

<https://doi.org/10.5281/zenodo.7800251>

Abdixalilova Gulshanoy Dilshodjon qizi

Xoshimova Gulasal Abror qizi

Siddiqova Zuxraxon Toxirjon qizi

Farg`ona Davlat Universiteti IV bosqich bakalavr talabalari

Farg`ona Davlat Universiteti katta o'qituvchisi

Tojimatov Isroil

Abstract - *Hozirgi tadqiqot NoSQL ma'lumotlar bazalarining mavjud bo'lgan turli shakllarni o'rganadi va farqlaydi. U NoSQLga bo'lgan ehtiyojni va uning relyatsion ma'lumotlar bazalariga qanday darajada bog'liqligini o'rganadi. NoSQL ma'lumotlar bazalarini to'rtta asosiy tasnifga bo'lish mumkin: asosiy qiymatlar do'konlari, grafik ma'lumotlar bazalari, keng ustunli do'konlar va hujjatlar do'konlari. Bu toifalar funksional va funksional bo'lmagan xususiyatlar asosida taqqoslanadi. Funksional bo'lmagan xususiyatlar unumdorlik, miqyoslilik, moslashuvchanlik, tuzilish va murakkablikni o'z ichiga oladi. Funksional xususiyatlar normalizatsiya, qo'shilish, markazlashganlik, yig'ish va kalitlarni o'z ichiga oladi. Keyinchalik tahlil qilish uchun ushbu toifalarning har biridan bitta ma'lumotlar bazasi tanlanadi, ya'ni MongoDB (hujjatlar do'konlari), Cassandra (keng ustunli do'konlar), Redis (asosiy qiymatlar do'konlari) va Neo4j (grafik ma'lumotlar bazalari). Tanlangan ma'lumotlar bazalari ularning ma'lumotlar modeli, CAP teoremasi, tarqatish xususiyatlari va boshqa omillar bo'yicha taqqoslanadi. Funksional bo'lmagan xususiyatlar bo'yicha taqqoslashni amalga oshirish orqali, agar yuqori unumdorlik, moslashuvchanlik va miqyoslilik talab qilinsa va biz JSON formatida ma'lumotlarni taqdim etgan bo'lsak, hujjatlar do'konidan foydalanish mumkinligi aniqlandi. Ustunlar do'konidan yuqori unumdorlik va miqyoslilikni talab qiluvchi yarim tuzilgan ma'lumotlar uchun foydalanish mumkin. Redis anin xotira ombori bo'lib, bitta parcha bilan ishlashda juda tez ishlaydi. Grafik ma'lumotlar bazalari o'zaro bog'langan ma'lumotlar va doimiy rivojlanayotgan ma'lumotlar modellari haqida gap ketganda foydalanish mumkin. MongoDB, Cassandra, Redis va Neo4j o'rtasidagi taqqoslash shuni ko'rsatdiki, ularning barchasi gorizontol o'lchovga amal qiladi va sxemasiz. Neo4j dan tashqari, boshqalar to'liq ACID xususiyatlariga ega emas. MongoDB, Redis va Cassandra ma'lumotlar bazalari uchun yozish va o'chirish operatsiyalari tez, Kassandrada esa o'qish jarayoni nisbatan sekin. Neo4j bo'lsa, REST ishlashi MongoDB ga o'xshaydi, o'rnatilgan esa nisbatan sekin. Shuningdek, biz ushbu ma'lumotlar bazalarining taqsimlangan muhitda qanday ishlashini muhokama qilamiz.*

Kalit so'zlar: *ma'lumotlar bazasi; NoSQL; taqqoslash; ma'lumotlar bazasi tizimlari; MongoDb; Cassandra; Redis; Neo4j.*

I. Kirish

Hozirgi kunda tarqalgan web-ilovalar va bulutli texnologiyalardagi so'nggi yutuqlar bitta tugunli tizimlar tomonidan boshqarishning imkoni bo'lmagan juda katta hajmdagi ma'lumotlarni yaratdi. Shunday qilib, saralab saqlash va yuqori tezlikdagi ishni ta'minlaydigan

va zarur bo'lgan darajdagi yechimlarni taklif qiladigan tizimlarga ehtiyojlar ortdi. Bunday tizimlarga (aloqaviy bo'lmagan saqlash) misollari Amazon tomonidan Dynamo va Google'ning Big Table ni keltirishimiz mumkin.

A. Relyatsion ma'lumotlar bazasi

Dastlab, har bir yozuv qo'lda saqlangan, ammo texnologiyaning paydo bo'lishi yillar davomida keskin o'zgarishlarga olib keldi. Ma'lumotlarni saqlashni osonlashtirish uchun ma'lumotlar bazalari yaratildi. Ma'lumotlar bazasi oddiy matnli hujjatdan ancha murakkab ma'lumotlar bazalarigacha o'zgaradi. Ushbu ma'lumotlar bazalari har qanday ortiqcha, nomuvofiq yoki ma'lumotlarni olib tashlash uchun vaqti-vaqti bilan takomillashtirilishi kerak samarali ishlashi uchun iflos ma'lumotlar. Ushbu ma'lumotlarni saqlashning eng keng tarqalgan, taniqli kontseptsiyasi relyatsion modeldir. Strukturaviy so'rovlar tili (SQL) ma'lumotlar bazasidan tegishli ma'lumotlarni chiqaradi.

Relyatsion ma'lumotlar bazalari soddaligi tufayli ma'lumotlar bazasining eng keng tarqalgan turi hisoblanadi. RDBMSda ma'lumotlar bir nechta jadvallarga bo'linadi, ularga jadvalga o'zgartirishlar kiritmasdan talablarga muvofiq kirish mumkin. Birlashtirish, yig'ish, qo'shish, yaratish, qidirish va o'chirish kabi operatsiyalar relyatsion ma'lumotlar bazalarida osonlik bilan amalga oshiriladi va mavjud jadvallarni kengaytirish yoki o'zgartirish ham juda oson.

Misollar: SQL Server, Oracle Database va MySQL

B. NoSQL ma'lumotlar bazalari nima uchun ishlatiladi?

O'sib borayotgan ma'lumotlarning asosiy muammosi uning bir xil emasligidir. Ushbu muammo tufayli so'nggi yillarda sanoatning ortib borayotgan ehtiyojlarini ko'paytirish uchun relyatsion bo'lmagan ma'lumotlar bazasi zarur va shu bilan birga yuqori samarali bo'lishi kerak. Bu yuqori darajada kengaytiriladigan, samarali va katta hajmdagi ma'lumotlarni saqlashi mumkin bo'lgan NoSQL ma'lumotlar bazalarini keltirib chiqardi.

RDBMS barcha uch turdagi ma'lumotlarni, ya'ni tuzilgan, yarim tizimli va tuzilmagan ma'lumotlarni boshqarishga qodir bo'lsada, ammo tuzilmagan va yarim tizimli ma'lumotlarni samarali saqlashga erishish uchun mehnat va tizimga moslashishni talab etadi. RDBMS tuzilgan ma'lumotlarni shunday saqlaydi, chunki ular allaqachon kerakli shaklda. Biroq, yarim tizimli ma'lumotlarni saqlash bir nechta murakkablikni o'z ichiga oladi. Yarim tuzilgan ma'lumotlar saqlashdan oldin relyatsion ma'lumotlarga aylantirilishi kerak. Bundan tashqari, tuzilmagan ma'lumotlar bo'lsa, ma'lumotlar blok ob'ekti sifatida saqlanadi va to'g'ridan to'g'ri saqlanmaydi.

Shunday qilib, ma'lumotlarning bir xil emasligini qondirish uchun ma'lumotlarni saqlashga yangi fikr berildi, bu esa NoSQL (nafaqat SQL) ma'lumotlar bazalarini yaratishga olib keldi.

NoSQL ma'lumotlar bazalari relyatsion ma'lumotlar bazalarining muhim o'rnini bosuvchi vosita sifatida paydo bo'ldi va biz ularni masshtablilik, mavjudlik va xatolarga chidamlilik kabi xususiyatlarga ko'ra tanlaymiz. Ular barcha RDBMSlar tomonidan qo'llaniladigan umumiy jadval/satr/ustun yondashuviga amal qilmaydi. NoSQLlar asosan taqsimlangan yoki aloqador bo'lmagan ma'lumotlar bazasi deb ataladi. Ular gorizontaal o'lchovni qo'llab-quvvatlaydi, shuning uchun vertikal o'lchov amalga oshiriladigan RDBMSda sodir bo'ladigan tizimning apparat ta'minotini yangilash o'rniga serverlar soni ko'paytiriladi.

C. NoSQL ning ahamiyati

NoSQL ma'lumotlar bazalari katta, xilma-xil va doimiy o'zgaruvchan ma'lumotlar to'ypamlarini boshqarishga qaratilgan. Ular ko'pincha taqsimlangan tizimlarda yoki bulutli ma'lumotlar bazalarida qo'llaniladi. In NoSQL ma'lumotlar bazalarining qat'iy sxemalari va boshqa ko'plab cheklovlardan qoching. Ular dastlab ma'lumotlar bazalari sifatida uzoq vaqtdan beri mavjud bo'lgan relyatsion ma'lumotlar bazalariga muqobil bo'lish uchun kiritilgan. NoSQL ma'lumotlar bazalari asosiy saqlash tizimlarining xizmatlari va imkoniyatlariga tayanadi. Ushbu NoSQL ma'lumotlar bazalari uchun masshtablik, xatolarga chidamlilik va mavjudlik qaror qabul qilishning eng muhim omillari hisoblanadi. Ular RDBMS ning qat'iy sxema yondashuviga amal qilmaydi NoSQL-lar relyatsion ma'lumotlar bazalaridan ma'lum ustunlikka ega, chunki ular katta ma'lumotlar bilan samarali kurashish, yuqori tezlikni ta'minlash va turli xil murakkablikdagi ma'lumotlarni qayta ishlashga qodir. Ular gorizontaal ravishda kengaytirilishi mumkin bo'lganligi sababli, ularni boshqarish ham sodda, buni klasterga yukni samarali boshqaradigan yangi tugunni qo'shish orqali amalga oshirish mumkin. Muvaffaqiyatsizlikka yo'l qo'ymaslik uchun ma'lumotlar bir nechta serverlar o'rtasida taqsimlanadi, shuning uchun bittasi ishlamay qolganda ham boshqalar ishlamaydi va shuning uchun noto'g'ri tugunning ishini osongina davom ettirishi mumkin. Bu ma'lumotlar bazasida bitta nosozlik nuqtasi mavjud emasligini kafolatlaydi va shuningdek, NoSQL ma'lumotlar bazasining haqiqiy xatolarga chidamliligini tasvirlaydi. Ma'lumotlar va funksiyalar bo'lsa, u o'rnatilgan ortiqcha qobiliyatidan ham foydalanadi.

NoSQL ma'lumotlar bazasining to'rtta umumiy turi mavjud bo'lib, ularda har bir ma'lumotlar bazasi o'ziga xos xususiyatlarga ega:

- Grafik ma'lumotlar bazasi: Ushbu turdagi ma'lumotlar bazalarining asosini grafik nazariyasi tashkil qiladi. Misollar: Neo4j va Titan.
- Key-value do'koni: Ushbu ma'lumotlar bazasida biz ma'lumotlarni ikki qismga, ya'ni kalit va qiymatga ajratamiz. Misollar: Redis, DyanmoDB, Riak.
- Ustunlarni saqlash: Bu erda ma'lumotlar ma'lumotlar ustunlari bo'limlari shaklida saqlanadi. Misollar: HBase, BigTable va Kassandra.
- Hujjat ma'lumotlar bazasi: Ushbu ma'lumotlar bazasi kalit-qiymat do'konlarining yuqori versiyasidir. Bu erda qiymatlar murakkab tuzilmalar (JSON kabi) ko'rinishidagi ma'lumotlar bo'lgan hujjatlar sifatida saqlanadi. Misollar: MongoDB va CouchDB.

CAP teoremasi barcha ma'lumotlar bazalarida qo'yilgan cheklovni tushuntiradi. Unda aytilishicha, har kim CAP sifatida qisqartirilgan uchta xususiyatdan faqat ikkitasini tanlashi mumkin, bunda C - izchillik, A - mavjudlik va P - bo'linga tolerantlik degan ma'noni anglatadi. Brewer teoremasining asosiy bayonotida aytilishicha, har qanday umumiy ma'lumotlar tizimi uchun ushbu xususiyatlardan maksimal ikkita xususiyat mavjud bo'lishi mumkin.

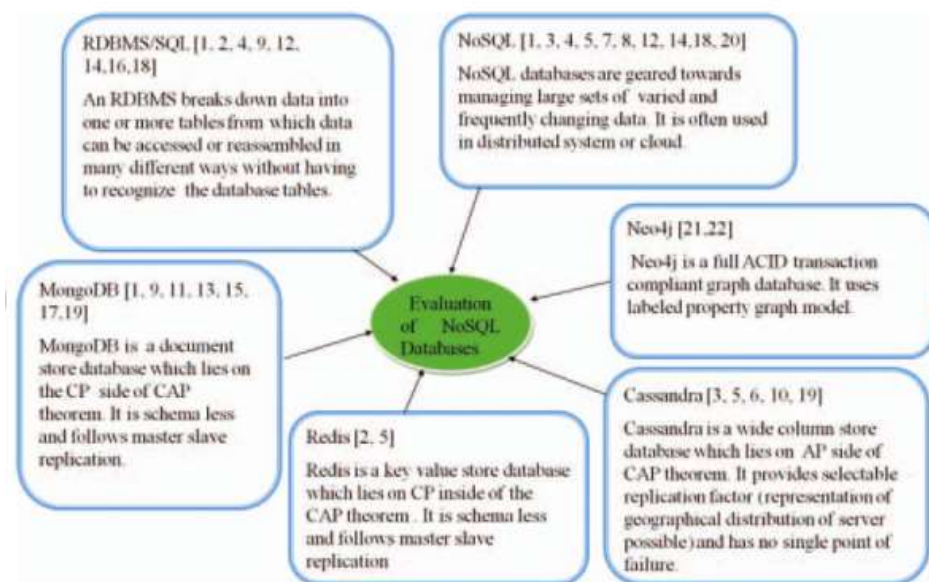
II. ADABIYOTLAR SHARXI:

1-rasmda biz VLDB, IEEE, ACM va SIGMOD kabi turli manbalardan olingan hujjatlarni o'rganganimizni ko'rsatadi. Ushbu maqolalar oltita guruhga bo'lingan (NoSQL,

SQL/RDBMS, Redis, MongoDB, Cassandra va Neo4j) va har bir guruhning tanqidiy tahlili o'tkazildi.

NoSQL: NoSQL, faqat SQL emas, ma'lumotlar bazasining relyatsion ko'rsatmalariga rioya qilmaydigan distributiv ma'lumotlar modelidir. U katta ma'lumotlarni saqlash, gorizontallik va massiv-parallel ma'lumotlarni qo'llab-quvvatlaydi qayta ishlash [9]. NoSQL, shuningdek, SQL ko'rinishida osongina ifodalanmaydigan ma'lumotlarni ham qo'llab-quvvatlaydi [17].

1-rasm: Tegishli qog'ozlarning tasnifi



Shuning uchun NoSQL ma'lumotlar bazalari odatiy SQL ma'lumotlar bazalarining o'rini bosuvchi vosita sifatida qabul qilingan, ayniqsa biz juda katta hajmdagi ma'lumotlarni boshqarayotgan stsenariyda [13]. NoSQL relyatsion ma'lumotlar bazalarining kamchiliklarini bartaraf etish uchun ishlab chiqilgan. Shuning uchun ko'plab kompaniyalar ushbu ma'lumotlar bazalari sohasini tadqiq qilishga sarmoya kiritdilar [9]. Hozirgi vaqtda ACID xususiyatlariga NoSQL ma'lumotlar bazalari tomonidan o'rta dastur yordamida ham erishish mumkin [3]. , NoSQL ma'lumotlar bazalarining qat'iy sxemalari va boshqa ko'plab cheklovlardan qoching. Ular dastlab ma'lumotlar bazalari sifatida uzoq vaqtdan beri mavjud bo'lgan relyatsion ma'lumotlar bazalariga muqobil bo'lish uchun kiritilgan. NoSQL ma'lumotlar bazalari asosiy saqlash tizimlarining xizmatlari va imkoniyatlariga tayanadi [8].

RDBMS/SQL

Relyatsion ma'lumotlar bazalari soddaligi tufayli ma'lumotlar bazasining eng keng tarqalgan turi hisoblanadi. RDBMSda [26] ma'lumotlar ko'proq samaradorlik uchun odatda normallashtirilgan shaklda bo'lgan bir nechta jadvallarga bo'linadi. Ma'lumotlarga kirishda uni foydalanuvchi talablariga muvofiq qayta yig'ish mumkin.

Strukturaviy so'rovlar tili (SQL) ma'lumotlarni aniqlash tili (DDL), ma'lumotlarni boshqarish tili (DCL) va ma'lumotlarni boshqarish tili (DML) bo'lgan to'rt turdagi so'rovlardan iborat. Har birida ma'lumotlarni aniqlash, ya'ni jadval yaratish, jadvalni o'zgartirish va hokazolar uchun bajariladigan o'z so'rovlari mavjud. yangilash, kiritish va hokazolar yordamida mavjud ma'lumotlarni talablarga muvofiq manipulyatsiya qilish. va orqaga

qaytarish, topshirish va h.k. kabi so'rovlar yordamida tranzaksiya nazoratini aniqlang. mos ravishda. NoSQL va RDBMS o'rtasidagi ma'lumotlarning haqiqiyliigi, so'rovlar tili, ma'lumotlar turi, ma'lumotlarni saqlash, sxema, moslashuvchanlik, miqyoslash va ACID muvofiqligi kabi xususiyatlar bo'yicha batafsil taqqoslash taqdim etilgan [40] Odatda NOSQLda faqat bitta rekord tranzaksiyalar va yakuniy izchillik takrorlash tizimi mavjud. qo'llab-quvvatlanadi, bunda tranzaksiyalar kommutativ deb taxmin qilinadi. Shunday qilib, ACID tranzaksiyalari ishlash uchun xavf tug'diradi [41].

Hujjatlar do'koni (MongoDB):

MongoDB CAP teoremasining CP tomonida joylashgan. NoSQL ma'lumotlar bazasining to'rtta umumiy turi mavjud bo'lib, ularda har bir ma'lumotlar bazasi o'ziga xos xususiyatlarga ega: Relyatsion ma'lumotlar bazalari soddaligi tufayli ma'lumotlar bazasining eng keng tarqalgan turi hisoblanadi. RDBMSda [26] ma'lumotlar ko'proq samaradorlik uchun odatda normallashtirilgan shaklda bo'lgan bir nechta jadvallarga bo'linadi. Ma'lumotlarga kirishda uni foydalanuvchi talablariga muvofiq qayta yig'ish mumkin. MongoDB BSON [37] formatini qo'llab-quvvatlaydi, ya'ni JSON [38] hujjatga o'xshash dinamik sxemalar bilan ma'lumotlar integratsiyasini oson va tezroq qiladi. MongoDB ning umumiy xususiyatlaridan biri shundaki, u hujjatlarga yo'naltirilgan saqlash qatlamiga ega va serverlar o'rtasida ma'lumotlarni takrorlash uchun u asinxron replikatsiyadan foydalanadi [15]. MongoDB va boshqa NoSQL ma'lumotlar bazalarida SQL ma'lumotlar bazalarida talab qilinmagan qo'shimcha qarorlar qabul qilinadi. Ushbu qarorlar ma'lumotlar bazalarining ishlashiga ta'sir qiladi[12]. MongoDB dan foydalanishning boshqa afzalliklari orasida oson replikatsiya, MapReduce va klasterlash kiradi.

Ustunlar do'koni (Kassandra):

Kassandra CAP teoremasining AP tomonida joylashgan. Bu Neo4j etiketli xususiyat grafik modelidan foydalanadi. Neo4j-da tugunlar va qirralar ular bilan bog'liq xususiyatlarga ega bo'lishi mumkin. o'z foydalanuvchilariga miqyoslilikni ta' minlaydi, chunki u chiziqli ravishda kengaytirilishi va ishlashiga putur etkazmasdan mavjudligi. Kassandra bir nechta tovar serverlarida ma'lumotlar yig'indisini osongina boshqarishga qodir, shu bilan birga hech qanday nosozliklarsiz yuqori mavjudlikni saqlab qoladi. [6][10]

Kalit qiymatlar ombori (Redis):

Redis CAP teoremasining CP tomonida joylashgan. Redis ma'lumotlarning mustahkamligi uchun tanlovga ega asosiy qiymat ma'lumotlar ombori. Bu xotiradagi NoSQL ma'lumotlar bazasi, Redis qatorlar, ro'yxatlar, to'plamlar, xeshlar va tartiblangan to'plamlar kabi turli xil ma'lumotlar tuzilmalari serverlarini qo'llab-quvvatlaydi. Uni relax master slave arxitekturasi yordamida takrorlash mumkin.

Grafik ma'lumotlar bazasi (Neo4j):

Neo4j etiketli xususiyat grafik modelidan foydalanadi. Neo4j-da tugunlar va qirralar ular bilan bog'liq xususiyatlarga ega bo'lishi mumkin. o'z foydalanuvchilariga miqyoslilikni ta'minlaydi, chunki u chiziqli ravishda kengaytirilishi va ishlashiga putur etkazmasdan mavjudligi. Tugunlarni o'z rollariga ko'ra turli xil toifalarga ajratadigan teglar bilan bog'lash mumkin. Neo4j to'liq ACID tranzaksiyalariga mos keladigan grafik ma'lumotlar bazasi. U mustaqil server sifatida (REST interfeysi) yoki o'rnatilgan shaklda foydalanish mumkin [22].

III. NOSQL MA'LUMOTLAR BAZALARINI QO`YISHI TAHLILI

A. NoSQL ma'lumotlar bazalarini funktsional va funktsional bo'lmagan talablar asosida taqqoslash.

1-jadval: Funktsional bo'lmagan xususiyatlar asosidagi turli xil NoSQL ma'lumotlar bazalari

| Data model | Performance of queries | Scalability of data | Flexibility of schema | Structure of database | Complexity of values |
|-----------------|------------------------|---------------------|-----------------------|---------------------------------|----------------------|
| Key-value store | High | High | High | Primary key with some value | None |
| Column Store | High | High | Moderate | row consisting multiple columns | Low |
| Document Store | High | Variable (High) | High | JSON in form of tree | Low |
| Graph Database | Variable | Variable | High | Graph – entities and relation | High |

1-jadvalda kalit-qiymatlar ombori, ustunlar do'koni, hujjatlar ombori va grafik ma'lumotlar bazasini so'rovlarning bajarilishi, ma'lumotlarning masshtabligi, sxemaning moslashuvchanligi, ma'lumotlar bazasining tuzilishi va qiymatlarning murakkabligi kabi funktsional bo'lmagan xususiyatlari asosida taqqoslanadi. kalitqiymat juftligi shaklida osongina ifodalanishi mumkin bo'lgan oddiy ma'lumotlar; asosiy qiymat do'koni tanlanishi mumkin, chunki u yuqori unumdorlik, kengayish va moslashuvchanlikni ta'minlaydi. Agar qiymat dan ustunida ifodalanishi mumkin bo'lsa va yarim tuzilgan bo'lsa, ustunlar do'koni mos ma'lumotlar bazasi hisoblanadi, chunki u yuqori unumdorlik va kengayishni ta'minlaydi. Agar ma'lumotlar JSON formatida taqdim etilishi mumkin bo'lsa, hujjatlar do'koniga ustunlik berish kerak, chunki u yuqori unumdorlik, moslashuvchanlik va odatda yuqori miqyoslilikka ega. Agar biz grafik nazariyasi yordamida ifodalanishi mumkin bo'lgan ma'lumotlarni saqlashimiz kerak bo'lsa yoki ma'lumotlar bir-biriga kuchli bog'liq bo'lsa, biz yuqori barqarorlikni ta'minlaydigan grafik do'kon modelidan foydalanamiz, lekin unumdorlik va kengayish o'zgaruvchan.

2-jadvalda NoSQL ma'lumotlar bazalarining to'rtta toifasi funktsional xususiyatlar asosida taqqoslanadi, masalan: Oddiylashtirish, Yagona agregat (bitta kalitga bir nechta kompozit kalitlarni qo'shish), Atomlik, Tartibga solinmagan kalitlar, Derived Jadval (jadval quyidagi sahifalarda tuzilishi mumkin). masterklassning asosi bu ko'p o'lchovli indekslar bo'yicha saralashga yordam beradi), Kompozit kalit, Kompozit yig'ish, agregatsiya, yig'ish va guruhlash,

qo'shnilik ro'yxatlari (har bir tugun bevosita ajdodlar yoki avlodlar massivlarini o'z ichiga olgan individual yozuv sifatida yaratilgan) , Ichki to'jplamlar va qo'yshilishlar.

Agar biz ma'lumotlar bazasida kompozit kalit, birlashma yoki olingan jadval operatsiyalaridan foydalanmoqchi bo'lsak, asosiy qiymatlarni saqlashdan qochish kerak.

Agar biz de dan foydalanmoqchi bo'lsak, Document Store'dan qochish kerak normallashtirish, tartibsiz kalit, kompozit kalit, kompozit yig'ish, ma'lumotlar bazasida birlashma yoki olingan jadval operatsiyalari.

Agar biz tartibsiz kalitlar, yig'ish va guruhlash, qo'shni ro'yxatlar, ichki o'rnatilgan to'plamlar yoki ma'lumotlar bazasida birlashma operatsiyalaridan foydalanmoqchi bo'lsak, keng ustunli saqlashdan qochish kerak.

Agar biz normalizatsiya qilishni xohlasak, Graph Store-dan foydalanish kerak.

| S. No | Features | Key Value Store | Document Store | Wide Column store | Graph Store |
|-------|---------------------------------|---------------------|----------------|-------------------|----------------|
| 1. | Denormalization | Applicable | Not Applicable | Applicable | Applicable |
| 2. | Single Aggregate | Applicable | Applicable | Applicable | Not Applicable |
| 3. | Atomicity | Applicable | Applicable | Applicable | Not Applicable |
| 4. | Unordered Keys | Applicable | Not Applicable | Not Applicable | Not Applicable |
| 5. | Derived Table | Not Applicable | Not Applicable | Applicable | Not Applicable |
| 6. | Composite Key | Not Applicable | Not Applicable | Applicable | Not Applicable |
| 7. | Composite Aggregation | Applicable(ordered) | Not Applicable | Applicable | Not Applicable |
| 8. | Aggregation | Applicable | Applicable | Applicable | Not Applicable |
| 9. | Aggregation and Group by | Applicable | Applicable | Not Applicable | Not Applicable |
| 10. | Adjacency Lists | Applicable | Applicable | Not Applicable | Not Applicable |
| 11. | Nested Sets | Applicable | Applicable | Not Applicable | Not Applicable |
| 12. | Joins | Not Applicable | Not Applicable | Not Applicable | Not Applicable |

2-jadval: Funktsional xususiyatlar asosida turli xil NoSQL ma'lumotlar bazalari

3-Jadval: Cassandra, MongoDB, Redis, Neo4j va boshqalarni farqlash

| S. yo'q. | Xususiyatlari | Keng ustunli do'kon (Kassandra) | hujjat (MongoDB) | Do'kon | Asosiy qiymat do'koni (Redis) | SQL (MySQL) | grafik Ma'lumotlar bazasi (Neo4j) |
|----------|-----------------------------|---|---|--------|--|--|--|
| 1 | Ma'lumotlar bazasi Model | Keng ustunli do'kon Hujjatlar do'koni | | | Asosiy qiymat do'koni | Relyatsion DBMS | grafik ma'lumotlar bazasi |
| 2 | Tavsif | Bu eng mashhur keng ustunli do'kon ma'lumotlar bazasidan biridir. U BigTable Key kosmik ustunlar oilasi | Bu taniqli hujjatlar do'koni ma'lumotlar bazasidan biridir | | Bu xotiradagi ma'lumotlar tuzilmasi do'koni va muhim kalit qiymatlar do'koni | Ochiq holda keng qo'llaniladi manba RDBMS | ochiq manba grafik ma'lumotlar bazasi |
| 3 JB 4 | | | | | | | |
| | stol | kontseptsiyasiga asoslangan | yig'ish | | Xesh to'plami, ro'yxat, to'plam, Saralangan to'plam va String | munosabat | Grafiklar yorliq |
| | qiymat | Qatorlar | Hujjatlar | | kalit qiymat juftligi | Qatorlar | Tugun va qirralar |
| 5 6 | o'qish Operatsiyalar | Sekin[4] | Tez[4] | | tez[5] | Sekin (Qo'shilishga bog'liq) | Ma'lumotlarga bog'liq |
| 7 | Yozing Operatsiyalar | Tez[4] | Tez[4] | | tez[5] | Sekin | ma'lumotlarga bog'liq |
| 8 | O'chirish Operatsiyalar | tez [4] | Tez[4] | | tez[5] | Sekin | ma'lumotlarga bog'liq |
| 9 | til | Java | C++ | | C[14] | C va C++ | Java, Scala |
| 10 | Litsenziya | ochiq manba | Ochiq manba | | ochiq manba | ochiq manba | ochiq manba |
| 11 | Ma'lumotlar sxemasi | Sxemasiz | Hech qanday aniq sxemaga amal qilinmaydi, lekin odatda bir xi hujjatlar | | Sxemasiz | Ha | Sxemasiz |
| 12 | Oldindan belgilangan turlar | ha; ASCII, int, blob, hisoblagich, o'nlik, ikkilik, ro'yxat, xarita, to'plam, matn, vaqt tamg'asi, varchar | konventsiyasining mazmuni o'xshash bo'lsa-da, majburiy Ha; Mantiqiy, sana, object_id, | | Qizman; qiymat satrlari uchun qo'llab-quvvatlanadigan String, Integer, Double, URI massivlari, giper jumalar, xeshlar, ro'yxatlar, to'plamlar, tartiblangan to'plamlar va geofazoviy indekslar | ha; int, float, double, sana, vaqt, bit, char, enum, binar, blob, Mantiqiy | ha; Mantiqiy, bayt, qisqa, int long, float, double, char, string |
| 13 | Server tomonidagi skriptlar | yo'q | JavaScript | | Lua | Ha | Ha |
| 14 | Triggerlar | Ha | yo'q | | yo'q | Ha | Ha |
| 15 | Bo'lish usullari | Sharding (Bu juda katta ma'lumotlar bazalari bor bo'lingan yoki ancha kichikroq va boshqariladigan bo'linmalarga bo'lingan) | Muvaffaqiyatsizlik nuqtasi bo'lmagan parchalanish | | Sharding | Gorizontal qismlarga ajratish, MySQL Cluster yoki MySQL Fabric bilan bo'lish | Neo4j-da qismlarga ajratishdan qo'dish kerak |
| 16 | Chet el kalitlari raqami | | Odatda, foydalanilmaydi, ammo DBRef bilan ekvivalent operatsiyani bajarish mumkin | | yo'q | Ha | Ha |
| 17 | tranzaksiya tushunchalar | Atomik va bor operatsiyalar uchun izolyatsiya qo'llab-quvvatlanadi | Atom operatsiyalari bitta hujjat doirasida amalga oshirilishi mumkin | | Optimistik quflash, buyruq bloklari va skriptlarning atomik bajarilishi Oddiy parolga | KISLOTA | KISLOTA |
| 18 | tushunchalar | Foydalanuvchilar uchun kirish huquqlari har bir ob'ekt uchun belgilanishi mumkin | Foydalanuvchilar va rolar uchun kirish huquqlari | | asoslangan kirishni boshqarish, | nozik Foydalanuvchilar avtorizatsiya tushunchalari bilan; www.mysql.com | Foydalanuvchilar, rolar va ruxsatlar |
| 19 | Veb-sayt | cassandra.apache.org www.mongo-db.org | | | redis.io | foydalanuvchi.guruhlari.yoki.redis.com | neo4j.com |
| 20 | dasturchi | Apache Dasturi 2008 yil | MongoDB Inc. | | Salvatore Sanfilippo | Oracle | Neo texnologiyasi |
| 21 | Yaratilgan Chiqarish | | 2009 yil | | 2009 yil | 1996 yil | 2007 yil |

B. NoSQL ma'lumotlar bazalarining toifalari asosida taqqoslash

3-jadvalda taqqoslash, ma'lumotlar bazasi modeli, tavsif, ma'lumotlar bazasi, jadval, qiymat, o'qish operatsiyalari, yozish operatsiyalari, o'chirish operatsiyalari, til litsenziyasi, ma'lumotlar sxemasi, oldindan belgilangan turlar, server tomoni skriptlari, triggerlar, qismlarga ajratish usullari, Replikatsiya usullari, masshtablash, xorijiy kalitlar, tranzaksiya tushunchalari, foydalanuvchi tushunchalari, veb-sayt, dasturchi, dastlabki nashr va joriy nashr

Joriy tadqiqot har biridan ma'lumotlar bazalarini oldi NoSQL ma'lumotlar bazalari toifasi, ya'ni Cassandra, MongoDB, Redis) va Neo4j. 3-jadvalda har bir toifadagi NOSQL ma'lumotlar bazalarining farqlanishi ko'rsatilgan. SQL - bu relyatsion ma'lumotlar bazasi; Cassandra BigTable va DynamoDB g'oyalari asoslangan Wide Column do'konlari toifasiga

kiradi. MongoDB - bu hujjatlar do'koni, Redis esa asosiy qiymatlar do'koni tushunchalariga amal qiladi.

Kassandrada SQL ma'lumotlar bazasiga o'xshash kalit maydoni va jadval o'rniga ustunlar oilasi mavjud. MongoDB to'plamni yaratadi, Redisda esa jadval o'rniga xeshlar, ro'yxatlar, to'plamlar va tartiblangan to'plamlar mavjud.

Kassandra va SQL-da o'qish operatsiyalari boshqa ikkitasiga qaraganda sekinroq. Yozish va o'chirish operatsiyalari uchun SQL barcha NoSQL ma'lumotlar bazalari bilan solishtirganda kam. Neo4j holatida, o'rnatilgan versiya sekin bo'lsa ham, RESTning ishlashi MongoDB ga deyarli o'xshaydi [21]. Bo'lish usullari uchun SQL Gorizonta Bo'limdan foydalanadi, qolganlari esa shardingdan foydalanadi.

Bundan tashqari, SQL xorijiy kalitlar kontsepsiyasidan foydalanadigan yagona hisoblanadi. Tranzaksiya tushunchalariga kelsak, SQL va Neo4j ACID xususiyatlariga amal qiladi. Yagona operatsiyalar uchun Kassandrada atomiklik va izolyatsiya qo'llab-quvvatlanadi. MongoDB-da uzoq hujjat ichida atom operatsiyalari mumkin, Redis esa optimistik qulflash va buyruq bloklari va skriptlarning atomik bajarilishini qo'llab-quvvatlaydi

MongoDB har xil turdagi foydalanuvchilar uchun kirish huquqlarini qo'llab-quvvatlaydi. Kassandra uchun har bir ob'ektga kirish huquqi o'rnatilishi mumkin. Redis murakkab bo'lmagan parolga asoslangan kirishni boshqarishni qo'llab-quvvatlaydi [44]. MongoDB da avtorizatsiya va autentifikatsiya sukut bo'yicha o'chirib qo'yilgan. Bu erda avtorizatsiya har bir ma'lumotlar bazasi darajasida rolga asoslangan yondashuvga rioya qilish orqa Har bir ma'lumotlar bazasi darajasida autentifikatsiyani ta'minlash asosiy MongoDB da mavjud bo'yilib, unda foydalanuvchilar, ayniqsa, bitta mantiqiy ma'lumotlar bazasi [42]. Neo4j [45] da avtorizatsiya va autentifikatsiya sukut bo'yicha yoqilgan.

C. Tarqatish xossalari asosida taqqoslash

| Feature | Wide Column Store (Cassandra) | Document Store (MongoDB) | Key Value pair Store (Redis) | Graph Database (Neo4j) |
|---------------------------|------------------------------------|-------------------------------|------------------------------|--|
| Sharding and Partitioning | Auto sharding and order preserving | Built in and order preserving | Auto sharding and no order | Supports sharding but should be avoided |
| Scaling | Horizontal | Horizontal | Horizontal | Horizontal |
| Replication | Selectable Replication Factor | Master slave | Relaxed Master slave | Causal Clustering using Raft protocol (master slave) |

4-jadvalda ma'lumotlar bazasi bir xil jismoniy joyda joylashgan yoki bo'lmasligi mumkin bo'lgan bir nechta kompyuterlarga tarqalganda to'rtta ma'lumotlar bazasi qanday ishlashini tushuntiradi. MongoDB bo'lsa, avtomatik parchalanish ma'lumotlarni saqlash tartibida bir nechta tugunlarga bo'lish uchun ishlatiladi. MongoDB gorizonta o'lchovni qo'llab-quvvatlaydi, bu unga bir nechta tugunlar bo'ylab ma'lumotlarni o'lchash imkonini beradi. Yuk tugunlar bo'ylab teng taqsimlanadi, agar muvozanat buzilgan bo'lsa, u avtomatik ravishda yukni teng ravishda qayta taqsimlaydi.

Kassandrada ma'lumotlarning katta miqdori ko'plab tugunlar bo'ylab bo'lingan, bu esa foydalanuvchiga juda yuqori va nosozliklarsiz foydalanish imkonini beradi. Shuningdek, u gorizontol masshtabni, tanlanadigan replikasiya faktorini va ma'lumotlar markazining o'zaro replikasiyasini qo'yllab-quvvatlaydi.

Redis master-slave arxitekturasidan foydalangan holda xotiradagi ma'lumotlar uchun mo'ljallangan. Umuman olganda, Redis master-qul replikasiyasining kamroq qat'iy amaliyotini qo'llab-quvvatlaydi, shu bilan birga har qanday ustadan olingan ma'lumotlar har qanday qul soniga osongina ko'chiriladi, qulning o'zi esa boshqa qullarga xo'jayin bo'lishi mumkin. Ushbu ma'lumotlar bazasi ma'lumotlarni tugunlar bo'ylab tartibni saqlaydigan tarzda taqsimlamaydi.

Neo4J miqyosidagi to'plam yuqori mavjudligi sifatida qayd etilgan. U qismlarga ajratishni qo'llab-quvvatlamaydi va to'liq ma'lumotlar to'plami butun klaster bo'ylab takrorlanadi.

IV. XULOSALAR

SQL ma'lumotlar bazalari vertikal (apparat), NoSQL ma'lumotlar bazalari esa gorizontol ravishda kengaytiriladi (server). Ushbu maqola yaqinda bozorda asosiy relyatsion ma'lumotlar bazasini boshqarish tizimlariga muqobil sifatida paydo bo'lgan NoSQL-larni batafsil ko'rib chiqish va joriy etishdan iborat. Birinchi yarmida munosabatlarga bog'liq bo'lmagan boshqaruv tizimlarini ishlab chiqish va ulardan foydalanish sabablari va sabablari muhokama qilinadi, keyingi yarmida esa NoSQLlar turlarga ajratiladi, xususan, hujjatlar do'konlari, kalit qiymatlar do'konlari va ustunlarga asoslangan do'konlar, so'ngra ularning modellari va modellari haqida ma'lumot beriladi. ishlar. Har bir ma'lumotlar bazasi turlicha ishlaydi va o'zini tutadi va ularning barchasi doimo rivojlanib boradi. Joriy tadqiqot har bir toifadagi ma'lumotlar bazalarini oldi: Cassandra (keng ustunli do'kon), Neo4j (Grafik ma'lumotlar bazasi), Redis (Asosiy qiymatlar juftligi do'koni) va MongoDB (hujjatlar do'koni) va ularni ma'lumotlar modellari, distributiv xususiyatlar va boshqalar asosida taqqosladi. Xususiyatlari. Tadqiqot ularni funktsional bo'lmagan xususiyatlari bo'yicha taqqoslaydi. Kalit qiymat shaklida osongina ifodalanishi mumkin bo'lgan oddiy ma'lumotlar uchun asosiy qiymatlar do'konini tanlash kerakligi aniqlandi, chunki u yuqori unumdorlik, kengayish va moslashuvchanlikni ta'minlaydi. Agar qiymat ustun ko'rinishida taqdim etilishi mumkin bo'lsa va yarim tuzilgan bo'lsa, ustunlar do'koni mos ma'lumotlar bazasi hisoblanadi, chunki u yuqori unumdorlik va o'lchovni ta'minlaydi. Agar ma'lumotlar JSON formatida taqdim etilishi mumkin bo'lsa, hujjatlar do'koniga ustunlik berish kerak, chunki u yuqori unumdorlikka, moslashuvchanlikka va odatda yuqori miqyosga ega. Agar grafika nazariyasi ma'lumotlarni ifodalasa, biz yuqori barqarorlikni ta'minlaydigan grafik do'kon modelidan foydalanamiz, ammo unumdorlik va kengayish o'zgaruvchan. Shundan so'ng, taqqoslash amalga oshiriladi funktsional xususiyatlari asosida. Agar ma'lumotlar bazasida kompozit kalit, birlashma yoki olingan jadval operatsiyalaridan foydalanish kerak bo'lsa, asosiy qiymatlarni saqlashdan qochish kerak degan xulosaga keldi. Agar ma'lumotlar bazasida normalizatsiya, tartibsiz kalit, kompozit kalit, kompozit yig'ish, birlashma yoki olingan jadval operatsiyalaridan foydalanish kerak bo'lsa, hujjatlar do'konidan qochish kerak. Ma'lumotlar bazasida tartibsiz kalitlar, yig'ish va guruhlash, qo'shni ro'yxatlar, o'rnatilgan to'plamlar yoki birlashma operatsiyalaridan foydalanmoqchi bo'lsak, keng ustunli saqlashdan qochish kerak. Agar biz normalizatsiya qilishni xohlasak, Grafik do'konidan foydalanish kerak. Redis maksimal xavfsizlik [43] uchun emas, balki

maksimal ishlash va soddalik uchun optimallashtirilgan. Stonebraker [41] NOSQL ma'lumotlar bazalarini qo'llab-quvvatlash uchun turli xil ishlash argumentlarini ko'rib chiqdi va ularning etarli emasligini kuzatdi. Shunday qilib, bu tizimlar ham turli cheklovlarga ega.

ADABIYOTLAR:

- [1] AvriiaFloratos, Nikhil Teletia , David J. DeWitt, Jignesh M. Patel, Donghui Zhang, “Can the Elephants Handle the NoSQL Onslaught?”, Proceedings of the VLDB Endowment, 2012
- [2] Tilmann Rabl, Mohammad Sadoghi, Hans-Arno Jacobsen, Sergio Gómez Villamor, Victor Muntés Mulero and Serge Mankovskii, “Solving Big Data Challenges for Enterprise Application Performance Management”, The 38th International Conference on Very Large Data Bases, August 27th - 31st 2012, Istanbul, Turkey. Proceedings of the VLDB Endowment, 2012
- [3] Bogdan George Tudorica, Cristian Bucur, “A comparison between several NoSQL databases with comments and notes”, RoEduNet International Conference 10th Edition: Networking in Education and Research, 2011
- [4] Yishan Li and Sathiamoorthy, “A performance comparison of SQL and NoSQL databases.”, IEEE Pacific Rim Conference on Communications, Computers and Signal Processing, 2013
- [5] Laurie Butgereit, “Four NoSQLs in Four Fun Fortnights: Exploring NoSQLs in a Corporate in a Corporate IT Environment”, SAICSIT '16 Proceedings of the Annual Conference of the South African Institute of Computer Scientists and Information Technologists, 2016
- [6] Avinash Lakshman, Prashant Malik, “Cassandra - A Decentralized Structured Storage System”, Operating Systems Review, 2010
- [7] John Klein, Ian Gorton, Neil Ernst, Patrick Donohoe, Kim Pham, Chriisjan Master, “A comparison between several NoSQL databases with comments and notes”, PABS '15 Proceedings of the 1st Workshop on Performance Analysis of Big Data Systems, 2015
- [8] Jiri Schindler, “I/O performance of NoSQL Databases(VLDB)”, SIGMETRICS '13 Proceedings of the ACM; SIGMETRICS/international conference on Measurement and modelling of computer systems, 2013
- [9] Amal W. Yassien, Amr F. Desouky, “RDBMS, NoSQL, Hadoop: A Performance-Based Empirical Analysis”, AMECSE '16 Proceedings of the 2nd Africa and Middle East Conference on Software Engineering, 2016
- [10] Hua Fan, Aditya Ramaraju, Marlon McKenzie, Wojciech Golab, Bernard Wong, “Understanding the Causes of Consistency Anomalies in Apache Cassandra”, Proceedings of the VLDB Endowment, 2015
- [11] Sumitkumar Kanoje, Varsha Powar, Debajyoti Mukhopadhyay, “Using MongoDB for Social Networking Website: Deciphering the Pros and Cons”, IEEE Sponsored 2nd International Conference on Innovations in Information Embedded and Communication Systems, 2015

- [12] Zachary Parker, Scott Poe, Susan V. Vrbsky, “Comparing NoSQLMongoDB to an SQL DB”, ACMSE '13 Proceedings of the 51st ACM Southeast Conference, 2013
- [13] Gansen Zhao, Weichai Huang, Shunlin Liang, Yong Tang, “Modelling MongoDB with Relational Model”, Fourth International Conference on Emerging Intelligent Data and Web Technologies, 2013
- [14] Rick Cattell, “Scalable SQL and NoSQL Data Stores”, SIGMOD Record, December 2010 (Vol. 39, No. 4), 2010
- [15] Lanjun Wang, Shuo Zhang, Juwei Shi, Limei Jiao, Oktie Hassanzadeh, Jia Zou, Chen Wangz, “Schema Management for Document Stores”, Proceedings of the VLDB Endowment, 2015
- [16] <http://www.planetcassandra.org/what-is-nosql/>
- [17] Concurrent Programming for Scalable Web Architectures (2012), Benjamin Erb Research Assistant, University of Ulm Distributed Systems (Cited by 14):http://berb.github.io/diplomathesis/original/061_challenge.html
- [18] Fay Chang, Jeffrey Dean, Sanjay Ghemawat, Wilson C. Hsieh, Deborah A. Wallach, Mike Burrows, Tushar Chandra, Andrew Fikes, Robert E. Gruber, "Bigtable: A Distributed Storage System for Structured Data", ACM Transactions on Computer Systems, Volume 26, issue 2, Article 4, June 2008
- [19] Seth Gilbert, Nancy Lynch, “Brewer's conjecture and the feasibility of consistent, available, partition-tolerant web services”, ACM SIGACT News Volume 33 Issue 2, June 2002 ,Pages 51-59 <http://dl.acm.org/citation.cfm?id=564601>
- [20] "Google Bigtable, Compression, Zippy and BMDiff". 2008-10-12. Archived from the original on 1 May 2013. Retrieved 14 April 2015..
- [21] Santosh S. Ravi, Kalyanaraman santhanam , “Performance of Neo4j versus MongoDB for social actions”; 2014 [22] Chad Vicknair, Michael Macias, Zhendong Zhao, Xiaofei Nan, Yixin Chen, Dawn Wilkins, “A Comparison of a Graph Database and a Relational”, ACM SE '10 Proceedings of the 48th Annual Southeast Regional Conference; 2010
- [23] DatabaseSQL Server by Microsoft: <https://www.microsoft.com/enus/sql-server/sql-server-2016/>
- [24] Oracle Official website: <https://www.oracle.com/database/index.html> [25] MySQL Documentation: <https://dev.mysql.com/doc/refman/5.7/en/whatis-mysql.html/>
- [26] Definition of RDBMS: <http://searchsqlserver.techtarget.com/definition/relational-databasemanagement-system/>
- [27] Neo4j Official Website: <https://neo4j.com/developer/graph-database/>
- [28] TITAN's (a distributed graph database) Official website: <http://titan.thinkaurelius.com/>
- [29] Redis Documentation: <https://redis.io/documentation>
- [30] Amazon DynamoDB Official Page: <http://docs.aws.amazon.com/amazondynamodb/latest/developerguide/Introduction.html>
- [31] Riak Official Website: <http://basho.com/products/>
- [32] Apache HBASE official website: <https://hbase.apache.org/>

- [33] Cassandra Documentation by Datastax Academy:
<https://academy.datastax.com/resources/brief-introduction-apachecassandra>
- [34] MongoDB Official Website: <https://www.mongodb.com/what-ismongodb>
- [35] Apache CouchDB Official Website: <http://couchdb.apache.org/>
- [36] Brewer's CAP Theorem: <http://www.royans.net/wp/2010/02/14/brewers-cap-theorem-ondistributed-systems/>
- [37] BSON Official Website and Documentation: <http://bsonspec.org/>
- [38] JSON Official Website and Documentation: <http://www.json.org/> [39] NoSQL Explanation by Datastax Academy:<http://www.planetcassandra.org/what-is-nosql/>
- [40] Antonios Makrisa, Konstantinos Tserpesa, Vassiliki Andronikoub, Dimosthenis Anagnostopouloa, “A classification of NoSQL data stores based on key design characteristics”, Cloud Futures: From Distributed to Complete Computing, CF2016, 18-20 October 2016. [41] Michael Stonebraker. 2010. SQL databases v. NoSQL databases. Commun. ACM 53, 4 (April 2010), 10-11. DOI: <https://doi.org/10.1145/1721654.1721659>.
- [42] “MongoDB Security Guide Release3.2.1”, MongoDB,Inc. February09, 2016,©MongoDB,Inc.2008–2015.