

TARTU ÜLIKOOL  
Arvutiteaduse instituut  
Informaatika õppekava

**Vladislav Buivol**  
**Andmebaasi juhtimissüsteemi võimekuse mõõtmine**  
**Bakalaureusetöö (9 EAP)**

Juhendaja: Vambola Leping

Tartu 2019

# **Andmebaasi juhtimissüsteemi võimekuse mõõtmine**

## **Lühikokkuvõte:**

Käesolevas töös võrreldakse PostgreSQLi ja MariaDBi juhtimissüsteeme. Võrreldakse järgmisi parameetreid: andmemaht, platvormi tugi ja turvalisus. Töö käigus mõõdeti jõudlust järgmistel andmebaasi juhtimissüsteemidel: PostgreSQL, MariaDB ja MariaDB, mille lähtekoodi oli muudetud. Jõudluse mõõtmiseks kasutati tööriista nimega HammerDB. Töö käigus kirjeldatakse andmebaasi juhtimissüsteemi seadistamist, jõudluse mõõtmise protsessi ja ette tulnud probleeme, nende lahendusi ja võimalikke põhjuseid.

## **Võtmesõnad:**

PostgreSQL, MariaDB, jõudluse mõõtmine, TPC-C, TPC-H, HammerDB

**CERCS:** P170 Arvutiteadus, arvutusmeetodid, süsteemid, juhtimine

# **Measurement of database management system capability**

## **Abstract:**

In this thesis PostgreSQL and MariaDB management systems are compared. The following parameters are compared: data volume, platform support and security. During the work, performance was measured for the following database management systems: PostgreSQL, MariaDB and MariaDB with the source code of which was changed. A tool called HammerDB was used to measure performance. This thesis describes how to set up a database management system, performance measurement process, encountered problems with their solutions and possible causes.

## **Keywords:**

PostgreSQL, MariaDB, measurement of database capability, TPC-C, TPC-H, HammerDB

**CERCS:** P170 Computer science, numerical analysis, systems, control.

# Sisukord

1. Ülevaade andmebaasi juhtimissüsteemidest: PostgreSQL ja MariaDB	8
1.1 PostgreSQL	8
1.2 MariaDB	8
1.3 MariaDB ja PostgreSQL andmemahu võrdlemine	9
1.4 Platvormi tugi	11
1.4.1 PostgreSQL platvormi tugi	11
1.4.2 MariaDB platvormi tugi	11
1.5 Turvalisus	12
1.6 Andmebaasi jõudluse mõõtmise meetodeid ja ühikud	13
1.6.1 Jõudluse mõõtmise ühikud ja meetodid	13
2. Andmebaasi juhtimissüsteemi jõudluse mõõtmine	15
2.1 HammerDB tööpõhimõtte kirjeldus	16
2.1.1 TPC-C töö põhimõtte ülevaade	17
2.1.2 TPC-H töö põhimõtte ülevaade	19
2.2 Andmebaasi juhtimissüsteemi seadistamine	20
2.2.1 Andmete genereerimine ja jõudluse mõõtmise käivitamine	21
2.3 Ette tulnud probleemid	22
3. Tulemuste analüüs	24
3.1 TPC-C tulemuste analüüs	24
3.2 TPC-H tulemuste analüüs	28
4. Kokkuvõte	30
5. Viidatud kirjandus	31
6. Lisad	35
I. Terminid	35
II. Transaktsioonid	38
III. Päringud	39
IV. Arvuti informatsioon	40
V. PostgreSQL parameetrid	41
VI. MariaDB parameetrid	42
VII. Driven Script	44
VIII. Sammud TPC-C käivitamiseks	45
IX. Muudetud failid	46

X. TPC-C tulemused	47
XI. TPC-H tulemused	48
XII. Litsents	49

## Sissejuhatus

Andmebaas on üks tähtsamaid tarkvara komponente infosüsteemides. Suurem osa tarkvarast kasutab, kogub ja hoiab andmeid, seetõttu kasutavad nad andmebaase. Viimaste loomiseks ja haldamiseks on olemas palju erinevaid andmebaaside juhtimissüsteeme (ABJS, ingl *DMBS*). Näiteks vajavad masinõppe ja tehisintellekti rakendused palju andmeid ning neid andmeid on vaja kusagil hoida ja kiiresti töödelda. Andmebaasi juhtimissüsteemi valimise mõju on pikaajaline ning selle otsuse muutmine võib hiljem osutada keeruliseks ja kalliks. Antud töö tegeleb andmebaaside juhtimissüsteemi jõudluse uurimisega, sest jõudlus on üks oluline näitaja, mis mõjutab süsteemi valikut.

Käesolevas töös mõõdetakse andmebaaside juhtimissüsteemi jõudluse mõõtmist vabavaraliste vahendite abil. Andmebaasi mõõtmise ja analüüsimise avalikud tulemused võivad rikkuda äri- lisi eesmärke, seega valiti mõõtmiseks mitte firmade tasulisi tooteid, vaid avatud lähtekoodiga tasuta andmebaaside juhtimissüsteeme PostgreSQL ja MariaDB. Kuigi tuntuim vabavaraline andmebaasi juhtimissüsteem on MySQL, valiti MySQLi asemel MariaDB sellepärast, et leidub vähe uuringuid MariaDBi jõudluse kohta ning MySQL on ainult osaliselt tasuta [1]. Kaaluti ka MongoDB<sup>1</sup> ja süsteemi Redis<sup>2</sup> analüüsimist, aga nende kohta on juba jõudluse uuringud olemas.

Antud bakalaureusetöö eesmärgid on:

1. Valida kaks tasuta avatud lähtekoodiga andmebaasi juhtimissüsteemi ja teha valitud andmebaasi juhtimissüsteemi ülevaade;
2. Selgitada, mis on andmebaasi juhtimissüsteemi jõudluse mõõtmine ja miks seda on vaja teha;
3. Tutvustada olemasolevaid mõõtmismeetodeid ja tööriistu, millega võib jõudlust mõõta;
4. Leida vabavaraline jõudluse mõõtmise tööriist;
5. Näidata, kuidas käib jõudluse mõõtmine valitud andmebaasi juhtimissüsteemi näitel;
6. Mõõta valitud andmebaaside juhtimissüsteemide jõudlust ja võrrelda tulemusi;
7. Uurida, kuidas muutub andmebaaside jõudlus pärast seadistamist.

---

<sup>1</sup>[https://www.researchgate.net/publication/317184986\\_Comparisons\\_between\\_MongoDB\\_and\\_MS-SQL\\_Databases\\_on\\_the\\_TWC\\_Website](https://www.researchgate.net/publication/317184986_Comparisons_between_MongoDB_and_MS-SQL_Databases_on_the_TWC_Website)

<sup>2</sup>[https://www.researchgate.net/publication/331840170\\_Architecture-Aware\\_High\\_Performance\\_Transaction\\_for\\_Persistent\\_Memory](https://www.researchgate.net/publication/331840170_Architecture-Aware_High_Performance_Transaction_for_Persistent_Memory)

Töö koosneb kolmest peatükist. Esimeses peatükis on antakse ülevaade teoreetilisest lähemisest jõudluse mõõtmisele. Selgitatakse, miks valiti just PostgreSQL ja MariaDB ning antakse ülevaade andmebaasi juhtimissüsteemist ja nende jõudluse mõõtmisest. Võrreldakse järgmisi parameetreid: hallatavate andmete maksimaalne hulk ehk mahtuvus, platvormi tugi ja turvalisus. Edasi selgitatakse, miks on vaja andmebaasi juhtimissüsteemi jõudlust mõõta. Samuti kirjeldatakse relatsioonilise andmebaasi juhtimissüsteemi jõudluse analüüsimise ja mõõtmise meetodeid ning jõudluse mõõtmise ühikuid.

Teises peatükis kirjeldatakse relatsioonilise andmebaasi juhtimissüsteemi jõudluse mõõtmist ja kasutatud tööriistu, selgitatakse tööriistade töötamise põhimõtteid ja mõõtmiste käigus ette tulnud probleeme. Põhjendatakse, miks valiti just sellised mõõtmise meetodid ja tööriistad. Tuuakse välja ka jõudluse mõõtmise mõisted.

Kolmandas peatükis kirjeldatakse käesoleva töö raames tehtud andmebaasi juhtimissüsteemi jõudluse mõõtmise ja jõudluse analüüsimise tulemusi.

Töoga seotud mõisted on lisas I.

# 1. Ülevaade andmebaasi juhtimissüsteemidest: PostgreSQL ja MariaDB

Selles peatükis kirjeldatakse, mis on PostgreSQL ja MariaDB, antakse nende andmebaaside juhtimissüsteemide ülevaade ja võrdlus ning selgitatakse, miks valiti just PostgreSQL ja MariaDB. Info MariaDB kohta on võetud MariaDB dokumentatsioonist. Info PostgreSQL kohta on võetud PostgreSQL dokumentatsioonist ja PostgreSQL wiki lehest. Ametlik dokumentatsioon ja PostgreSQL wiki leht valiti kui kõige usaldusväärsemad allikad.

PostgreSQL valiti sellepärast, et see on üks populaarsemaid avatud lähtekoodiga andmebaase [2]. MariaDB valiti, kuna selle populaarsus kasvab kiiresti ja praegu on see samuti üks enim kasutatavaid avatud lähtekoodiga andmebaase [2].

## 1.1 PostgreSQL

PostgreSQL on avatud lähtekoodiga objekt-relatsiooniline andmebaasi juhtimissüsteem, mis kasutab ja laiendab SQL-keelt paljude funktsioonidega, mis turvaliselt talletavad andmeid ja kannatavad suure andmevoo koormusi [3].

Süsteem pärineb 1986. aastast, mis loodi POSTGRESi projekti käigus California Ülikoolis Berkeleys [3]. PostgreSQL on pälvinud hea maine oma asjakohase arhitektuuri, usaldusväärsuse, laiendatavuse (ingl *scalability*, vt lisa I) poolest ja tarkvara taga oleva kogukonna pühendumisele avatud lähtekoodi täiendamisele, et pakkuda järjepidevalt tulemuslikke ja uuenduslikke lahendusi [3]. PostgreSQL, mis töötab kõigis peamistes operatsioonisüsteemides, on alates 2001. aastast ACID (ingl *Atomicity, Consistency, Isolation, Durability*, vt lisa I) ühilduv [3]. PostgreSQL avaldatakse PostgreSQLi litsentsi alusel, mis on avatud lähtekoodiga litsents. PostgreSQLi ei avaldata GNU üldise avaliku litsentsi alusel, sest eelistatakse PostgreSQL litsentsi, ning seda valikut ei plaanita muuta [4].

## 1.2 MariaDB

MariaDB Server on üks populaarsemaid andmebaasiservereid maailmas [2]. MariaDB Sihtasutus (ingl *Foundation*) toetab MariaDB järjepidevust. Sihtasutus on mittetulundusühing Dela-



ware'is USAs, mida rahastavad ettevõtjad ja üksikud sponsorid [5]. MariaDBi andmebaasi juhtimissüsteem on loodud MySQLi algsete arendajate poolt ja on tagatud, et see jääb avatud lähtekoodiga serveriks [5]. Selle süsteemi tuntud kasutajad on Wikipedia, WordPress ja Google [5]. MariaDB uusimad versioonid sisaldavad ka GIS- ja JSON-funktsioone (vt lisa I) [5]. Andmebaasi MariaDB levitatakse GPLi (vt lisa I) litsentsi versiooni 2 alusel [6].

### 1.3 MariaDB ja PostgreSQL andmemahu võrdlemine

Kuni versioonini 10.1 kasutas MariaDB salvestusmootorit Percona XtraDB (vt lisa I). Alates versioonist 10.2 vaikimisi kasutatakse salvestusmootorit InnoDB [7]. Percona XtraDB on täiustatud versioon InnoDBi salvestusmootorist, mis on mõeldud kaasaegsele riistvarale paremaks laiendatavuseks ja sisaldab mitmeid muid funktsioone, mis on kasulikud suure jõudlusega keskkonnas [8].

MariaDBi hallatavate andmete hulk sõltub sellest, mis versiooni kasutatakse [9]. Tabelis 2 vaadeldakse salvestusmootori InnoDBi mahtuvust, sest alates MariaDB versioonist 10.2 kasutatakse InnoDBi salvestusmootorit. Nii InnoDBi kui ka XtraDBi tabelid võivad olla kuni 64TB suurused [10].

MariaDB tabeli maht sõltub lehekülje (vt lisa I) suuruselt [11]. Vaikimisi lehekülje suurus on 16KB ja tabeli maht 64TB. Lehekülje suurus võib muuta ja siis muutub lubatud tabeli maht [12]. Tabelis 1 on esitatud lehekülje suurus ja maksimaalne tabeli maht [12].

Tabel 1: Tabeli mahu sõltuvus lehekülje suuruselt [12]

InnoDB Lehekülje suurus	Maksimaalne tabeli maht
4KB	16TB
8KB	32TB
16KB	64TB
32KB	128TB
64KB	256TB

Tavaliselt saavutatakse kasutatava mälu füüsilise suuruse ja operatsioonisüsteemi ruumi piirid enne MariaDBi sisemiste piiride saavutamist [10].

Tabelis 2 on esitatud andmebaaside PostgreSQLi ja InnoDBi mahtuvused.

Tabel 2: PostgreSQL ja MariaDB mahtuvuse võrdlemine

	<b>PostgreSQL</b> [13]	<b>InnoDB</b> [10]
<b>Maksimaalne andmebaasi suurus</b>	Piirdub vaba kettaruumi ja mäluruumiga	Piirdub vaba kettaruumi ja mäluruumiga
<b>Suurim lubatud tabeli maht</b> (vaikimisi suurus)	Vaikimisi 32TB	Vaikimisi 64TB.
<b>Maksimaalne rea suurus</b>	1,6TB	MariaDB kehtestab kõigi veergude kombineeritud suuruste jaoks 65535 baidise rea suuruse piiri.
<b>Maksimaalne välja suuruse</b>	1 GB	MariaDB kehtestab kõigi veergude kombineeritud suuruste jaoks 65535 baidise rea suuruse piiri.
<b>Maksimaalne kirjapanekute maht tabelis</b>	Piiranguid ei ole. Piirdub vaba kettaruumi ja mäluruumiga	InnoDB salvestusmootoril ei ole ridade arvu piiratud, kuid tabeli mahu piirmäär on 64 terabaiti.
<b>Maksimaalne veergude arv tabelis</b>	250–1600, sõltuvalt veergude tüübidest	1017
<b>Maksimaalne indekstite arv tabelis</b>	Piirdub vaba kettaruumi ja mäluruumiga	64

MariaDB kehtestab kõigi veergude kombineeritud suuruste jaoks 65535-baidise rea suuruse piiri ehk maksimaalne kirje pikkus on 65535 baiti. Kui tabel sisaldab BLOB (ingl *Binary Large Object*) - või TEXT (vt lisa I) veerge, siis arvestatakse nende väljade laiuseks ainult 9-12 baiti, sest nende sisu salvestatakse eraldi. InnoDBi tabelitel võib olla maksimaalselt 64 sekundaarset indeksit (vt lisa I) [10, 45]. InnoDBi mitme veeru indeks võib kasutada maksimaalselt 16 veergu [12]. Kui üritatakse luua mitme veeru indeksit (vt lisa I), mis kasutab rohkem kui 16 veergu, annab MariaDB vea 1070 [12].

PostgreSQLis on kasutusel üks ACID MVCC (ingl *Multiversion concurrency control*, vt lisa D) mootor [14]. PostgreSQLi tabeli maksimaalne suurus ja maksimaalne veergude arv sõltub ploki suurusest ja neid suurendatakse koos ploki suuruse suurendamisega [13]. Tabelis ei tohi olla rohkem kui 1600 veergu. Tegelikult on efektiivne limiit madalam [15].

## 1.4 Platvormi tugi

Selles peatükis tutvustatakse ja võrreldakse peamisi platvorme, mida PostgreSQL ja MariaDB toetavad.

### 1.4.1 PostgreSQL platvormi tugi

Platvorm loetakse PostgreSQL arendusringkonna poolt toetatuks, kui kood sisaldab sätteid sellel platvormil töötamiseks ja on kontrollitud, et selle platvormi regressiooni testimine (vt lisa D) on läbitud [16]. Praegu teostatakse enamik platvormide ühilduvuse testimist PostgreSQL Build Farm testimisseadmete abil [16]. PostgreSQL Build Farm on levinud süsteem PostgreSQL lähtekoodi muutuste automaatseks testimiseks paljudel erinevatel platvormidel [17]. See server on nende testide tulemuste keskne hoidla [14].

Üldiselt võib eeldada, et PostgreSQL töötab järgmistel CPU arhitektuuridel: x86, x86\_64, IA64, PowerPC, PowerPC 64, S / 390, S / 390x, Sparc, Sparc 64, ARM, MIPS, MIPSEL ja PA-RISC. Platvormide M68K, M32R ja VAX jaoks on olemas kooditugi, kuid neid ei ole hiljuti testitud. Sageli on võimalik tugineda toetamata CPU tüübile, kuid jõudlus võib olla halb [16].

PostgreSQL puhul võib eeldada nende operatsioonisüsteemide toimimist: Linux, Windows (Win2000 SP4 ja uuemad), FreeBSD, OpenBSD, NetBSD, MacOS, AIX, HP / UX ja Solaris. Teised Unixi sarnased süsteemid võivad töötada, kuid neid pole testitud [16].

### 1.4.2 MariaDB platvormi tugi

MariaDB Foundation püüab toetada võimalikult palju erinevaid operatsioonisüsteeme, Linuxi eri versioone ja protsessori arhitektuure [18]. Kui operatsioonisüsteem lõpetab turvavärskenduste saatmise, siis MariaDB reegli kohaselt platvorm lõpetab selle jaoks binaarsete pakettide

pakkumise [18]. Lisateavet MariaDB tarkvaraga seotud põhimõtete kohta on võimalik leida Engineering Policy lehel [18].

## 1.5 Turvalisus

Andmebaaside turvalisus viitab meetmetele, mida kasutatakse andmebaasi kaitsmiseks ebaseadusliku kasutamise, ohtude ja rünnakute eest [19]. Andmebaaside turvalisus on lai mõiste, mis sisaldab paljusid protsesse, tööriistu ja meetodikaid, mis tagavad turvalisuse andmebaasi keskkonnas [19].

Tabelis 3 on kirjas meetodid, mida kasutavad andmebaasid PostgreSQL ja MariaDB 10.1 selleks, et tagada andmebaasi turvalisust.

Tabel 3: Turvameetodid [20]

<b>PostgreSQL</b>	<b>MariaDB</b>
Andmebaasil PostgreSQL on kliendi / serveri kommunikatsiooni krüpteerimiseks olemas natiiivne SSL-tugi [20]. Andmebaasil PostgreSQL on luba ja päritud rollid õiguste seadmiseks ja säilitamiseks [21]. Lisaks sellele on andmebaasil PostgreSQL kaasas sisseehitatud lisaseade SE-PostgreSQL, mis pakub täiendavat juurdepääsu kontrolli, mis põhineb SELinuxi turvapoliitikal [22].	Selleks, et tagada turvalisust, kasutab MariaDB siseturvalisuse ja parooli kontrolli [23]. See annab võimaluse rakendada kahe faktoriga autentimist. PAM (vt lisa I) ja LDAP autentimine (vt lisa I) võimaldavad seadistada andmebaasi keskkonda, kus kasutajad saavad jagada paroole tavapärastest sisselogimisest [23]. Lisaks olemas Kerberos <sup>3</sup> , kasutaja rollid ja andmebaasi krüpteerimine [23].

PostgreSQL ja MariaDB mõlemad kasutavad kasutaja rolle, sest see on üks tõhusamaid turvameetodeid.

<sup>3</sup> <https://mariadb.atlassian.net/browse/MDEV-4691>

## 1.6 Andmebaasi jõudluse mõõtmise meetodeid ja ühikud

Andmebaasi jõudluse mõõtmine on tarkvara mittefunktsionaalse testimise vorm, mis keskendub sellele, kuidas süsteem töötab etteantud koormuse all ning on mõeldud süsteemi valmiduse määramiseks [24].

Jõudluse mõõtmine määrab kindlaks, kas tarkvara vastab eeldatavate töö koormuste korral laiendatavuse ja stabiilsuse nõuetele [25]. Ebatäpsete või halbade näitajatega turule sisenevad rakendused saavad tõenäoliselt halva maine ja ei vasta oodatavatele müügieesmärkidele [26].

### 1.6.1 Jõudluse mõõtmise ühikud ja meetodid

Põhilised jõudluse mõõtmise meetodid on koormuse testimine (ingl *load testing*) ja stressitestimine (ingl *stress testing*) [27].

Jõudluse mõõtmise ühikud sõltuvad süsteemide eripärasusest ja mõõtmise eesmärkidest. Põhilised mõõtmise ühikud andmebaaside juhtimissüsteeme mõõtmiseks on järgnevad [29]:

1. Aktiivsete seansside maksimaalne arv (ingl *maximum active sessions*) on maksimaalne seansside arv, mis võivad korraga olla aktiivsed;
2. Mälu kasutamine (ingl *memory use*) on arvuti protsessidele kättesaadava füüsilise mälu maht;
3. Päringute arv (ingl *requests per second*) sekundi kohta näitab, mitu päringut käivitati ühe sekundi jooksul;
4. Päringu täitmise aeg.

Koormuse testimine kontrollib süsteemi käitumist tavapäraustes koormuste tingimustes ning see on tegeliku töökoormusega katsetamine või simuleerimine ja ei riku süsteemi [28].

Koormuse testimine tuvastab tavaliselt [28]:

1. rakenduse maksimaalse töövõime;
2. määrab, kas praegune infrastruktuur on rakenduse käivitamiseks piisav;
3. selgitab välja samaaegsete kasutajate arvu, mida rakendus saab toetada, ja laiendatavuse, et võimaldada rohkematel kasutajatel seda kasutada.

Koormuse testimise peamine näitaja on keskmine reageerimisaeg, hallatavate andmete hulk, süsteemi samaaegsete kasutajate arv ja päringute arv sekundis ning päringute täitmise aeg [29].

Hallatavate andmete hulga mõõtmine on raske ülesanne, sest hallatavate andmete hulk tavaliselt sõltub süsteemi füüsiliselt piiridest (nt: maksimaalne mälu, operatsioonisüsteemi mälu, protsessorite arv).

Stressitestimine on süsteemi käitumise testimine äärmuslikes tingimustes ja seda tehakse kuni süsteemi rikke tekkimiseni. Stressitestimine üritab süsteemi katkestada, mis toimub tagamaks, et süsteem ei läheks kriisiolukordades katki [30].

## 2. Andmebaasi juhtimissüsteemi jõudluse mõõtmine

Selles peatükis kirjeldatakse andmebaasi juhtimissüsteemi jõudluse mõõtmise protsessi. Jõudluse mõõtmise jaoks kasutatakse koormuse testimismeetodeid. Stressitestimise meetodeid ei kasutata sellepärast, et selleks on vaja arvutit, mis talub äärmuslikku koormust kuni süsteemi rikke tekkimiseni.

Jõudluse mõõtmiseks kasutatakse teste, mis on tehtud TPCi (ingl *Transaction Processing Performance Council*) jõudlustestide baasil. TPC on mittetulundusühing, mis on keskendunud andmepõhiste võrdlusstandardite väljatöötamisele [31]. Jõudluse mõõtmiseks valiti jõudlustestid TPCi baasil, sest TPC on andmebaasi tööstuse kõige tuntum viis, et määratleda andmebaasi jõudluse standard, mida tunnustavad kõik juhtivad andmebaasi müüjad [32]. TPC jaotab oma võrdlusaluseid avalikkusele tasuta [32]. TPCi liikmed on tehnoloogiaettevõtted nagu Alibaba group, Dell, Huawei, IBM, Intel, Lenovo, Microsoft ja Oracle [33].

Koormuse testimist võib teha kahel viisil:

1. Iseseisvalt luua tööriist, mille abil mõõta andmebaasi jõudlust;
2. Kasutada olemasolevaid tööriistu.

Iseseisvalt tööriista tegemine nõuab palju aega, seega kasutatakse olemasolevaid tööriistu. Otsiti tööriista, mis on tasuta, piisavalt dokumenteeritud ja lihtne kasutada. Tööriistal peab samuti olema aktiivne kogukond (ingl *community*), piisav arenemine ning toetama PostgreSQLi ja MariaDBi.

Siinse töö jaoks leiti 2 tööriista, mis sobivad kõige paremini [34]:

1. Tsung;
2. HammerDB.

Info HammerDB kohta on võetud HammerDBi dokumentatsioonist.

Andmebaasi jõudluse mõõtmiseks valiti HammerDB. HammerDB valiti sellepärast, et see on avatud lähtekoodiga võrdlusuuringute ja koormuse testimise tarkvara ning sellel on aktiivne

kogukond. Lisaks sellele kasutavad seda vahendit kõik juhtivad andmebaasi- ja tehnoloogiaettevõtted nagu Oracle, IBM, Intel, Dell / EMC HPE, Huawei ja Lenovo [35].

See toetab Oracle'i andmebaasi, SQL Serverit, IBM Db2, MySQL, MariaDB, PostgreSQL ja Redis [35]. Tööriistu HammerDB kasutatakse testiskeemi loomiseks, andmete laadimiseks ja paljude virtuaalsete kasutajate töökoormuste simuleerimiseks andmebaasi suhtes nii transaktsioonide (vt lisa I) kui ka analüütiliste stsenaariumide puhul. Lisaks leidub HammerDBi jaoks detailne dokumentatsioon [35].

HammerDB peamised eelised Tsungi ees on parem dokumentatsioon, aktiivsem kogukond ning see, et suured ettevõtted kasutavad seda rohkem kui Tsungi.

## 2.1 HammerDB tööpõhimõtte kirjeldus

HammerDB töökoormus põhineb TPC-C (vt lisa I) ja TPC-H (vt lisa I) spetsifikatsioonil. HammerDBi eesmärk on mõõta suhtelist, mitte absoluutset andmebaasi jõudlust süsteemide vahel [36]. See tähendab seda, et kui teha test ühe kindla riist- ja tarkvara konfiguratsiooni vastu ning käivitada sama test täpselt sama konfiguratsiooni vastu, tuleb täpselt sama tulemus juhusliku valiku piires, mis tavaliselt erineb kuni 1% võrra [36].

HammerDB ei rakenda täielikke TPC-C ja TPC-H (vt lisa I) võrdlusnäitajaid, sest täielik simulatsioonide läbiviimise võimalus on piiratud, eelkõige nõutava andmepaketi suuruse tõttu, seega HammerDBi tulemusi ei saa mingil viisil võrrelda ametlikult avaldatud TPC-C ja TPC-H võrdlusalustega [35]. TPC-C ja TPC-H võrdlusaluste spetsifikatsioonil põhinev HammerDB rakendus on loodud selleks, et jäädvustada TPC-C ja TPC-H olemust sellisel kujul, mida saab kasutada madalate kuludega mis tahes süsteemis [35]. HammerDB pakub professionaalset, usaldusväärset ja prognoositavat koormustesti kõikidele andmebaasi keskkondadele [35]. HammerDB annab kaks jõudluse parameetrit TPMi ( ingl *Transactions per Minute*) ja NOPMi (uusi tellimusi minutis, ingl *New Orders per Minute*) süsteemide võrdlemiseks [35] .

TPM on konkreetse andmebaasi transaktsiooni mõõtmine, mis on tavaliselt defineeritud kui kasutajate päringute sooritamise (ingl *commits*) ja / või tagasivõtmiste (ingl *rollbacks*) arv [37].



NOPM on toimivusmõõdik, mis ei sõltu ühestki konkreetse andmebaasi rakendamisest ja on soovitatav peamine meetrika [37]. Erinevate andmebaaside võrdlemiseks on vaja kasutada NOPMi näitajat [38].

Ajalooliselt alustas HammerDB Hammerorana ja toetas ainult Oracle'i andmebaasi, nii et see oli vajalik ainult Oracle TPM esitamiseks. Hiljem lisati tugi teistele andmebaasidele [38]. Kuna andmebaasid salvestavad TPM erinevalt, siis ei saa TPM näitajat võrrelda erinevate andmebaaside, näiteks Oracle ja SQL Serveri vahel [38].

### **2.1.1 TPC-C töö põhimõtte ülevaade**

Allolev ülevaade on tehtud HammerDBi ja TPCi dokumentatsioonide järgi. Täieliku TPC-Ci kirjelduse võib leida TPC dokumentatsioonis<sup>4</sup>.

HammerDB sisaldab endas TPC-C võrdlusaluse spetsifikatsiooni rakendamist [32]. TPC-Ci spetsifikatsiooni rakendamisel on oluline eelis, sest see näitab, et test on usaldusväärne ja tagab, et saadakse täpseid, korduvaid [32].

TPC-C hõlmab viit erinevat tüüpi ja keerukate samaaegsete transaktsioonide kombinatsiooni, mis on teostatud võrgus või mis on järjekorda pandud edasilükkamiseks [39]. TPC-C väärtust mõõdetakse transaktsioonides minutis (tpmC) [39].

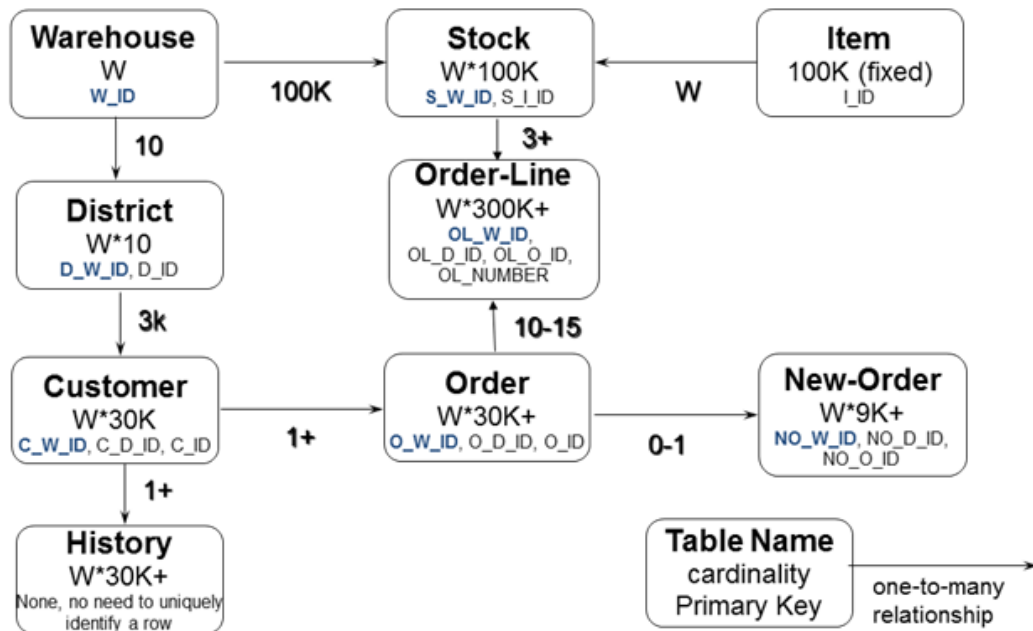
TPC-C koosneb üheksast tabeli tüübist, millel on palju erinevaid kirjeid [32, 39]. Joonis 1 näitab, kuidas kõikide tabelite ridade arv, välja arvatud fikseeritud ITEM-tabel, sõltub skeemi loomiseks valitud ladude ("Warehouse" tabel) arvust. Termin "Warehouse" („Ladu”) TPC-C kontekstis ei ole seotud "Data Warehousing" („Andmete ladustamine”) töökoormusega, kuna TPC-C määratleb transaktsioonipõhise süsteemi, mitte otsuste tuge [40, 41].

Skeem simuleerib ettevõtet, mis müüb 100 000 toodet („Item” tabel) ja hoiab neid laos („Warehouse” tabel) [40]. Igal laol on 10 müügipiirkonda („District” tabel) ja iga piirkond teenindab 3000 klienti („Customer” tabel) [40]. Kliendid helistavad ettevõttele, kelle operaatorid telli-

---

<sup>4</sup>[http://www.tpc.org/tpc\\_documents\\_current\\_versions/current\\_specifications.asp](http://www.tpc.org/tpc_documents_current_versions/current_specifications.asp)

muse esitavad, millest iga tellimus sisaldab mitmeid punkte [40]. Tellimused tarnitakse tavaliselt kohalikust laost, kuid väike arv objekte ei ole konkreetsel ajahetkel laos ja neid tarnib alternatiivne ladu [40]. Oluline on märkida, et ettevõtte suurus ei ole fikseeritud ja ettevõtte kasvades võib ladusid ja müügipiirkondi lisanduda [40]. Sel põhjusel võib testiskeem olla kuitahes suur, kuid suurem skeem vajab suuremate transaktsioonide töötlemiseks võimsamat arvutisüsteemi [40].



Joonis 1: TPC-C skeem [40].

Lisaks tellimuste esitamiseks kasutatavale arvutisüsteemile võimaldab see ka tellimuste eest maksmist ja kättetoimetamist ning võimet küsida laovarude taset [40].

Töökoormuse määrab viie transaktsiooni kombinatsioon, mis valitakse juhuslikult vastavalt protsendilise väärtuse tasakaalule, mis on näidatud järgmiselt [40] :

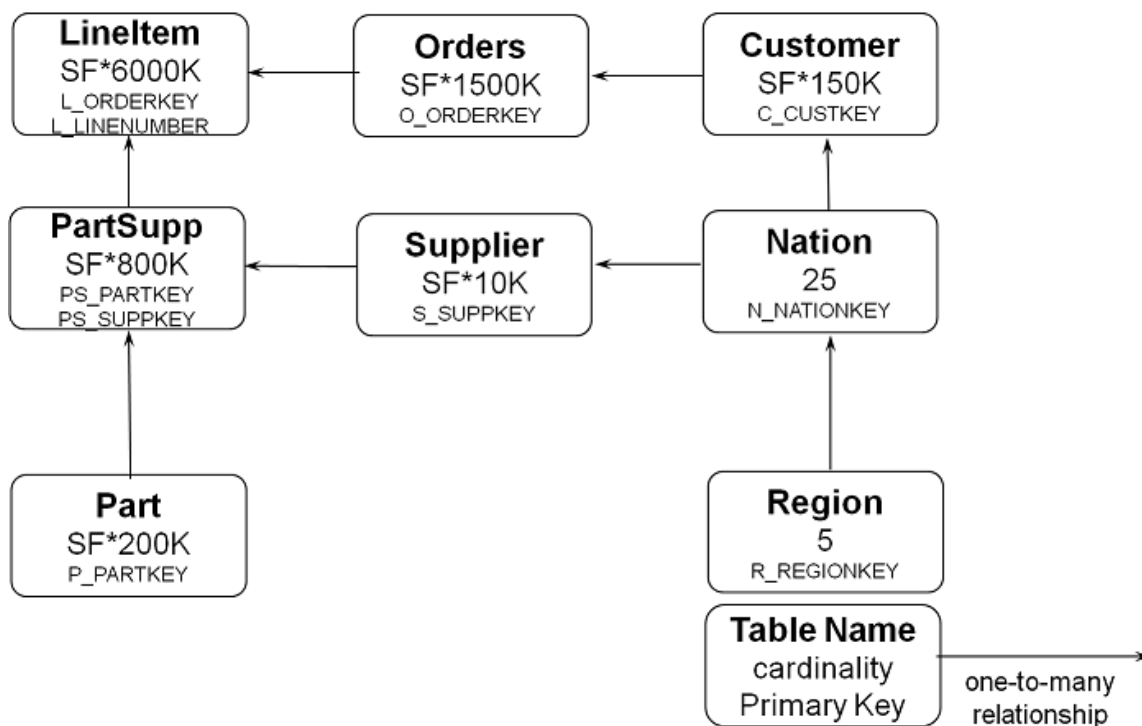
1. Uus tellimus: saate uue tellimuse kliendilt: 45%;
2. Makse: maksete salvestamiseks värskendage klientide saldot: 43%;
3. Kohaletoimetamine: tellimuste asünkroonne edastamine: 4%;
4. Tellimuse staatus: allalaaditavate klientide staatus: 4%;
5. Varude tase: tagastage lao inventari staatus: 4%.

Transaktsioonide näide on lisas number II.

## 2.1.2 TPC-H töö põhimõtte ülevaade

Allolev ülevaade on tehtud HammerDB ja TPC dokumentatsioonide järgi. Täielik TPC-Hi kirjeldusi võib leida TPC dokumentatsioonis<sup>5</sup>.

TPC-H on jõudlusnäitaja, mis koosneb ärile orienteeritud päringutest ja samaaegsete andmete muutmisest [42]. TPC-H on spetsialiseerunud ärialaste päringute kogum ja seda mõõdetakse nende päringute täitmiseks kuluva ajaga [42]. TPC-H näitab päringute arvu tunnis (QphH), mida arvutatakse konkreetse valemiga<sup>6</sup> [43]. HammerDB genereerib skeemi, mis on näidatud joonisel 2.



Joonis 2: TPC-H skeem [42].

Skeemi suurus ei ole fikseeritud ja sõltub Scale Factor'ist (vt lisa I). Skeemil on Scale Factor näidatud nagu SF. Antud jõudlustestimise meetod käivitab 22 päringud. HammerDB annab võimalust vaadata iga päringu täitmise aeg eraldi [42]. Käesolevas töös kasutatakse TPC-Hi meetodit selleks, et leida iga päringu täitmise aeg. Käivitatud päringud on ära toodud lisa number III.

<sup>5</sup>[http://www.tpc.org/tpc\\_documents\\_current\\_versions/current\\_specifications.asp](http://www.tpc.org/tpc_documents_current_versions/current_specifications.asp)

<sup>6</sup><https://www.hammerdb.com/docs/ch09s02.html>

## 2.2 Andmebaasi juhtimissüsteemi seadistamine

Alljärgnevalt kirjeldatakse jõudluse mõõtmise protsessi ja andmebaasi juhtimissüsteemi seadistamist enne jõudluse mõõtmise käivitamist. Andmebaaside versioonid on järgmised: PostgreSQL 11 (64 bit) ja MariaDB 10.3 (64 bit). Sellised versioonid valiti, sest need on kõige uuemad [44, 45]. Jõudluse mõõtmine koosneb kahest etapist. Esimene mõõtmine toimus pärast andmebaasi juhtimissüsteemi paigaldamist, mis oli automaatselt seadistatud. Informatsioon arvuti kohta, kus toimus andmebaasi juhtimissüsteemi jõudluse mõõtmine, on ära toodud lisas number IV.

Samuti mõõdetakse kompileeritud andmebaasi MariaDB, mille lähtekoodi oli muudetud. Lähtekoodis oli muudetud viivituse kordaja, kuna eelnev kordaja võib põhjustada aeglast tööd [46]. Seda on soovitatav teha, kui masin kasutab protsessorit arhitektuuriga Skylake või `ut_delay` võtab liiga palju protsessori (CPU) ressursi [46, 47]. MariaDBi lähtekoodis oli muudetud delay (viivituse) kordaja failis `utOut.cc` reas

```
for (i = 0; i < delay * 50; i++). Viivituse kordajaks oli pandud 5,  
for (i = 0; i < delay * 5; i++). Pärast seda kompileeriti binaarfailid7.
```

MariaDB kompileerimiseks kasutati järgmiseid käske:

```
cmake .. -G "Visual Studio 15 2017 Win64"  
cmake --build . --config relwithdebinfo --target package [46, 47].
```

PostgreSQLi koodi ei olnud muudetud, sest polnud piisavalt informatsiooni ja võimalust jõudluse parandamiseks koodi muutmise kaudu. Teine mõõtmine toimus pärast juhtimissüsteemi seadistamisest.

PostgreSQLi seadistamine toimub failis `postgresql.conf`, mis asub kaustas `PostgreSQL\data\pg11` [48]. Oma parameetrid on vaja lisada faili lõpus pärast kommentaare “# Add settings for extensions here” [48]. MariaDBi seadistamine toimub failis `my.ini`, mis asub kaustas `MariaDB 10.3\data` [49]. Iga masina jaoks on vaja panna eraldi parameetrid ning on

---

<sup>7</sup> <https://superuser.com/questions/52705/what-are-the-binaries-of-a-file>

vaja arvestada sellega, et mida kiiremini andmebaasi juhtimissüsteem töötab, seda rohkem ressursi see kasutab ja seda vähem ressursi jääb teisele protsessile. Pärast parameetrite muutumist on vaja taaskäivitada (ingl *restart*) andmebaaside serverid, et uued sätted jõustuksid.

TPC-C jaoks Postgre ja MariaDB parameetrid oli valitud PostgreSQLi, MariaDBi ja HammerDBi juhendite järgi [49, 50]. Parameetrid, mis oli lisatud faili *postgresql.conf* (vt lisa V) [51]. Parameetrid, mis oli lisatud faili *my.ini*, asuvad lisas number VI [46].

PostgreSQLi jaoks TPC-H-põhise testi käivitamiseks kasutati järgmisi parameetreid [52]:

1. `work_mem = 1000MB;`
2. `max_worker_processes = 16;`
3. `max_parallel_workers_per_gather = 16;`
4. `force_parallel_mode = on.`

Parameetrid, mis kasutati MariaDBi jaoks TPC-H-põhise testi käivitamiseks, asuvad lisas number VI.

### **2.2.1 Andmete genereerimine ja jõudluse mõõtmise käivitamine**

Enne jõudluse mõõtmist on vaja luua andmebaas ja genereerida andmeid. Loodi kolm serverit<sup>8</sup> erinevatega paroolidega, selleks et vältida ühendust vale andmebaasiga. Andmete genereerimiseks kasutati andmebaasi HammerDB.

TPC-C jaoks oli loodud skeem, millel on 250 ladu ja „Virtual Users to Build Schema” parameeter oli pandud 2. See on soovitatud ladude arv [53].

Ühe andmebaasi suurus on umbes 25 GB. Genereerimise aeg sõltub andmebaasi suurusest ja masinast, kus toimub andmebaasi genereerimine. Pärast skeemide genereerimist oli loodud „Driven Script” ja käivitatud „Autopilot”. Autopilot töötas igal käivitamisel umbes 1 - 1.5 tundi. Samal ajal ei tohi käivitada teisi protsesse, sest need võivad mõjutada lõpptulemust. Sammud TPC-Ci käivitamiseks on lisas number VIII (TPC-Hi käivitamiseks on analoogsed sammud).

---

<sup>8</sup> [https://www.webopedia.com/TERM/D/database\\_server.html](https://www.webopedia.com/TERM/D/database_server.html)

TPC-Hi jaoks ei olnud infot vajaliku andmebaasi suuruse kohta, seega oli genereeritud andmebaas, kus „Scale Factor” on 1 ja „Virtual Users to Build Schema” on 2.

Kasutatud „*Driven Script*” on esitatud lisas number VII.

## 2.3 Ette tulnud probleemid

Selles peatükis kirjeldatakse jõudluse mõõtmisel tekkivaid probleeme ja lahendusi. Probleemid on seotud HammerDBi seadistamisega.

Esimene probleem, mis tekkis, oli see, et HammerDB ei pakkunud MariaDBi valimist. Ametlikult veebilehelt leiti, et MySQLi saab asendada MySQL-põhiste andmebaasidega nagu MariaDB [46]. Segaduse vältimiseks oli MySQL ümber nimetatud. MySQLi asemel kirjutati MariaDB.

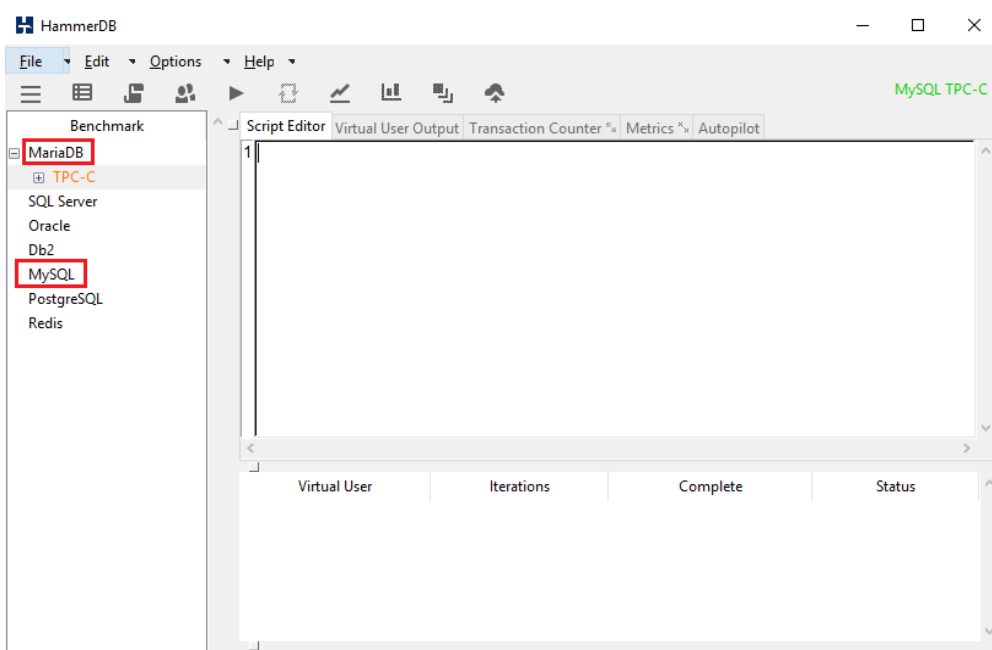
Kui valesti seadistada faili *database.xml*, tuleb HammerDBi konsoolis viga „*Unmatched Background Error - can't read "configmariadb": no such variable*”, mis tekkis virtuaalsete kasutajate arvu muutumisel. Viga seisneb selles, et `<name> </name>` siltide vahel on kirjas MariaDB, aga pidi olema MySQL.

Allpool on näidatud koodi jupp, mis sisaldab vigu.

```
<mysql>
<name>MariaDB</name>
<description>MariaDB</description>
<prefix>mysql</prefix>
<library>mysqldb</library>
<workloads>TPC-C TPC-H</workloads>
<commands>mysql::sel mysqluse mysqlescape mysqlsel mysqlnext mysqlseek mysqlmap
mysqlexec mysqlclose mysqlinfo mysqlresult mysqlcol mysqlstate mysqlinsertid mysqlquery
mysqlendquery mysqlbaseinfo mysqlping mysqlchangeuser mysqlreceive</commands>
</mysql>
```

Selleks, et antud kood töötaks, on vaja muuta „mysql” siltideks „mariadb”, teha uus fail *config* kaustas nimega *mariadb.xml*. Seadistamise jaoks peab kood asuma `<mariadb></mariadb>` siltide vahel. Kaustas *src* on vaja teha kaust *mariadb*, mis sisaldab *.tcl* laiendiga failie. Failid *.tcl*

laiendiga võib võtta *mysql* kaustast ja vajadusel muuta (võib võtta *mysql* kaustast, kuna MariaDB on tehtud MySQL baasil). Muudetud failide näide asub lisas number IX. Alltoodud joonis 3 näitab muutusi liideses. Liideses on asendatud “MariaDB” väärtusega “MySQL”.



Joonis 3: Muudatused kasutajaliideses

Selleks, et vältida “*Unmatched Background Error*” viga, ei tohi muuta tähendust, mis asub `<name></name>` siltide vahel. Selleks on vaja sisestada soovitud kirjeldus siltide `<description>` ja `</description>` “*database.xml*” failis [54].

Teine probleem tekkis TPC-H põhise jõudlustestimise käivitamisel PostgreSQL’is. Pääringute täitmise aeg oli umbes 10 tundi. Täpne põhjus ei selgunud. Võimalik põhjus on halb PostgreSQL’i või pääringute optimeerimine.

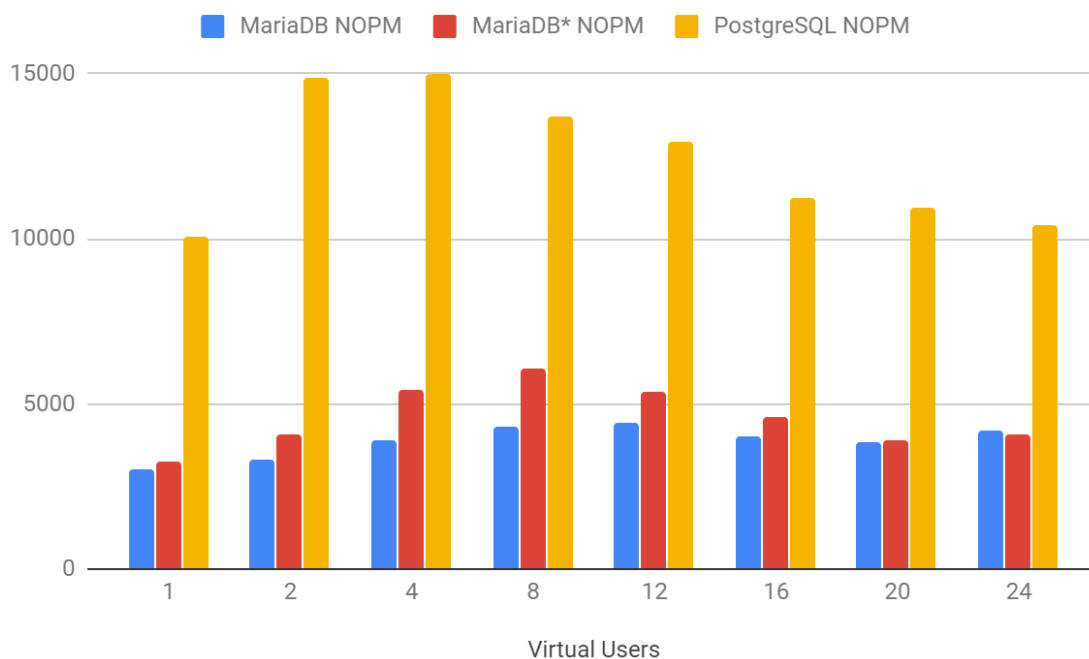
### 3. Tulemuste analüüs

Antud peatükis kirjeldatakse saadud tulemusi, viiakse läbi nende analüüs ning võrreldakse andmebaaside juhtimissüsteemide jõudlust. Käesoleva töö raames oli tehtud MariaDB ja PostgreSQL jõudluse mõõtmine. MariaDBi lähtekoodi oli muudetud, seega edaspidi „MariaDB\*“ tähendab, et MariaDB lähtekoodi oli muudetud (viivituse kordaja on muudetud).

#### 3.1 TPC-C tulemuste analüüs

Joonisel 4 on esitatud diagramm, kus on esimese mõõtmise tulemused. Seal on näidatud MariaDBi, MariaDBi\* ja PostgreSQLi tulemused. Andmebaasi juhtimissüsteemi parameetrid ei olnud seadistatud.

Diagrammilt on näha, et PostgreSQL teeb uut tellimust minutis (NOPM) kiiremini, kui MariaDB ja MariaDB\*. PostgreSQL saavutab maksimumi, kui on 2 kuni 4 virtuaalkasutajat. Keskmiselt PostgreSQL on kiirem kui MariaDB 3,19 korda ja kiirem kui MariaDB\* 2,69 korda. Kõik tulemused on lisas number X.



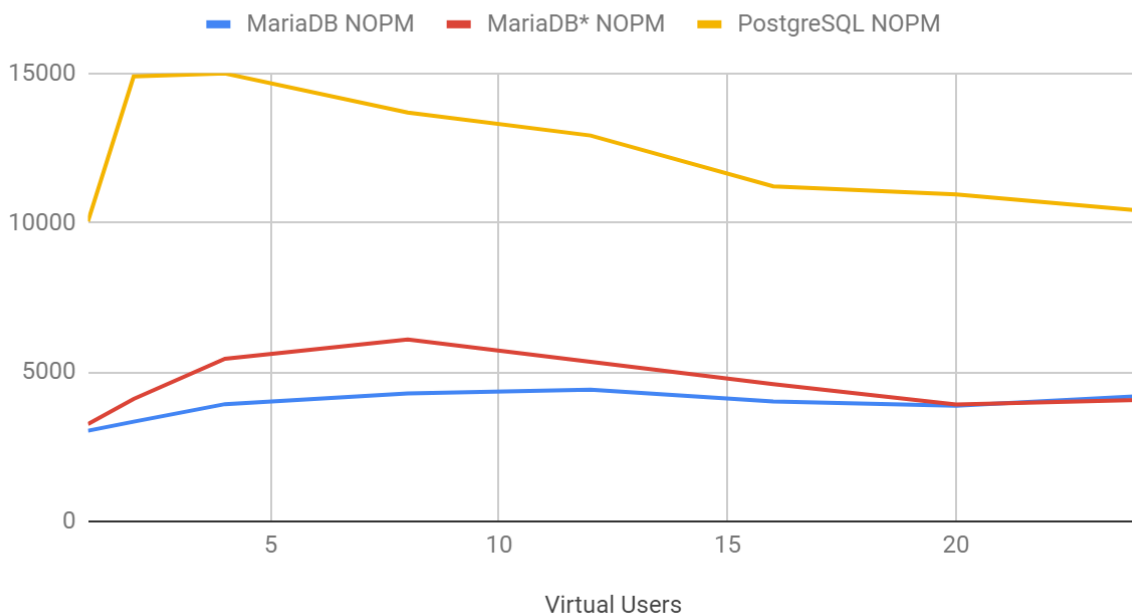
Joonis 4: Esimese mõõtmise tulemused.



Vaadeldakse nüüd andmebaasi MariaDB eraldi. Diagrammilt 4 on näha, et MariaDB ja MariaDB\* saavutab maksimumi, kui on 8 virtuaalkasutajat. MariaDBi\* tulemused on lähedased MariaDBi omadega, kui virtuaalkasutajate arv on 2 ja 16 vahel. Keskmiselt on MariaDB\* 1,18 korda kiirem ehk viivituse kordaja muutumine tõstis MariaDBi jõudlust 1,18 korda.

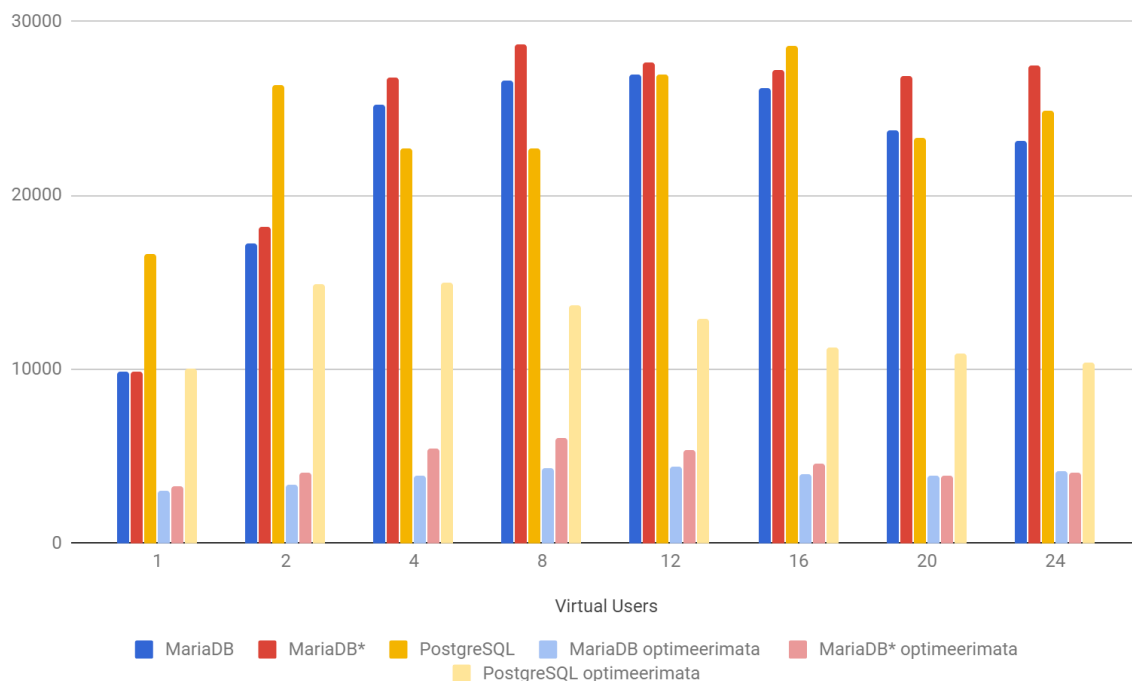
Joonisel 5 on esitatud graafik, mis näitab, kuidas muutub NOPMi näitaja. Graafikult on näha, et PostgreSQLi NOPMi näitaja kasvab alguses kiiresti ja siis, kui virtuaalkasutajate arv on suurem kui 4, väheneb. MariaDBi ja MariaDBi\* näitajad kasvavad ja langevad sujuvamalt kui PostgreSQLi näitaja.

MariaDB NOPM, MariaDB\* NOPM и PostgreSQL NOPM



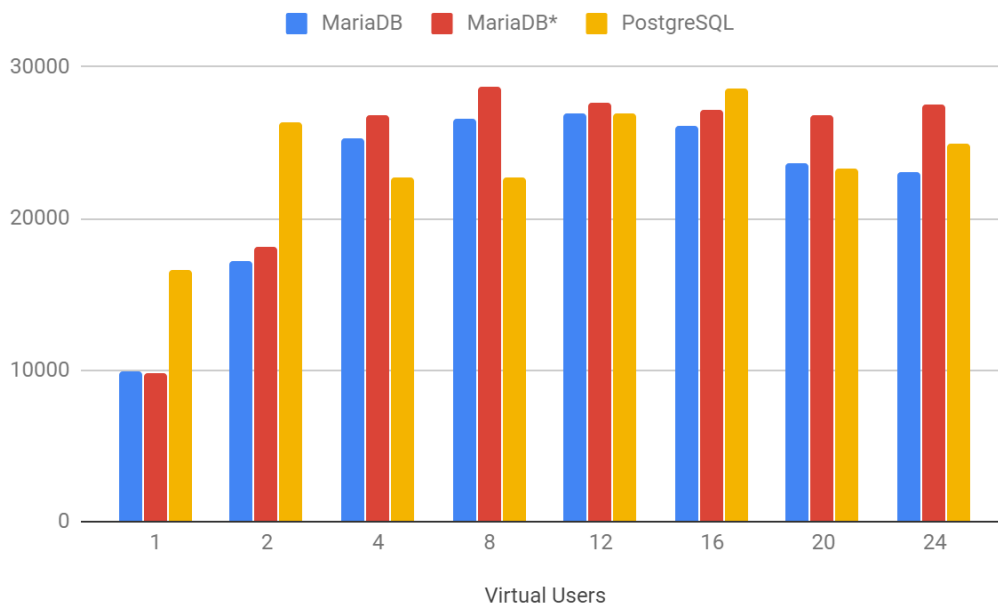
Joonis 5: NOPM näitaja sõltuvus virtuaalkasutajate arvust

Joonisel 6 on esitatud tulemused pärast andmebaasi juhtimissüsteemi parameetrite muutumist koos tulemusega, mis oli saadud ilma parameetrite muutmiseta. Diagrammilt on näha, et tulemused kasvasid mitu korda. MariaDBi tulemused kasvasid keskmiselt 5,76 korda, MariaDBi\* tulemused kasvasid keskmiselt 5,24 korda ja PostgreSQLi tulemused kasvasid keskmiselt 1,94 korda. Seega andmebaasi juhtimissüsteemi jõudlust võib tõsta parameetrite muutmisega (lisatud parameetrid on lisas number V ja VI).



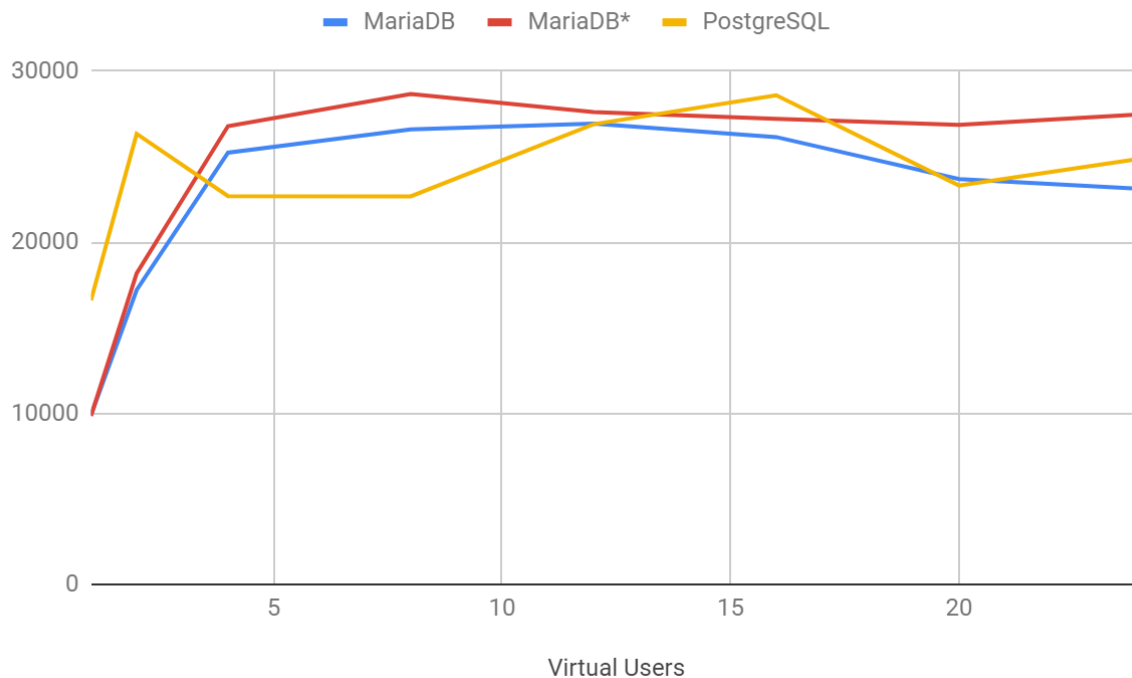
Joonis 6: Võrdlemise esimesed ja teised mõõtmise tulemused.

Vaatleme saadud tulemusi eraldi joonisel 7. Jooniselt 7 on samuti näha, et andmebaasi juhtimissüsteemide jõudlused erinevad vähem kui varem. Kõige paremat tulemust näitas MariaDB\*. MariaDB töötas keskmiselt natuke (1.0032 korda) kiiremini kui PostgreSQL. Kuna HammerDB mõõdab täpsusega 1%, aga antud juhul MariaDB\* näitas tulemusi, mis on 0,3% suurem, võib öelda, et MariaDB\* ja PostgreSQL andsid sama tulemuse. Keskmiselt näitas MariaDB\* 1,08 korda paremat tulemust kui MariaDB. PostgreSQL saavutas maksimumi, kui oli 2 või 16 virtuaalkasutajat. MariaDB ja MariaDB\* saavutasid maksimumi, kui oli 8 virtuaalkasutajat.



Joonis 7: Teise mõõtmise tulemused.

Joonisel 8 on esitatud graafik, mis näitab, kuidas muutub NOPM näitaja.



joonis 8: NOPM näitaja sõltuvus virtuaalkasutajate arvust

Graafikust on näha, et PostgreSQL võrreldes MariaDB ja MariaDB\* andmebaasidega ei tööta ühtlaselt. PostgreSQL näitaja kasvab järsku ja siis langeb ühtlaste intervallide kaupa.

Saadetud tulemusest on nähe, et PostgreSQL ja MariaDB jõudlust sõltub seadistamist. Õige seadistamine võib tõsta andmebaasi juhtimissüsteemi jõudlust mitmekordselt. PostgreSQL ja MariaDB\* näitasid võrdseid tulemusi. MariaDBi ja MariaDBi\* tulemustes ei olnud palju erinevusi.

### 3.2 TPC-H tulemuste analüüs

Päringute täitmise aeg tuli väga erinev, seega andmebaaside vahel võrreldakse ainult summaarset käivitamise aega. Võrreldakse, kuidas muutus summaarne täitmise aeg pärast andmebaasi juhtimissüsteemi optimeerimist. Käivitatud päringud on lisas number III. TPC-Hi tulemused on lisas number XI. Käivitatud päringud on lisas number III. Tabelis 4 on summaarne täitmise aeg sekundites (sek). „MariaDB\*“ tähendab, et MariaDB lähtekoodi oli muudetud.

Tabel 4: Summaarne täitmise aeg

	MariaDB* optimeerimata	MariaDB* optimeeritud	MariaDB optimeerimata	MariaDB optimeeritud	PostgreSQL optimeerimata	PostgreSQL optimeeritud
<b>Summaarne täitmise aeg (sek)</b>	177,011	62,336	1226,556	1114,892	41973,578	37670,284

Tabelist 4 on näha, et PostgreSQLis summaarne täitmise aeg on kõige suurem, mida põhjustasid päringud 17, 20 ja 21 (vt lisa III). Need päringud (17, 20, 21) moodustavad 99,86% kogu täitmise ajast. Täpne põhjus, miks see juhtus, on teadmata. TPC-H-põhine jõudlustest oli käivitatud PostgreSQLis jaoks uuesti, aga saadi samad tulemused.

Tabelis 5 on näidatud päringute täitmise aeg, kui pole võetud arvesse 3. päringut, mille täitmise aeg kõige suurem.

Tabel 5: Täitmise aeg.

	MariaDB* optimeerimata	MariaDB* optimeeritud	MariaDB optimeerimata	MariaDB optimeeritud	PostgreSQL optimeerimata	PostgreSQL optimeeritud
<b>Täitmise aeg</b>	99,091	32,735	319,997	318,726	60,962	54,61

Tabel 6 näitab, kuidas muutus täitmise aeg pärast optimeerimist sekundites (sek) ja protsentides (%).

Tabel 6: Summaarne täitmise aja muutmine.

	<b>MariaDB*</b>	<b>MariaDB</b>	<b>PostgreSQL</b>
<b>Vahe (sek)</b>	114,675	111.664	4303,294
<b>Vahe (%)</b>	64.78%	9%	10,25%

Tabelist 6 on näha, et MariaDB\* ja MariaDB päringute täitmise aja vähendamine sekundis sarnane.

Pärast optimeerimist päringute täitmise aeg on vähenenud (vt lisa XI). TPC-H-põhisel jõudlustestil näitas MariaDB\* parimat tulemust, seega võib öelda, et viivituse kordaja muutumist võib väheneda päringu täitmise aeg.

## 4. Kokkuvõte

Käesolevas töös võrreldakse PostgreSQLi ja MariaDBi juhtimissüsteeme. Võrreldakse järgmiseid parameetreid: andmemaht, platvormi tugi ja turvalisus.

Info andmebaaside juhtimissüsteeme võrdlemiseks on võetud ametlikust dokumentatsioonist ja ametlikult wiki lehelt.

Töö käigus mõõdeti jõudlust järgmistel andmebaasi juhtimissüsteemidel: PostgreSQL, MariaDB ja MariaDB, mille lähtekoodi oli muudetud. MariaDBi lähtekoodis oli muudetud viivituse kordaja, kuna leiti informatsiooni, et viivituse kordaja võib põhjustada aeglast tööd. Lähtekoodis viivituse kordaja 50 asemel pandi väärtus 5 ja seejärel kood kompileeriti uuesti. Töö käigus kirjeldati ka andmebaasi juhtimissüsteemi seadistamist ja näidati, kuidas kompileerida andmebaasi MariaDB. Samuti kirjeldatakse esinenud probleeme, nende lahendusi ja võimalikke põhjuseid.

Jõudluse mõõtmiseks kasutati tööriista nimega HammerDB. Töö raames kirjeldatakse jõudluse mõõtmise protsessi, seletatakse sellega seotud mõisteid nagu TPC-C ja TPC-H ning kirjeldatakse HammerDB töö põhimõtteid. Näidatakse, kuidas genereerida andmeid HammerDB abil.

Jõudluse mõõtmine oli tehtud TPC-C- ja TPC-H-põhiste testidega. TPC-Ci väärtus näitab transaktsioonide arvu minutis. TPC-Hi meetodit kasutati, et leida iga päringu täitmise aeg. Jõudluse mõõtmise järel analüüsiti tulemusi.

Käesoleva töö jaoks on mitmeid potentsiaalseid edasiarendusi. Selleks võib uurida, kuidas päringuid optimeerida, teha HammerDBi koodi läbivaatamine (ingl *code review*) ja kirjeldada antud tööriista tehnilisi näitajaid ning kirjeldada stressitestimise protsessi.

## 5. Viidatud kirjandus

- [1] “MySQL :: MySQL Downloads.” [Võrgumaterjal]. <https://www.mysql.com/downloads/>. (03.05.2019).
- [2] “DB-Engines Ranking - popularity ranking of database management systems,” DB-Engines Ranking. [Võrgumaterjal]. <https://db-engines.com/en/ranking>. (03.05.2019).
- [3] “PostgreSQL: About.” [Võrgumaterjal]. <https://www.postgresql.org/about/>. (03.05.2019).
- [4] “PostgreSQL: License.” [Võrgumaterjal]. <https://www.postgresql.org/about/licence/>. (03.05.2019).
- [5] “About MariaDB - MariaDB.org.” [Võrgumaterjal]. <https://mariadb.org/about/>. (03.05.2019).
- [6] “Licensing FAQ - MariaDB Knowledge Base.” [Võrgumaterjal]. <https://mariadb.com/kb/en/library/licensing-faq/>. (03.05.2019).
- [7] “Why does MariaDB 10.2 use InnoDB instead of XtraDB? - MariaDB Knowledge Base.” [Võrgumaterjal]. <https://mariadb.com/kb/en/library/why-does-mariadb-102-use-innodb-instead-of-xtradb/>. (03.05.2019).
- [8] “About XtraDB - MariaDB Knowledge Base.” [Võrgumaterjal]. <https://mariadb.com/kb/en/library/about-xtradb/>. (03.05.2019).
- [9] “MariaDB Limitations - MariaDB Knowledge Base.” [Võrgumaterjal]. <https://mariadb.com/kb/en/library/mariadb-limitations/>. (03.05.2019).
- [10] “which is the max storage capacity of Version 5.5 MariaDB? - MariaDB Knowledge Base.” [Võrgumaterjal]. <https://mariadb.com/kb/en/library/which-is-the-max-storage-capacity-of-version-55-mariadb/>. (03.05.2019).
- [11] “database - MySQL: What is a page? - Stack Overflow.” [Võrgumaterjal]. <https://stackoverflow.com/questions/4401910/mysql-what-is-a-page>. (03.05.2019).
- [12] “InnoDB Limitations - MariaDB Knowledge Base.” [Võrgumaterjal]. <https://mariadb.com/kb/en/library/innodb-limitations/>. (03.05.2019).
- [13] “FAQ - PostgreSQL wiki.” [Võrgumaterjal]. [https://wiki.postgresql.org/wiki/FAQ#What\\_is\\_the\\_maximum\\_size\\_for\\_a\\_row.2C\\_a\\_table.2C\\_and\\_a\\_database.3F](https://wiki.postgresql.org/wiki/FAQ#What_is_the_maximum_size_for_a_row.2C_a_table.2C_and_a_database.3F). (03.05.2019).
- [14] “MVCC - PostgreSQL wiki.” [Võrgumaterjal]. <https://wiki.postgresql.org/wiki/MVCC>. (03.05.2019).
- [15] “PostgreSQL: Documentation: 10: CREATE TABLE.” [Võrgumaterjal]. <https://www.postgresql.org/docs/10/sql-createtable.html>. (03.05.2019).

- [16] “PostgreSQL: Documentation: 10: 16.6. Supported Platforms.” [Võrgumaterjal]. <https://www.postgresql.org/docs/10/supported-platforms.html>. (03.05.2019).
- [17] “PostgreSQL BuildFarm.” [Võrgumaterjal]. <https://buildfarm.postgresql.org/>. (03.05.2019).
- [18] “MariaDB Deprecation Policy - MariaDB Knowledge Base.” [Võrgumaterjal]. <https://mariadb.com/kb/en/library/deprecation-policy/#current-package-platforms>. (03.05.2019).
- [19] “What is Database Security? - Definition from Techopedia.” [Võrgumaterjal]. <https://www.techopedia.com/definition/29841/database-security>. (03.05.2019).
- [20] “PostgreSQL: Documentation: 11: 18.9. Secure TCP/IP Connections with SSL.” [Võrgumaterjal]. <https://www.postgresql.org/docs/current/ssl-tcp.html>. (03.05.2019).
- [21] “PostgreSQL: Documentation: 11: Chapter 21. Database Roles.” [Võrgumaterjal]. <https://www.postgresql.org/docs/11/user-manag.html>. (03.05.2019).
- [22] “SEPostgreSQL SELinux Overview - PostgreSQL wiki.” [Võrgumaterjal]. [https://wiki.postgresql.org/wiki/SEPostgreSQL\\_SELinux\\_Overview](https://wiki.postgresql.org/wiki/SEPostgreSQL_SELinux_Overview). (03.05.2019).
- [23] “MariaDB & Database Security | MariaDB.” [Võrgumaterjal]. <https://mariadb.com/resources/blog/mariadb-database-security/>. (03.05.2019).
- [24] “Performance Testing Types, Steps, Best Practices, and Metrics.” [Võrgumaterjal]. <https://stackify.com/ultimate-guide-performance-testing-and-software-testing/>. (03.05.2019).
- [25] A. B. Bondi, “Characteristics of scalability and their impact on performance,” in Proceedings of the second international workshop on Software and performance - WOSP '00, New York, New York, USA, 2000, pp. 195–203.
- [26] “Performance Testing Tutorial: What is, Types, Metrics & Example.” [Võrgumaterjal]. <https://www.guru99.com/performance-testing.html>. (03.05.2019).
- [27] “Database Testing Performance (Load, Stress).” [Võrgumaterjal]. [https://www.tutorialspoint.com/database\\_testing/database\\_testing\\_performance.htm](https://www.tutorialspoint.com/database_testing/database_testing_performance.htm). (03.05.2019).
- [28] “Load Testing Tutorial: What is? How to? (with Examples).” [Võrgumaterjal]. <https://www.guru99.com/load-testing-tutorial.html>. (03.05.2019).
- [29] “Load Testing Metrics Explained - LoadStorm.” [Võrgumaterjal]. <https://loadstorm.com/load-testing-metrics/>. (03.05.2019).
- [30] “What is STRESS Testing in Software Testing? Tools, Types, Examples.” [Võrgumaterjal]. <https://www.guru99.com/stress-testing-tutorial.html>. (03.05.2019).
- [31] “TPC - About the TPC.” [Võrgumaterjal]. <http://www.tpc.org/information/about/abouttpc.asp>. (03.05.2019).



- [32] “2. What is the TPC and TPC-C?” [Vörgumaterjal]. <https://www.hammerdb.com/docs/ch03s02.html>. (03.05.2019).
- [33] “TPC - Who We Are.” [Vörgumaterjal]. <http://www.tpc.org/information/who/howeare.asp#auditors>. (03.05.2019).
- [34] “Open Source Database Testing Tools.” [Vörgumaterjal]. <https://www.softwaretesting-magazine.com/tools/open-source-database-testing-tools/>. (03.05.2019).
- [35] “HammerDB.” [Vörgumaterjal]. <https://www.hammerdb.com/>. (03.05.2019).
- [36] “3. HammerDB Transactional TPC-C based workloads.” [Vörgumaterjal]. <https://www.hammerdb.com/docs/ch03s03.html>. (03.05.2019).
- [37] “4. Comparing HammerDB results.” [Vörgumaterjal]. <https://www.hammerdb.com/docs/ch03s04.html>. (03.05.2019).
- [38] “Why both TPM and NOPM Performance Metrics? – HammerDB Blog.” [Vörgumaterjal]. <https://www.hammerdb.com/blog/uncategorized/why-both-tpm-and-nopm-performance-metrics/>. (03.05.2019).
- [39] “TPC-C - Homepage.” [Vörgumaterjal]. <http://www.tpc.org/tpcc/>. (03.05.2019).
- [40] “5. Understanding the TPC-C workload.” [Vörgumaterjal]. <https://www.hammerdb.com/docs/ch03s05.html>. (03.05.2019).
- [41] “TPC BENCHMARKTM C” [Vörgumaterjal]. [http://www.tpc.org/tpc\\_documents\\_current\\_versions/pdf/tpc-c\\_v5.11.0.pdf](http://www.tpc.org/tpc_documents_current_versions/pdf/tpc-c_v5.11.0.pdf). (03.05.2019).
- [42] “1. What is TPC-H?” [Vörgumaterjal]. <https://www.hammerdb.com/docs/ch09s01.html>. (03.05.2019).
- [43] “TPC BenchmarkTM H.” [Vörgumaterjal], [http://www.tpc.org/tpc\\_documents\\_current\\_versions/pdf/tpc-h\\_v2.18.0.pdf](http://www.tpc.org/tpc_documents_current_versions/pdf/tpc-h_v2.18.0.pdf). (03.05.2019)
- [44] “PostgreSQL: PostgreSQL 11 Released!” [Vörgumaterjal]. <https://www.postgresql.org/about/news/1894/>. (03.05.2019).
- [45] “Changes & Improvements in MariaDB 10.3 - MariaDB Knowledge Base.” [Vörgumaterjal]. <https://mariadb.com/kb/en/library/changes-improvements-in-mariadb-103/>. (03.05.2019).
- [46] “HammerDB MySQL and MariaDB Best Practice for Performance and Scalability – HammerDB Blog.” [Vörgumaterjal]. <https://www.hammerdb.com/blog/uncategorized/hammerdb-mysql-and-mariadb-best-practice-for-performance-and-scalability/>. (03.05.2019).
- [47] “Building MariaDB on Windows - MariaDB Knowledge Base.” [Vörgumaterjal]. [https://mariadb.com/kb/en/library/Building\\_MariaDB\\_on\\_Windows/](https://mariadb.com/kb/en/library/Building_MariaDB_on_Windows/). (03.05.2019).
- [48] “PostgreSQL: Documentation: 11: Chapter 19. Server Configuration.” [Vörgumaterjal]. <https://www.postgresql.org/docs/11/runtime-config.html>. (03.05.2019).

- [49] “Configuring MariaDB with Option Files - MariaDB Knowledge Base.” [Vörgumaterjal]. <https://mariadb.com/kb/en/library/configuring-mariadb-with-option-files/>. (03.05.2019).
- [50] “Optimization and Tuning - MariaDB Knowledge Base.” [Vörgumaterjal]. <https://mariadb.com/kb/en/library/optimization-and-tuning/>. (03.05.2019).
- [51] “HammerDB Best Practice for PostgreSQL Performance and Scalability – HammerDB Blog.” [Vörgumaterjal]. <https://www.hammerdb.com/blog/uncategorized/hammerdb-best-practice-for-postgresql-performance-and-scalability/>. (03.05.2019).
- [52] “2. Installation and Configuration.” [Vörgumaterjal]. <https://www.hammerdb.com/docs/ch10s02.html>. (03.05.2019).
- [53] “1. Test Network Configuration.” [Vörgumaterjal]. <https://www.hammerdb.com/docs/ch04s01.html>. (03.05.2019).
- [54] “HammerDB Concepts and Architecture – HammerDB Blog.” [Vörgumaterjal]. <https://www.hammerdb.com/blog/uncategorized/hammerdb-concepts-and-architecture/>. (03.05.2019).

## 6. Lisad

### I. Terminid

**ACID** (*Atomicity, Consistency, Isolation, Durability*)

Andmebaasi transaktsioonide omaduste kogum.

**Atomaarsus** (*Atomicity*) Transaktsiooni väljakutsuja (klient, rakendus) seisukohalt on transaktsioon loogiline tervik, mis täidetakse kas täielikult või jäetakse täielikult täitmata.

**Terviklikkus**, konsistentsus (*Consistency*) Transaktsioon viib andmebaasi ühest korrektsest seisundist teise.

**Isoleeritus** (*Isolation*) Transaktsioonid ei saa piiluda üksteise vahetulemusi.

**Püsivus**, jätkuvus (*Durability*) Andmebaasisüsteem garanteerib transaktsiooni lõppemise järel selle tehtud muudatuste püsiva salvestamise.

**BLOB** (*Binary Large Object*)

Binaarne suur objekt. BLOB on suur fail, tavaliselt pildi- või helifail.

**Esmane indeks**

On väljade kogum, mis sisaldab unikaalset esmast võtit ja ei sisalda kordusi.

**Esmane võti**

On andmebaasi juhtimissüsteemi võti, mis on iga kirje jaoks unikaalne.

**GIS**

On Geograafiline infosüsteem süsteem, mis on ette nähtud kõikide geograafiliste andmete kogumiseks, salvestamiseks, manipuleerimiseks, analüüsimiseks, haldamiseks ja esitlemiseks.

**GNU**

Üldine avalik litsents, mida sageli lühendatakse GNU GPL (või lihtsalt GPL), loetleb tingimused ja tingimused vaba tarkvara kopeerimiseks, muutmiseks ja levitamiseks.

**JSON** (*JavaScript Object Notation*)

on lihtsustatud andmevahetusvorming.

**Laiendatavust**

On süsteemi võime õigesti töötada suureneva koormusega, nt. suurenenud päringute arv.

**LDAP autentimine**

On kasutajanime ja parooli kombinatsiooni serveriga kinnitamise protsess.

**Lehekülg**

Lehekülg on virtuaalse mälu plokk. Lehekülgede kohta tuleb meeles pidada, et need on virtuaalsed plokid ja neil on fikseeritud suurus.

**Lehekülje suurus**

Arvutite puhul viitab lehekülje suurus lehekülje suurusele. Lehekülje suurus mõjutab vajalikku mälumahtu ja ruumi, mida kasutatakse programmide käivitamisel. Enamik operatsioonisüsteeme võimaldab määrata lehekülje suuruse, kui programm käivitub, mis võimaldab arvutada selle programmi kasutamise ajal kõige efektiivsemat mälu kasutamist.

**Mitme veeruline indeks**

On indeks, mida saab määratleda tabelis rohkem kui ühe veeruga.

**Mittefunktsionaalne testimine**

Mittefunktsionaalsed testid keskenduvad detailidele, mis ei ole funktsionaalsusega seotud. Näiteks üldine toimimine, vastupidavus, usaldusväärsus jms. Ehk siis kas süsteem jääb töökorda kui seda kasutab palju kasutajaid samaaegselt.

**MVCC (*Multiversion concurrency control*)**

On meetod, mida PostgreSQL kasutab andmete järjepidevuse käsitlemiseks, kui mitmed protsessid kasutavad sama tabelit.

**OLTP**

On tarkvaraprogrammide klass, mis suudab toetada transaktsioonidele orienteeritud rakendusi Internetis.

**Transaktsioon**

On loogiline üksus, mida teostatakse sõltumatult andmete otsimiseks või uuendamiseks

**TEXT**

Veerg maksimaalse pikkusega 65535 tähemärki.

**TPC-C**

On jõudlusnäitaja, mis simuleerib online-tellimist.

TPC-C hõlmab viit erinevat tüüpi ja keerukate samaaegsete transaktsioonide kombinatsiooni, mis on teostatud võrgus või mis on järjekorda pandud edasilükkamiseks.

**TPC-H**

On jõudlusnäitaja, mis koosneb ärile orienteeritud päringutest mille käigus toimub samaaegne andmete muutmine.

**PAM**

On autentimisraamistik, mida kasutavad Linux, FreeBSD, Solaris ja teised Unixi sarnased operatsioonisüsteemid.

**Percona XtraDB**

On täiustatud versioon InnoDB salvestusmootorist, mis on mõeldud kaasaegsele riistvarale paremaks laiendatavuseks ja sisaldab mitmeid muid funktsioone, mis on kasulikud suure jõudlusega keskkonnas.

**Platvorm**

On CPU arhitektuur ja operatsioonisüsteemide kombinatsioon.

**Regressioonitestimine**

On üldnimetus kõikide tarkvara testimise tüüpidele, mis on suunatud vigade avastamisele varem testitud rakenduses või programmikoodis.

**Salvestusmootor**

On tarkvaramoodul, mida andmebaasi juhtimissüsteem kasutab andmebaaside andmete salvestamiseks, lugemiseks ja uuendamiseks.

**Scale Factor**

Antud töö kontekstis on parameeter, millest sõltub andmebaasi suurus.

**Sekundaarne indeks**

On indeks, mis ei ole esmane indeks ja võib sisaldada duplikaate.

**Virtual Users to Build Schema**

on HammerDB parameeter.

## **II. Transaktsioonid**

Transaktsioonide näidet asuvad tööga kaasas olevas failis “transaktsioonid.txt”.

### **III. Päringud**

Päringud asuvad tööga kaasas olevas failis "päringud.txt".

## **IV. Arvuti informatsioon**

Arvuti informatsioon asub tööga kaasas olevas failis "info.nfo".



## V. PostgreSQL parameetrid

<b>Parameeter</b>	<b>Tähendus</b>
max_connections	257
shared_buffers	4000MB
huge_pages	on
temp_buffers	2000MB
work_mem	2000MB
maintenance_work_mem	512MB
autovacuum_work_mem	-2
max_stack_depth	3MB
max_files_per_process	4001
synchronous_commit	off
wal_buffers	512MB
max_locks_per_transaction	257
max_pred_locks_per_transaction	266

## VI. MariaDB parameetrid

Parameeter	Tähendus
max_connections	4000
table_open_cache	8000
table_open_cache_instances	16
back_log	1500
ssl	0
performance_schema	OFF
max_prepared_stmt_count	128000
skip_log_bin	1
character_set_server	latin1
collation_server	latin1_swedish_ci
transaction_isolation	REPEATABLE-READ
innodb_log_files_in_group	32
innodb_open_files	4000
innodb_buffer_pool_size	6400M
innodb_buffer_pool_instances	16
innodb_log_buffer_size	64M
innodb_doublewrite	0
innodb_thread_concurrency	0
innodb_flush_log_at_trx_commit	0
innodb_max_dirty_pages_pct	90
innodb_max_dirty_pages_pct_lwm	10
join_buffer_size	32K
sort_buffer_size	32K
innodb_use_native_aio	1
innodb_stats_persistent	1
innodb_spin_wait_delay	6
innodb_max_purge_lag_delay	300000
innodb_max_purge_lag=0	0
innodb_flush_method	O_DIRECT_NO_FSYNC

innodb_checksum_algorithm	none
innodb_io_capacity	4000
innodb_io_capacity_max	20000
innodb_lru_scan_depth	9000
innodb_change_buffering	none
innodb_read_only	0
innodb_page_cleaners	4
innodb_undo_log_truncate	off
innodb_adaptive_flushing	1
innodb_flush_neighbors	0
innodb_read_io_threads	16
innodb_write_io_threads	16
innodb_purge_threads	4
innodb_adaptive_hash_index	0

## **VII. Driven Script**

Driven Script asub tööga kaasas olevas RAR failis “driven\_script.rar”.

## **VIII. Sammud TPC-C käivitamiseks**

Sammud TPC-C käivitamiseks asuvad tööga kaasas olevas RAR failis "sammud.rar".

## **IX. Muudetud failid**

Muudetud failid asub tööga kaasas olevas RAR failis “hammerdb\_muudetud\_failid.rar”,

## **X. TPC-C tulemused**

TPC-C tulemused asuvad tööga kaasas olevas failis "Tulemused\_tpcc.pdf".

## **XI. TPC-H tulemused**

TPC-H tulemused asuvad tööga kaasas olevas failis päringu\_täitmise\_aeg.pdf



## **XII. Litsents**

### **Lihtlitsents lõputöö reprodutseerimiseks ja üldsusele kättesaadavaks tegemiseks**

Mina, Vladislav Buivol,

*(autori nimi)*

1. annan Tartu Ülikoolile tasuta loa (lihtlitsentsi) enda loodud teose

Andmebaasi juhtimissüsteemi võimekuse mõõtmine,

*(lõputöö pealkiri)*

mille juhendaja on Vambola Leping,

*(juhendaja nimi)*

reprodutseerimiseks eesmärgiga seda säilitada, sealhulgas lisada digitaalarhiivi DSpace kuni autoriõiguse kehtivuse lõppemiseni.

2. Annan Tartu Ülikoolile loa teha punktis 1 nimetatud teos üldsusele kättesaadavaks Tartu Ülikooli veebikeskkonna, sealhulgas digitaalarhiivi DSpace kaudu Creative Commons'i litsentsiga CC BY NC ND 3.0, mis lubab autorile viidates teost reprodutseerida, levitada ja üldsusele suunata ning keelab luua tuletatud teost ja kasutada teost ärieesmärgil, kuni autoriõiguse kehtivuse lõppemiseni.

3. Olen teadlik, et punktides 1 ja 2 nimetatud õigused jäävad alles ka autorile.

4. Kinnitan, et lihtlitsentsi andmisega ei riku ma teiste isikute intellektuaalomandi ega isikuandmete kaitse õigusaktidest tulenevaid õigusi.

Vladislav Buivol

**10.05.2019**