaže TPC-H test performansi, ali i zahteve krajnjeg korisnika. Kao primer, možemo uzeti zahtev testa performansi TPC-H gde se prilikom svakog testiranja radi tzv. „cold restart“ klastera. On predstavlja potpuni restart sistema i njegovo uključivanje sistema u rad od samog početka. Time se, kao jedan od primera faktora uticaja, eliminiše uticaj keš memorije (engl. cache memory) na performanse klastera. Sa strane korisnika, tj. subjekta koji raspisuje tender, pretpostavka je da se u dva ciklusa izmerene performanse mogu uzeti kao merodavna vrednost. Ove vrednosti procenjuje stručna komisija kod ispunjenja uslova i dostignutih performansi u okviru ocenjivanja predloženih rešenja od strane učesnika tendera.

3. **Metodologija hipoteze** - Ono što će biti eksplicitno testirano eksperimentom i vodi određivanju parametara performansi na predloženom klasteru i njegovoj arhitekturi. Na osnovu ove metodologije, napravljene su osnovne pretpostavke za izgradnju klastera (Slika 5, Slika 6, Slika 7, Slika 8), podešavanje računarske mreže, konfiguraciju celog sistema u smislu akvizicije podataka, pažljivi izbor parametara servera kao i softverskih tehnologija koje su korišćene.

4. **Komparativna metoda** - Koristiće se za poređenje rezultata dobijenih tokom različitih merenja sa precizno definisanim postupkom, kao npr. „cold restart“ prilikom ponavljanja svakog ciklusa merenja, kako bi došlo do eliminacije uticaja keš memorije na test performansi. Ovom metodom smo na osnovu fleksibilnosti klastera, njegove tzv. horizontalnog skaliranja, povećali relativno efikasno broj čvorova u okviru definisane arhitekture.

5. **Eksperimentalna metoda** – u neposrednoj je vezi sa hipotezom. Ovde se radi o pripremanju eksperimenata na osnovu definisanih hipotetičkih principa mogućeg rešenja koje uzimamo u proceduru testiranja performansi. Na primer, možemo pretpostaviti pre početka bilo kakvog testiranja performansi, da naše rešenje može zadovoljiti klaster koji ima minimalan ali istovremeno apsolutno neophodan broj čvorova klastera. Kao praktičan primer, ako uzmemo minimalni broj čvorova klastera na kome ćemo smeštati podatke i testirati performanse tog dela rešenja, minimalan i apsolutno neophodan broj čvorova je tri. Ovaj broj se vezuje za visoku dostupnost klastera i ispunjenje tzv. K-Safety = 2.

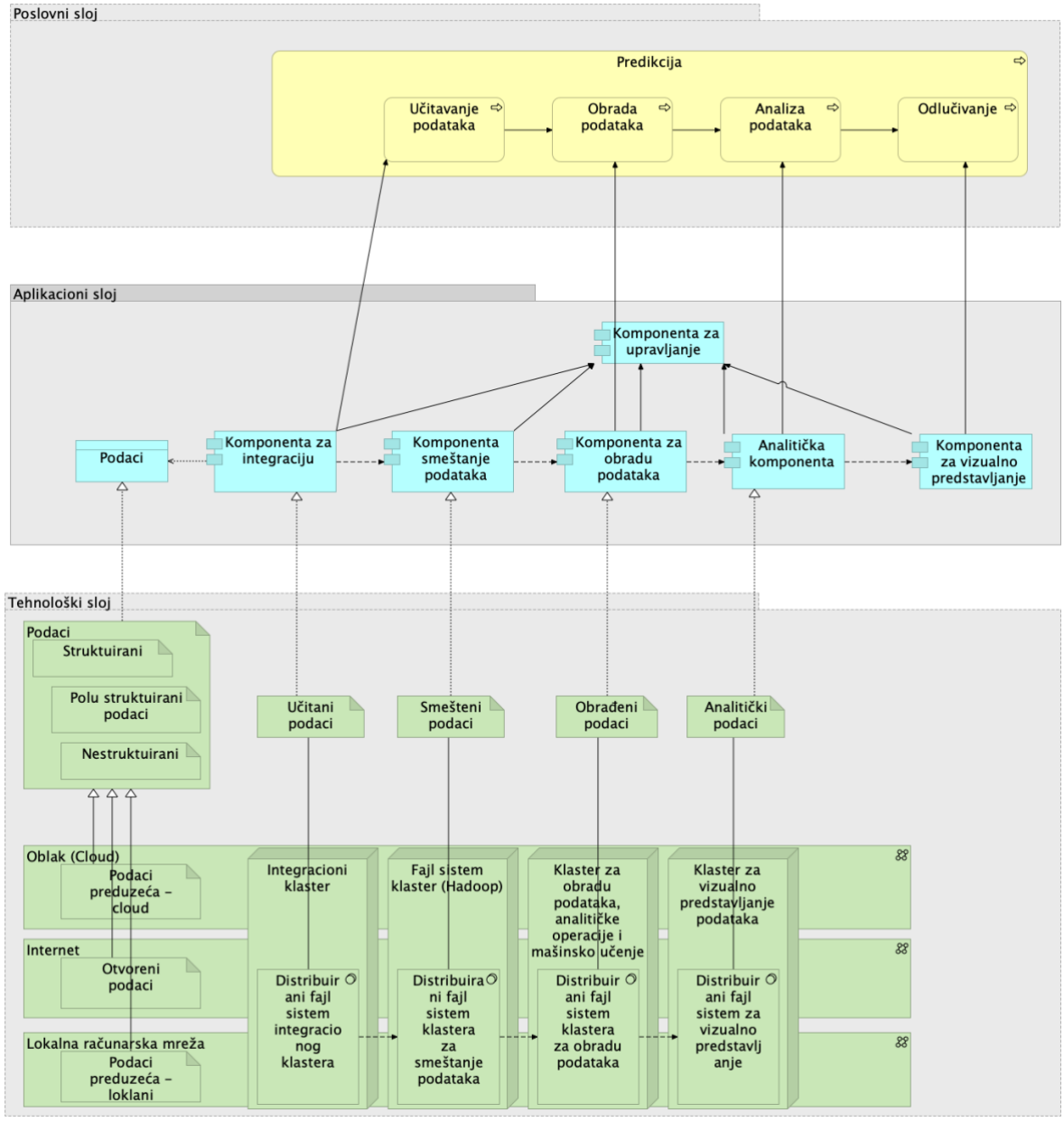
6. **Metoda evaluacije** – potvrda zaključaka predstavlja postupak koji se odnosi na upoređivanje rezultata na osnovu realizovanog testiranja performansi, tj. pojedinačnih merenja. Prilikom merenja dobijeni rezultati se napred podvrgavaju validnosti merenja. Merenje se realizuje na definisanoj platformi u okviru određenog ciklusa, prilikom koga je jedino dozvoljena operacija „cold restart“. Time je omogućeno da pod istovetnim uslovima dođemo do rezultata, koje evaluiramo, tj. uzajamno upoređujemo. Uzajamno upoređivanje rezultata, vodi do izbora traženog, željenog rešenja, koje ulazi u finalni izbor za predstavljanje rezultata korisniku u okviru tendera. Tender predstavlja formalizovani postupak za izbor subjekta, izvođača projekta sa strane korisnika koji je raspisao tender.

7. **Metoda sinteze** - Individualni rezultati, tj. pojedinačno dobijeni rezultati merenja će biti kombinovani prilikom kreiranje finalnog rešenja. Delimičnim rezultatima i merenjima, koje ne sme imati uticaja na celokupno merenje, možemo doći do zaključka kako se naš klaster koga smo zamislili u početnoj fazi može poboljšati u smislu performansi.

5.2 Model univerzalne arhitekture

Model univerzalne arhitekture namenjen je za obradu velike količine podataka na Big Data klasterima. Ima vrlo visoke performanse i koristi se za specijalizovanu ili generalnu obradu podataka, koji imaju tri sloja: poslovni sloj, aplikacioni sloj, tehnološki sloj. Model univerzalne arhitekture omogućava izgradnju specijaliziranog klastera bilo u kojim oblastima modernog društva, kao što je na primer zdravstvo, proizvodnja, velikoprodaja, maloprodaja, transport, komunikacije, finansije, transport, bezbednost i bankarstvo i sl.

Na sledećoj slici (Slika 4) je predstavljen model univerzalne arhitekture Big Data klastera. U današnje vreme, i dalje postoje tradicionalni sistemi kao što su sistemi poslovne inteligencije. Međutim, u novije doba sve više preduzeća ima zahteve za smeštanje i upotrebu sve veće količine podataka. Pored toga, preduzeća i poslovna inteligencija organizacija, ima potrebe za bržom obradom podataka, kao i uvođenja prediktivnih modela na osnovu obrade podataka, donošenja odluka i upotrebom veštačke inteligencije. To sve prate i zahtevi koji se vezuju za digitalizaciju organizacije, preduzeća kao i automatizaciju u punoj meri prilikom izvršavanje poslovnih procesa.



Slika 4 - Model generalizovane arhitekture Big Data klastera

Tehnološki sloj predstavlja hardver, sistemski softver, specijalizovani softver, poslovni procesi, dok su na najnižem nivou podaci. Podaci su obično smešteni lokalno. Mogu biti preuzeti iz opštih internih izvora u vidi otvorenih (Open Data) ili drugih podataka ili iz sredine oblaka (Cloud Data). Između njih se koriste odnosi za modeliranje npr. informacije, roba, novac i sl. U našem slučaju, kod modela univerzalne arhitekture imamo odnose na nivou informacija, tj. podataka. Oni se na osnovu tipova učitavaju, smeštaju, obrađuju, analiziraju i prikazuju.

Aplikacioni sloj predstavlja enkapsulaciju funkcionalnosti aplikacije usklađene sa strukturom implementacije. Struktura implementacije je modularna i prilagodljiva na osnovu potrebe preduzeća i njegovog poslovnog menadžmenta. Aplikacioni sloj se sastoji od aplikacionih komponenti. Kao takve se mogu nezavisno rasporediti ili zameniti. Aplikaciona komponenta može biti dodeljena jednoj ili više funkcija. Ima jedan ili više interfejsa koje omogućavaju njenu upotrebu i funkcije koje ona obezbeđuje.

Poslovni sloj sadrži više raznorodnih procesa kojima se postižu određeni rezultati kao što je npr. definisani skup proizvoda ili poslovnih usluga. Poslovni proces opisuje interno ponašanje koje obavlja poslovna uloga u okviru poslovnog sloja. Pored poslovnih procesa postoje i složeni (kompleksni) poslovni procesi, koji predstavljaju skup detaljnijih poslovnih procesa.

Kod svake platforme za obradu velike količine podataka, Big Data platforme, možemo razlikovati sledeće generalizovane komponente za:

1. Prikupljanje/Unošenje ili akviziciju podataka (Data Sources).
2. Prikupljanje poruka (Message Ingestion).
3. Smeštanje podataka (Data Storage).
4. Upravljanje klasterom (Cluster Management).
5. Serijska obradu podataka (Batch Processing).
6. Obradu podataka u realnom vremenu (Stream Processing).
7. Analitiku i smeštanje analitičkih podataka (Analytical Data Storage).

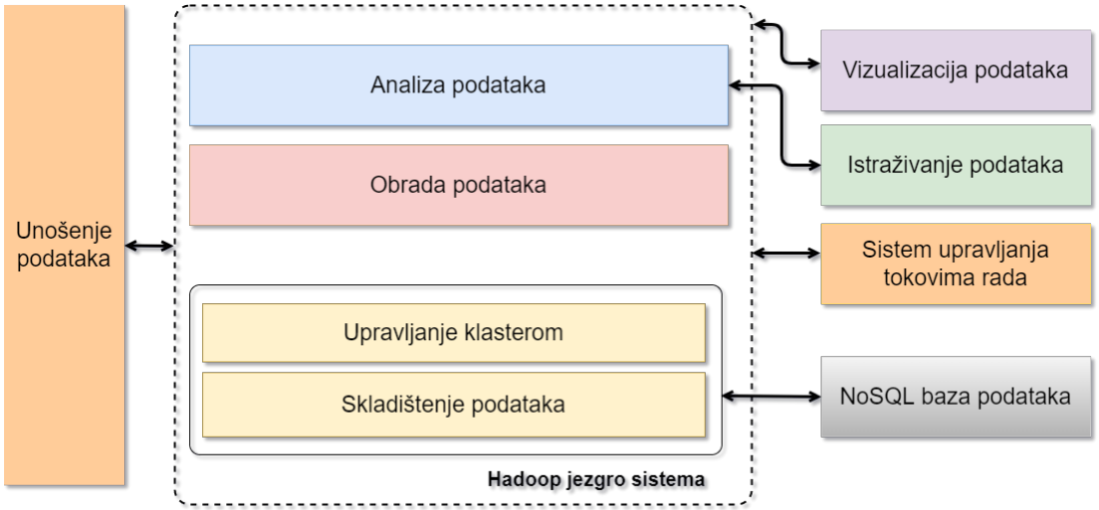
8. Analizu i vizualno predstavljanje podataka (Analytics and Reporting).

9. Upravljanje tokovima podataka klastera (Cluster Data Flow Management).

10. Istraživanje podataka (Data Exploration, Data Harvesting).

11. Vizualizaciju podataka (Data Visualization).

Na slici (Slika 5) prikazan je generalni model koji sadrži prethodno navedene komponente. Ono što je značajno na njemu je uzajamna relacija ovih komponenti prilikom izgradnje celokupnog rešenja za obradu velike količine podataka.



Slika 5 - Generalni model Big Data klastera

Ukoliko pogledamo neke od komponenti koje su vrlo često bile u upotrebi prilikom realizacije ovakvih ili sličnih rešenja, možemo videti, da većina komponenti (Slika 5 i Slika 6) je postavljena na softverskim komponentama ASF - Apache Software Foundation (https://www.apache.org/foundation) ali i komercijalnim komponentama kao što su npr. Vertica, Qlik, Tableau, Looker i sl. Dalje, ove komponente su vrlo često dostupne i poznatim sredinama za računarstvo u oblaku, kao što je Google, Oracle, Amazon, Microsoft Azure i sl.

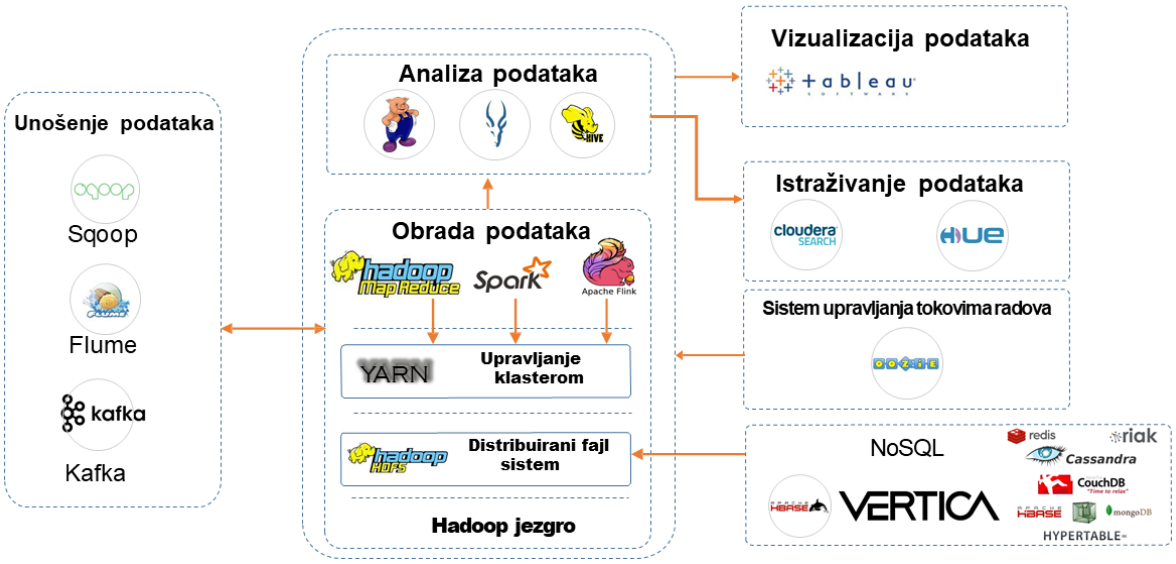
Slika 6 prikazuje dijagram realnih softverskih komponenti platforme za obradu velike količine podataka. Celokupno rešenje je osmišljeno kao hibridno. Delovi ovog rešenja su implementirani u sredini za računarstvo u oblaku (Microsoft, Amazon, Google) a drugi deo komponenti je implementiran u lokalnoj sredini (Open Stack, Hadoop). Akvizicija podataka je organizovana tako što se ulazni stream podaci obrađuju komponentom za stream podatke Apache Kafka. U sledećom koraku vrši se dalja obrada ETL procedurom ali i tzv. „in memory“

transformacijom i obradom podataka na nivou Apache Spark (https://spark.apache.org) komponente. Nakon toga, obrađeni podaci se smeštaju u odgovarajući format (npr. Parquet https://parquet.apache.org, Avro https://avro.apache.org, ORC https://orc.apache.org) radi lakše obrade kolonski uređenih podataka. Kao dalji primer, podaci mogu biti smešteni u specijalizovani deo za smeštanje velike količine podataka (npr. Cloud Storage) koji je visoko skalabilan i predstavlja pouzdanu infrastruktura za smeštanje velike količine podataka sa malim kašnjenjem. Sledeći nivo predstavlja lokalnu infrastrukturu u kojoj se vrši obrada pripremljenih podataka, kako bi se postigla njihova obrada u realnom vremenu. Istovremeno ovakva platforma omogućava upotrebu pogodnih programskih jezika, kao što su npr. Java, Scala, Python, R, C++, SQL. U završnoj fazi, ovako pripremljene i obrađene podatke je potrebno vizualno predstaviti radi lakše percepcije na strani krajnjeg korisnika. Na osnovu tako predstavljenih podataka, moguće je doneti različite odluke, koje omogućavaju dalje automatizovane korake, koji poboljšavaju efikasnost sistema vezano za npr. poslovnu inteligenciju, bezbednost u informacionim tehnologijama, telekomunikacije, finansijske operacije i sl.

Slika 6 - Integrisani sistem - Otvorena arhitektura

Na slici (Slika 7) predstavljeno je rešenje koje je u celosti realizovano kao „on site“, tj. rešenje u lokalnom okruženju korisnika. Akvizicija podataka je prilagođena na osnovu tipa podataka (strukturni, nestrukturni ili polu strukturni). Nakon njihove obrade, podaci se smeštaju u fajl sistem koji je prilagođen za smeštanje velike količine podataka Hadoop. Nakon toga, dolazi do serijske obrade ovih podataka uz pomoć komponente Hadoop Map Reduce, ili njihove obrade u memoriji kako bi se postigle visoke performanse. Deo podataka koji je

pogodan za smeštanje u kolonskim bazama podataka se tamo smešta i obrađuje, kako bi došlo do njihove pripreme za vizualizaciju nakon npr. smeštanja u fajl sistem Hadoop koji je pogodan za to. Nakon analize i takve obrade podataka, vrši se vizualizacija podataka na nivou specijalizovanog alata.



Slika 7 – Primer Big Data komponente -„ekosistem“

Upoređivanjem platformi koje prikazuje Slika 6 i Slika 7 možemo doći do sledećeg zaključka:

1. Princip akvizicije podataka i njihov tretman u smislu obrade je identičan. Podaci su smešteni na različitim lokacijama.

2. Format podataka je identičan, uvek prilagođen potrebama radi dalje obrade i na kraju smešten u centralu komponentu za smeštanje velike količine podataka.

3. Komponente koje se koriste za serijsku ili memorijsku obradu podataka su identične.

4. Smeštanje više uvedenih komponenti za serijsku ili memorijsku obradu podataka kod platforme za računarstvo u oblaku (engl. Cloud) kao i kod lokalne implementirane platforme je uvek „blizu“ velike količine podataka. Ovde je potrebno naglasiti da je su zahtevi za mrežnom komunikacijom izuzetno visoki (1-10 Gb/s), te stoga ove komponente moraju biti implementirane u okviru jedne lokalne mreže (Cloud ili lokalna).

5. Komponente za vizualizaciju podataka zahtevaju manju briznu prenosa podataka, te stoga celokupno hibridno rešenje omogućava relativno lakšu grafičku prezentaciju već obrađenih podataka na osnovu njihove pripremljenosti.

Na osnovu uporedne analize iz ovog poglavlja, gde smo upoređivali platforme sa različitih aspekata, dolazimo do tzv. generalizovanog modela na osnovu koga bi smo mogli predložiti konkretno rešenje platforme za obradu velike količine podataka. Generalno, možemo razlikovati logičke celine za:

1. Prikupljanje tj. akviziciju podataka
2. Transformaciju podataka
3. Smeštanje podataka
4. Obradu podataka sa komponentom za poboljšanje performansi prilikom obrade
5. Administraciju platforme
6. Monitoring
7. Vizualizaciju podataka
8. Integraciju sa ostalim sistemima

5.3 Pregled konkretne implementacije arhitekture

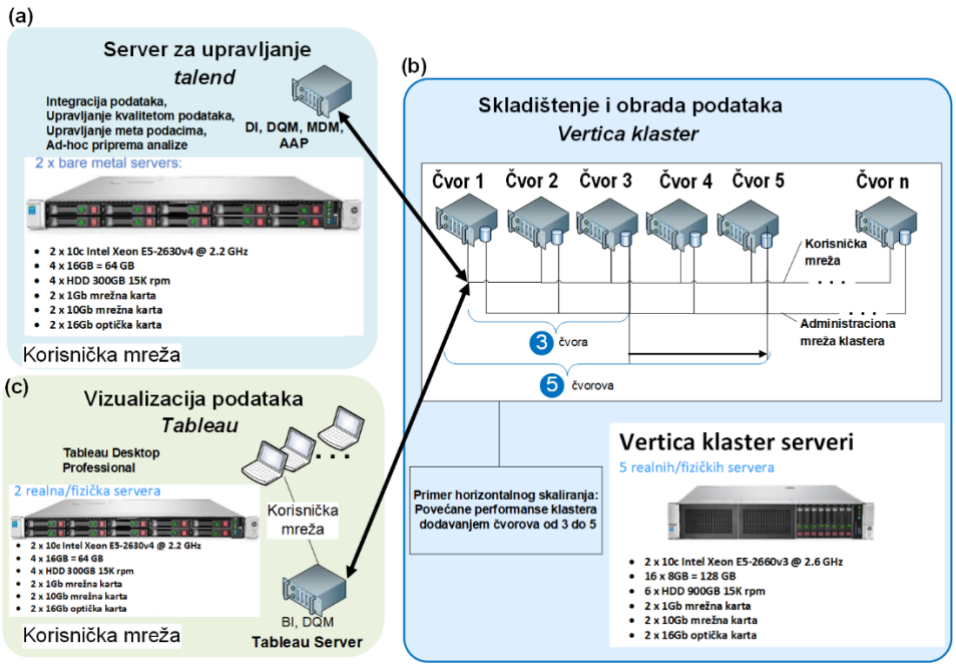
U ovom poglavlju predstavljena je platforma za obradu velike količine podataka, koja predstavlja distribuiranu, visoko skalabilnu platformu sa performansama visokog nivoa na tzv. uobičajenom hardveru (engl. Commodity Hardware). Ovde se radi o uobičajenim, ekonomski dostupnim serverima koji su povezani u mrežu preko računarske mrežne kartice, obično brzine 1Gb/s ili 10Gb/s. Pomoću ovakvih komponenti ćemo graditi klaster rešenje, BDA (engl. Big Data Analytics) platformu. Dodavanjem pojedinačnih, tradicionalnih, običnih servera, postižemo promenu, tj. povećanje performansi sistema. Jedno od takvih rešenja na više uvedenim principima je bilo implementirano kao Big Data analitički klaster za potrebe Instituta za zdravstvene informacione sisteme i statistiku Češke Republike (IHIS).

Na slici (Slika 8) je prikazana BDA platforma [67] koja objedinjuje tri ključne komponente:

1. **Talend** (verzija 6.4) – komponenta za akviziciju i transformaciju podataka.

2. **Vertica** (verzija 9.0.1) – komponenta za smeštaj svih tipova podataka i njihovu obradu u tzv. NoSQL bazi podataka. Vertica [68-70] NoSQL baza podataka, izgrađena na pet čvorova (engl. nodes), obezbeđuje skladištenje i obradu podataka.

3. **Tableau Desktop i Server** (verzija 10.5) – specijalizovana komponenta za vizualizaciju podataka.



Slika 8 - Arhitektura i infrastruktura komponenti Big Data klastera

5.3.1 Integracioni deo podataka klasterskog rešenja

Integracioni deo podataka DI (engl. Data Integration Layer) predstavlja posebno ulaznu komponentu koja omogućava integraciju Big Data analitičkog klastera sa okolnim svetom. Ova komponenta sadrži funkcije koju omogućavaju manipulaciju, transformaciju kao i jednostavnu obradu podataka uključujući i podešavanja hijerarhije, ali i paralelnu obradu tokom prenošenja podataka do centralnog dela BDA klastera određenog za smeštanje podataka.

5.3.2 Skladištenje podataka na klasteru za obradu velikih podataka

Skladištenje podataka (engl. Data Storage) predstavlja sistemski modul koji sadrži horizontalno skalabilan klaster baziran na fizičkoj arhitekturi izgrađenoj na NoSQL Vertica bazama podataka. Skladištenje podataka se odvija na hardveru sa mogućnostima distribuiranog skladištenja, što omogućava masovnu paralelnu obradu (engl. Massive Parallel Processing) preko celokupnog zbira podataka. Sistem za skladištenje podataka čuva podatke u formatu kolone u dva kontejnera, Write Optimized Store (WOS) i Read Optimized Store (ROS), radi postizanja najboljih performanse. Svaki klaster je kolekcija čvorova sa softverom za izgradnju NoSQL baze podataka Vertica. Svaki čvor je konfigurisan za pokretanje Vertica NoSQL baze podataka kao član klastera baze podataka, koji podržava redundantnost, visoku dostupnost i horizontalnost skalabilnost, obezbeđujući efikasne i kontinuirane performanse. Ova infrastruktura omogućava oporavak od bilo kog potencijalnog kvara čvora dozvoljavajući drugim čvorovima da preuzmu kontrolu. Za predstavljeno rešenje (Slika 8), postavili smo toleranciju greške K-Safety = 2 [69]. Komponente integracionog nivoa određuju koliko kopija sačuvanih podataka Vertica NoSQL baza podataka treba da kreira u bilo kom trenutku.

5.3.3 Modul upravljanja kvalitetom podataka na nivou klastera

Modul upravljanja kvalitetom podataka DQM (engl. Data Quality Management) podržava kontrolu kvaliteta podataka uključujući trendove i strukture podataka. Ovaj modul generiše složene modele za krajnje korisnike koji podržavaju analizu podataka za otkrivanje i ispravljanje grešaka, kao i sofisticiranu vizuelizaciju i izveštavanje potrebne za zadatke kontrole kvaliteta. On kreira, sortira, grupiše i traži pravila validacije u strukturiranom obliku. Pravila validacije mogu da se izvrše preko korisnički definisanog skupa podataka i da se njima upravlja centralno.

5.3.4 Modul za upravljanje meta podacima na nivou klastera

Modul za upravljanje meta podacima MDM (engl. Meta Data Management) podržava upravljanje korisničkim, tehničkim i operativnim meta podacima. MDM centralno obrađuje meta podatke iz svake komponente klaster sistema, smeštene zajedno u skladištu podataka.

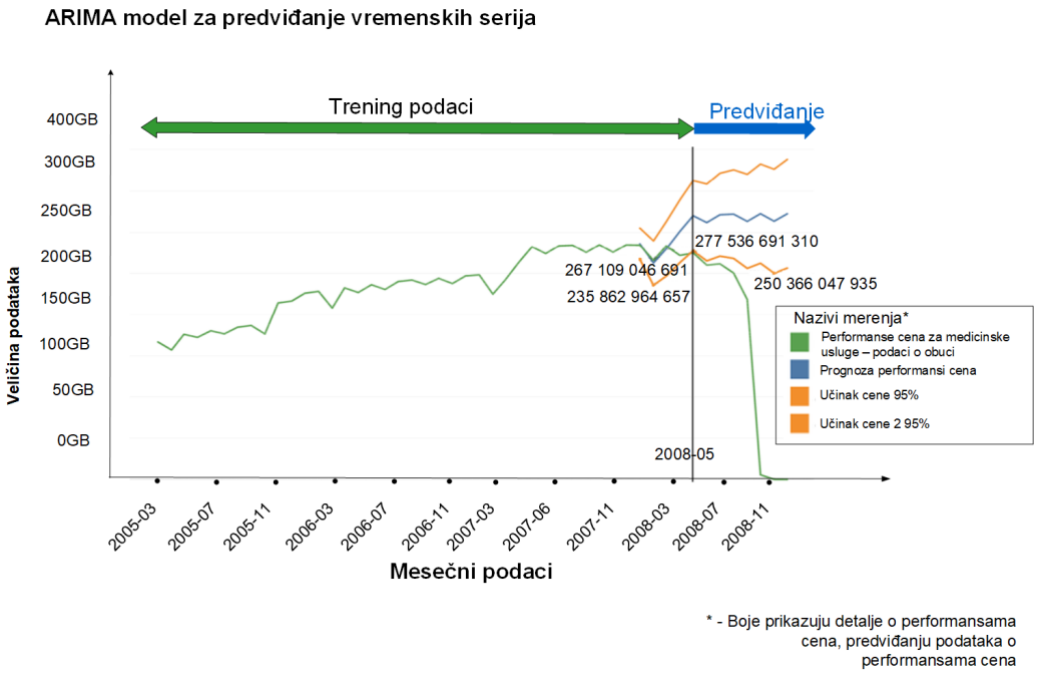
MDM može da uporedi različite verzije meta podataka i izlaznih podataka na osnovu unetih komandi, uključujući vizuelizaciju namenjenu izveštavanju o podacima. MDM je u

stanju da kreira dinamičke, aktivne grafikone i tabele, omogućavajući višedimenzionalne i interaktivne komande. MDM koristi tzv. „sandboxing“ za testiranje privremenih ulaza i izlaza i može da generiše izlaze u HTML, PDF i PPT formatima. MDM komponenta koristi operacije onlajn analitičke obrade (OLAP) preko višedimenzionalnog modela podataka. Pored toga, sadrži rečnik pojmova i veze za koncepte kako bi se omogućila analiza uticaja i porekla.

5.3.5 Modul za pripreme ad-hoc analize na klasteru za obradu podataka

Za proces pripreme ad-hoc analize AAP (engl. Ad-hoc Analysis Procedure) programirali smo dve različite verzije u alatu za integraciju po nazivom (engl. Talend Open Studio). U prvoj verziji, MHDA (engl. Massive High Data Analytics) koriste se komponente za transformacije i učitavanja ekstrakcije i učitavanja alata za integraciju (ETL). Ove komponente čitaju podatke iz struktura skladišta podataka (dimenzije i tabele činjenica) u memoriju klastera. Zatim, komponente filtriranja i agregacije obrađuju podatke u izlaznu tabelu. Druga verzija koristi ELT (engl. Extract Load and Transform) komponente alata za integraciju. Kako ETL tako i ELT komponente su u stanju da generišu nemodifikovani SQL jednostavan za korišćenje jezika za manipulaciju podacima DML (engl. Data Manipulation Language) u pozadini. AAP modul ubrzava vreme obrade podataka bez potrebe da učitava velike količine meta podataka u programsku memoriju izgrađenog klastera.

Sledeća Slika 9 predstavlja primer prediktivnog modela za predviđanje vremenskih serija na osnovu dostavljenih test podataka iz IHIS. Primer pokazuje deo klastera za obradu velike količine podataka i vizuelizaciju (Slika 3 i Slika 9), koji prikazan kao snimak ekrana napravljen u softverskom proizvodu Tableau Desktop (verzija 10.5). Slika prikazuje predviđanje na istorijskom skupu za testiranje podataka koji je dostavio IHIS, gde smo testirali model ARIMA [71] za pristup vremenskim serijama u bazi podataka. Ovaj model se može kreirati ili direktno u bazi podataka NoSQL Vertica, koja podržava prediktivno modeliranje, ili u posebnom statističkom alatu kao što je Tableau, koji će uzeti podatke iz baze podataka i vratiti kreirani model napisan u PMML (engl. Predictive Model Markup Language) ili drugi format koji baza podataka podržava.



Slika 9 - Primer prediktivnog modela - ARIMA model

Uključivanje algoritama mašinskog učenja i analize podataka u bazu podataka često dovodi do povećanih zahteva za obradu na BDA platformama. Što se tiče vizuelizacije podataka, Tableau Server (verzija 10.5, neograničene licence) obezbeđuje vizuelizaciju preko grafičkog korisničkog interfejsa GUI (engl. Graphical User Interface) i veb pretraživača za standardne krajnje korisnike. Tableau Desktop, međutim, pruža dodatnu funkcionalnost namenjenu analitičarima podataka i korisničkim profilima naučnika.

5.3.6 Vizualizacija podataka na klasteru za obradu velike količine podataka

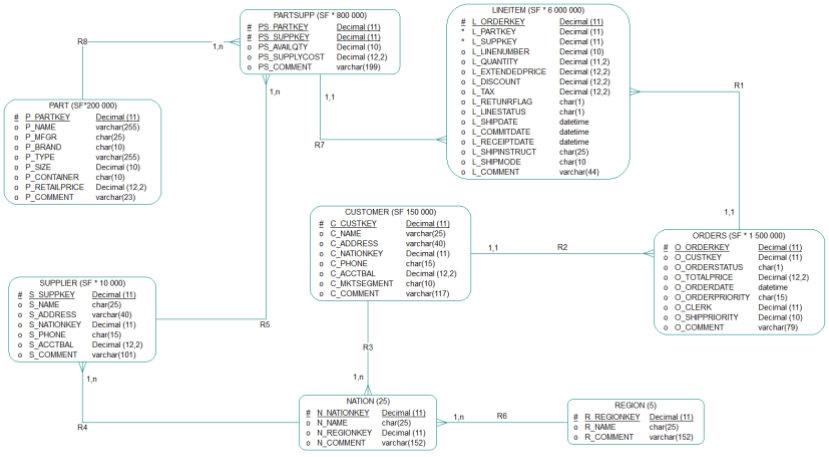
Modul vizuelizacije podataka DV (engl. Data Visualization) sadrži alate za opisivanje perspektiva podataka i otkrivanje znanja iz podataka. Komponente DV vizuelno predstavljaju podatke i meta podatke i daju tumačenja za moguće uvide. Pored toga, ugradili smo DV komponente u Tableau platformu za vizualizaciju kako bismo obezbedili vizuelizaciju podataka i meta podataka pomoću grafikona i slika. Tableau je popularan interaktivni alat za analizu podataka kao i za njihovu vizualizaciju, koji može pomoći da se „sirovi“ tj. neobrađeni podaci pojednostave u lako razumljive kontrolne tabele i radne listove. Na primer, Slika 10 predstavlja primer dijagnoze iz stvarnog života pacijenata mlađih od 10 godina kao regionalna vizuelizacija podataka u Češkoj Republici pomoću Tableau Desktop (verzija 10.5). Pored integracije podataka, funkcionalnost geo-mapiranja, obezbeđuje mapiranje pandemije, epidemije skoro u realnom vremenu, kao i njeno praćenje, širenje i vizuelizaciju podataka o

riziku koji iz toga proističe. Slika 10 prikazuje deo vizuelizacije podataka iz jednog od IHIS slučajeva studije preklapanjem sa geografskom kartom Češke Republike.

Slika 10 - Primer vizuelizacije dijagnoze iz stvarnog života mlađih od 10 godina

5.4 TPC-H konfiguracija klastera za obradu velike količine podataka

TPC-H test performansi zahteva da se podaci generišu u okviru osam tabela korišćenjem faktora za skaliranje SF (engl. Scale Factor), koji određuje približnu količinu podataka u gigabajtima (Slika 11). Koristili smo TPC-H test performansi, koji meri propusnosti/vreme odgovora niza od 22 upita. Vertica podržava ANSI SQL-99 standard i svi upiti se primenjuju bez promena sintakse. Skupovi testnih podataka kreirani su programom TPC-H pod nazivom DBGEN [20] . U našim testovima smo otkrili da su upiti Q9 i Q21 složeniji u poređenju sa uobičajeno očekivanim upitima.



Slika 11 – Komponente TPC-H koje se sastoje od osam tabela

5.5 Način implementacije

U okviru pripreme rešenja za implementaciju u okruženju firme, koja radi generalnu isporuku celokupnog rešenja, možemo prepoznati aktivnosti koje se vezuju za izbor tradicionalnog hardvera (servera koji predstavlja čvor klastera, a koji će biti uključeni u klaster okruženje), njihovu instalaciju i povezivanje uz pomoć računarske mreže u unifikovano rešenje. Dalje, podešavanje bezbednosti sistema i instalacije ključnih softverskih komponenti koje se tiču pojedinačnih funkcija. Kao primer toga može biti izbor komponente koja se tiče integracije ili dela klastera koji je odgovoran za smeštanje podataka, transformaciju, vizualizaciju i sl. Ovde je važno upozoriti na to da svaki parametar mora biti ponaosob pažljivo izabran i podešen, kao npr. brzina mrežnih kartica koje se koriste za administrativnu komunikaciju između čvorova klastera. Dalje, brzina hard diskova koji se koriste za smeštanje i obradu podataka, količina RAM memorije, izbor CPU i sl.

Implementacije kompletnog rešenja se sastoji od nekoliko etapa, koje možemo označiti kao:

1. Priprema „on premise“ arhitekture.
2. Priprema okruženja za računarstvo u oblaku.
3. Izvršenje testa performansi.
4. Implementaciju u okruženju korisnika.

Na osnovu principa pružanja usluga konzumiranja resursa kao službe („resource as a service“), potrebno je sagledati mogućnosti implementacije sistema u ovakvom okruženju. Razlog je taj, što firma koja pruža usluge može na vrlo efikasan i fleksibilan način da izgradi klaster u okruženju za računarstvo u oblaku uz poštovanje bezbednosnih zahteva krajnjeg korisnika. Resurse, koje je moguće iskoristiti za obradu velike količine podataka, nude vrlo široku upotrebu, ali istovremeno mogućnost njihovog skaliranja. U budućnosti, ukoliko se pokaže potreba za povećanjem kapaciteta sistema, povećanje je moguće postići na vrlo efikasan način sa relativno malim učešćem ljudskih resursa. Na taj način postižemo fleksibilnost celokupnog rešenja. Ovo može ići do nivoa da krajnji korisnik, ukoliko ima dovoljno edukovan svoj ljudski kapital i odgovarajuće vreme, smanji svoju zavisnost od firme koja je isporučila rešenje. Sa strane ekonomskog aspekta, cena produkcije može na taj način biti postignuta na

maksimalni nivo finansijske isplativosti ovakvog rešenja. Pošto je već bilo navedeno, da celokupno rešenje može biti implementirano i u okruženju za računarstvo u oblaku, potrebno je voditi računa o bezbednosti sistema kao takvog. U oblasti bezbednosti sistema, kod implementacija rešenja u sredini za računarstvo u oblaku za zemlje članice EU, potrebno je naglasiti da je potrebno poštovati smernice ENISA (European Network and Information Security Agency) (http://www.enisa.europa.eu).

Etapa testa performansi se odnosi na infrastrukturu nezavisno da li je on-premise ili u okruženju za računarstvo u oblaku i izvršavanje specijalizovanih upita (22 upita Q1-Q22, Slika 12) nad već pripremljenim podacima smeštenim u bazu podataka klaster rešenja. Ovde je važno upozoriti da je potrebno ispuniti sve uslove za ispravno merenje performansi, tj. test performansi koje se odnosi na celokupnu infrastrukturu pred pokretanje 22 SQL upita, gde se merenje radi po tzv. principu „cold restart“. To znači da prilikom pokretanja merenja između prvog i drugog ciklusa je potrebno napraviti restart celokupnog klastera. Time se isključuje mogućnost uticaja keš memorije na sve performanse celokupnog rešenja izgrađene arhitekture, koja je subjekt testa performansi.

Implementacija u okruženju korisnika se odnosi na prenošenje iskustava tokom projektovanja klastera, testa performansi, merenja njegovih performansi i implementacije u sredini korisnika, ukoliko je to moguće na skroz identičan način. Finalnoj implementaciji u sredini korisnika, prethodila je implementacija klaster sistema i merenja njegovih performansi po unapred definisanim kriterijumima kako u „on-premise“ ali i u sredini za računarstvo u oblaku.

6. Evaluacija performansi arhitekture predloženog rešenja

U narednim poglavljima, predstavljeni su konkretni rezultati merenja, komparacija rezultata kao i generalizacija zaključaka performansi klaster sistema.

Kao granični parametar za određivanje broja čvorova klastera bilo je određeno ukupno vreme učitavanja podataka količine 3TB, koje je bilo definisano na maksimalnih 6h kao obavezan kriterijum (Tabela 3). Količina podataka je bila definisana iz razloga dovoljne fleksibilnosti prilikom razvoja cele platforme u smislu definisanja njegove minimalne veličine sa strane hardvera arhitekture i njenih komponenti. Kako je bilo potrebno obezbediti veći broj ponavljanja testiranja i izvršiti ponavljanje merenja, ova veličina podataka je bila uzeta kao referentna vrednost za ispunjenost zahteva. U zahtevu koji se vezuje za ukupno vreme učitavanja podataka u okviru 6h uzeti su u obzir sledeći parametri sistema:

1. Najkraće moguće vreme za održavanje sistema u vremenskom periodu 24.00 – 06.00,
2. Sve operacije koje su vezane za rad sistema kao i njegovo održavanje moraju biti obezbeđene u okviru više definisanog vremenskog okvira.
3. Sve integracione procedure sa okolnim sistemima, koje su vezane za serijski prenos podataka, moraju biti obavljene u okviru ovog vremenskog okvira isključujući on-line interfejse.
4. Prilikom budućeg povećanja podataka, ovaj vremenski period mora ostati nepromenjen.
5. Dostupnost u režimu 24/7.

6.1 Rezultati merenja

Učinak koji je postignut korišćenjem podataka, koje je unapred definisao standard testa performansi TPC-H [20] (www.tpc.org), pokazuje da je razvijeni i predloženi sistem (kao PoC) nadmašio druge konkurente sa sličnim karakteristikama proizvoda [58, 72, 73]. Koristeći predstavljenu arhitekturu platforme (Slika 8), korisnik, tj. IHIS je takođe testirao performanse (Tabela 3, Tabela 4, Slika 12 i Slika 13) celokupnog rešenja. Razvijeni sistem je instaliran

unutar granica Češke Republike u centralizovanom lokalnom režimu korišćenjem kanala komunikacije podataka koji su fizički odvojeni od postojeće Internet infrastrukture.

Tabela 3 predstavlja TPC-H parametre definisane zahtevom IHIS za inicijalni import podataka (1TB i 3TB), kao i granične vrednosti 1 i 2 ciklusa i njihovih pojedinačnih merenja (* - Initial Import; ** - TPC-H Benchmark).

Parameter	Limit [hours]	Achieved results [hours]
Initial import TPC-H 1 TB	24	2.94*
Initial import TPC-H 3 TB	96	5.99*
Power test TPC-H 1TB – 1st run	1.5	1.4**
Power test TPC-H 1TB – 2nd run	1.5	1.36**
Power test TPC-H 3TB – 1st run	5	4.2**
Power test TPC-H 3TB – 2nd run	5	4.17**

Tabela 3 - TPC-H parametri definisani IHIS za inicijalni unos podataka

Veličina podataka	1TB data			3 TB data		
Tabela	Br. redova	Trajanje u [s]	Trajanje u [h]	Br. redova	Trajanje u [s]	Trajanje u [h]
Customer	150,000,000	1,185.00	0.33	450,000,000	4,100.00	1.14
Nation	25	0.10	0.00	25	0.20	0.00
Orders	1,500,000,000	2,533.00	0.70	4,500,000,000	5,423.00	1.51
Part	200,000,000	272.00	0.08	600,000,000	865.00	0.24
Part Supp	800,000,000	1,342.00	0.37	2,400,000,000	4,240.00	1.18

Veličina podataka	1TB data			3 TB data		
Region	5	0.07	0.00	5	0.07	0.00
Supplier	10,000,000	105.00	0.03	30,000,000	266.00	0.07
Line item	5,999,989,709	10,594.00	2.94	18,000,048,306	21,548.00	5.99
Ukupno trajanje učitavanja podataka u [h]	16,031.17(s) **2.94(h)**			36,442.27(s) 5.99 **(h)**		

Tabela 4 - Izmereni rezultati za podatke generisane SW DBGEN

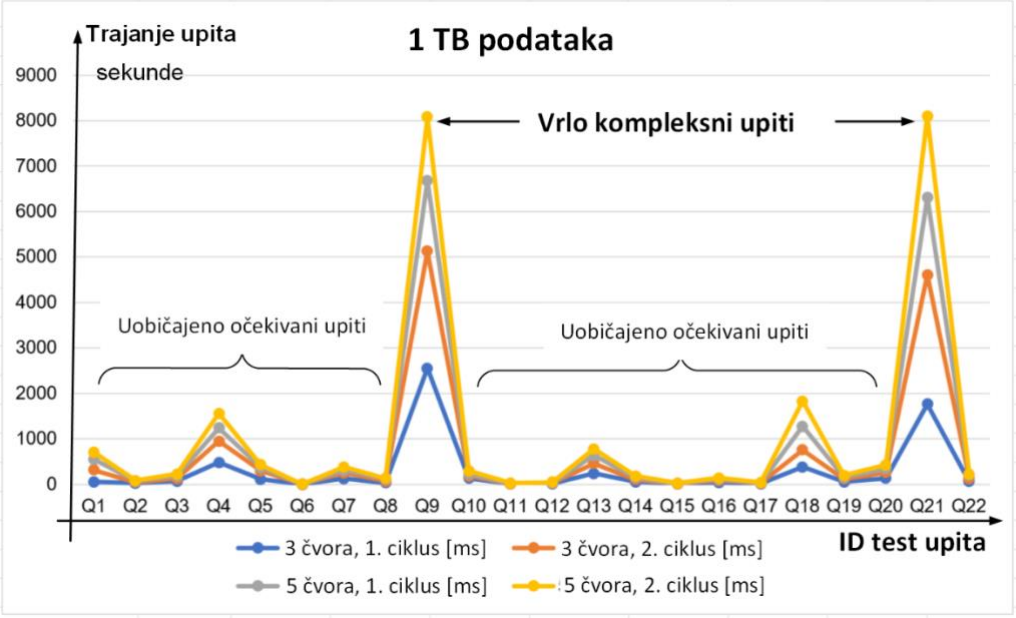
Tabela 5 prikazuje za svaki TPC-H benchmark upit izmeren pojedinačni izlaz koji je prikazan za Q1 - Pricing Summary Report Query, Q2 - Minimum Cost Supplier Query, Q3 - Shipping Priority Query, Q4 - Order Priority Checking Query, Q5 - Local Supplier Volume Query, Q6 - Forecasting Revenue Change Query, Q7 - Volume Shipping Query, Q8 - National Market Share Query, Q9 - Product Type Profit Measure Query, Q10 - Returned Item Reporting Query, Q11 - Important Stock Identification Query, Q12 - Shipping Modes and Order Priority Query, Q13 - Customer Distribution Query, Q14 - Promotion Effect Query, Q15 - Top Supplier Query, Q16 - Parts/Supplier Relationship Query, Q17 - Small-Quantity-Order Revenue Query, Q18 - Large Volume Customer Query, Q19 - Discounted Revenue Query, Q20 - Potential Part Promotion Query, Q21 - Suppliers Who Kept Orders Waiting Query, Q22 - Global Sales Opportunity Query.

Veličina podataka	1 TB data				3 TB data			
Upit br..	3 čvora, 1. krug	3 čvora, 2. krug	5 čvorova, 1. krug	5 čvorova, 2. krug	3 čvora, 1. krug	3 čvora, 2. krug	5 čvorova, 1. krug	5 čvorova, 2. krug
Q1	51	267	232	161	427	383	441	457
Q2	22	23	19	15	52	42	36	40
Q3	65	64	55	40	128	109	121	125

Veličina podataka	1 TB data				3 TB data			
Upit br..	3 čvora, 1. krug	3 čvora, 2. krug	5 čvorova, 1. krug	5 čvorova, 2. krug	3 čvora, 1. krug	3 čvora, 2. krug	5 čvorova, 1. krug	5 čvorova, 2. krug
Q4	480	470	287	320	918	897	914	900
Q5	114	177	71	70	484	462	465	454
Q6	0.7	0.6	0.8	0.5	1.2	1	1.4	1
Q7	129	119	65	65	144	129	140	140
Q8	34	37	40	22	375	361	270	263
Q9	2551	2576	1555	1397	16015	15173	3791	3824
Q10	130	52	44	72	65	58	64	64
Q11	7	6	5	3.8	13	10	10	11
Q12	13	13	8	11	24	20	23	23
Q13	237	221	180	136	296	251	325	301
Q14	55	49	41	36	111	102	105	105
Q15	7	7	4	4	9	7	9	10
Q16	42	42	29	30	87	78	86	88
Q17	12	11	8	6	23	19	23	22
Q18	380	376	517	554	741	721	743	746
Q19	58	58	41	41	112	104	111	110

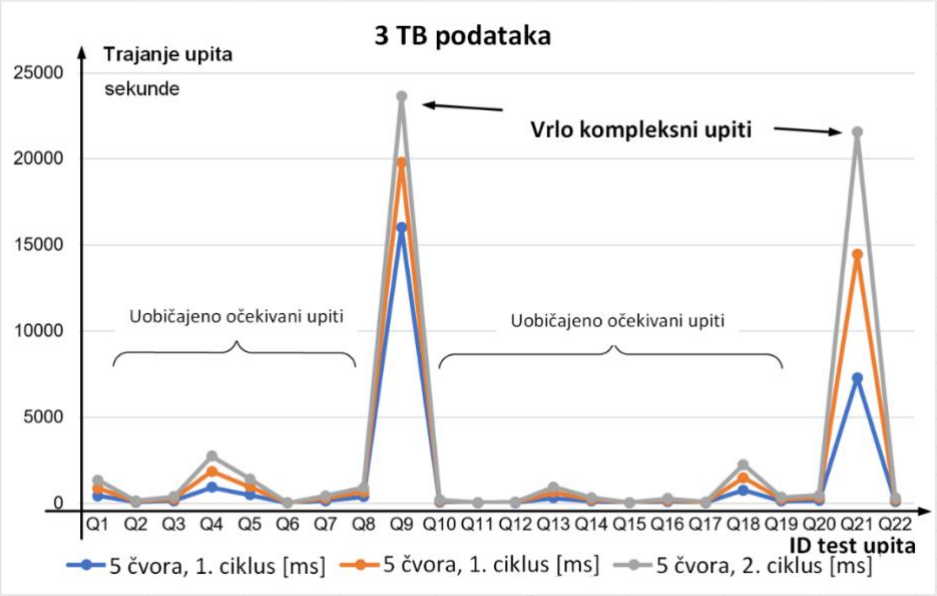
Veličina podataka	1 TB data				3 TB data			
Upit br..	3 čvora, 1. krug	3 čvora, 2. krug	5 čvorova, 1. krug	5 čvorova, 2. krug	3 čvora, 1. krug	3 čvora, 2. krug	5 čvorova, 1. krug	5 čvorova, 2. krug
Q20	131	129	87	73	156	137	150	147
Q21	1763	2850	1703	1787	7278	7021	7196	7085
Q22	60	67	62	35	107	88	110	106
Rezultat u [s]	6341.7	7614.6	5053.8	4879.3	27566.2	26173	15134.4	15022
Rezultat u [h]	1.76	2.12	1.4	1.36	7.66	7.3	4.2	4.17

Tabela 5 - TPC-H benchmark upit za Q1 do Q22



Slika 12 - Trajanje TPC-H upita na bazi podataka od 1 TB (od Q1 do Q22)

Performanse TPC-H testova koji izvedeni na Vertica klasteru, bili su izmereni za veličine baze podataka od 1 TB i 3 TB (Tabela 4). Oni su prikazani na slikama (Slika 12 i Slika 13), ukazujući na slična iskustva vremenskog trajanja kod kompleksnih i neuobičajenih SQL upita.



Slika 13 - Trajanje TPC-H upita na bazi podataka od 3 TB (od Q1 do Q22)

Prema zahtevu klijenta, takođe je bilo neophodno da u naš izveštaj uključimo dva probna rada na istim hardverskim konfiguracijama. Prvi skup rezultata, vremena izvršavanja TPC-H upita je završen nakon ponovnog pokretanja sistema po izvršenju „cold restart“. Drugi skup rezultata probnih izvršenja daje indikaciju poboljšanja performansi samo nakon ponovnog pokretanja baze podataka.

Nadgledanje I/O zahteva za precizno snimanje ponašanja radnog opterećenja važno je za dizajn, implementaciju i optimizaciju podsistema za skladištenje. TPC-H kolekcija praćenja na kojoj smo sproveli analizu prikupljena je na Vertica 9.0.1 klasteru baza podataka koji radi na CentOS Linux 7.3 (instaliran na ekt4 sistemu datoteka), pet čvorova, 2 10 jezgara CPU Intel® Xeon E5-2660v3@2.66 GHz, 16 8 GB = 128 GB RAM, 6 HDD 900 GB (@15K o/min), 2 1 Gb Ethernet, 2 10 Gb Ethernet, 2 16 Gb Fiber Channel adapter. Zbog odnosa performansi i cene koju je odredio klijent, ne bismo mogli da preporučimo brži disk sa stanovišta broja I/O operacija.

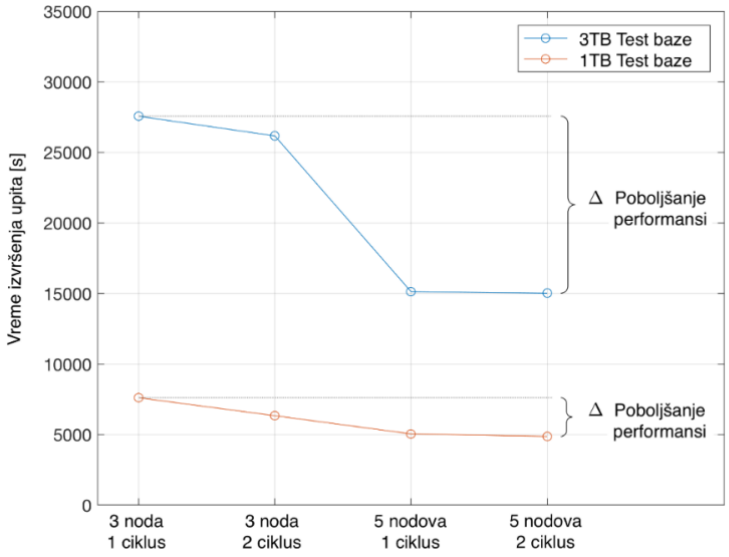
6.2 Komparacija rezultata

Kao što je predstavljeno, TPC-H se takođe može koristiti kao metrika za razmatranje više aspekata sposobnosti NoSQL Vertica sistema baze podataka da obrađuje upite. Aspekti poboljšanja performansi za različite veličine baza podataka i proširenje sistema prikazani su zajedno u Tabela 6 i slikama (Slika 14, Slika 15 i Slika 16). Iz njih je moguće zaključiti da očekivane potrebe za budućim nadogradnjama sistema i očekivanim performansama. Na osnovu dokaza iz izmerenih poboljšanja performansi sa tri do pet čvorova testiranih na 1 TB i 3 TB baze podataka do ovog zaključka možemo vrlo lako doći. Tabela 6 predstavlja rezime TPC-H upita na konfiguracijama sa tri i pet čvorova u prvom i drugom pokretanju na test bazama podataka od 1 TB i 3 TB.

	3 čvora	**3 čvora**	**5 čvorova**	**5 čvorova**
	1st kolo	2nd kolo	1st kolo	2nd kolo
Rezultati u [s] za 1 TB [s]	7614,6	6431,7	50503,8	4879,3
Rezultati u [s] za 3 TB [s]	33.099,26	27.566,2	15.134,4	15.022,0

Tabela 6 - Rezime TPC-H upita

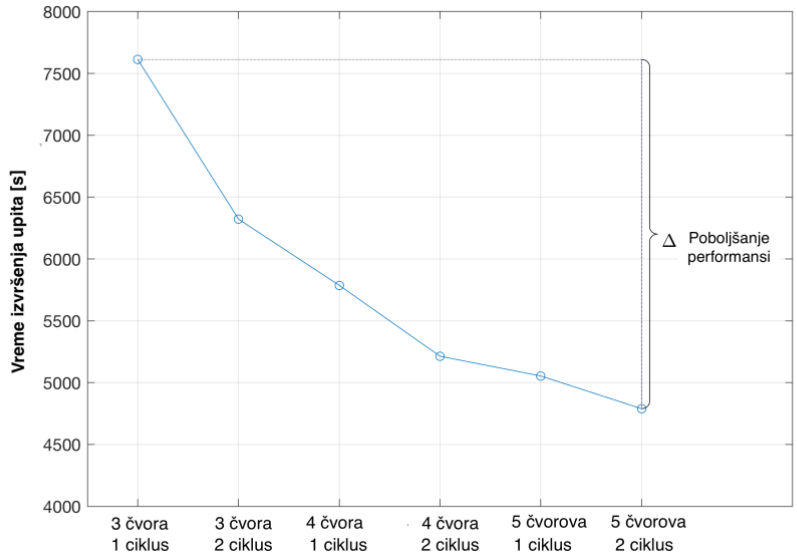
Slika 14 predstavlja vizuelizaciju poboljšanja performansi za vreme izvršavanja TPC-H upita, kao i njihova poređenja nakon prvog i drugog pokretanja na tri i pet čvorova u Vertica klasteru za veličine test baza podataka od 1 TB i 3 TB.



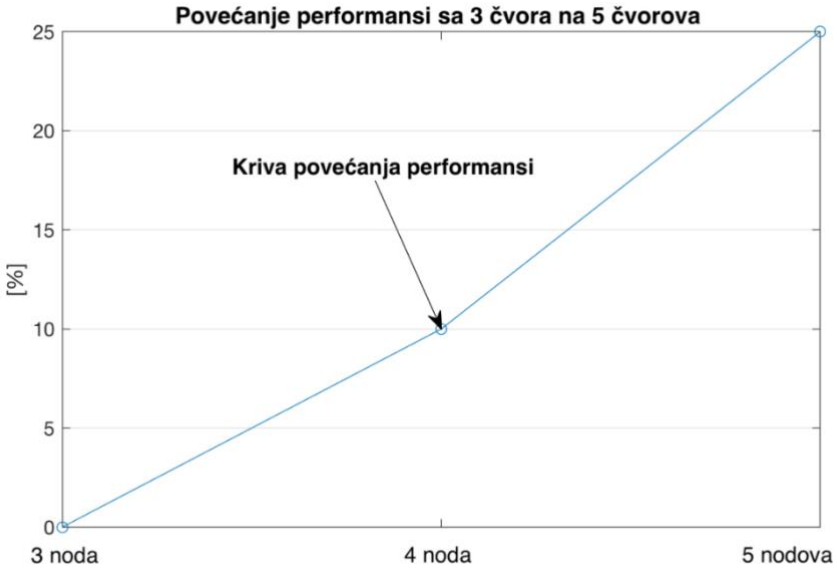
Slika 14 – Uporedni test performansi za podatke od 1 TB i 3 TB

U poređenju poboljšanja performansi i perspektiva skalabilnosti, rezultati pokazuju povećanje performansi od najmanje 25% sa 3 na 5 čvorova (Slika 16) na veličini baze podataka od 1 TB koristeći jeftin hardver.

Slika 15 predstavlja ukupan zbir 22 TPC-H test vremena izvršenja upita do pet čvorova u Vertica klasteru za test bazu podataka od 1 TB. Poboljšanja performansi su primetna nakon drugog pokretanja zbog ponovnog pokretanja baze podataka i nakon horizontalnog skaliranja sa dodatnim čvorovima.



Slika 15 – Poboljšanje performansi u zavisnosti od broja čvorova



Slika 16 - Poređenje performansi sa tri do pet čvorova u Vertica klasteru (1 TB)

6.3 Generalizacija zaključaka performansi klaster sistema

Prikazana arhitektura sistema omogućava postizanje zahtevanih performansi sistema na osnovu implementacije sledećih ključnih osobina:

1. Upotreba tradicionalnog hardvera.
2. Uvođenje horizontalne arhitekture sistema, povezivanjem čvorova u klaster.
3. Postizanje skoro linearne zavisnosti karakteristike performansi sistema.
4. Visoka bezbednost sistema, uključujući i visoku dostupnost podataka uz postizanje visoke propusnosti predloženog rešenja.
5. Saglasnost sa zahtevima TPC-H standarda.
6. Podrška za smeštanje veoma velike količine podataka i njegove skalabilnosti od TB pa do nivoa PB.
7. Slojevita organizacija sistema, gde razlikujemo deo za upravljanje sistemom, deo za smeštanje podataka i deo za vizualizaciju odnosno prikazivanje podataka.
8. Podrška kompletne zahtevane analize podataka i analitičkih servisa u okviru same baze podataka, uključujući predikativni model, mašinsko učenje koje podržava npr. regresiju, klasifikaciju podataka kao i analizu vremenskih serija ili analizu sekvenci podataka.
9. Mogućnost upotrebe rešenja u bilo kom domenu moderne civilizacija gde ima potrebe za obradom velike količine podatka uz ispunjenje više navedenih osobina i zahteva.

7. Praktični primer platforme, upotreba i preporuke

Analitika velike količine podataka (BDA) napravila je u zdravstvu značajnu kvalitativnu razliku medicinskih informacionih sistema. Uvođenjem analitičkih sposobnosti, uključujući veštačku inteligenciju (engl. AI – Artificial Intelligence) primenjenu u zdravstvenim ustanovama i organizacijama dovodi se do smanjenja operativnih troškova, gde dolazi do povećanja kvaliteta medicinske nege i usluga pruženim pacijentima.

Implementacija opisane platforme ima za cilj unapređenje postojećeg zdravstvenog sistema Češke Republike, čijim je uvođenjem došlo do kvalitativnog poboljšanja pruženih medicinskih usluga prema pacijentima, prema ostalim državnim javnim informacionim sistemima. Celokupno rešenje se oslanja se na generalnu arhitekturu i rezultate analize dobijenih prilikom razvoja prethodno navedene platforme, a koji su prikazani u prethodnom poglavlju.

Pored pružanja analitičkih sposobnosti platforme, sposobnosti koje podržavaju osnovne integracione principe sa ostalim sistemima, kao i trenutnu i buduću veštačku inteligenciju sa mašinskim učenjem i rudarenjem, tj. istraživanjem podataka, postoji potreba za etičkim razmatranjima podataka koja nalažu nove načine očuvanja privatnosti, a koja su uslovljena sve većim brojem propisa i očekivanja. Predstavljena BDA platforma, ispunila je sve zahteve (N > 100, N-broj zahteva), uključujući sve standarde vezane za zdravstvene informacione sisteme koji su bili veoma zahtevni. Merilo za podršku u odlučivanju za performanse obrade transakcija su postavljenje na osnovu TPC-H testa performansi uz usaglašenost sa zakonodavstvom Evropske unije (EU) i Češke Republike.

Predstavljeni koncept na osnovu koga je bilo izgrađeno produkciono rešenje, proizilazi iz generalizovanog modela za obradu velike količine podataka stvorenim prilikom razvoja konkretnog sistema na osnovu unapred definisanih zahteva. Predloženo rešenje predstavlja nadogradnju generalizovanog okruženja, koje objedinilo izolovane informacione sistema zdravstvenog informacionog sistema Češke Republike. Predstavljena BDA platforma, artefakti i koncept koji je ovde predstavljen se može preneti na sisteme zdravstvene zaštite drugih zemljama zainteresovanih za razvoj ili nadogradnju sopstvene nacionalne zdravstvene infrastrukture. Ovaj koncept zdravstvenu zaštitu čini isplativu, bezbednu, skalabilnu uz postizanje obrade podataka, njihovu prezentaciju na sistemu, tj. platformi visokih performansi.

7.1 Zahtevi Instituta za zdravstvene informacione sisteme i statistiku

Zahtevi IHIS-a mogu se grupisati prema sledećim aspektima:

1. **Skalabilnost:** sistem mora omogućiti poboljšanje performansi putem dodatnih i dostupnih računarska tehnologija, uključujući hardverske proizvode.

2. **Modularnost, otvorenost i interoperabilnost:** komponente sistema moraju biti integrisane preko specificiranih interfejsa, prema tačnim specifikacijama zahteva. Takođe je neophodno da različiti proizvođači mogu bez većih poteškoća da iskoriste komponente sistema.

3. **Mogućnost zamene:** rešenje sistem mora da podržava instalaciju operativnih sistema otvorenog koda, kao i da sadrže alate za neprofitne i obrazovne svrhe. Sistem mora biti u skladu sa standardnim sistemom skladišta podataka (engl. DWH). Neke komponente moraju biti zamenljive sa Komponente sistema (engl. MHDA - Massive High Data Analytics).

4. **Proširivost:** svi alati i komponente sistema moraju da obezbede prostor za buduće nadogradnje, uključujući unapređenje funkcionalnosti i mogućnosti.

5. **Osiguranje kvaliteta:** alat za kontrolu kvaliteta podataka i integriteta meta podataka je potreban da bi se to osiguralo obrađeni podaci ostaju tačni tokom čitavog postupka analize.

6. **Bezbednost:** sistem mora da radi na lokalnim serverima, bez oslanjanja na računarstvo u oblaku (engl. Cloud) ili angažovanje spoljnih rezervnih sistema. Neophodno je da sistem obezbedi sigurnost svih podataka od eksternih ili unutrašnjih pretnji u smislu IT bezbednosti. Stoga su autorizacija, pristup skladištu i komunikacija od najveće važnosti. Prava pristupa korisnika su morala biti postavljena na nivo baze podataka, tabele ili kolone da bi se ograničili podaci pristup ograničenom broju naprednih korisnika. Sistem mora da evidentira sva izvršenja i pročita operacije za buduće revizije. Takođe, mora da podržava alate za kontrolu verzija i razvoj, dok ispunjava zahteve za kvaliteta meta podatka i verziranije podataka, pravljenje rezervnih kopija i arhiviranje.

7. **Jednostavnost:** sistem mora omogućiti paralelnu timsku saradnju na svim procesima, protoku podataka i šeme baze podataka. Svi zadaci moraju biti u potpunosti uređivani, omogućavajući urezivanje i vraćanje promena u podacima i meta podaci. Neophodno je da Sistem bude jednostavan i lak za korišćenje, kao i stabilan i otporan na ispade podsistema.

8. **Performanse:** Sistem mora biti dizajniran za navedeni minimalni broj istovremenih korisnika. Razmatraju se grupna obrada izvora podataka i sofisticirane analize rudarenja podataka suštinski. Kompletna integraciona obrada tromesečnih podataka ne sme biti duža od jednog sata.

7.2 Izazovi i mogućnosti

U slučaju velike količine podataka, tradicionalni algoritmi za obradu analitičkih operacija, nemaju dovoljnu efikasnost prilikom obrade heterogenih podataka. Za rešavanje ovog problema potreban je veoma efikasan proces čišćenja podataka, što predstavlja veliki izazov svakog sistema pre nego što bude sposoban za njihovu analizu.

1. **Promena količine:** Tehnologije kao što je Hadoop nam omogućavaju da upravljamo velikim količinama podataka sa relativno niskom cenom izgradnje takve platforme. Međutim, učestalost promena veličine podataka može uticati na izvršenje celokupne analize. Korišćenje hibridne infrastrukture za računarstvo u oblaku (engl. Cloud) u kombinaciji sa smeštanjem podataka na lokalnom nivou, može predstavljati vrlo značajan izazov prilikom izgradnje celokupnog rešenja.

2. **Poverljivost:** Poverljivost podataka koji se analiziraju i koji najčešće pripadaju pojedinačnim korisnicima i kupcima čine rukovanje velikim podacima veoma diskutabilnim pitanjem. Zakoni koji su uokvireni zahtevom za zaštitu prava na privatnost, njihovi podaci analiziraju imaju direktan uticaj na poverljivost podataka.

3. **Performanse:** Sa opcijama kao što je analiza podataka kao usluga (engl. DaaS – Data as a Service) dostupnim za eksterno angažovanje analiza velike količine podataka, performanse i kašnjenje postaju veliki izazov. Uspešan načini za smeštaj *stream* podataka ispunjava princip ili zahtev da platforme koje se koriste za dopremanje i

smeštaj podataka moraju imati visoku dostupnost, efektivnost i omogućiti njihovu upotrebu u realnom vremenu.

7.3 Komponente sistema

U okviru konkretnog predloženog rešenja zdravstvenoj instituciji IHIS, možemo prepoznati da su zastupljene sve komponente razvijene platforme (poglavlje 5.2), osim komponente za obradu podataka u realnom vremenu, koja je predmet daljeg razvoja ove platforme.

7.3.1 Sloj integracije podataka predloženog rešenja

Sloj integracije podataka (engl. Data Integration) predstavlja sistemski modul koji omogućava parametrizovane podatke funkcije manipulacije, uključujući transformaciju podataka, kontrolu obrade i hijerarhiju, čitanje, pisanje, i paralelna ili sekvencijalna obrada zadataka, tj. niti. Koristimo termin „meta podaci“ da opišemo rezultirajuće zadatke statistike, klasifikacije ili agregacije podataka. Integracionim slojem obezbeđujemo meta podatke za razvoj, test i proizvodno okruženje cele platforme. Sloj integracije podataka takođe pruža vizuelizaciju njegovih procesa u oblik dijagrama toka podataka. Drugi alat specifičan za sloj integracije podataka generiše izlaze iz prethodno obrađenih podataka. Ovaj alat takođe podržava brz razvoj procesa, uključujući odabir i transformaciju velikih količina primarnih podataka u paralelnom izvršavanju sa više vlakana. U bliskoj budućnosti ćemo morati da se više pozabavimo tehničkim pitanjima i operativnim izazovima. Sloj za integraciju podataka takođe sadrži alat za otklanjanje grešaka za razvoj softvera, testiranje i održavanje.

7.3.2 Skladište podataka predloženog rešenja

Skladištenje podataka (engl. DS – Data Storage) predstavlja sistemski modul koji sadrži horizontalno skalabilan klaster fizičke arhitekture izgrađene na NoSQL Vertica bazi podataka. Sloj za skladištenje podataka radi na hardveru sa mogućnostima distribuiranog skladištenja podataka, što omogućava masovnu paralelnu obradu (engl. MPP - Massive Parallel Processing) preko celokupnih podataka smeštenih u njemu. Sloj za skladištenje podataka čuva podatke u formatu kolone i to u dva kontejnera (engl. WOS - Write Optimized Store) i (engl. ROS - Read Optimized Store), radi postizanja najboljih performansi. Svaki klaster predstavlja kolekciju servera (čvorova) sa softverskim paketom NoSQL baze podataka Vertica. Svaki čvor je

konfigurisan za pokretanje Vertica baze podataka kao član specifičnog klastera baze podataka, koji podržava redundantnost, visoku dostupnost i horizontalnost skalabilnost, obezbeđujući efikasne i kontinuirane performanse. Ova infrastruktura omogućava oporavak bilo kog potencijalnog kvara čvora dozvoljavajući drugim čvorovima da preuzmu kontrolu. Za predstavljeno rešenje (Slika 3), postavili smo toleranciju greške K-Safety = 2 [69]. Komponente sloja za smeštanje podataka određuju koliko kopija sačuvane podatke Vertica treba da kreira u bilo kom trenutku. Pošto se radi o platformi koja podleže licenciranju, konkretno rešenje za potrebe testiranje, prvenstveno testiranja performansi koje je usaglašeno sa definisanom količinom podataka, sadrži u sebi licence Vertica NoSQL baze podataka za ovu veličinu podataka.

7.3.3 Upravljanje kvalitetom podataka

Modul upravljanja kvalitetom podataka (engl. Data Quality Management) podržava kontrolu kvaliteta podataka uključujući trendove i strukturu podataka. Modul upravljanja kvalitetom podataka generiše složene modele za krajnje korisnike koji podržavaju analizu podataka za otkrivanje i ispravljanje grešaka, kao i sofisticirana vizuelizacija podataka i njihovo izveštavanje za potrebe kvaliteta podataka. Ovaj modul kreira, sortira, grupiše i traži pravila validacije uneta u strukturiranom obliku. Pravila validacije mogu da se izvrše preko korisnički definisanog skupa podataka i da se njima upravlja centralno.

7.3.4 Upravljanje meta podacima

Modul za upravljanje meta podacima (engl. MDM – Master Data Management) podržava upravljanje korisničkim, tehničkim i operativni meta podaci. Sloj za upravljanje meta podacima centralno obrađuje meta podatke iz svake komponente sistem, smeštene zajedno u skladištu podataka. Sloj za upravljanje meta podacima može da uporedi različite verzije meta podataka i izlaze prikaza, uključujući vizuelizaciju namenjen za izveštavanje podataka. On je u stanju da kreira dinamičke, aktivne grafikone i tabele, time što omogućava višedimenzionalne i interaktivne poglede. Dalje, koristi princip tzv. sandboxing za testiranje privremenih ulaza i izlazi i može da generiše izlaze u HTML, PDF i PPT formatu. Jedna njegova komponenta koristi onlajn analitičke operacije obrade (engl. OLAP – Online Analytical Processing) preko višedimenzionalnog modela podataka. Pored toga sadrži rečnik pojmova i njihovih veza kako bi se omogućila analiza uticaja.

7.3.5 Ad-hoc priprema analize predloženog rešenja

Za procese pripreme ad-hoc analize (engl. AAP – Ad-hoc Analysis Preparation) programirali smo dve različite verzije u Talend Open Studio alatu za integraciju. U prvoj verziji, MHDA koristi transformaciju ekstrakcije i učitavanje (engl. ETL – Extract Transfer and Load) komponente alata za integraciju. Ove komponente čitaju podatke iz struktura skladišta podataka (dimenzije i tabele činjenica) u memoriju. Zatim, komponente filtriranja i agregacije obrađuju podatke u izlaznu tabelu. Druga verzija koristi komponente (engl. ELT – Extract, Load, Transform) od alata za integraciju. I komponenta ETL, kao i ELT su u stanju da generišu nemodifikovani SQL upit jednostavan za korišćenje, kao jezika za manipulaciju podacima (engl. DML – Data Manipulation Language) u pozadini. AAP sloj predložene platforme ubrzava vreme obrade podataka bez potrebe za učitavanjem velikih količina meta podataka u memoriju programa. Slika 4 prikazuje predviđanje na istorijskom skupu podataka testa koji je dostavio IHIS, gde smo testirali ARIMA model [71] za pristup vremenskim serijama u bazi podataka. Ovaj model se može kreirati ili direktno u NoSQL Vertica bazi podataka, pošto ona podržava predikativno modeliranje, ili u zasebnom statističkom alatu kao npr. Tableau, koji će uzeti podatke iz baze podataka i vratiti kreirani model (napisan u engl. PMML - Predictive Model Markup Language) ili drugi format koji baza podataka podržava.

7.3.6 Vizuelizacija podataka predloženog rešenja

Sloj za vizuelizaciju podataka (engl. Data Visualization) sadrži alate za opisivanje perspektiva podataka, njihovo vizualno predstavljanje kao i dalje obradu koja može biti povezana sa elementima veštačke inteligencije. DV komponente vizuelno predstavljaju podatke i meta podatke i daju interpretacije za moguće uvide. Pored toga, ugradili smo DV komponente u Tableau da bismo obezbedili podatke i vizuelizacije meta podataka u grafikonima i slikama. Tableau je popularna interaktivni analitički alat za vizuelizaciju podataka. Ovaj alat može pomoći da se sirovi podaci pojednostave u lako razumljive kontrolne tabele i radne listove. Na primer, Slika 10 prikazuje deo vizualnog predstavljanja podataka iz jednog od IHIS studija sa preklapanjem geografske karte Slika 9. Primer prediktivnog modela za predviđanje vremenskih serija na osnovu dostavljenih test podataka iz IHIS. Primer pokazuje deo sloja za obradu podataka i vizuelizaciju, prikazan kao snimak ekrana napravljen u Tableau Desktop (verzija 10.5).

7.4 Postupak za izgradnju koncepta predloženog rešenja

Postupak za izgradnju koncepta predloženog rešenja oslanja se na izgradnju klastera za obradu velike količine podataka je organizovan u više koraka koji imaju logičku vezu stvarajući jednu celinu za postizanje zahtevanog rešenja. Kao prvo, moramo pripremiti bazu podataka, njenu strukturu tj. model baze podataka u koji ćemo uneti podatke. Podaci su definisani za potrebe izgradnje i testiranja klastera u smislu njegove funkcionalnosti ali i provere postignutih performansi. Takođe, moramo imati pripremljen alat za integracioni deo koji se koristi za čitanje i unos podataka u bazu, koju imamo na klaster platformi. Sledeći korak predstavlja mapiranje ulaznih podataka i provera ispravnosti unetih podataka u bazu podataka uz merenje vremena potrebnog za unos podataka. Pored toga, moramo obezbediti i postupak validacije tj. provere ispravnosti ulaznih podataka koje ćemo na automatizovani način uz pomoć integracionih alata uneti u bazu. Nakon završenog dela dizajna automatizovanog procesa za učitavanje podataka, moramo obezbediti sistemsku kontrolu podataka nakon njihovog unošenja u bazu, na osnovu čega može doći do pokretanja sledećeg koraka vezanog za celu proceduru izgradnje i testiranja klastera za obradu velike količine podataka. Pored svega, moramo obezbediti i kontrolu meta podataka kao sastavni deo cele procedure, koja kasnije omogućava razne analize koje se vezuju za meta podatke i njihovo upravljanje. Sledeći sastavni deo rešenja je kontrola kvaliteta podataka, koji će biti smešteni u bazu podataka, a koja predstavlja skladište podataka koje se vezuje za deo upravljanja kvalitetom podataka. Priprema podataka za izvršenje ad-hoc analize, koji se tiču podataka za konkretnu analizu uključuje funkcije kao što je filtriranje podataka, agregaciju, sortiranje i sl.

Jedan od važnih koraka u okviru izrade klastera za obradu velike količine podataka je pripremanje više dimenzionalnog modela podataka radi njihove analize i grafičke prezentacije u specijalizovanom alatu za takve operacije, kao što su npr. alati za poslovnu inteligenciju.

Postupak za izgradnju klastera za obradu velike količine podataka je koncipiran u okviru 9 koraka, koji su opisani u sledećim poglavljima.

7.4.1 Korak 1 – Priprema skladišta podataka

1. Izrada modela baze podataka prema predatom zadatku.

Cilj	Kreirati strukture podataka u DB prema predloženoj DDL definiciji (ili njihovom semantički kompatibilnom ekvivalentu) za druge delove testa.
	Proveriti funkcionalnost procedura za rad sa modelima podataka i šemama.
Ulaz	Kompletan opis modela podataka (Slika 11).
Izlaz	V1.1.a Kreirati model podataka u DB.
	V1.1.b Opis i objašnjenje razlika u strukturi i tipovima podataka u DB u poređenju sa zadatkom.
	V1.1.c Opis koraka potrebnih za kreiranje modela u DB (korišćeni alati, komande, itd.) za mogućnost replikacije tj. ponavljanja procedure, uključujući parametre za skladištenje podataka (indeksi, particionisanje, segmentacija, strani ključevi, itd.).
Procedura	DDL koju šemu učitava alat za rad sa SQL-om.
	Alternativno, relevantne strukture se generišu iz projekta u Visual Paradigm (https://www.visual-paradigm.com).
Reference	P1.1.a Model baze podataka (ERD) u PDF-u.
	P1.1.b ERD u alatu za modeliranje Visual Paradigm.
	P1.1.c DDL šema u ANSI-SQL i odabrani dijalekti SQL baza podataka.
Praćeni parametri i zahtevi	1) Model je kompatibilan sa specifikacijom.
	2) Nije bilo potrebno praviti semantičke izmene modela podataka, ručnu optimizaciju modela i atributa, kompletiranje nagoveštaja, izvedenih atributa, tabela, pogleda, particija itd.

2. Dizajn i implementacija procesa integracije podataka.

Cilj	Dizajn i implementacija (programiranje) procesa integracije u isporučenom alatu sloja za integraciju podataka.
	Verifikacija funkcionalnosti isporučenog alata za integraciju u svim koracima unapred pripremljenog procesa integracije.
	Dizajnirati i opisati optimizaciju procesa integracije dostavljenih podataka u smislu efikasnosti kreiranja (optimizacija dizajna), kao i u procesu (optimizacija brzine).

Ulaz	Mapiranje od izvora do cilja ulaznih atributa u atribute rezultirajućeg dimenzionalnog modela.
	Tehnička dokumentacija obrađenog procesa u DI alatu Talend Open Studio za integraciju podataka (TOSDI).
Izlaz	V1.2.a Kreiran proces integracije u isporučenom DI alatu koji odgovara predatom zadatku.
	V1.2.b Tehnička dokumentacija procesa.
	V1.2.c Opis koraka potrebnih za pokretanje koraka integracije.
Proces	Koristite isporučeni DI alat za dizajniranje, kreiranje i testiranje procesa integracije podataka, uključujući osnovne validacije sadržaja i proces sastavljanja procedura hospitalizacije prema priloženim dokumentima, odnosno ceo korak 2.
	U slučaju procesa sastavljanja slučajeva hospitalizacije, nije potrebna predložena implementacija ciklusa u protoku podataka (ekvivalent kursorima baze podataka), što je samo jedna od bonus karakteristika sistema. Ukoliko ponuđeno rešenje ne podržava ove cikluse, potrebno je osmisliti najjednostavniji mogući način implementacije ovog ciklusa (npr. korisničku proceduru).
Reference	P1.2.a Interfejs podataka za ulazne podatke.
	P1.2.b Mapiranje od izvora do cilja.
	P1.2.c Proces integracije uzorka.
Praćeni parametri i zahtevi	1) Implementirani proces u logici integracije u potpunosti se poklapa sa dostavljenim šablonom.
	2) Uslovi za DI alat date SRS su ispunjeni.
	3) Proverava se jasnoća i jednostavnost evidentiranja procesa integracije podataka i njegove dokumentacije.

3. Početno preuzimanje sadržaja skladišta podataka.

Ovo su podaci svih kvartalnih izvoza van poslednjeg kvartala u skladište podataka, odnosno period 2005/01 - 2008/09. Ulaz za ovaj korak su CSV datoteke, kreirane kao slika tabela skladišta podataka („dump“ baza podataka), koje se učitavaju u rezultujuću bazu podataka u produkcionom okruženju 1:1.

Cilj	Popuniti skladište podataka potrebnom količinom podataka.
Ulaz	Ulazni podaci CSV datoteke slika skladišta podataka za izvoz van poslednjeg kvartala.
Izlaz	Popunjene tabele činjenica i dimenzije skladišta podataka.
	V1.3.a Spisak svih tabela činjenica i dimenzija sa brojem zapisa u njima.
Kurs	Vreme obrade se neće procenjivati za ovaj deo testa.
	Čitaju se samo podaci koji odgovaraju strukturi skladišta podataka.
	Moguće je koristiti bilo koji metod preuzimanja podataka koji podržava baza podataka ili koristiti drugi alat za integraciju podataka.
Reference	P1.1.a Model baze podataka (ERD) u PDF formatu.
	P1.3.a Kompresovani CSV podaci sa slikom skladišta podataka 2005/01 - 2008/09.
Praćeni parametri i zahtevi	1) Podaci u listi svih tabela odgovaraju očekivanim vrednostima.

7.4.2 Korak 2 – Standardizovani proces integracije podataka za kvartalni izvoz

Procesi u ovom koraku će biti procenjeni za vreme njihovog izvršenja. Vreme će biti izračunato od početka procesa integracije podataka (7.4.2) i skladištenja rezultata u skladištu podataka, uključujući moguću optimizaciju njihovog skladištenja (particionisanje, indeksi, itd.), obavljanje provera kvaliteta podataka (korak 7.4.2) do generisanja standarda izveštaja (korak 2.3). Učitavanje svakog tromesečnog izvoza se ocenjuje posebno uz ograničenje za obradu. Predviđeno vreme je 12 sati, uključujući sve neophodne korake, važi za sve korake koji su vezani za učitavanje podataka.

1. Inkrementalno učitavanje izvoza, ulaza u skladište podataka.

Cilj	Testirati rad sa tabelama činjenica, dinamičkim dimenzijama, integracijom podataka zasnovanom na meta podacima i funkcionalnosti predloženog procesa integracije podataka.

	Testirati specifične funkcije kao što su SCD2 i „late arriving“ stavki dimenzija i proces sastavljanja ulaznih zapisa u više jedinice agregacije - boravišne rečenice. Testirati efikasnost i brzinu implementiranog procesa integracije kada integrišete određene kvartalne podatke sa unapred popunjenim skladištem podataka.
Ulaz	Datoteke sa ulaznim podacima veta_ab i veta_ac u CSV obliku za poslednji kvartal (meseci 2008/10 - 2008/12).
Izlaz	Dodate tabele činjenica i dinamičke dimenzije skladišta podataka iz ulaznih podataka poslednjeg kvartala. V2.1.a Dnevnik sa tokom procesa integracije podataka, uključujući vremensku evidenciju, iz koje se vidi da se odvijao prema specifikaciji u tački 1.2. V2.1.b Spisak svih tabela činjenica i dimenzija sa brojem zapisa u njima.
Proces	Odvojeni početak procesa integracije za svaki paket podataka. Parametri su nazivi ulaznih datoteka (AB i AC teorema). Dinamičke dimenzije kao što su PoC_patient_dim i PoC_item_dim se dopunjuju odvojenim nitima iz ulaznih podataka – izračunavaju se dodatni atributi, stavke koje imaju kašnjenje u isporuci od strane pošiljaoca, proveravaju se i popunjavaju kao SCD2 dimenzije, i sl. Tabele činjenica se čitaju odvojenim nitima iz ulaznih podataka. Rezidentni zapisi se dodaju u tabelu činjenica PoC_f_pojit, koji se kompajliraju iz ulaznih zapisa prema navedenim pravilima, zatim se povezuje tabela činjenica o stavkama za boravak. Proces mora tačno slediti standardnu proceduru predloženu u tački 1.2 i ne sme biti nestandardnih optimizacija.
Reference	P1.2.a Interfejs podataka za ulazne podatke. P1.2.b Mapiranje od izvora do cilja. P2.1.a Ulazni podaci za proces integracije podataka za period 2008/10 - 2008/12.
Praćeni parametri i zahtevi	1) Ulazni podaci iz eksporta su ispravno učitani u skladište podataka.

2) Rezultat se obrađuje u roku od 12 sati za sve tri osiguravajuće firme od početka celog Koraka 2.

2. Kontrola kvaliteta uvezenih podataka.

Izvršite sledeće provere kvaliteta podataka za svaki obrađeni izvoz, tj. eksport podataka.

a) Inkrementalno učitavanje eksporta ulaznih podataka u skladište podataka.

b) Kontrola kvaliteta učitanih podataka (tiče se DQM alata, podrška analitičke baze podataka).

c) Generisanje unapred određenih izveštaja.

7.4.3 Korak 3 - Ispravka nevažećih podataka u DWH

1. Brisanje nevažećeg eksporta podataka iz skladišta podataka.

2. Učitavanje ispravljene verzije podataka i njihova priprema za import u skladište podataka.

3. Kontrola kvaliteta nad učitanim podacima u skladište podataka.

4. Generisanje unapred definisanih izveštaja.

7.4.4 Korak 4 – Kontrola kvaliteta podataka u celom skladištu podataka

Počevši od ovog koraka, sledeći zadaci su deo procenjenog parametra „poc_speed_benchmark“. Tačnije, ceo korak počinje u dve situacije:

1. Kao deo kontrole kvaliteta za implementaciju svakog procesa integracije podataka, gde su eksplicitno navedeni period obuke i osiguravajućeg društva čiji se podaci uzimaju u obzir. U ovom slučaju, vreme za obavljanje zadatka je uključeno u ograničenje za obavljanje celog procesa integracije podataka (12 sati) i nije uključeno u parametar „poc_speed_benchmark“.

2. Kao poseban zadatak, koji se izvodi preko celog skladišta podataka sa podrazumevanim parametrima. U ovom slučaju, vreme za završetak zadatka je

uključeno u ukupno vreme testa brzine („poc_speed_benchmark“). U ovom slučaju maksimalno vreme za završetak koraka 4 je 1 sat.

7.4.5 Korak 5 – Priprema materijala za ad-hoc analize

U ovoj fazi biće testiran alat (engl. AAP - Ad-hoc Analysis Preparation). Cilj je da se kompleksnijim procesom transformacije podataka uskladištenih u skladištu podataka kreiraju podaci za konkretnu analizu, koji podrazumeva filtriranje, spajanje, agregaciju, sortiranje i sl.

Pratiće se jednostavnost i brzina kreiranja i modifikacije procesa u AAP alatu, kao i brzina same transformacije i ispravnost rezultata. Uslov implementacije je upotreba standardnih AAP komponenti, bez potrebe za pisanjem ad-hoc SQL upita ili drugih skript procedura (uključujući filter za ulazne podatke), čija semantika ne bi bila eksplicitno ilustrovana analizom roda (engl. Lineage Analysis). Svi atributi i transformacije procesa moraju biti jasno predstavljeni u „lineage analizi“. Prvo je potrebno implementirati proces opisan u zadatku 5.1.

Zadatak 5.1

Cilj	Verifikacija rada sa alatom za pripremu podataka iz skladišta podataka za naknadnu analitičku obradu. Provera vremenske pripreme podataka preko velike ulazne datoteke.
Ulaz	Dokumentacija procesa pripreme podataka za ATC grupu. Tabele skladišta podataka: PoC_f_item, PoC_f_residential, PoC_patient_dim, PoC_lecivo_dim. Ulazni parametar ATC kod (atc_l5_code). Izvođenje za ATC grupe L01KSE03, J01CR02 i H02AB02.
Izlaz	V5.1.a Tehnička dokumentacija AAP procesa za ATC grupu. V5.1.b Tabela podataka sa sačuvanim zapisima o dužini praćenja pacijenata lečenih proizvodima iz grupe date ATC kodom. V5.1.c rezultira BI izveštajem koji pokazuje prosečno vreme praćenja za datu ATC grupu u zavisnosti od simptoma smrti.

	V5.1.d Analiza porekla celog procesa i vreme praćenja parametara.
Proces	Proces izračunava vreme praćenja za svakog pacijenta lečenog proizvodom definisanim parametrom za unos ATC koda. Ovaj period se određuje kao period od početka lečenja pacijenta ili od prve primene ovog proizvoda (prijavljivanje predmeta) do smrti ili do kraja lečenja ili poslednje primene proizvoda. Proces takođe čuva informacije o pacijentu da li je pacijent umro (vrednost simptoma 1) ili ne (vrednost simptoma 0). Rezultat se čuva u tabeli baze podataka, uključujući kod koji je unela ATC grupa.
Reference	P5.a Dokumentacija procesa pripreme AAP podataka (za ATC grupe).
Praćeni parametri i zahtevi	1) Kreirani proces ispunjava navedene uslove. 2) Rezultat se dobija uz poštovanje vremenskog ograničenja (20 minuta za ceo korak 5). 3) Dobijeni izveštaj sa vrednostima prosečnog vremena praćenja odgovara pretpostavkama. 4) Analiza loze pokazuje tačno sve korake procesa i izračunavanje vremena praćenja parametara.

U ukupno vreme testiranja brzine („poc_speed_benchmark“) uključeno je vreme izvršenja navedenih procesa u AAP alatu i vizuelizacija rezultata u BI alatu. Maksimalno vreme za završetak koraka 5 je 20 minuta.

7.4.6 Korak 6 – Rad sa alatom za poslovnu inteligenciju

Preduslov je unapred pripremljen dimenzionalni model u BI (engl. Business Intelligence) alatu koji sadrži sve osnovne činjenice i dimenzije prema DWH (engl. Data Warehouse) modelu podataka. Uslov je da koristite standardni BI korisnički interfejs bez potrebe za pisanjem SQL, MDQS ili drugih upita.

Zadatak se odvija u dve situacije:

1. Kao deo izvršavanja svakog procesa integracije podataka, gde su ulazni parametri statičkog izveštaja eksplicitno navedeni. U ovom slučaju, vreme za završetak zadatka se ne računa u ukupno vreme testiranja brzine („poc_speed_benchmark“), već u ograničenje za završetak celog procesa integracije podataka (12 sati).

2. Kao poseban zadatak, izvodi se preko celog skladišta podataka sa unapred podešenim parametrima. U ovom slučaju, vreme za završetak zadatka je uključeno u ukupno vreme testa brzine („poc_speed_benchmark“). Maksimalno vreme za izvođenje koraka 6 je 1 sat.

Vreme potrebno za kreiranje definicije izveštaja, odnosno radno vreme korisnika, nije uključeno u izmerena vremena.

Rezultat zadataka su interaktivni (dinamički veb) izveštaji i izvedeni statički izveštaji, sačuvani kao PDF ili HTML fajl. Statički izveštaji se optimalno kreiraju samo kao statički izvoz iz njegove onlajn verzije sa unapred podešenim parametrima. Definicija oflajn (engl. off-line) verzije izveštaja ima samo one parametre navedene u tabeli koji se razlikuju od onlajn verzije izveštaja, ostali parametri su identični. Termin "SUM" u definicijama izveštaja označava najviši nivo agregacije vrednosti, odnosno svih vrednosti.

7.4.7 Korak 7 –Test performansi baze podataka

Cilj je da se testira brzina unetih opštih upita preko unapred pripremljenog dimenzionalnog modela i drugih podataka (npr. iz faze 5). Pratiće se brzina izvršavanja upita i tačnost rezultata. Upiti se mogu sintaksički modifikovati samo u slučaju drugog SQL dijalekta baze podataka koja se koristi, sve promene moraju biti eksplicitno navedene kao deo dokumentacije za testiranje. Ni upiti ni baza podataka ne mogu biti optimizovani na bilo koji ad-hoc način da bi se povećao učinak ovih konkretnih upita.

Vreme izvršenja ovih upita je uključeno u ukupno vreme testiranja brzine („poc_speed_benchmark“). Maksimalno vreme za izvođenje celog koraka 7 je 1 sat.

Cilj	Testiranje brzine unetih SQL upita preko DS-a.
Ulaz	Lista SQL upita.
Izlaz	V7.1.a Tačna sintaksa izvršenih upita sa objašnjenjem mogućih promena u odnosu na unos.

	V7.1.b Opis izvršenja upita i merenja vremena. V7.1.c Dnevnik sa trajanjem pojedinačnih upita i brojem vraćenih redova. V7.1.d Rezultirajući CSV fajl sa izlaznim rezultatima izvršenih upita.
Napredak	Pokrenite upite u njihovom originalnom obliku ili nakon potrebnog uređivanja sintakse.
	Sačuvajte rezultate upita u CSV datoteke.
	Priložite dnevnik sa trajanjem svakog upita i brojem vraćenih vrednosti.
Reference	P7.1.a SQL za test brzine DB.
Praćeni parametri i zahtevi	1) Semantičke promene u upitu ili njihova optimizacija nisu bile neophodne za izvršavanje upita. 2) Rezultati upita odgovaraju zadatku. 3) Postignuto vreme ne sme prekoračiti dozvoljenu granicu (60 minuta).

7.4.8 Korak 8 – Analiza u bazi podataka i vizuelizacija u alatima poslovne inteligencije

Zadatak ovog koraka je izvođenje zadataka navedenih u nastavku, koji verifikuju i potvrđuju ispravnost funkcija analize baze podataka (kalkulacije u bazi podataka). Testira tri osnovne oblasti podrške za analitičke zadatke:

a) Opis navedene grupe podataka.

b) Zadatak klasifikacije prema pripremljenom modelu.

c) Kreiranje modela predviđanja vremenske serije i njegova primena za validacije zapisa u skladištu podataka.

Svi zadaci moraju biti izvedeni direktno u bazi podataka, kao proširenje preko standardnog mehanizma baze podataka: ili kao proširenje SQL sintakse (poželjno), ili kao poseban jezik upita za obavljanje analitičkih zadataka (npr. proširenje jezika R za podršku analize u bazi podataka).

Za svaki od navedenih zadataka kreirati jednostavan BI izveštaj, koji vizualizuje izračunate podatke u odgovarajućem tabelarnom i grafičkom obliku u vidu onlajn ili off-line izlaza.

Svi zadaci se mere u vremenu i doprinose rezultujućem parametru "poc_speed_benchmark". Izmereno vreme uključuje svo mašinsko vreme za pripremu podataka, proračun i vizuelizaciju, ali ne i ljudsko vreme utrošeno na pripremu i izvođenje analize i izlaza. Vreme kreiranja predikativnog modela u zadatku takođe nije uključeno u izmereno vreme. Maksimalno vreme za izvođenje koraka 8 je 1 sat.

7.4.9 Korak 9 - Modifikacija skladišta i alati za upravljanje meta podacima

Ovaj korak ima za cilj da verifikuje tok aktivnosti u razvoju skladišta podataka, uključujući njegovu dokumentaciju, rad sa razvojnim i proizvodnim okruženjem i rad sa komponentom za upravljanje meta podacima (engl. MDM) - poslovni rečnik, analiza roda, verzija, operativni meta podaci itd.

Svi zadaci u ovom koraku su vremenski ograničeni. Izmereno vreme uključuje svo mašinsko vreme za pripremu podataka, proračun i vizuelizaciju, ali ne i vreme koje je utrošeno na pripremu i izvođenje analize i izlaza. Maksimalno vreme za izvođenje celog koraka 9 je 6 sati.

1. Priprema razvojnog okruženja.

Cilj	Pokazati kako se radi sa razvojnim okruženjem. Pripremiti razvojnog okruženja za druge zadatke. Demonstracija mogućnosti rada sa MDM alatom.
Ulaz	Podaci iz poslednjeg ispravljenog izvoza snimljenog u proizvodnom okruženju (volumen približno 18 GB podataka). Unosite izveštaj u koraku 6.2.
Izlaz	V9.1.a - Zapis o napretku učitavanja poslednjeg izvoza u razvojno okruženje iz proizvodnog okruženja. V9.1.b - Izlaz BI izveštaja 6.2 Pregled troškova lekova prema ATC-u za odabrane dijagnoze.

	V9.1.c - Opis cene poslovnog termina, koji se odnosi na atribut cene u tabeli činjenica poc_f_item. V9.1.d - Izlaz analize porekla za zbirnu metričku vrednost (poc_f_item.price) u ovom izveštaju.
Napredak	Učitajte podatke iz poslednjeg ispravljenog izvoza u razvojno okruženje iz proizvodnog okruženja - sve činjenice datog izvoza i elementi dimenzija na koje se te činjenice odnose.
	Generišite izlaz izveštaja van mreže iz koraka 6.2.
	Izvršite analizu porekla metrike zbira (poc_f_item.price) u ovom izveštaju.
Linkovi Praćeni parametri i zahtevi	1) Učitajte podatke iz proizvodnog okruženja u razvojno okruženje ne zahteva poseban napor ili neočekivano dugo vreme za izvršenje. 2) Izlaz izveštaja iz koraka 6.2 odgovara očekivanim vrednostima. 3) Rezultati iz poslovnog rečnika i izvršena analiza loze pokazuju tačan opis korišćenih termina i njihovog porekla.

2. Modifikacija procesa u razvojnom okruženju

Cilj	Pokazati način razvoja u razvojnom okruženju.
	Demonstrirati proces promene modela podataka i proces integracije podataka, proces rada sa MDM alatom.
Ulaz	Podaci iz poslednjeg ispravljenog izvoza zabeleženog u proizvodnom okruženju.
	Opis toka ovog koraka u nastavku.
	Dokumentacija modifikovanog modela podataka.
	Dokumentacija modifikovanog procesa integracije uzoraka podataka.
Izlaz	V9.2.a Modifikovani model podataka (dokumentacija i DDL).
	V9.2.b Dokumentacija modifikovanog procesa integracije podataka.

	V9.2.c Izlaz BI izveštaja iz koraka 6.2 Pregled troškova lekova prema ATC-u za odabrane dijagnoze.
	V9.2.d Opis izmenjene cene poslovnog termina, koji se odnosi na atribut cena tabele činjenica poc_f_item.
	V9.2.e Izlaz analize porekla za novo izračunatu vrednost zbirne metrike (poc_f_item.price) u ovom izveštaju.
Napredak	Izmenite model podataka razvojnog okruženja skladišta podataka, gde će novi atribut cena_orig biti dodat u tabelu poc_f_item, promeniti poslovni rečnik za termin cena.
	Izmenite prvobitni proces integracije podataka da biste sačuvali originalnu prijavu vrednosti proizvoda u atributu price_orig i da biste uskladištili u atributu cena vrednost koja je ispravljena za maksimalno plaćanje sa odgovarajuće liste kodova proizvoda (ako postoji, koristite minimalnu cenu, inače koristite poc_f_item.price_orig).
	Kreirajte automatski generisanu dokumentaciju ovih promena.
	Kreirajte novi proces integracije podataka koji modifikuje već obrađene podatke u skladištu podataka.
	Izvršite gornju modifikaciju podataka u DWH razvojnom okruženju.
	Generišite ažuriranu verziju izlaznog izveštaja iz koraka 6.2 i analize porekla, opisane u koraku 9.1.
Linkovi	P9.2.a Modifikovani model podataka baze podataka (ERD).
	P9.2.b Dokumentacija modifikovanog procesa integracije uzoraka podataka.
Praćeni parametri i zahtevi	1) Modifikacija modela podataka ili procesa integracije podataka ne zahteva poseban napor ili neočekivano dugo vreme da se izvrši.
	2) Modifikovani model podataka i proces integracije podataka ispunjavaju navedene zahteve.
	3) Izlaz generisanog izveštaja iz koraka 6.2 sadrži očekivane vrednosti.
	4) Rezultat analize loze pokazuje razliku u izračunavanju metričke vrednosti u ovom izveštaju.

3. Prenos promena iz razvojnog u proizvodno okruženje

Cilj	Pokazati kako napraviti modifikacije napravljene iz jednog okruženja u drugo. Demonstrirati mogućnosti alata za upravljanje meta podacima (MDM) u različitim verzijama i operativnom upravljanju meta podacima.
Ulaz	Modifikovana verzija modela podataka i procesa integracije podataka u razvojno okruženje.
Izlaz	V9.3.a BI izveštaj izlaz iz koraka 6.2 Pregled troškova lekova prema ATC-u za izabrane dijagnoze. V9.3.b Poređenje verzija originalnog i modifikovanog procesa integracije podataka. V9.3.c Izlaz izveštaja sa zapisom o napretku svih početaka procesa integracije podataka i izveštajem o svim generisanim BI izveštajima.
Proces	Prenesite promene iz razvojnog okruženja u proizvodno – modifikovan model i proces integracije podataka i proces modifikacije postojećih podataka. Napravite pripremljenu modifikaciju postojećih podataka u proizvodnom okruženju. Generišite izveštaj iz koraka 6.2 u proizvodnom okruženju. U MDM-u, uporedite originalnu i novu verziju procesa integracije podataka i vizualizujte rezultat tako da su razlike između dve verzije jasno vidljive. Napravite izveštaj koji opisuje napredak svih pokrenutih procesa integracije podataka, koji će sadržati informacije o vremenima i trajanju svakog pokretanja, ulaznim parametrima, verziji, vremenu izvršenja svakog koraka, veličini obrađenih podataka i rezultatu procesa. Napravite izveštaj sa opisom napretka svih generisanih unapred pripremljenih izveštaja u BI alatu, vremenom i trajanjem njihovog generisanja i mestom skladištenja.
Linkovi Praćeni parametri i zahtevi	1) Prenos promena iz razvojnog u proizvodno okruženje ne zahteva poseban napor ili neočekivano dugo vreme za implementaciju.

2) Izlaz generisanog izveštaja iz koraka 6.2 sadrži očekivane vrednosti.

3) Poređenje originalne i modifikovane verzije procesa integracije podataka pokazuje razlike između ovih verzija na jasan i razumljiv način.

4) Izveštaji sa listom svih tekućih procesa integracije podataka i svi generisani BI izveštaji sadrže sve važne informacije.

8. Diskusija

U ovoj doktorskoj disertaciji opisan je novi pristup rešavanju problema vezanog za izgradnju klastera za obradu velike količine podataka u realnom vremenu. Pristup je jedinstven jer obezbeđuje postizanje kompromisa između podešavanja veličine klastera i njegove cene vezane za njegovu izgradnju, kao i upotrebljenih hardverskih, softverskih tehnologija. Takođe, ovaj pristup omogućava dalji razvoj i unapređenje rešenja, kako na osnovu povećanja podataka, tako i na osnovu zahteva za povećanjem performansi klastera. Razvijeni model je primenljiv u različitim domenima kao što su na primer saobraćaj, finansije, zdravstvo, energetika i sl. Osim opisa pristupa za definisanje sistema visokih performansi i sistema za generalnu obradu velike količine podataka na Big Data klasterima, ova doktorska disertacija sadrži i predlog konkretne arhitekture.

Upotreba tehnologije koja je namenjena unapređenju sistema zdravstvene zaštite može se proceniti u vidu postignutih performansi, privatnosti, bezbednosti, interoperabilnosti, usklađenosti, troškova i buduće provere kao što su skalabilnost do inkrementalnih hardverskih integracija, analitičkih alata i eksponencijalno povećanja podataka. U slučaju zdravstvenog sistema Češke Republike, nezavisno od isporučioca rešenja, morao je biti zadovoljen veliki broj zahteva koji obuhvataju sve navedene kriterijume namenjenih modernizaciji nacionalnog zdravstvenog sistema u okviru Evropske unije. Predstavljeno rešenje koje je prihvatio IHIS ispunilo je sve zahteve i pokazalo je rezultate performansi sistema koji znatno premašuju potrebne zahteve.

Sve veći obim medicinske dokumentacije pacijenata, kao i podataka generisanih na primer iz IoT i mobilnih uređaja, u bliskoj budućnosti nalažu uređajima usvajanje analize velike količine podataka u zdravstvenoj zaštiti, njoj srodnim kontekstima ali i u mnogim ostalima domenima savremenog društva. Kao deo nacionalne strategije za usvajanje analitičke platforme za obradu velike količine podataka u zdravstvu, Češki institut za zdravstvene informacione sisteme i statistiku uskladio je svoju strategiju sa Evropskom unijom na državnom nivou Češke Republike, kao zemlje članica Evropske unije. Sa preko 100 kompleksnih zahteva, između ostalog bili su definisani i zahtevi za uključivanje daljih kriterijuma u pogledu performansi, isplativosti, robusnosti i toleranciji grešaka, a u skladu sa zakonskom regulativom Češke Republike. Takvo rešenje, sistem ili platforma, koja radi na softveru otvorenog koda (engl. Open Source) zasnovanom na Linuxu, ali i Talend Open Studio,

Python, R, Java, Scala okruženju, moralo je biti sposobno da postigne konkurentan i iznad očekivanog praga rezultate u vezi sa procenom ukupne performanse sistema, zasnovanih na odluci TPC-H industrijskog standarda za test performansi, kao merilo podrške.

8.1 Okruženje predložene arhitekture sistema visokih performansi

Opisana platforma u ovom doktorskom radu, predstavlja momenat, koji je premašen očekivanim radom na TPC-H referentnim testovima specifičnim za oblast zdravstva. Platforma i njena kontrola, upravljanje istom su na kraju celog procesa bili implementirani kao informacioni sistem u okviru IHIS. Time je, između ostalog došlo do objedinjena izolovanih zdravstvenih sistema u jedan eSistem. Pored demonstriranih testova i testova performansi u stvarnom životu, trenutni sistem kao takav ima veliki potencijal da unapredi nacionalnu zdravstvenu zaštitu u Češkoj Republici. Pored toga ima ambicije da ispuni očekivanja koja se trenutno razvijaju na nivou Češke Republike. Proizvedeni sistem, zasnovan na Vertica softveru za upravljanje analitičkom bazom podataka je otporan na budućnost u smislu obrade toka i velike količine, skalabilnosti (zasnovanoj na produkcionom hardveru) i toleranciji grešaka (npr. ispadanjem čvora ili čvorova klastera ne bi izazvalo gubitak podataka). Korišćenjem uobičajenog hardvera, horizontalni testovi skalabilnosti pokazuju poboljšanje performansi od preko 25% sa povećanjem broja čvorova klastera sa tri na pet čvorova. Na osnovu predloženih rezultata, zaključili smo da poboljšanje performansi klastera za obradu velike količine podataka nema linearnu osobinu porasta performansi ali istovremeno da postoji direktna zavisnost u odnosu na broj čvorova u klasteru predloženog rešenja tj. njegove arhitekture.

Trenutno, proizvedeni zdravstveni sistem je fizički izolovan od Internet infrastrukture tako što je instaliran u lokalnoj mreži instituta IHIS, unutar nacionalnih geografskih granica. Stoga, smatra se visoko bezbednim sistemom, istovremeno podržavajući industrijske standarde u vezi sa bezbednošću podataka i protokolima. Ovakav sistem (engl. BDA Healthcare eSistem) podržava niz softvera otvorenog koda, uključujući različite Linux distribucije sa sve većim brojem biblioteka za mašinsko učenje i integraciju komercijalnih alata kao što je npr. Tableau. U svetlu nedavne epidemije koronavirusa (COVID-19), predstavljeni sistem čini jedan od ključnih sistema u okviru Češke Republike, koji između ostalog, pruža vizuelizaciju geografskih podataka Češke Republike u roku od nekoliko malo sekundi i može da razmenjuje podatke sa drugim zdravstvenim eSistemima Češke Republike. Pored integracije podataka,

funkcija geo-mapiranja obezbeđuje praćenje pandemije/epidemije skoro u realnom vremenu, praćenje širenja epidemije i vizuelizaciju podataka o rizicima.

Ova doktorska disertacija može služiti ka osnova za transformaciju drugih zdravstvenih sistema, osiguravajućih firmi i sl. Ukoliko ovaj opisani princip generalizujemo na podignemo na viši nivo uopštenosti transformacije ili postupaka za izgradnju klaster sistema, on se može upotrebiti u bilo kome domenu razvoja savremenog društva. Pored toga, hteo bih da naglasim važnost osobine ovakvog rešenja u smislu njegove skalabilnosti tokom povećanja količine podataka i performansi, smeštaj algoritama za mašinsko učenje u bliskoj budućnosti i analitičkih alata, bezbednosnih i strateških za planiranje zdravstvene zaštite.

NoSQL baza podataka Vertica, izgrađena na klaster principu, ključna je komponenta opisanog rešenja. Ona je bila praktično upotrebljena kao BDA platforma a može biti implementirana u raznim okruženjima. Radi u okruženjima kao što su Amazon, Microsoft Azure, Google i VMware oblaci, pružajući korisnicima agilnost i mogućnosti za brzo uspostavljanje, prilagođavanje i integraciju raznih softverskih alata. Vertica omogućava prelazak skladišta podataka u okruženju za računarstvo u oblaku i lokalno smeštanje podataka, pružajući fleksibilnost za početak maloj količini podataka, kao i njihov rast zajedno sa poslovnim zahtevima kupaca. U ovom slučaju, naš klijent (IHIS) postavio je uslove za implementaciju predloženog rešenja po „on-premise“ principu. Rešenje je imalo za cilj da bude fizički izolovano od Interneta i nije bilo moguće predložiti rešenje zasnovano na računarstvu za obradu u oblaku (cloud) principu. Bez obzira na to, Vertica pruža sveobuhvatnu sigurnost uz podršku za standardne protokole u industriji, tako da verujemo da će se budućnost infrastrukture razvijati kao multi-cloud i hibridno rešenje, tj. kombinacija lokalnih i okruženja za računarstvo u oblaku. Takvi pristupi analitici podataka i upravljanju nisu predviđeni da budu ograničeni samo na jednu vrstu okruženja. Na primer, Vertica je objavila dostupnost Eon režima za Pure Storage (https://www.vertica.com/purestorage/) kao prvo industrijsko analitičko rešenje baze podataka sa odvajanjem računarske i memorijske arhitekture za lokalnu raspodela radnog opterećenja.

Druge dostupne skalabilne tehnologije u okviru obrade velike količine podataka [69–73] uključuju radno okruženje kao što je Hadoop. Predstavlja ekosistem otvorenog koda (sa vlasničkim sistemom datoteka HDFS). Njegova ključna komponenta MapReduce predstavlja programski okvir koji je zasnovan na Java programskom jeziku i određena je za skladištenje i grupnu obradu velike količine podataka. Apache Spark [74] je takođe dizajniran da se dobro

uklapa u ekosistem za obradu velike količine podataka skoro u realnom vremenu. Apache Spark je, na primer, poznat po čuvanju velike količina podataka (engl. Resilient Distributed Data) u memoriji i koji pruža bolje performanse klaster sistema od Hadoop[48] (u redosledu od desetina do sto puta). Međutim, Apache Spark i njegov računarski mehanizam za obradu podataka u memoriji ne obavlja skladištenje podataka po principu „ključ/vrednost“ kao što to radi Hadoop na HDFS ili NoSQL [75, 76] bazama podataka u okviru svog okvira. Apache Spark i NoSQL baze podataka su često povezane u jedan ekosistem na vrhu Hadoop-a instalacija. Uzimajući u obzir osobine Apache Spark i Hadoop ekosisteme, ne smemo zaboraviti da postoje troškovi kašnjenja prilikom obrade podataka. Štaviše, takvi ekosistemi zahtevaju dodatne administrativne napore, posebno u slučajevima odvojenih klastera i dupliranja podataka u smislu njihove upotrebe, administracija i produkcije.

Kao deo predstavljenog rešenja, potrebno je naznačiti važnost proizvoda otvorenog koda Talend Open Studio (verzija 6.4). On je korišćen sa namerom integracije podataka, izdvajanje-prenos-i-učitavanje (ETL) u različite izvore podataka (uključujući sisteme datoteka, Hadoop, NoSQL, RDBM) na način obrade u serijama ili u realnom vremenu. Preporučeni operativni sistem za Vertica BDA platformu je Linux CentOS 7.3. Vertica takođe ima podršku za druge operativne sisteme zasnovane na Linuxu, kao što su (po redosledu preferencija autora): Red Hat Enterprise Linux (RHEL) 7.3, Oracle Enterprise Linux (OEL) 7.3, SUSE 12 SP2, Debian 8.5 i Ubuntu 14.04 LTS. Za naš eSistem implementiran u prostorijama IHIS-a, dodatno smo instalirali open-source softver Nagios Core (verzija 4.1) za potrebe mrežne infrastrukture i praćenja klastera.

Upotrebom gore navedenih novih komponenti i njihovim uključenjem u klaster rešenje, možemo doći do visoko specijalizovanih sistema, koji se mogu upotrebiti generalno u bilo kom domenu, kao što je npr. transport, zdravstvo, proizvodnja, automobilska industrija, energetika, finansijski i bankarski sektor i sl.

8.2 Preporuke prilikom izgradnje i projektovanja klaster sistema

Prilikom izgradnje i projektovanja klaster sistema, potrebno je pretpostaviti da tokom razvoja celokupnog sistema i rešenja koje je na njemu izgrađeno, vrlo često u budućnosti, dolazi do postavljanja daljih zahteva koji se tiču performansi sistema ali i optimizacije troškova koji se vezuju za njegovu upotrebu. Zbog toga prilikom izgradnje klaster sistema, potrebno je uzeti u obzir i dati odgovore na sledeća pitanja:

1. Kako povećati performanse predloženog sistema?

2. Kako postići sistem sa optimalnim troškovima prilikom njegove izgradnje ali i produkcione upotrebe?

BDA rešenje koje je ovde prikazano u vremenu premašuje očekivani rad na TPC-H referentnim testovima specifičnim za zdravstvenu zaštitu. BDA rešenje i njegova kontrola je prebačen u IHIS, koji je u proteklih sedam meseci objedinio izolovane zdravstvene sisteme u jedan sistem. Pored demonstriranih testova i performansi u stvarnom životu, aktuelni sistem ima veliki potencijal za unapređenje nacionalne zdravstvene zaštite u Češkoj Republici, kao i za smeštaj evoluirajuća očekivanja i buduće potrebe za podacima. Proizveden sistem baziran na Vertica analitičkoj bazi podataka softver za upravljanje je spreman za budućnost u smislu obrade toka i velike količine podataka, skalabilnosti (hardverski i softverski deo) i toleranciji grešaka (npr. isključivanje čvorova klastera ne bi prouzrokovao gubitak podataka). Horizontalni testovi skalabilnosti korišćenjem produkcionog hardvera pokazuju poboljšanje performansi od preko 25% povećanjem broja čvorova klastera sa tri na pet, obezbeđivanje dovoljnih dokaza o skaliranom dizajnu zasnovanom na isplativom produkcionom hardveru.

Predstavljena arhitektura sistema pokazuje kako je moguće postići očekivanu arhitekturu sistema tj. rešenja sa sledećim karakteristikama:

1. Horizontalni skalabilni sistem.

2. Približna skoro linearna skalabilnost.

3. Rezultati visokih performansi zasnovani na produkcionom hardveru i arhitekturi zasnovanoj na klasterima.

4. Visoko bezbedna i pouzdana platforma sa performansama visoke propusnosti.

5. Usklađenost sa zahtevima standarda TPC-H.

6. Podržano skladištenje, razmena i obrada velike količine podataka (skalabilno u smislu TB do desetine PB).

7. Slojevite operacije razdvojene na (1) Sloj upravljanja podacima; (2) Čuvanje i obrada podataka; (3) Sloj vizuelizacije podataka.

8. Kompletna integracija analitičke podrške u bazi podataka, kao i kreiranje i testiranje prediktivnih analitičkih modela (uključujući podršku mašinskog učenja za regresivne modele, stabla klasifikacije i regresije, analizu vremenskih serija, analizu sekvence).

9. Mogućnost korišćenja u bilo kom domenu današnjeg digitalnog sveta i moderne civilizacije.

8.3 Moguća unapređenja predložene arhitekture sistema visokih performansi

Što se tiče planova za dalje unapređenje i razvoj našeg rešenja u 2023. godini, razmatramo dalja poboljšanja za nacionalnu zdravstvenu zaštitu i zaštitu privatnosti putem obrade podataka sa zdravstvenih IoT uređaja i mobilne aplikacije (uključujući uređaje koji se mogu nositi kao što su senzori pametnih satova). Realizacijom više vrsta testova u razvojnom okruženju proširenjem komponenti i upotrebom drugih platforma, kao što je broker otvorenog koda za prenos podataka sa IoT uređaja korišćenjem MQTT protokola (engl. Mosquitto) [77, 78]). Pribavljeni test podaci sa IoT uređaja se prenose kao strim podaci preko MQTT Mosquitto brokera (https://mosquitto.org), transformisani pomoću Apache Spark-a (https://spark.apache.org) i sačuvani za buduće svrhe rada sa podacima u Hadoop-u. Sa ovog sloja podaci se dalje obrađuju u Vertica NoSQL klasteru baze podataka za smeštaj i analitičku obradu podataka. Za potrebe upravljanja IoT platformom, koristimo radno okruženje Node.js (https://nodejs.org) za izgradnju brze i skalabilne mrežne aplikacije kroz radno okruženje za aplikacioni deo, Angular platformu (https://angular.io), koja bi se koristila za razvoj mobilne i desktop aplikacije.

9. Zaključak

Ova doktorska disertacija odnosi se na projektovanje uopštene arhitekture sistema visokih performansi i sistema za generalnu obradu podataka na Big Data klasterima. Predložena i obrađena arhitektura sistema visokih performansi omogućila je efikasnu akviziciju podataka, njihovo optimalno smeštanja kao i obradu velike količine podataka uz njihovo vizualno i alfanumeričko predstavljanje. Osnovna ideja za postizanje ovakvog rešenja bila je u postizanju konkretnog, univerzalnog sistema koji bi mogao biti upotrebljen u različitim domenima savremenog društva uz njegovu optimalnu ekonomsku opravdanost. Poseban osvrt prilikom izgradnje uopštene arhitekture sistema visokih performansi bio je posvećen njegovom performansama na osnovu testa performansi koji predstavlja međunarodni standard TPC-H [20]. U ovoj doktorskoj disertaciji prikazano je praktično rešenje tj. arhitektura sistema visokih performansi za obradu velike količine podataka u zdravstvu u okviru Češke Republike [28].

Predložena arhitektura obezbedila je efikasno planiranje sistema, njegovo upotrebu, ispunjene promena koje nastaju prilikom upotrebe klaster rešenja za obradu velike količine podataka. Ova univerzalna arhitektura pružila je mogućnost podešavanja performansi celokupnog rešenja u zavisnosti od količine podataka, njihove bezbednosti, stabilnosti klastera u toku izvršavanja obrade velike količine podataka. Predložena platforma tj. njena arhitektura kao takva omogućila je integraciju podataka, njihovo efikasno smeštanje i obezbeđenje dostupnosti podataka, upravljanje kvalitetom podataka, meta podacima, ad-hoc analitičke operacije ali i vizualizaciju uz obezbeđenje podataka na svim nivoima na gde su oni prisutni.

Kao jedan od važnih doprinosa u okviru predloženih rezultata je princip poboljšanja performansi arhitekture klastera predloženog rešenja. Tom prilikom došlo je do povećanja broja čvorova klastera, kao i njihova direktna zavisnost u smislu povećanja performansi, koja je bila skoro linearna.

9.1 Pravci daljeg istraživanja

Ova doktorska disertacija daje mogućnost sledećih pravaca istraživanja:

1. Koja je optimalna arhitektura za postizanje sistema za realnu obradu podataka?
2. Kako unaprediti arhitekturu i time omogućiti obradu stream podataka u realnom vremenu kao što su npr. IoT podaci ili podaci senzorskog tipa?

Sledeći koraci u budućem razvoju predstavljene platforme u oblasti zdravstva obuhvataju: **(1)** proširenja platforme u smislu podrške streaming podataka (medicinske prirode) IoT-a i mobilnih aplikacija, tako da postojeće rešenje ostaje nepromenjeno u smislu arhitekture; **(2)** podrška odlukama zasnovanim na podacima tokom događaja sa velikim prometom (Data Driven Decision Making - DDDM); **(3)** tekuće horizontalno skaliranje i povećanje sa 100 TB na 1 PB obrade podataka; **(4)** novi pristupi čišćenju, skladištenju i preuzimanju podataka sa minimalnim kašnjenjem; **(5)** integracija sa drugim nacionalnim registrima (npr. za upravljanje ili olakšavanje logistike kod distribucije lekova); i **(6)** strateško planiranje korišćenjem zdravstvenih podataka.

Definisati okvir za poboljšanje ne samo predloga unapređenja arhitekture, već i na smanjenju krivulje za prikupljanje kvalitetnijih podataka, izgradnju ML modela i kreiranje aplikacija na tim modelima, uključujući moderne DevOps procedure. Dostizanje AI/ML potencijala u preduzeću u okviru autonomne inteligencije, tj. autonomnog vođenog vozila i autonomnih mobilnih robota ili potpomognute inteligencije, tj. praćenje zdravlja i identifikacija rizika. Hibridna arhitektura predlaže velike podatke i kako će budući veliki podaci biti raspoređeni i/ili podeljeni u ovom okruženju. Uključujući upravljanje i nove trendove virtualizacije na licu mesta ili u okruženju za računarstvo u oblaku (tj. Docker, Kubernetes kao deo okruženja za računarstvo u oblaku ili lokalne strategije).

Dalje istraživanje može da se kreće i u pravcu proširenja dela za akviziciju strim (engl. stream) podataka koji bi mogli biti prikupljani u realnom vremenu. U tom slučaju je potrebno proveriti i ostale delove arhitekture, prvenstveno u delu koji se tiče smeštanja ali i obrade podataka.

Literatura

[1] G. Dicuonzo, G. Galeone, M. Shini, and A. Massari, "Towards the Use of Big Data in Healthcare: A Literature Review," *Healthcare,* vol. 10, no. 7, p. 1232, 2022. [Online]. Available: https://www.mdpi.com/2227-9032/10/7/1232.

[2] F. Xhafa, A. Aly, and A. A. Juan, "Allocation of applications to Fog resources via semantic clustering techniques: with scenarios from intelligent transportation systems," *Computing,* vol. 103, no. 3, pp. 361-378, 2021, doi: https://doi.org/10.1007/s00607-020-00867-w.

[3] D. Broneske, A. Drewes, B. Gurumurthy, I. Hajjar, T. Pionteck, and G. Saake, "In-Depth Analysis of OLAP Query Performance on Heterogeneous Hardware," *Datenbank-Spektrum,* vol. 21, no. 2, pp. 133-143, 2021.

[4] J. Huang, H. Obracht-Prondzynska, D. Kamrowska-Zaluska, Y. Sun, and L. Li, "The image of the City on social media: A comparative study using "Big Data" and "Small Data" methods in the Tri-City Region in Poland," *Landscape and Urban Planning,* vol. 206, p. 103977, 2021, doi: https://doi.org/10.1016/j.landurbplan.2020.103977.

[5] H. Zhang, Z. Zang, H. Zhu, M. I. Uddin, and M. A. Amin, "Big data-assisted social media analytics for business model for business decision making system competitive analysis," *Information Processing & Management,* vol. 59, no. 1, p. 102762, 2022, doi: https://doi.org/10.1016/j.ipm.2021.102762.

[6] M. Štufi and B. Bačić, "Designing a Real-Time IoT Data Streaming Testbed for Horizontally Scalable Analytical Platforms: Czech Post Case Study," *Proceedings of the 11th International Conference on Sensor Networks - SENSORNETS,* 2021, doi: https://doi.org/10.5220/0010788300003118.

[7] W. Li *et al.*, "A comprehensive survey on machine learning-based big data analytics for IoT-enabled smart healthcare system," *Mobile Networks and Applications,* vol. 26, no. 1, pp. 234-252, 2021.

[8] C. Petrov. "25+ Impressive Big Data Statistics for 2021." https://techjury.net/blog/big-data-statistics/#gref (accessed 08.10.2022).

[9] J. C. Couto, J. Damasio, R. Bordini, and D. Ruiz, "New trends in big data profiling," in *Science and Information Conference*, 2022: Springer, pp. 808-825.

[10] T. Erl, W. Khattak, and P. Buhler, *Big Data Fundamentals: Concepts, Drivers & Techniques*. Prentice Hall Press, 2016.

[11] R. Kulkarni. "Big Data Goes Big." https://www.forbes.com/sites/rkulkarni/2019/02/07/big-data-goes-big/?sh=36b1118320d7 (accessed 08.10.2022).

[12] M. Babar and F. Arif, "Real-time data processing scheme using big data analytics in internet of things based smart transportation environment," *Journal of Ambient Intelligence and Humanized Computing,* vol. 10, no. 10, pp. 4167-4177, 2019.

[13] M. Mohsin. "10 Google Search Statistics You Need To Know in 2022 " https://www.oberlo.com/blog/google-search-statistics (accessed 10 March, 2022).

[14] K. Santosh and S. Ghosh, "Covid-19 imaging tools: How big data is big?," *Journal of Medical Systems,* vol. 45, no. 7, pp. 1-8, 2021.

[15] J. R. Montoya-Torres, S. Moreno, W. J. Guerrero, and G. Mejía, "Big Data Analytics and Intelligent Transportation Systems∗∗This work was supported by Universidad de La Sabana, Colombia (grant INGPHD-10-2019)," *IFAC-PapersOnLine,* vol. 54, no. 2, pp. 216-220, 2021, doi: https://doi.org/10.1016/j.ifacol.2021.06.025.

[16] M. Naeem *et al.*, "Trends and future perspective challenges in big data," in *Advances in intelligent data analysis and applications*: Springer, 2022, pp. 309-325.

[17] T. P. Smith, *How Big is Big and How Small is Small: The Sizes of Everything and Why*. OUP Oxford, 2013.

[18] T. R. Rao, P. Mitra, R. Bhatt, and A. Goswami, "The big data system, components, tools, and technologies: a survey," *Knowledge and Information Systems,* vol. 60, no. 3, pp. 1165-1245, 2019.

[19] W. Wang and C. Lu, "Visualization analysis of big data research based on Citespace," *Soft Computing,* vol. 24, no. 11, pp. 8173-8186, 2020.

[20] *TPC Benchmark™ standard specification revision 2.18.0*, Transaction Processing Performance Council, 2018. [Online]. Available: http://www.tpc.org/tpc_documents_current_versions/pdf/tpc-h_v2.18.0.pdf

[21] P. Boncz, T. Neumann, and O. Erling, "TPC-H analyzed: Hidden messages and lessons learned from an influential benchmark," in *Technology Conference on Performance Evaluation and Benchmarking*, 2013: Springer, pp. 61-76.

[22] V. Persico, A. Pescapé, A. Picariello, and G. Sperlí, "Benchmarking big data architectures for social networks data processing using public cloud platforms," *Future Generation Computer Systems,* vol. 89, pp. 98-109, 2018/12/01/ 2018, doi: https://doi.org/10.1016/j.future.2018.05.068.

[23] G. C. Deka, "A survey of cloud database systems," *IT Professional,* vol. 16, no. 2, pp. 50-57, 2014.

[24] D. Abadi, P. Boncz, S. Harizopoulos, S. Idreos, and S. Madden, "The design and implementation of modern column-oriented database systems," *Foundations Trends® in Databases,* vol. 5, no. 3, pp. 197-280, 2013.

[25] X. Amatriain, "Big & personal: Data and models behind Netflix recommendations," in *Proceedings of the 2nd international workshop on big data, streams and heterogeneous source mining: Algorithms, systems, programming models and applications*, 2013: ACM New York, NY, USA, 2013, pp. 1-6, doi: https://doi.org/10.1145/2501221.2501222.

[26] R. Assunção and K. Pelechrinis, "Sports analytics in the era of big data: Moving toward the next frontier," *Big Data,* vol. 6, no. 4, 2018, doi: https://doi.org/10.1089/big.2018.29028.edi.

[27] O. Badawi *et al.*, "Making big data useful for health care: A summary of the inaugural MIT critical data conference," *JMIR Med Inform,* vol. 2, no. 2, p. e22, 2014 2014, doi: https://doi.org/10.2196/medinform.3447.

[28] M. Štufi, B. Bačić, and L. Stoimenov, "Big data analytics and processing platform in Czech Republic healthcare," *Applied Sciences,* vol. 10, no. 5, p. 1705, 2020, doi: https://doi.org/10.3390/app10051705.

[29] H. Chen, R. H. Chiang, and V. C. Storey, "Business intelligence and analytics: From big data to big impact," *J MIS quarterly,* vol. 36, pp. 1165-1188, 2012.

[30] P. Cohen, R. Hahn, J. Hall, S. Levitt, and R. Metcalfe, "Using big data to estimate consumer surplus: The case of Uber," National Bureau of Economic Research, Working Paper 22627, 2016. [Online]. Available: http://www.nber.org/papers/w22627

[31] European Commission. "Big data: Digital single market policy." https://ec.europa.eu/digital-single-market/en/policies/big-data (accessed 08.10.2022).

[32] R. D. Raut, V. S. Yadav, N. Cheikhrouhou, V. S. Narwane, and B. E. Narkhede, "Big data analytics: Implementation challenges in Indian manufacturing supply chains," *Computers in Industry,* vol. 125, p. 103368, 2021, doi: https://doi.org/10.1016/j.compind.2020.103368.

[33] A. Arooj, M. S. Farooq, A. Akram, R. Iqbal, A. Sharma, and G. Dhiman, "Big data processing and analysis in internet of vehicles: architecture, taxonomy, and open research challenges," *Archives of Computational Methods in Engineering,* pp. 1-37, 2021.

[34] A. Luckow, G. Chantzialexiou, and S. Jha, "Pilot-streaming: A stream processing framework for high-performance computing," *2018 IEEE 14th International Conference on e-Science,* pp. 177-188, 2018, doi: 0.1109/eScience.2018.00033.

[35] A. Ed-daoudy and K. Maalmi, "A new Internet of Things architecture for real-time prediction of various diseases using machine learning on big data environment," *Journal of Big Data,* vol. 6, no. 1, p. 104, 2019, doi: 10.1186/s40537-019-0271-7.

[36] B. Socolow, "Wearables technology data use in professional sports," *World Sports Law Report,* vol. 14, no. 4, pp. 12-15, 2016.

[37] S. Vijayarani and S. Sudha, "An efficient clustering algorithm for predicting diseases from hemogram blood test samples," *Indian Journal of Science and Technology,* vol. 8, no. 17, p. 1, 2015.

[38] J. Warren and N. Marz, *Big Data: Principles and best practices of scalable realtime data systems*. Simon and Schuster, 2015.

[39] C. Niyizamwiyitira and L. Lundberg, "Performance evaluation of SQL and NoSQL database management systems in a cluster," *IJDMS,* vol. 9, no. 6, pp. 1-24, 2017.

[40] A. Pavlo *et al.*, "A comparison of approaches to large-scale data analysis," in *Proceedings of the 2009 ACM SIGMOD International Conference on Management of data*, 2009: ACM, pp. 165-178.

[41] A. Gupta, A. Deokar, L. Iyer, R. Sharda, and D. Schrader, "Big data & analytics for societal impact: Recent research and trends," *Information Systems Frontiers,* vol. 20, no. 2, pp. 185-194, 2018.

[42] L. Oukhouya, B. Er-raha, H. Asri, and N. Laaz, "A Proposed Big Data Architecture Using Data Lakes for Education Systems," in *International Conference on Networking, Intelligent Systems and Security*, 2023: Springer, pp. 53-62.

[43] M. I. Razzak, M. Imran, and G. Xu, "Big data analytics for preventive medicine," *Neural Computing and Applications,* vol. 32, no. 9, pp. 4417-4451, 2020, doi: https://doi.org/10.1007/s00521-019-04095-y.

[44] R. Cortés, X. Bonnaire, O. Marin, and P. Sens, "Stream processing of healthcare sensor data: studying user traces to identify challenges from a big data perspective,"

Procedia Computer Science, vol. 52, pp. 1004-1009, 2015, doi: https://doi.org/10.1016/j.procs.2015.05.093.

[45] H. Koziolek, S. Grüner, and J. Rückert, "A comparison of MQTT brokers for distributed IoT edge computing," 2020.

[46] A. Labhansh, N. Parth, T. Sandeep, and G. Vasundhra, "Business intelligence tools for big data," *JBAR,* vol. 3, no. 6, pp. 505-509, 2016.

[47] Z. Dafir, Y. Lamari, and S. C. Slaoui, "A survey on parallel clustering algorithms for big data," *Artificial Intelligence Review,* vol. 54, no. 4, pp. 2411-2443, 2021.

[48] "Apache Hadoop." The Apache Software Foundation. https://hadoop.apache.org/ (accessed 09.10.2022).

[49] A. Abouzeid, K. Bajda-Pawlikowski, D. J. Abadi, A. Rasin, and A. Silberschatz, "HadoopDB: An Architectural Hybrid of MapReduce and DBMS Technologies for Analytical Workloads," *Proc. VLDB Endow.,* vol. 2, pp. 922-933, 2009.

[50] D. R. Olson, K. J. Konty, M. Paladini, C. Viboud, and L. Simonsen, "Reassessing Google flu trends data for detection of seasonal and pandemic influenza: A comparative epidemiological study at three geographic scales," *PLoS computational biology,* vol. 9, no. 10, p. e1003256, 2013.

[51] A. E. Azzaoui, S. K. Singh, and J. H. Park, "SNS big data analysis framework for COVID-19 outbreak prediction in smart healthy city," *Sustainable Cities and Society,* vol. 71, p. 102993, 2021.

[52] N. Ihde *et al.*, "A Survey of Big Data, High Performance Computing, and Machine Learning Benchmarks," in *Technology Conference on Performance Evaluation and Benchmarking*, 2021: Springer, pp. 98-118.

[53] A. Ross. "Gartner: Top 10 data and analytics technology trends for 2019." Information Age,. https://www.information-age.com/gartner-data-and-analytics-technology-trends-123479234/ (accessed 08.10.2022).

[54] D. Laney, "3D data management: Controlling data volume, velocity and variety," *META group research note,* vol. 6, no. 70, p. 1, 2001.

[55] B. Geerdink, "A Reference Architecture for Big Data Solutions," 2017.

[56] T. Sajana, C. S. Rani, and V. Narayana, "A survey on clustering techniques for big data mining," *Indian journal of Science and Technology,* vol. 9, no. 3, pp. 1-12, 2016.

[57] M. Zaharia, "An Architecture for Fast and General Data Processing on Large Clusters," 2014.

[58] J. M. Hellerstein, M. Stonebraker, and J. Hamilton, "Architecture of a database system," *Foundations Trends® in Databases,* vol. 1, no. 2, pp. 141-259, 2007.

[59] M. Focus. "What is Performance Testing?" Micro Focus. https://www.microfocus.com/en-us/what-is/performance-testing (accessed 19, June, 2022).

[60] M. Thieme, "Performance Testing Methodology Standardization," pp. 1069-1074, 2022, doi: https://doi.org/10.1007/978-0-387-73003-5_233.

[61] R. Nambiar, N. Wakou, F. Carman, and M. Majdalany, "Transaction processing performance council (TPC): State of the council 2010," in *Technology Conference on Performance Evaluation and Benchmarking*, 2010: Springer, pp. 1-9.

[62] A. Thusoo *et al.*, "Data warehousing and analytics infrastructure at Facebook," in *Proceedings of the 2010 ACM SIGMOD International Conference on Management of data*, 2010: ACM, pp. 1013-1020.

[63] J. Ginsberg, M. H. Mohebbi, R. S. Patel, L. Brammer, M. S. Smolinski, and L. Brilliant, "Detecting influenza epidemics using search engine query data," *Nature,* vol. 457, no. 7232, p. 1012, 2009.

[64] M. Štufi and B. Bačić, "Designing a Real-time IoT Data Streaming Testbed for Horizontally Scalable Analytical Platforms: Czech Post Case Study," presented at the Proceedings of the 11th International Conference on Sensor Networks - SENSORNETS, Wien, 2022, 978-989-758-551-7. [Online]. Available: https://www.scitepress.org/Link.aspx?doi=10.5220/0010788300003118.

[65] S. Borisova and A. Zein, "Disadvantages of Relational DBMS for Big Data Processing," in *International Conference for Information Systems and Design*, 2021: Springer, pp. 279-290.

[66] "The Global Big Data Market is expected to grow by $ 247.30 bn during 2021-2025, progressing at a CAGR of almost 18% during the forecast period." https://www.globenewswire.com/news-release/2021/07/05/2257756/0/en/The-Global-Big-Data-Market-is-expected-to-grow-by-247-30-bn-during-2021-2025-progressing-at-a-CAGR-of-almost-18-during-the-forecast-period.html (accessed 11 March, 2022).

[67] M. Štufi, B. Bačić, and L. Stoimenov, "Big data architecture in Czech Republic healthcare service: requirements, TPC-H benchmarks and Vertica," *ArXiv,* vol. abs/2001.01192, 2020.

[68] C. Bear, A. Lamb, and N. Tran, "The Vertica database: SQL RDBMS for managing big data," in *Proceedings of the 2012 workshop on Management of big data systems*, 2012: ACM, pp. 37-38.

[69] A. Lamb *et al.*, "The Vertica analytic database: C-store 7 years later," *Proceedings of the VLDB Endowment,* vol. 5, no. 12, pp. 1790-1801, 2012.

[70] S. Srivastava, "Top 10 countries & regions leading the big data adoption in 2019," in *Analytics Insight* ed, 2019.

[71] X. Liu and P. S. Nielsen, "A hybrid ICT-solution for smart meter data analytics," *Energy,* vol. 115, pp. 1710-1722, 2016.

[72] M. Stonebraker *et al.*, "C-store: A column-oriented DBMS," in *Proceedings of the 31st international conference on Very large data bases*, 2005: VLDB Endowment, pp. 553-564.

[73] C. J. Tauro, S. Aravindh, and A. Shreeharsha, "Comparative study of the new generation, agile, scalable, high performance NoSQL databases," *Int J Comput Appl,* vol. 48, no. 20, pp. 1-4, 2012.

[74] "Apache Spark." The Apache Software Foundation. https://spark.apache.org/ (accessed 08.10.2022).

[75] M. Stonebraker, "SQL databases v. NoSQL databases," *Commun ACM,* vol. 53, no. 4, pp. 10-11, 2010.

[76] A. Moniruzzaman and S. A. Hossain, "NoSQL database: New era of databases for big data analytics-classification, characteristics and comparison," *IJDTA,* vol. Vol. 6, No. 4. 2013, 2013.

[77] *MQTT Version 5.0*, OASIS, 2019. [Online]. Available: https://mqtt.org/mqtt-specification/

[78] Solutia. "mqtt-performance." https://github.com/stufim/mqtt-performance (accessed 08.10.2022).

Dodatak A – Tabela za procenu predloženog rešenja

Tabela za procenu predloženog rešenja ili tabela za ispunjenje tehničkih uslova, koristi se radi izbora najoptimalnijeg rešenja. Razlikujemo nekoliko obaveznih koraka koje je potrebno obezbediti. Ova tabela odnosi se na formalno definisanje tehničkih uslova, koje primarno služi tenderskoj komisiji za procenu ispunjenja i kvaliteta predloženog rešenja. U ovom slučaju u okviru dodataka (Dodatak A, Dodatak B, Dodatak C) radi se o prilagođenim tabelama, koje predstavljaju univerzalni postupak prilikom ocene ispunjenosti i kvaliteta predloženog rešenja a koje proizilaze iz tendera, kojem je predložena platforma bila podvrgnuta.

Prva se odnosi na kontrolu obaveznih tehničkih uslova koji se odnosi na sistemske zahteve, hardvera, softvera, parametara produkcije, postupaka za sigurnosno kopiranje i vraćanje podataka i svih ostalih funkcija definisanih za sistema kao takav. Dalje procena kvaliteta, tj. optimalnosti sistema se odnosi na bonus funkcije. Sledeća tabela se odnosi na definisanje tehničkih kriterijuma i njihove ispunjenosti u relaciji za limitom određene cene celokupnog predloženog rešenja. U tabeli za ocenu predloženog rešenja proveravamo ispunjenost tehničkog rešenja, usklađenosti parametara faze 1, usklađenja graničnih vrednosti faze 2 u odnosu sa cenom i bonus funkcijama. U okviru sekcije analiziramo cenu predloženog rešenja sa cenom najboljeg rešenja, kako bi smo došli do zaključka o kvalitetu rešenja. Poslednja sekcija omogućava analizu bonus parametara predloženog rešenja na osnovu težine koja je unapred određena za svaku od njih pojedinačno.

Obavezni tehnički uslovi

SRS ref.	Kratak opis uslova Ispunjenost uslova (0/1)	Ispunjenost uslova [0/1]
1	Zahtevi za celo rešenje	
1.1	Osnovni sistemski zahtevi	
1.1.a	Arhitektura rešenja barem za DS izgrađen (ili prenosiv) na MPP	
1.1.b	Potpuna garancija i post-garantni servis na licu mesta za hardver i softver tokom 3 godine	

SRS ref.	Kratak opis uslova Ispunjenost uslova (0/1)	Ispunjenost uslova [0/1]
1.1.b	Hardverske ispravke i softverske ispravke grešaka do kraja sledećeg radnog dana	
1.1.b	Kritične softverske greške se ispravljaju odmah ili najkasnije u roku od 30 dana	
1.1.c	Obuka najmanje u obimu navedenom u tabeli kada su ispunjeni zahtevi	
1.1.d	Podrška barem u meri navedenoj u tabeli kada su ispunjeni zahtevi	
2	Hardver i operativni sistem	
2.a	Najmanje jedan server uključen u isporuku	
2.b	Kapacitet diska> = 3 TB korisničkih podataka u DB	
2.b	DB na nezavisnim diskovima sa faktorom replikacije > = 2	
2.c	10 Gbit Ethernet	
2.c	Fiber Channel 16Gb / InfiniBand	
2.d	Linux ili Windows Server OS uključujući alate i licence	
2.f	Sopstveni UPS sa dovoljnim kapacitetom	
2.f	Sopstveni orman klijenta ili ugradnja u klijentov orman	
2.g	Sopstveni alati za praćenje i kontrolu rada hardvera i operativnog sistema ili podrška za SNMP 0	
2.h	Najmanje 5 korisničkih naloga administratora, 5 korisničkih licenci	
3	Softver	
3.1	Opšti zahtevi za SV alate	
3.1.a	Instaliran kao lokalno rešenje na isporučenom hardveru u izolovanoj mreži klijenta	
3.1.a	Ugovor uključuje instalaciju, konfiguraciju i testiranje tokom testa prihvatanja	
3.1.b	Višekorisničko i paralelno okruženje aplikacija	

SRS ref.	Kratak opis uslova Ispunjenost uslova (0/1)	Ispunjenost uslova [0/1]
3.1.c	Aplikacije integrisane u jedan paket, međusobno povezane na nivou meta podataka i funkcije	
3.1.d	Podržava potrebne metode bezbednosti i zaštite podataka	
3.1.e	Alati za hot-backup na infrastrukturi klijenta (backup-to-disc)	
3.1.e	Alati za oporavak sa hladnom rezervnom kopijom	
3.1.f	Alati i procedure za kontrolu verzija za podatke i meta podatke	
3.1.g	Proširivost sistema dodatnim funkcijama, skriptovima i spoljnim alatima	
3.1.h	Lako definisanje razvojnog, testnog i proizvodnog okruženja	
3.1.i	Sopstveni alati za praćenje i kontrolu rada SV-a ili SNMP podrška	
3.1.j	Generisanje automatski konfigurativne dokumentacije iz svih komponenti	
3.1.k	Podrška za obradu podataka od 50/500/3000 GB u pojedinačnim okruženjima	
3.1.l	Parametri performansi ekvivalentni za svako okruženje	
3.1.m	Sistem će podneti tipično maksimalno opterećenje navedeno u zahtevima	
3.1.n	Dostupna je kompletna dokumentacija i korisnički forum	
3.1.o	Sve licence i registracije su prenosive na druge zaposlene	
3.1.o	Fiksne licence ili plutajuće licence sa lokalnim serverom licenci	
3.1.p	Komponente alata i licence nisu vremenski ograničene	
3.2	Integracija podataka (DI)	
3.2.a	Funkcije koje se mogu parametrizirati za manipulaciju podacima prema zahtevu	
3.2.b	Funkcije podataka koje se mogu koristiti u okviru navedenih funkcija transformacije	
3.2.c	Funkcije za kontrolu procesa obrade podataka	

SRS ref.	Kratak opis uslova Ispunjenost uslova (0/1)	Ispunjenost uslova [0/1]
3.2.d	Podrška za čitanje i pisanje navedenog DB	
3.2.e	Uvođenje hijerarhije procesa obrade podataka, ciklusi, paralelne i sekvencijalne niti	
3.2.f	Obrada zadataka kontrolisana meta podacima	
3.2.g	Alati za otklanjanje grešaka DI procesa	
3.2.h	Upravljanje verzijama procesa integracije uključujući poređenje	
3.2.i	Meta podaci DI procesa dostupni u komponenti MDM	
3.2.j	Jednostavan prenos između razvojnog, okruženja za testiranje i proizvodnog okruženja	
3.2.k	Vizuelizacija rezultujućeg procesa kao konfiguracionih dijagrama toka procesa i toka podataka	
3.2.m	Licenca za najmanje 4 korisnika ukupno, 2 istovremeno	
3.3	Priprema podataka za analitičke rezultate (AAP - Ad-hoc Analysis Preparation)	
3.3.a	Alat se koristi za brzu pripremu izlaza iz već obrađenih podataka	
3.3.b	Podržava brzi proces razvoja prototipa procesa	
3.3.c	Omogućava veoma brz izbor i transformaciju velikih količina primarnih podataka	
3.3.c	Nema potrebe da se ručno piše ad-hoc kod	
3.3.d	Pokretanje više niti i podrška za više korisnika	
3.3.e	Rezultujući proces je opisan meta podacima u komponenti MDM	
3.3.f	Vizuelizacija rezultujućeg procesa kao konfiguracionih dijagrama toka procesa i toka podataka 0	
3.3.h	Licenca za najmanje 4 korisnika ukupno, 2 istovremeno rade	
3.4	Skladištenje podataka (DS)	
3.4.a	Standardni SQL interfejs koji podržava funkcije prema SQL-u: 2003	
3.4.a	DB dostupan preko ODBC i/ili izvornih JDBC drajvera	

SRS ref.	Kratak opis uslova Ispunjenost uslova (0/1)	Ispunjenost uslova [0/1]
3.4.b	Deterministički rezultati upita i ispunjavanja ACID svojstava transakcije	
3.4.c	Skladištenje podataka optimizovano za složene upite preko dimenzionalne šeme	
3.4.d	Potpuno automatske metode skladištenja ili optimizacija ili na osnovu automatski generisanih preporuka	
3.4.e	Sačuvajte podatke sa faktorom replikacije ≥ 2	
3.4.f	Podržava veoma brzu obradu složenih ad-hoc upita nad dimenzionalnim podacima	
3.4.g	Podržava izračunavanje statističkih i analitičkih zadataka preko velikih podataka direktno u bazi podataka	
3.4.h	Podržava izračunavanje standardnih deskriptivnih statističkih metoda prema PMML odeljku Ugrađene funkcije	
3.4.i	Podržava standardni proračun	
3.4.j	Podržava paralelne zadatke za bodovanje zasnovane na bazi podataka u odnosu na unapred pripremljene modele (najmanje 6 od 8)	
3.4.k	DB podržava kreiranje pomenutih prediktivnih modela ili je dopunjen statističkim alatom za njihovo kreiranje	
3.4.l	Sadrži GUI klijent za upravljanje DB-om	
3.4.m	Sadrži GUI klijent za upravljanje modelima podataka preko ovog DB-a	
3.4.n	Svi meta podaci moraju biti dostupni MDM komponenti	
3.4.o	Primenjeni TPC-H benchmark ispunjava zahteve za trajanje pojedinačnih koraka	
3.4.k	Licenca za najmanje 6 korisnika koji rade direktno na DB, 3 rade istovremeno	
3.4.k	Licenca za najmanje 25 korisnika ukupno koji rade sa DB preko drugih komponenti sistema, 12 rade istovremeno	
3.4.r	Licenca za najmanje 4 korisnika u ukupno GUI za upravljanje modelom podataka, 2 istovremeno rade	

SRS ref.	Kratak opis uslova Ispunjenost uslova (0/1)	Ispunjenost uslova [0/1]
3.5	Upravljanje kvalitetom podataka (DQM)	
3.5.a.1	Pravila podrške za osnovnu proveru sintaksičkih podataka	
3.5.a.2	Pravila podrške za osnovnu proveru semantičkih podataka	
3.5.a.3	Podrška pravilima za statističke provere trendova i struktura podataka	
3.5.a.4	Podrška za pravila kontrole korisničkih podataka	
3.5.a.5	Podrška za složene analitičke modele za kontrolu podataka	
3.5.b	Podrška za izradu pravila, analizu podataka, otkrivanje grešaka, upozorenja i mehanizme ispravljanja	
3.5.c	Sveobuhvatna vizuelizacija i izveštavanje o toku i rezultatu kontrole kvaliteta	
3.5.d	Pravila validacije uneta u strukturiranom obliku, upotrebljiva i za korisnike koji nisu IT	
3.5.d	Pravila validacije se mogu sortirati, grupisati, pretraživati	
3.5.e	Pravila validacije se mogu pokrenuti preko korisnički definisanog skupa podataka	
3.5.f	Centralizovano upravljanje svim vrstama pravila validacije opisanih u obliku meta podataka	
3.5.i	Licenca za najmanje 4 korisnika ukupno, 2 koja rade istovremeno	
3.6	Upravljanje meta podacima (MDM)	
3.6.a	Podrška za upravljanje korisničkim, tehničkim i operativnim meta podacima	
3.6.b	Obrađuje meta podatke iz svih komponenti isporučenog sistema	
3.6.c	Podržava preuzimanje meta podataka za najmanje 4 od 5 navedenih tehnologija	
3.6.d	Meta podaci iz ovih eksternih izvora mogu se koristiti za definisanje vaših meta podataka	
3.6.e	Može da uporedi različite verzije meta podataka i prikaže razlike u jasnom formatu	

SRS ref.	Kratak opis uslova Ispunjenost uslova (0/1)	Ispunjenost uslova [0/1]
3.6.f	Omogućava upravljanje korisničkim meta podacima, uključujući rečnik pojmova i veze između termina	
3.6.g	Omogućava analizu uticaja i loze	
3.6.h	Licenca za najmanje 4 korisnika ukupno, 2 koja rade istovremeno	
3.7	Poslovna inteligencija (BI)	
3.7.a	Jasan i intuitivan korisnički interfejs za izveštavanje o unapred pripremljenim podacima, interfejs za prevlačenje i ispuštanje	
3.7.b	Učitavanje podataka iz skladišta podataka i dodatnih ad-hoc izvora, koristeći sandbox za privremene ulaze i izlaze	
3.7.c	Izlazi na mreži (html) i van mreže (posebno PDF, eventualno PPT koji se može uređivati)	
3.7.d	Rezultati u obliku tekstova, tabela i grafikona	
3.7.e	Podržava najmanje sledeće uobičajene tipove grafikona	
3.7.f	Osnovne funkcije agregacije i funkcije za rad sa podacima	
3.7.g	Statički i dinamički filteri, filteri koji sadrže veliki broj zapisa	
3.7.h	Osnovne OLAP operacije na dimenzionalnom modelu	
3.7.i	Dinamički aktivni grafikoni i tabele za rad sa činjenicama i dimenzijama, uspostavljanje veza između izveštaja itd.	
3.7.j	Definicije izveštaja su dostupne u obliku alata za upravljanje meta podacima (MDM)	
3.7.l	Licenca za najmanje 8 korisnika ukupno pri dizajniranju izveštaja, 2 rade istovremeno tj. konkurentno	
3.7.l	Licenca za najmanje 20 korisnika ukupno kada se pokreću izveštaji, od kojih 4 moraju omogućiti aktivni rad u konkurentnom režimu	

Tabela 7 – Tabela za procenu komisije – Obavezni tehnički uslovi

REZIME OBAVEZNIH ZAHTEVA: _____

UKUPNO ispunjenje obaveznih tehničkih kriterijuma: _____

Funkcije - Sistem bonusa

SRS ref.	Kratak opis uslova Ispunjenost uslova (0/1)	Ispunjenost uslova [0/1]
1.2	Bonus karakteristike sistema	
1.2.a.1	MPP arhitektura DS uključena u isporuku	
1.2.a.1	MPP AAP arhitektura uključena u isporuku	
1.2.a.1	MPP BI arhitektura uključena u isporuku	
1.2.a.1	MPP arhitektura DKM uključena u isporuku	
1.2.a.1	MPP arhitektura DI uključena u isporuku	
1.2.b	Potpuno automatska optimizacija skladištenja podataka u DS	
1.2.c	Potpuna analitička podrška u bazi podataka	
1.2.d	Najmanje jedno od navedenih svojstava za komponentu AAP	
1.2.e	Neograničene licence za horizontalno DS skaliranje na i hardveru	
1.2.f	Neograničene licence za vertikalnu skalabilnost DS-a na isporučenom hardveru	
1.2.g	Podrška za cikluse u toku podataka bez potrebe za skript programom	
1.2.h	Integracija kreiranja, testiranja i primene naprednih prediktivnih modela u BI	
1.2.i	Generisanje PPT prezentacija koje se mogu uređivati	

Tabela 8 – Funkcija sistema bonusa

REZIME BONUS KARAKTERISTIKE: _____

UKUPNO ispunjenje bonus karakteristika sistema: _____

Faza 1 – Definicija tehničkih kriterijuma

Ograničenje	Vrednosti parametra	Limit	Ispunjeno [DA/NE]
Cena procenjenog rešenja (CZK)		5.500.000 [CZK]	
Obavezni tehnički kriterijumi (broj ispunjenih kriterijuma)	0	106 bodova	
TPC-H benchmark (vreme u satima)			
Početni uvoz TPC-H 1 TB		24 [h]	
Početni uvoz TPC-H 3 TB		96 [h]	
Test performansi TPC-H 1 TB - 1. krug		1,5 [h]	
Test performansi TPC-H 3 TB - 1. krug		5 [h]	
Test performansi TPC-H 1 TB - 2. krug		1,5 [h]	
Test performansi TPC-H 3 TB - 2. krug		5 [h]	
Ispunjenje svih kriterijuma Faze 1			

Tabela 9 - Faza 1 – Definicija tehničkih kriterijuma

Faza 2 – Kriterijumi za ocenu ponuđenog rešenja

Kriterijumi za ocenu ponuđenog rešenja – 2. faza

Parametar	Težina (Preračun)	Ocena	Težina ocene (Preračunavanje)
Ispunjenost osnovnih kriterijuma za celu 1. fazu (tehnički kriterijumi)			FALSE
Usklađenost sa graničnim vrednostima iz faze 1 (TPC-H benchmark)			FALSE
Usklađenost sa graničnim vrednostima za celu fazu 2			FALSE
Cena	70%	0%	0%

Parametar	Težina (Preračun)	Ocena	Težina ocene (Preračunavanje)
Bonus funkcije	15%	0%	0%
Takmičenje u brzini	15%	0%	0%
Konačna ocena ukupno			0%

Tabela 10 - Kriterijumi za ocenu ponuđenog rešenja – 2. faza

Cena

Ograničenje	Vrednosti	Parametar	Ispunjeno [DA/NE]
Cena procenjenog rešenja	0	FALSE	
Cena najboljeg rešenja	5.500.000		
Rezultat procenjenog rešenja	0%		

Tabela 11 – Izračunavanje cene predloženog rešenja

Bonus karakteristike

Parametar	Težina	Sadrži parametar (0/1)		Konačna ocena
MPP arhitektura uključena				
MPP DS komponente uključujući analitičku podršku	30%	0	0%	
MPP komponente AAP	10%	0	0%	
MPP komponente BI	10%	0	0%	
MPP komponente DKM	10%	0	0%	
MPP komponente DI	10%	0	0%	
Potpuno automatska optimizacija skladištenja podataka u DS	20%	0	0%	
Kompletna analitička podrška u bazi podataka	20%	0	0%	

Parametar	Težina	Sadrži parametar (0/1)		Konačna ocena
Najmanje jedno od navedenih svojstava za AAP komponentu	20%	0	0%	
Neograničene licence za horizontalno DS skaliranje na nivou hardvera	20%	0	0%	
Neograničene licence za vertikalnu skalabilnost DS-a na nivou hardvera	5%	0	0%	
Podrška ciklusa protoka podataka bez skript programa	5%	0	0%	
Integracija kreiranja, testiranja i primene naprednih prediktivnih modela u BI	5%	0	0%	
Generisanje PPT prezentacija koje se mogu uređivati	5%	0	0%	

Tabela 12 - Bonus karakteristike

Dodatak B – Ispunjenje minimalnih zahteva

Korak 2 - Standardni proces integracije podataka kvartalnog izvoza

1. korak (Zdravstvena osiguravajuća organizacija – šifra 901)

Korak	Vreme (h)	Ograničenje (h)	Ispunjeno
Korak 2.1 - Inkrementalno učitavanje izvoza ulaza u skladište podataka			
Korak 2.2. - Kontrola kvaliteta uvezenih podataka			
Korak 2.3 - Generisanje unapred definisanih izveštaja			
Korak 2 - Standardni proces inicijalnog učitavanja podataka u DWH za poslednji kvartal UKUPNO	0,00	12,00	FALSE

Tabela 13 - 1. Korak – Zdravstvena osiguravajuća organizacija – šifra 901

2. korak (Zdravstvena osiguravajuća organizacija – šifra 922)

Korak	Vreme (h)	Ograničenje (h)	Ispunjeno
Korak 2.1 - Inkrementalno učitavanje izvoza ulaza u skladište podataka			
Korak 2.2. - Kontrola kvaliteta uvezenih podataka			
Korak 2.3 - Generisanje unapred definisanih izveštaja			
Korak 2 - Standardni proces inicijalnog učitavanja podataka u DWH za poslednji kvartal UKUPNO	0,00	12,00	FALSE

Tabela 14 - 2. Korak - Zdravstvena osiguravajuća organizacija – šifra 922

3. korak (Zdravstvena osiguravajuća organizacija – šifra 955)

Korak	Vreme (h)	Ograničenje (h)	Ispunjeno
Korak 2.1 - Inkrementalno učitavanje izvoza ulaza u skladište podataka			
Korak 2.2. - Kontrola kvaliteta uvezenih podataka			

Korak	Vreme (h)	Ograničenje (h)	Ispunjeno
Korak 2.3 - Generisanje unapred definisanih izveštaja			
Korak 2 - Standardni proces inicijalnog učitavanja podataka u DWH za poslednji kvartal UKUPNO	0,00	12,00	FALSE

Tabela 15 - 3. Korak - Zdravstvena osiguravajuća organizacija – šifra 955

4. korak (Zdravstvena osiguravajuća organizacija – šifra 955)

Korak	Vreme (h)	Ograničenje (h)	Ispunjeno
Korak 3.1. Brisanje nevažećeg izvoza iz skladišta podataka			
Korak 3.2. Otpremite ispravljenu verziju za izvoz			
Korak 3.3. Kontrola kvaliteta uvezenih podataka			
Korak 3.4. Generisanje unapred definisanih izveštaja			
Korak 3 - Ispravite pogrešne podatke u DWH UKUPNO	0,00	12,00	FALSE

Tabela 16 - 4. Korak - Zdravstvena osiguravajuća organizacija – šifra 955

Korak 9 - Izmenite skladište podataka

Korak	Vreme (h)	Ograničenje (h)	Ispunjeno
Korak 9.1. Priprema razvojnog okruženja			
Korak 9.2. Modifikacija procesa u razvojnom okruženju			
Korak 9.3. Prenos promena iz razvojnog u proizvodno okruženje			
Korak 9 - Modifikacija skladišta podataka UKUPNO	0,00	12,00	FALSE

Tabela 17 - 9. Korak - izmena skladišta podataka

Dodatak C – Test performansi

Takmičenje u brzini (poc_speed_benchmark)

Ocenjeno rešenje

1. Test performansi - ciklus 1

Korak	Vreme koraka (min)	Ograničenje (min)	Završeno
Korak 4 – Kontrola kvaliteta	0,00	60,00	FALSE
Korak 5.1 – Vreme praćenja pacijenata sa propisanim lekovima	0,00		
Korak 5.2 – Vreme praćenja pacijenata sa određenim procedurama	0,00		
Korak 5 – Priprema dokumenata za ad-hoc analize UKUPNO	0,00	20,00	FALSE
Korak 6.1 – Trend prijavljenih stavki nakon dijagnoze	0,00		
Korak 6.2 – Pregled troškova ATC lekova za odabrane dijagnoze	0,00		
Korak 6.3 – Epidemiološko opterećenje za odabrane dijagnoze	0,00		
Korak 6 – Rad sa BI [UKUPNO]	0,00	60,00	FALSE
Korak 7 - Test brzine DB	0,00	60,00	FALSE
Korak 8.1 - Opis slučajeva hospitalizacije	0,00		
Korak 8.2 – Zadatak klasifikacije prema pripremljenom modelu	0,00		
Korak 8.3 - Predviđanje vremenske serije	0,00		
Korak 8 – Analiza u bazi podataka i vizuelizacija u BI alatu UKUPNO	0,00	60,00	FALSE
1. Benchmark [UKUPNO]	0,00		FALSE

Tabela 18 – Prvi krug testa performansi

2. Test performansi - ciklus 2

Korak	Vreme koraka (min)	Ograničenje (min)	Završeno
Korak 4 – Kontrola kvaliteta	0,00	60,00	FALSE
Korak 5.1 – Vreme praćenja pacijenata sa propisanim lekovima	0,00		
Korak 5.2 – Vreme praćenja pacijenata sa određenim procedurama	0,00		
Korak 5 – Priprema dokumenata za ad-hoc analize [UKUPNO]	0,00	20,00	NETAČNO
Korak 6.1 – Trend prijavljenih stavki nakon dijagnoze	0,00		
Korak 6.2 – Pregled troškova ATC lekova za odabrane dijagnoze	0,00		
Korak 6.3 – Epidemiološko opterećenje za odabrane dijagnoze	0,00		
Korak 6 – Rad sa BI [UKUPNO]	0,00	60,00	FALSE
Korak 7 - Test brzine DB	0,00	60,00	FALSE
Korak 8.1 - Opis slučajeva hospitalizacije	0,00		
Korak 8.2 – Zadatak klasifikacije prema pripremljenom modelu	0,00		
Korak 8.3 - Predviđanje vremenske serije	0,00		
Korak 8 – Analiza u bazi podataka i vizuelizacija u BI alatu UKUPNO	0,00	60,00	FALSE
2. Benchmark [UKUPNO]	0,00		FALSE

Tabela 19 - Drugi krug testa performansi

Prvi krug

Korak	Vreme koraka (min)	Ograničenje (min)	Završeno
Korak 4 – Kontrola kvaliteta	0,00	60,00	FALSE

Korak	Vreme koraka (min)	Ograničenje (min)	Završeno
Korak 5.1 – Vreme praćenja pacijenata sa propisanim lekovima	0,00		
Korak 5.2 – Vreme praćenja pacijenata sa određenim procedurama	0,00		
Korak 5 – Priprema dokumenata za ad-hoc analize [UKUPNO]	0,00	20,00	FALSE
Korak 6.1 – Trend prijavljenih stavki nakon dijagnoze	0,00		
Korak 6.2 – Pregled troškova ATC lekova za odabrane dijagnoze	0,00		
Korak 6.3 – Epidemiološko opterećenje za odabrane dijagnoze	0,00		
Korak 6 – Rad sa BI [UKUPNO]	0,00	60,00	FALSE
Korak 7 - Test brzine DB	0,00	60,00	FALSE
Korak 8.1 - Opis slučajeva hospitalizacije	0,00		
Korak 8.2 – Zadatak klasifikacije prema pripremljenom modelu	0,00		
Korak 8.3 - Predviđanje vremenske serije	0,00		
Korak 8 – Analiza u bazi podataka i vizuelizacija u BI alatu [UKUPNO]	0,00	60,00	FALSE
3. Benchmark [UKUPNO]	0,00		FALSE

Tabela 20 – Test performansi tabela prvi krug

Rezime procenjenog rešenja

Protokol	Vreme testiranja kruga (min)
1. Benchmark	0,00
2. Benchmark	0,00
3. Benchmark	0,00
Prosečna vrednost	0,00

Tabela 21 - Rezime procenjenog rešenja

Konačna ocena takmičenja u brzini

Parametar	Vrednost parametra
Najgore moguće rešenje (min)	260,00
Ocenjeno rešenje (min)	0,00
Najbolje rešenje (min)	260,00
Rezultat procenjenog rešenja (%, max. 100%)	0%

Tabela 22 - Konačna ocena testa performansi

Biografija autora

Martin Štufi je rođen 25.10.1973. godine u Nišu. Završio je osnovnu školu „Ivo Andrić“ u Nišu sa prosečnom ocenom 5.00 u toku školovanja. Nakon osnovne škole završio je elektrotehničku školu „Mija Stanimirović“ u Nišu sa prosečnom ocenom 5.00 u toku školovanja. Elektronski fakultet u Nišu upisao je 01.10.1992. godine, a diplomirao juna 1998. godine sa ocenom 10 na temu „Oracle baza podataka“ i stekao zvanje diplomiranog inženjera za računarsku tehniku i informatiku. Po završetku osnovnih studija, decembra 1999. godine se preselio u Prag, Češka Republika, gde i danas živi i radi. Doktorske studije na Elektronskom fakultetu u Nišu upisao je oktobra 2012. godine. U toku doktorskih studija položio sve predviđene ispite sa prosečnom ocenom 10.

U periodu od 2000. godine do 2003. godine, radio je kao softver inženjer i IT konsultant u međunarodnim firmama Sitronics TS, CMG, Telefonica O2. Bio je uključen na raznim komercijalnim i javnim projektima kao glavni inženjer za migraciju baza podataka, njihovo projektovanje kao i integraciju u okviru informacionih sistema na svim nivoima. Od 2004. godine do 2006. godine radio je na vladinim projektima Češke Republike u oblasti informacionih tehnologija u toku pripreme pristupanja Evropskoj uniji, kao i nakon njenog pristupanja. Tokom 2004. godine osnovao je dve IT kompanije koje se bave razvojem softvera i pružanjem usluga u okviru savremenih informacionih sistema. Od 2004. godine radi kao osnivač, vlasnik i izvršni direktor kompanije Solutia s.r.o. sa sedištem u Pragu.

Interesovanja njegovog istraživanja i rada u oblasti računarstva su u razvoju i primeni informacionih sistema kompanija zasnovanih na modernim tehnologijama za obradu velike količine podataka, rešenja za računarstvo u oblaku, IT bezbednosti, mobilnosti korisnika uz upotrebu modernih informatičkih rešenja, kao i IoT tehnologija i Internet stvari. Učestvovao je u kreiranju modernih softverskih rešenja razvijenih primarno na modernim Java tehnologijama kao što su Java Vaadin Framework, Java Spring Framework. Na ovim tehnologijama u kompaniji Solutia s.r.o. razvijeni su napredni sistemi za akviziciju, smeštanje, obradu i

prikazivanje velike količine podataka, telemetrije, kao i potpune personalizacije svakog rešenja i prilagođavanja problema klijenata.

Takođe, nosilac je međunarodno priznatih sertifikata za upravljanje projektima i programima u oblasti informacionih tehnologija, upravljanje rizicima u IT projektima, arhitekturu IT sistema, etičkog testiranja bezbednosti softvera, sistema za obradu velike količine podataka. U svojoj karijeri održao je više od 100 međunarodnih kurseva kao sertifikovani trener za obradu velike količine.

Kao autor, objavio je nekoliko naučnih i stručnih radova u međunarodnom časopisu kao i na nacionalnim i međunarodnim konferencijama.

Izjava o autorstvu

Izjavljujem da je doktorska disertacija, pod naslovom

„PREDLOG ARHITEKTURE SISTEMA VISOKIH PERFORMANSI ZA GENERALNU OBRADU PODATAKA NA KLASTERIMA ZA PODATKE VELIKOG OBIMA“

koja je odbranjena na Elektronskom fakultetu Univerziteta u Nišu:

- rezultat sopstvenog istraživačkog rada;
- da ovu disertaciju, ni u celini, niti u delovima, nisam prijavljivao na drugim fakultetima, niti univerzitetima;
- da nisam povredio autorska prava, niti zloupotrebio intelektualnu svojinu drugih lica.

Dozvoljavam da se objave moji lični podaci, koji su u vezi sa autorstvom i dobijanjem akademskog zvanja doktora nauka, kao što su ime i prezime, godina i mesto rođenja i datum odbrane rada, i to u katalogu Biblioteke, Digitalnom repozitorijumu Univerziteta u Nišu, kao i u publikacijama Univerziteta u Nišu.

U Nišu, _____

Potpis autora disertacije:

Dr Martin T. Štufi

Izjava o istovetnosti elektronskog i štampanog oblika doktorske disertacije

Naslov disertacije:

„PREDLOG ARHITEKTURE SISTEMA VISOKIH PERFORMANSI ZA GENERALNU OBRADU PODATAKA NA KLASTERIMA ZA PODATKE VELIKOG OBIMA“

Izjavljujem

da je **elektronski oblik moje doktorske disertacije**, koju sam predao za unošenje u Digitalni repozitorijum Univerziteta u Nišu, **istovetan štampanom obliku**.

U Nišu, _____

Potpis autora disertacije:

Dr Martin T. Štufi

Izjava o korišćenju

Ovlašćujem Univerzitetsku biblioteku „Nikola Tesla“ da u Digitalni repozitorijum Univerziteta u Nišu unese moju doktorsku disertaciju, pod naslovom:

„PREDLOG ARHITEKTURE SISTEMA VISOKIH PERFORMANSI ZA GENERALNU OBRADU PODATAKA NA KLASTERIMA ZA PODATKE VELIKOG OBIMA“

Disertaciju sa svim prilozima predao sam u elektronskom obliku, pogodnom za trajno arhiviranje.

Moju doktorsku disertaciju, unetu u Digitalni repozitorijum Univerziteta u Nišu, mogu koristiti svi koji poštuju odredbe sadržane u odabranom tipu licence Kreativne zajednice (Creative Commons), za koju sam se odlučio.

1. Autorstvo **(CC BY)**
2. Autorstvo – nekomercijalno **(CC BY-NC)**
3. Autorstvo – nekomercijalno – bez prerade **(CC BY-NC-ND)**
4. Autorstvo – nekomercijalno – deliti pod istim uslovima **(CC BY-NC-SA)**
5. Autorstvo – bez prerade **(CC BY-ND)**
6. Autorstvo – deliti pod istim uslovima **(CC BY-SA)**

U Nišu, _____

Potpis autora disertacije:

Dr Martin T. Štufi