

NoSQL მონაცემთა ბაზები: განვითარების პერსპექტივები და პრობლემები მართვის საინფორმაციო სისტემებში

გია სურგულაძე, გიორგი კვიციანი, ბექა კახელი

საქართველოს ტექნიკური უნივერსიტეტი

რეზიუმე

განხილულია მართვის საინფორმაციო სისტემებში განსაკუთრებით დიდი მონაცემების დამუშავების და შენახვის თანამედროვე მეთოდები და ინსტრუმენტული საშუალებები. კლასიკური რელაციური მონაცემთა ბაზების გვერდით წარმოდგენილია დღეისათვის ერთ-ერთი აქტუალური ტექნოლოგია - NoSQL, რომელიც ხშირად ახალი ტიპის არარელაციურ მონაცემთა ბაზების ოჯახის სახით განიხილება. აგრეთვე მოცემულია მონაცემთა ჰიბრიდული ბაზების სისტემები, სადაც ინტეგრირდება რელაციური, NoSQL-ის და გრაფული ბაზების კონცეფციები. შედარებულია რელაციური და არარელაციური (MongoDB) ბაზების გამოყენების მახასიათებლები მართვის საინფორმაციო სისტემებში. განსაზღვრულია ასეთი მონაცემთა ბაზების ეფექტური გამოყენების სფეროები და პირობები. ნაშრომში ყურადღება გამახვილებულია აგრეთვე NewSQL ტიპის მონაცემთა ბაზებზე, MySQL და MariaDB სისტემებსა და Hadoop – „დიდი მონაცემთა“ ტექნოლოგიაზე.

საკვანძო სიტყვები: მონაცემთა ბაზა. SQL. NoSQL. MongoDB. NewSQL. MySQL. MariaDB. Linux ოპერაციული სისტემა. Hadoop ტექნოლოგია.

1. შესავალი

მართვის საინფორმაციო სისტემების (Management Information Systems) ერთ-ერთი მნიშვნელოვანი კომპონენტია მონაცემთა ბაზა. იგი, შეიძლება ითქვას, მისი გულია! მონაცემთა ბაზის მართვის სისტემებს საკმაო ისტორია აქვს, დაწყებული გასული საუკუნის 70-ანი წლებიდან. პირველი ბაზები იყენებდა მონაცემთა იერარქიულ და ქსელურ მოდელებს (IMS/2, OKA, ADABAS, CODASYL და სხვ.), შემდეგ კი, პროფესორ ედგარ კოდის მიერ რელაციური მოდელის და მონაცემთა ბაზის შესახებ პირველი სტატიების გამოქვეყნებით (1972-1974 წწ.), რადიკალურად შეიცვალა პროგრესულად მოაზროვნე მეცნიერთა საქმიანობის ორიენტაცია [1]. შედეგად მივიღეთ Oracle, MsSQL Server, MsAccess, MySQL, MariaDB, PostgreSQL და მრავალი სხვა, რომლებიც დღეისათვის აქტიურად გამოიყენება თითქმის ყველა სფეროსა და ორგანიზაციაში, რელაციური ბაზები იპყრობს პროგრამული სისტემების ბაზრის 80 % -ს.

პრობლემები, რომლებიც დიდ კორპორაციათა მართვის საინფორმაციო სისტემების მონაცემთა საცავებს (Data warehouses) ეხება, პირველ რიგში მათი სწრაფქმედების, წარმადობისა და საიმედოობის მახასიათებლებით ფასდება. რთული მოთხოვნების შესრულება რელაციურ ბაზებში დაკავშირებულია ბაზის სქემის შესაბამისად ცხრილების (Tables) კომპლექსურ დამუშავებაში SQL ენის ოპერაციებით, რაც ხშირად საგრძნობლად დიდ დროს თხოულობს და არაეფექტურია. ამიტომაც მეცნიერებმა დაიწყეს ასეთი სისტემების სრულყოფის საკითხებზე ინტენსიური მუშაობა. იგი ეხება რელაციურ ბაზებში მოთხოვნათა სწრაფი დამუშავების ალგორითმების ოპტიმიზაციას ან ასეთ სისტემებში არარელაციური (NoSQL) ბაზების გამოყენების პრინციპების განვითარებას [2].

წინამდებარე სტატიაში განიხილება რელაციური და არარელაციური მონაცემთა ბაზების თანამედროვე მდგომარეობა და მათი მომავალი განვითარების ტენდენციები „დიდი მონაცემთა დამუშავების“ თვალსაზრისით.

2. ძირითადი ნაწილი

2.1. დოკუმენტ-ორიენტირებული მონაცემთა ბაზა

დოკუმენტ-ორიენტირებული მონაცემთა ბაზა (Document-Oriented Database) არის მონაცემთა ბაზების მართვის სისტემა (მბმს), რომელიც გამოიყენება დოკუმენტების (მონაცემთა იერარქიული სტრუქტურების) შესანახად და რეალიზებულია NoSQL მიდგომით [2-5].

დოკუმენტ-ორიენტირებული მბმს-ას საფუძვლად უდევს დოკუმენტების საცავი (document Store), რომელსაც აქვს ხის სტრუქტურა. იგი იწყება ფესვური კვანძით და შეიძლება შეიცავდეს რამდენიმე შიგა კვანძს და ფოთლების კვანძს. ფოთლების კვანძი შეიცავს მონაცემებს, რომლებიც დოკუმენტის დამატების დროს შეიტანება ინდექსებში, რაც უზრუნველყოფს, რთული სტრუქტურების შემთხვევაშიც კი, მოიძებნოს გზა საჭირო მონაცემებისკენ [6].

მოთხოვნის საფუძველზე API (Application Programming Interface) ახორციელებს დოკუმენტების და მათი ნაწილების ძებნას. დოკუმენტები შეიძლება იყოს ორგანიზებული (დაჯგუფებული) კოლექციებში. ეფექტური ინდექსირების მიზნით სასურველია კოლექციებში მსგავსი სტრუქტურების დოკუმენტების გაერთიანება.

დოკუმენტ-ორიენტირებული მონაცემთა ბაზები გამოიყენება შინაარსის მართვის სისტემებში (CMS -Content Management System), საგამომცემლო საქმეში, დოკუმენტების საძიებო სისტემებში და სხვ. ასეთი ბაზების მართვის სისტემების მაგალითებია: MongoDB, CouchDB, Couchbase, MarkLogic, eXist, IBM Lotus Notes და სხვ.[7-11].

დოკუმენტ-ორიენტირებული მონაცემთა ბაზების მთავარ ცნებაა „დოკუმენტი“, რომელიც განისაზღვრება როგორც მონაცემთა ინკაფსულაცია ინფორმაციის კოდირების სტანდარტული ფორმატებისა და მეთოდების გამოყენების საფუძველზე. ასეთი ფორმატებია: XML, JSON, BSON, YAML. ზოგ შემთხვევაში შესაძლებელია PDF, Ms Office და მსგავსი დოკუმენტების ბინარული ფორმატით შენახვაც [4].

დოკუმენტი მონაცემთა ბაზაში მისამართდება უნიკალური გასაღების საშუალებით. ხშირად ეს გასაღები მარტივი სტრიქონია, რომელიც შეიძლება იყოს URI (Unified Resource Identifier) ან გზა (path) დოკუმენტამდე. ასეთი გასაღების ან მისი ინდექსის საშუალებით მოიძებნება დოკუმენტი ბაზაში და შესაძლებელია მისი სწრაფად ამოღება.

დოკუმენტური ბაზის დამახასიათებელია სიტყვა-გასაღების (მნიშვნელობის-გასაღების) მარტივად განსაზღვრა მოთხოვნილი დოკუმენტების მოსაძებნად. მონაცემთა ბაზას აქვს სპეციალური API ანუ მოთხოვნების ენა, რომელიც უზრუნველყოფს დოკუმენტების მიღებას მათი შინაარსის (content) მიხედვით. API არის აპლიკაციების დაპროგრამების ინტერფეისი. იგი შეიცავს მზა კლასების, პროცედურების, ფუნქციების, სტრუქტურებისა და კონსტანტების ერთობლიობას, რომელსაც წარმოადგენს დანართი (ბიბლიოთეკა ან სერვისი) ან ოპერაციული სისტემა. იგი გამოიყენება პროგრამისტების მიერ აპლიკაციის შექმნისას.

მომდევნო ნაწილში ჩვენ მიმოვიხილავთ NoSQL ტიპის მონაცემთა კონკრეტულ ბაზებს, შედარებით დეტალურად შევეხებით MongoDB პაკეტის შესაძლებლობებს.

2.2. გრაფულ-ორიენტირებული მონაცემთა ბაზა

გრაფული მონაცემთა ბაზის მართვის სისტემებისთვის დამახასიათებელია მონაცემთა გრაფული მოდელი, ანუ ინფორმაციის შენახვა ხდება არა „ცხრილებით“ (tables) და ატრიბუტებით (როგორც რელაციურ ბაზებში), არამედ გრაფული სტრუქტურებით, ანუ კვანძებით (nodes) და მათ შორის კავშირებით (გრაფის წიბოები – edges). ასეთი კავშირები

შეიძლება რამდენიმე დონეს მოიცავდეს (ღრმა კავშირები) და ისინი ძალზე აქტუალურია დიდი სოციალური პროექტების (ქსელების), ბიოინფორმატიკის, რთული მარშრუტების, სემანტიკური ქსელის (Web, HTTP გვერდებით) და სხვა სფეროს ამოცანების გადასაწყვეტად [2,8,9].

გრაფული მონაცემთა ბაზა არის ქსელური მოდელის (ან RDF-მოდელის) რეალიზაციის ნაირსახეობა. მისი კონცეფცია ჯერ კიდევ 80-იან წლებში გამოჩნდა, ხოლო პირველი გრაფული რეალიზაცია 2007 წელს, Neo4j სისტემის სახით [9]. დღეისთვის უკვე არსებობს რამდენიმე ათეული ასეთი ბაზებისა, მაგალითად: ArangoDB, OrientDB, MarkLogic, Oracle Spatial and Graph და სხვ. [9,12-16].

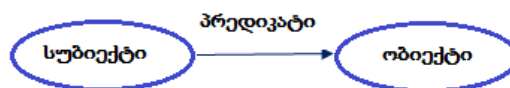
RDF (Resource Description Framework) – რესურსის აღწერის გარემო შეიქმნა WWW კონსორციუმის მიერ როგორც მონაცემთა აღწერის მოდელი – მეტამონაცემებით.

რესურსი RDF-ში შეიძლება იყოს ნებისმიერი არსი, როგორც ინფორმაციული (მაგალითად, ვებ-გვერდი, გამოსახულება), ასევე არაინფორმაციული (მაგალითად, ადამიანი, მანქანა ან აბსტრაქტული ცნება). რესურსის შესახებ გამონათქვამის მტკიცებას აქვს სამეულის (Triple Stores - სამადგილიანი შენახვა) სახე:

„სუბიექტი — პრედიკატი — ობიექტი“

მაგალითად, მტკიცება „დოლიძე არის პროფესორი“, RDF-ის ტერმინოლოგიით ჩაიწერება ასე (ნახ.1):

სუბიექტი – „დოლიძე“, პრედიკატი – „აქვს თანამდებობა“, ობიექტი – „პროფესორი“.



ნახ.1. RDF-ის სამეული

ამგვარად, RDF-ის მტკიცებულებები (ფაქტები) ქმნის ორიენტირებულ გრაფს, რომელშიც კვანძებია სუბიექტები და ობიექტები, ხოლო წიბოები – მათ შორის მიმართებები. RDF არის მონაცემთა აბსტრაქტული მოდელი, ანუ იგი აღწერს მოცემულ სტრუქტურას, დამუშავების ხერხებს და მონაცემთა ინტერპრეტაციებს.

ქვემოთ მოცემულია NoSQL ტიპის ოჯახის ზოგიერთი პოპულარული მმს, რომლებიც გრაფულ ან ჰიბრიდულ ბაზებს მიეკუთვნება [8].

AllegroGraph – დამუშავებულია W3C სტანდარტით Common Lisp ენაზე Triple Store (პრედიკატული სამეული) სახით მონაცემთა RDF მოდელისთვის, Windows, Linux და Mac OSX - თვის, კლიენტის ინტერფეისებით: Java, Python, Ruby, Perl, C#, Clojure და Common Lisp.

ArangoDB – დაწერილია C++ და JavaScript ენებზე მულტიმოდელური ბაზის სახით. გამოიყენება როგორც key/value, document, და graph data ბაზები და აქვთ ერთი საერთო მოთხოვნების ენა [10]. 2011 წლამდე გამოდიოდა AvocadoDB სახელით.

DataStax Enterprise Graph (DSE) – აგებულია Java ენაზე Web-საიტებისა და მობილური ტექნიკისთვის. სერვერის მხარეს Backend-ის სახით იყენებს Apache Cassandra-ს. შეუძლია დაამუშაოს წამში პეტაბაიტი ინფორმაცია და ერთდროულად მოემსახუროს ათას მომხმარებელს. ბაზა განაწილებულია კვანძების კლასტერებში და აქვს მასშტაბირებადი არქიტექტურა [2,13]. მასში ჩადგმულია OLAP ანალიზის და გრაფში ძებნის მხარდაჭერა. აქვს უსაფრთხოების დამატებითი პარამეტრები კონფიდენციალური მონაცემებისთვის.

MarkLogic – მულტიმოდელური ბაზაა სემანტიკური გრაფით და RDF სამეულით. ინახავს დოკუმენტებს JSON (JavaScript Object Notation) და XML ფორმატებში [2,14]. აქვს ჩადგმული

სამიერო სისტემა, ACID მახასიათებლების მქონე ტრანზაქციები – მაღალი წვდომადობა და ავარიული აღდგენის უნარი, გარანტირებული უსაფრთხოება, მოქნილობა და მასშტაბურობა.

Neo4j – კომპანია Neo Technology-ის პროდუქტი, დამუშავებულია java ენაზე და ერთ-ერთი ყველაზე პოპულარული გრაფული ბაზაა [2,15]. აპლიკაციების დაპროგრამების ინტერფეისი მონაცემთა ბაზისთვის რეალიზებულია მრავალ ენისთვის, მათ შორის: Java, Python, Ruby, PHP და სხვ.

OrientDB – გრაფული- და დოკუმენტ-ორიენტირებული ბაზის სისტემა, დამუშავებულია Java ენაზე Orient Technologies LTD ფირმის მიერ Windows, Linux, Mac და სხვა ოპერაციული სისტემებისთვის [2,16]. მოთხოვნების ენისათვის აქვს SQL-ის მხარდაჭერა (ამიტომაც მას NewSQL ბაზასაც მიაკუთვნებენ). იგი არ იყენებს JOIN ოპერაციას. მის მაგივრად აქვს სუპერ-სწრაფი მულტიმიდი მიმთითებლები ჩანაწერებს შორის, რომლებიც გრაფული ბაზებისთვისაა დამახასიათებელი. ეს უზრუნველყოფს ჩანაწერების ცალკეული ან მთლიანი ხეების და გრაფების გადასინჯვას რამდენიმე მილიწამის ფარგლებში.

Stardog – არის კროსპლატფორმული, სემანტიკური გრაფული ბაზის სისტემა, რეალიზებული Java ენაზე, RDF-ის და OWL (Web Ontology Language)-ის მხარდაჭერით [18]. OWL ენით აღიწერება კლასები და მიმართებები მათ შორის, რომლებიც დამახასიათებელია ვებ-დოკუმენტებისა და აპლიკაციებისთვის [19].

2.3. NewSQL მონაცემთა ბაზები

NewSQL არის თანამედროვე რელაციური ბაზების მართვის სისტემების კლასი, რომელიც გაფართოებული ფუნქციონალობის საფუძველზე უზრუნველყოფს NoSQL ბაზების მსგავს მწარმოებლურობას ტრანზაქციების ოპერატიული დამუშავებისათვის [20,21]. ამასთანავე იგი ინარჩუნებს ACID პრინციპებს [2].

განსაკუთრებით საყურადღებოა აქ მონაცემთა შენახვის პრინციპულად ახალი პლატფორმების შექმნა, რომლებიც ორიენტირებულია განაწილებული არქიტექტურის და მრავალნაკადურ სისტემებზე.

ასეთი მიდგომის ერთ-ერთი პოპულარული მონაცემთა ბაზაა MemSQL (რელაციური ბაზა



ნახ.2

ოპერატიული მეხსიერებით [In-Memory Storage] Linux ოპერაციული სისტემისთვის) [24] (ნახ.2). იგი იყენებს SQL ენას. კოდის გენერაცია სრულდება C++ ენაზე. ანუ MemSQL სერვერზე გაგზავნილი მოთხოვნები გარდაიქმნება C++-ზე და კომპილირდება GCC-ს დახმარებით.

MemSQL თავსებადია MySQL-თან. აპლიკაციები შეიძლება შეერთდეს MemSQL სისტემასთან ODBC/JDBC სტანდარტებით, ასევე დრაივერებით და MySQL-ის მომხმარებლებით. გარდა ზემოთ აღნიშნულისა, ლიტერატურულ წყაროებში განიხილავენ NewSQL-ის ტიპის შემდეგ ბაზებს: NuoDB, VoltDB, OrientDB, Clustrix, ScaleDB, dbShards და სხვ [24,25] (ნახ.3).



ნახ.3

მე-4 ნახაზზე მოცემულია მონაცემთა ტრადიციული SQL ბაზების, NoSQL და NewSQL ბაზების შედარება ოთხი ძირითადი თვისებით (Properties) [24]. როგორც ვხედავთ, NewSQL მონაცემთა ბაზა აერთიანებს ტრადიციული (რელაციური) და NoSQL (არარელაციური) ბაზების საუკეთესო თვისებებს, ამიტომაც იგი ძალზე პერსპექტიულია სამომავლო პროექტებისათვის. ერთ-ერთი საინტერესო გადაწყვეტა ამ თვალსაზრისით არის MySQL და NoSQL ბაზების ინტეგრაციის საკითხი, რომელსაც Oracle კორპორაცია ავითარებს.

COMPARISON :			
PROPERTIES	TRADITIONAL SQL	NOSQL	NEWSQL
ACID PROPERTY	✓	✗	✓
IN MEMORY DB	✗	✓	✓
BIG DATA	✗	✓	✓
RDBMS	✓	✗	✓

ნახ.4

2.4. ახალი მონაცემთა ბაზა MariaDB და მისი ინსტალაცია



მონაცემთა ბაზა MariaDB არის MySQL-ის ახალი ალტერნატიული free-ვარიანტი, რომელიც შექმნა მაიკლ ვიდენიუსმა 2009 წელს. იგი იყო ავტორი MySQL-ისაც (1995), რომელიც Oracle კორპორაციამ შეიძინა 2008 წელს და გახდა იგი კომერციული პროდუქტი [27]. (მ. ვიდენიუსის უფროსი ქალიშვილია მიაა -MySQL, ხოლო უმცროსი - მარია).

MariaDB თავსებადია MySQL-თან, უზრუნველყოფს შესაბამისობას API-სთან და MySQL-ის ბრძანებებთან. მასში დამატებულია მონაცემთა შენახვის (storage engine) ქვესისტემა XtraDB, რომელიც ცვლის MySQL-ის InnoDB-ს [27-29].

იმისათვის, რომ დავაყენოთ მონაცემთა ბაზა, უნდა შესრულდეს შემდეგი პირობები:

1) გამართული უნდა იყოს ლინუქსის ოპერაციული სისტემა ფიზიკურ ან ვირტუალურ მანქანაზე (2 CORE , 2 GB RAM , 20 GB HDD). რაც შეეხება ლინუქსის დისტრიბუციას, არჩევანი დიდია (მაგალითად, CentOS-ს, რომელიც დიდი პოპულარობით სარგებლობს) [30,31];

2) გამართულ ოპერაციულ სისტემას წვდომა უნდა ჰქონდეს ინტერნეტში (ბაზის დაყენების დროს).

მას შემდეგ რაც ოპერაციული სისტემა გამართულია და ინტერნეტში წვდომაც გვაქვს, შეგვიძლია დავიწყოთ მონაცემთა ბაზის დაყენება (აღნიშნული ინსტრუქცია გათვლილია „CentOS 6 64-bit“-სთვის) [29,30].

ოპერაციულ სისტემაში შევდივართ „root“ მომხმარებლით და ტერმინალში ვწერთ შემდეგ ბრძანებებს:

1) touch /etc/yum.repos.d/MariaDB.repo რაც შექმნის MariaDB.repo ფაილს /etc/yum.repos.d/ დირექტორიაში;

2) vi/etc/yum.repos.d/MariaDB.repo vi ედიტორით გავხსნათ MariaDB.repo ფაილი კლავიატურაზე „i“ ღილაკის გამოყენებით გადავიდეთ „insert“ რეჟიმში და ფაილში ჩავწეროთ შემდეგი :

- [mariadb]
- name = MariaDB
- baseurl = http://yum.mariadb.org/5.5/centos6-amd64
- gpgkey=https://yum.mariadb.org/RPM-GPG-KEY-MariaDB
- gpgcheck=1

კლავიატურაზე „Esc“ ღილაკის გამოყენებით გადავიდეთ ბრძანების რეჟიმში ვწერთ „wq“ და ვაწვებით „Enter“ ღილაკს, რის შემდეგადაც ჩვენი შეყვანილი ინფორმაცია შეინახება „MariaDB.repo“ ფაილში.

3) yum -y install MariaDB MariaDB-server (დაიწყება ბაზის ინსტალაცია).

4) /etc/init.d/mysql start (მონაცემთა ბაზის გაშვება)

5) როდესაც მონაცემთა ბაზა გაეშვება „mysql“ ბრძანებით შეგვიძლია შევიდეთ მონაცემთა ბაზის ტერმინალში სადაც გაუშვებს mysql ის ბრძანებებს:

- show databases;
- quit;

6) მას შემდეგ, რაც მონაცემთა ბაზას წარმატებით დავუკავშირდით და ყველაფერმა იმუშავა, აუცილებელია უსაფრთხოების პარამეტრების გამართვა, რისთვისაც ვუშვებთ შემდეგ ბრძანებას და დაკვირვებით გავივლით შემდგომ ეტაპებს:

mysql_secure_installation

```

/usr/bin/mysql_secure_installation: line 379: find_mysql_client: command not found
NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MariaDB
SERVERS IN PRODUCTION USE! PLEASE READ EACH STEP CAREFULLY!

In order to log into MariaDB to secure it, we'll need the current
password for the root user. If you've just installed MariaDB, and
you haven't set the root password yet, the password will be blank,
so you should just press enter here.

Enter current password for root (enter for none):
OK, successfully used password, moving on...

Setting the root password ensures that nobody can log into the MariaDB
root user without the proper authorisation.

Set root password? [Y/n] Y
New password:
Re-enter new password:
Password updated successfully!
Reloading privilege tables..
... Success!

Remove anonymous users? [Y/n] y
... Success!

Normally, root should only be allowed to connect from 'localhost'. This
ensures that someone cannot guess at the root password from the network.

Disallow root login remotely? [Y/n] y
... Success!

By default, MariaDB comes with a database named 'test' that anyone can
access. This is also intended only for testing, and should be removed
    
```

before moving into a production environment.
 Remove test database and access to it? [Y/n] y
 - Dropping test database...
 ... Success!
 - Removing privileges on test database...
 ... Success!
 Reloading the privilege tables will ensure that all changes made so far
 will take effect immediately.
 Reload privilege tables now? [Y/n] y
 ... Success!
 Cleaning up...
 All done! If you've completed all of the above steps, your MariaDB
 installation should now be secure.
 Thanks for using MariaDB!

- 7) /etc/init.d/mysql restart (კონფიგურაციის გავლის შემდეგ აუცილებელია მონაცემთა ბაზის გადატვირთვა).
- 8) chkconfig mysql on (სისტემის ჩატვირთვისას მონაცემთა ბაზა ავტომატურად რომ გაეშვას)
- 9) mysql -u root -p (მონაცემთა ბაზის ტერმინალში შესასვლელად -u პარამეტრით გადაცემთ მომხმარებლის სახელს, ხოლო -p პარამეტრით პაროლს).

2.5. MongoDB ბაზის ინსტალაცია



MongoDB არის დოკუმენტზე ორიენტირებული NoSQL მონაცემთა ბაზა [32]. მისი დაყენებისთვის აუცილებელია იგივე წინაპირობები და ეტაპები, რაც MariaDB-სთვის:

- 1) სისტემაში შევდივართ „root“ მომხმარებლით
- 2) touch /etc/yum.repos.d/mongodb.repo
- 3) vi /etc/yum.repos.d/mongodb.repo vi ედიტორის საშუალებით mongodb.repo ფაილში ჩავწერთ შემდეგი :
 - [mongodb]
 - name=MongoDB Repository
 - baseurl=http://downloads-distro.mongodb.org/repo/redhat/os/x86_64/
 - gpgcheck=0
 - enabled=1
- 4) yum -y install mongo-10gen mongo-10gen-server (მონაცემთა ბაზის დაყენება)
- 5) service mongod start (მონაცემთა ბაზის სერვისის გაშვება)
- 6) service mongod status (მონაცემთა ბაზის სტატუსის შემოწმება)
- 7) mongostat (მონაცემთა ბაზაში მიმდინარე პროცესების ნახვა)

ამით mongoDB-ს დაყენება დამთავრებულია. იმისათვის, რომ მონაცემთა ბაზაში მუშაობა შევძლოთ, ოპერატიული სისტემის ტერმინალში უნდა ავკრიფოთ mongo და შევიდეთ მონაცემთა ბაზის ტერმინალში, სადაც უშუალოდ mongo-ს ბრძანებების გაშვებას შევძლებთ. mongo (მონაცემთა ბაზის კლიენტი) ბრძანების გაშვებისას ოპერატიული სისტემა ავტომატურად მიმართავს „localhost:27017“ და ცდის ბაზასთან დაკავშირებას.

2.6. Hadoop – „დიდ მონაცემთა“ ტექნოლოგია

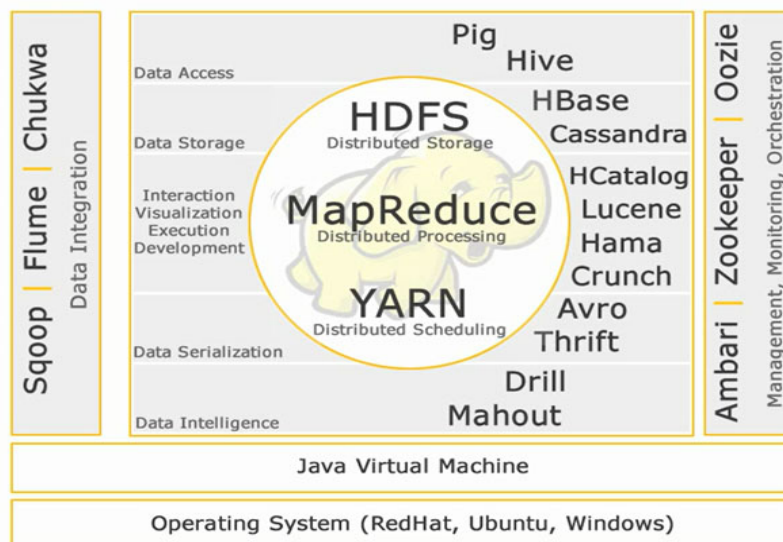
ჩვენ ვცხოვრობთ ინფორმაციის ეპოქაში. 2013 წლისთვის ციფრული სამყაროს ზომა 4.4 ზეტაბაიტი იყო, 2020 წლისთვის კი ნავარაუდებია ინფორმაციის მოცულობის ათმაგი ზრდა, 44 ზეტაბაიტამდე (44 მილიარდი ტერაბაიტი). [33]

ამ მოცულობის ინფორმაციიდან გარკვეული ნაწილი ისეთ კორპორაციებზე მოდის, როგორცაა ნიუ იორკის საფონდო ბირჟა – 4-5 ტერაბაიტი დღეში [34], Facebook.com – ჯამში 240 მილიარდი ფოტო, თვეში 7 პეტაბაიტი ზრდის მახასიათებლით [35], Ancestry.com – 10 პეტაბაიტი მოცულობის გენეალოგიის ბაზა [36].

ეს ადამიანის შექმნილი მონაცემებია, მაგრამ ბოლო ათწლეულში ტენდენცია შეიცვალა, დღეს უკვე ინფორმაციის უდიდეს ნაწილს ადამიანების ნაცვლად კომპიუტერული ტექნიკა აგენერირებს.

ბოლო წლებში აქტუალური გახდა ტერმინი IOT (Internet Of Things), რაც თავის თავში მოიცავს ყველა იმ აპარატს და კომპიუტერულ ტექნიკას, რაც მიმდინარე დროში დიდი რაოდენობით მონაცემებს აგენერირებს. მაგალითად, მანქანის GPS სისტემები, სხვადასხვა სენსორები და ყველა ის ტექნიკა, რაც ძირითადი ფუნქციონირების პარალელურად წარმოშობს დიდი რაოდენობის დამხმარე ინფორმაციას (metadata).

ამ რაოდენობის მონაცემების დამუშავებას სრულიად განსხვავებული სისტემა სჭირდება არა მხოლოდ რელაციური და არარელაციური ბაზების დონეზე, არამედ იმ სერვერული არქიტექტურის დონეზე, სადაც ვაყენებთ მონაცემთა ბაზებს. დღეისათვის საუკეთესო გამოსავალი დააპროექტა Apache Software Foundation-მა, სახელით Hadoop. Hadoop არის უფასო, ჯავაზე დაფუძნებული პლატფორმა, რომელიც შექმნილია დიდი ზომის მონაცემთა ნაკადის დასამუშავებლად (ნახ.5) [37].



ნახ. 5

Hadoop ეკოსისტემაში სხვადასხვა პროდუქტებია გაერთიანებული, ბირთვად კი სამი ძირითადი კომპონენტი აქვს:

- HDFS (Hadoop Distributed File System) - განაწილებული ფაილური სისტემა მონაცემების შესანახად;
- Map Reduse - მთავარი კომპონენტი განაწილებული გამოთვლების ჩასატარებლად;
- YARN (Yet Another Resource Negotiator) - განაწილებული გარემოს მართვა.

დასკვნა

ბოლო ათწლეულში მონაცემთა ბაზების მართვის სისტემების განვითარება ხორციელდებოდა რამდენიმე მიმართულებით. მთავარი იყო რელაციური ბაზების, როგორც უმრავლესი მართვის საინფორმაციო სისტემების ძირითადი კომპონენტის სრულყოფა მწარმოებლურობის თვალსაზრისით. ვითარდებოდა მოთხოვნების სწრაფად დამუშავების ოპტიმიზაციის ალგორითმები. მეორე მიმართულება გახდა NoSQL („არარელაციური“ ან „არა მხოლოდ რელაციური“) ტიპის ბაზების მართვის სისტემების შექმნა და სრულყოფა, განსაკუთრებით „დიდ მონაცემთა“ დასამუშავებლად, სადაც რელაციური ბაზები არაეფექტურია, მათი Join და სხვა რელაციური ოპერაციებით ცხრილების დამუშავების პროცესის დიდი დროის გამო. ასეთია კორპორაციებში დოკუმენტების დასამუშავებელი სისტემები. მესამე მიმართულება ეხება გრაფული ბაზების შექმნას და განვითარებას. ბოლოს, შეიძლება ვახსენოთ ჰიბრიდული მონაცემთა ბაზების მართვის სისტემები, სადაც ინტეგრირდება რელაციური, NoSQL-ის და გრაფული ბაზების კონცეფციები. ყველა მიმართულება აქტუალურია და რომელი ტიპის ბაზას აირჩევს მომხმარებელი, დამოკიდებულია კონკრეტული მართვის საინფორმაციო სისტემის მიზნებსა და ფუნქციებზე.

ლიტერატურა:

1. Codd E. F. (1970). A Relational Model for Large Shared Data Banks, Comm. ACM, Vol.13, No. 6, June '70. Relational Model for Database Management - Version 2, Addison-Wesley 1990
2. NoSQL For Dummies®. Published by: John Wiley & Sons, Inc., 111 River Street, Hoboken, NJ 07030-5774, www.wiley.com Copyright © 2015, New Jersey
3. NoSQL. <https://en.wikipedia.org/wiki/NoSQL>
4. Гранков М.В., Жуков А.И. (2013). Системы управления Базами данных. Донской гос.техн. Университет. Ростов-на-Дону.
5. ქოროლიშვილი ვ. NoSQL'n CAP. Just Development. Computer sciences. 2014. <http://vakhokor.blogspot.com/2014/10/nosql-n-cap.html>
6. Document-Oriented Database. https://en.wikipedia.org/wiki/Document-oriented_database
7. Document-oriented Database. Clusterpoint. Retrieved on 2015. <https://www.clusterpoint.com>
8. https://en.wikipedia.org/wiki/Graph_database
9. Lepinsky R. (2013). Graph Databases Overview and Applications. University of Winnipeg April 29. <https://rodgersnotes.files.wordpress.com/2014/06/rlgraphdb08.pdf>
10. www.arangodb.com
11. <https://en.wikipedia.org/wiki/NewSQL>
12. Morgan A., Lord M. NoSQL with MySQL 2014. <http://www.drdobbs.com/database/nosql-with-mysql/240167115>
13. https://docs.datastax.com/en/latest-dse/datastax_enterprise/graph/dseGraphAbout.html
14. MarkLogic. <https://en.wikipedia.org/wiki/MarkLogic>
15. Neo4j. <https://en.wikipedia.org/wiki/Neo4j>
16. OrientDB. <https://en.wikipedia.org/wiki/OrientDB>
17. სურგულაძე გ. დაპროგრამების ჰიბრიდული ტექნოლოგიები და მონაცემთა მენეჯმენტი. (2016). სტუ. „IT-კონსალტინგის ცენტრი“. თბილისი.
18. StardogDB. <https://en.wikipedia.org/wiki/Stardog>
19. https://en.wikipedia.org/wiki/Web_Ontology_Language
20. <https://en.wikipedia.org/wiki/NewSQL>
21. Stonebraker M. (2011). New SQL: An Alternative to NoSQL and Old SQL for New OLTP Apps. <http://cacm.acm.org/blogs/blog-cacm/109710-new-sql-an-alternative-to-nosql-and-old-sql-for-newoltp-apps/fulltext>
22. Kim W., Garza J.F., Ballou N., Woelk D.. <http://dl.acm.org/citation.cfm?id=627402>

23. Eisenberg A. (1996). New Architecture of the ORION Next-Generation Database System. 1990 Standard for Stored Procedures in SQL. ACM SIGMOD Record 25.12.
24. Choudhary S. (2014). NewSQL: The Best of Both "OldSQL" and "NoSQL"<http://www.slideshare.net/SUSHANTBCHOUDHARY/newsql-the-best-o>.
25. Glushkov I. (2015). NewSQL overview. www.slideshare.net/IvanGlushkov/newsql-overview.
26. Memcached. <https://en.wikipedia.org/wiki/Memcached>
27. MariaDB. <https://en.wikipedia.org/wiki/MariaDB>
28. Web-site of MariaDB. <https://mariadb.com/kb/en/>
29. MariaDB: <https://mariadb.com/kb/en/>
30. Nemeth E., Snyder G., Hein T.R., Whaley B. (2010). UNIX and LINUX System Administration Handbook fourth edition. Prentice Hall.
31. Cobbaut P. (2015). Linux Fundamentals. <http://linux-training.be/linuxfun.pdf>
32. MongoDB manual – <http://docs.mongodb.org/manual/>
33. The Digital Universe of Opportunities: Rich Data and the Increasing Value of the Internet of Things - http://bit.ly/digital_universe
34. Groenfeldt T. (2013). At NYSE, The Data Deluge Overwhelms Traditional Databases - <http://www.forbes.com/sites/tomgroenfeldt/2013/02/14/at-nyse-the-data-deluge-overwhelms-traditional-databases/#644feb072eb7>
35. Miller R. Facebook Builds Exabyte Data Centers for Cold Storage - http://bit.ly/facebook_exabyte
36. Ancestry.com - Company Facts. www.ancestry.com/corporate/about-ancestry/company-facts
37. Allaka S. (2015). Hadoop Ecosystem and its components, April 23. <http://www.edupristine.com/blog/hadoop-ecosystem-and-components>.

NOSQL DATABASES PROSPECTS OF DEVELOPMENT AND IMPLEMENTATION PROBLEMS FOR MIS

Surguladze Gia, Kiviladze Giorgi, Kakheli Beka
Georgian Technical University

Summary

The current methods and technologies are considered for Big Data management. One of the current technologies nowadays often viewed as a family of non-relational databases - NoSQL is considered next to the classical relational database. Hybrid database systems integrating concepts of relational, NoSQL and graph databases are also provided. A comparative analysis of the characteristics using relational and non-relational (MongoDB) database for information management systems is represented. The scope and terms of the effective use of these databases are determined. The paper is also focused on NewSQL databases, MySQL / MariaDB and the Hadoop ecosystem for "Big data".

№SQL БАЗЫ ДАННЫХ: ПЕРСПЕКТИВЫ ИХ РАЗВИТИЯ И ПРОБЛЕМЫ РЕАЛИЗАЦИИ ДЛЯ АСУ

Сургуладзе Г., Кивиладзе Г., Кахели Б.
Грузинский Технический Университет

Резюме

Рассматриваются современные методы и инструментальные средства особенно больших данных для АСУ. Рядом с классическими реляционными базами данных представлена одна из актуальных на сегодня технологий – NoSQL, которая часто рассматривается как семейство нереляционных баз данных нового типа. Предложены также системы гибридных баз данных, в которых интегрированы концепций реляционных, NoSQL и графовых баз данных. Дается сравнительный анализ характеристик использования реляционных и нереляционных (MongoDB) баз данных в информационных системах управления. Определены сферы и условия эффективного использования таких баз данных. В работе уделяется внимание также NewSQL базам данных, пакетам MySQL/MariaDB и Hadoop технологии “big data”.