

## RİYAZİYYAT, MEXANİKA, FİZİKA VƏ ASTRONOMİYA

### ƏLAQƏLİ VERİLƏNLƏR BAZALARI

Əli SƏBZƏLİYEV 

Naxçıvan Dövlət Universiteti, Naxçıvan, Azərbaycan

\*Yazışılan müəllif: [elisebzeliyev@ndu.edu.az](mailto:elisebzeliyev@ndu.edu.az)

#### NƏŞR TARİXİ:

*Qəbul edilmə tarixi:*

03.09.2025

*Nəşr edilmə tarixi:*

28.10.2025

#### AÇAR SÖZLƏR:

baza,  
sxem,  
əlaqə,  
struktur,  
informasiya,  
sorgu

#### XÜLASƏ

Verilənlər bazaları müasir kompüter sistemlərində məlumatların saxlanması və idarə edilməsi üçün istifadə olunur. Məlumat bazaları həyatımızın demək olar ki, bütün sahələrində iştirak edir: təhsil müəssisələrində, biznes və ya istehsalın idarə edilməsində, bank işində, elmi, mühəndislik və tibbi problemlərin həllində. Verilənlər bazaları böyük həcmli verilənləri mütəşəkkil və səmərəli şəkildə idarə etmək zərurətindən yaranmışdır. Metaverilənlər verilənlərin xüsusiyyətlərini və məlumatları birləşdirən əlaqələr toplusunu ehtiva edir. Məsələn, metaverilən hər bir məlumat elementinin adını, dəyərlərin növünü (rəqəm, tarix və ya mətn) saxlayır. Beləliklə, metaverilən verilənlər bazasında verilənlərin daha dolğun təsvirini təqdim edir. Metaverilən tərifini nəzərə alsaq, verilənlər bazası “özünü təsvir edən məlumat toplusu” kimi müəyyən edilə bilər. Məqalədə verilənlər bazası ilə inteqrasiya üsullarını araşdırmaq, verilənlərin emalı və saxlanmasını sürətləndirmək üçün onun sxeminin və sorgularının optimallaşdırılması ilə əlaqəli verilənlər bazasından verilənlərin NoSQL tipli verilənlər bazası formatına çevrilməsi üçün rəsmiləşdirilmiş üsullar toplusunun hazırlanmasından bəhs edilir.

### GİRİŞ

Əlaqəli verilənlər bazası (ƏVB) 1969-cu ildə Edqar F.Codd [39] tərəfindən təklif edilmiş relyasion modelə əsaslanır. O vaxtdan bəri, o, kommertiya tətbiqləri üçün dominant verilənlər bazası modelinə çevrildi (iyerarxik və şəbəkə modelləri kimi digər verilənlər bazası modelləri ilə müqayisədə).

Əlaqəli verilənlər bazasında əsas struktur vahidləri bir-biri ilə əlaqələndirilə bilən cədvəllərdir. Bu gün Oracle, IBM DB2 və Microsoft SQL Server kimi bir çox kommertiya əlaqəli verilənlər bazası idarəetmə sistemləri mövcuddur. MySQL, mSQL (mini-SQL) və quraşdırılmış JavaDB (Apache Derby) kimi bir çox pulsuz və açıq mənbə verilənlər bazası idarəetmə sistemləri də mövcuddur.

*Əlaqəli verilənlər bazalarının üstünlükləri:*

1. Məlumatlara universal giriş: ƏVB verilənləri emal etmək üçün SQL sorgu dilindən istifadə edir. SQL verilənlər bazası idarəetmə sistemlərinin əksəriyyətinin riayət etdiyi standartlara malik sorgu dilidir.

2. Verilənlərin tamlığı: əlaqəli verilənlər bazası sisteminin ən vacib xarakteristikası. Məlumatların bütövlüyü mexanizmi təcrid olunmuş və ya əlaqəsi olmayan qeydlərdən qaçmağa kömək edir.

3. Sorğuların optimallaşdırılması: əksər əlaqəli verilənlər bazası idarəetmə sistemlərində sorğuların qrammatik və sintaktik təhlilini və toplanmış statistika nəzərə alınmaqla sorğu planlarının optimallaşdırılmasını həyata keçirən sorğu optimallaşdırıcısı var.

4. ACID (Atomicity, Consistency, Isolation, Durability) xassələri: verilənlər bazasındakı məlumatların bütövlüyünü, ardıcılığını, təhlükəsizliyini və davamlılığını təmin etmək üçün hər bir əməliyyatın malik olmalı olduğu xüsusiyyətlər toplusu.

*Mənfi cəhətləri:*

1. Miqyaslaşdırılma: Əksər əlaqəli verilənlər bazaları paylanmış məlumatların işlənməsini və saxlanmasını dəstəkləmir, bu da böyük həcmli məlumatlarla işləyərkən əhəmiyyətli problemlərə səbəb olur. Çözümlərdən biri, böyük həcmdə məlumatı emal etmək üçün güclü hesablama imkanlarına malik serverdən istifadə etmək ola bilər, lakin bu, bahalı bir həlldir.

### **NoSQL verilənlər bazası**

2000-ci illərin əvvəllərindən web texnologiyalarının, sosial şəbəkələrin, mobil cihazların və Əşyaların İnternetinin inkişafı strukturlaşdırılmış, yarı strukturlaşdırılmış və strukturlaşdırılmamış məlumatların artmasına səbəb olmuşdur. Bu verilənlərlə işləyən proqramlar sorğuların işlənməsi sürətinin artması; əlavə resurslardan istifadə etmək imkanı; yüksək məlumat mövcudluğu; nasazlıqlara, aparat və proqram təminatının nasazlığına, hadisələrin yenilənməsinə qarşı müqavimət; əməliyyatların etibarlılığı; verilənlərin ardıcılığını qorumaq və verilənlər bazası sxemini dəyişdirərkən xərcləri minimuma endirmək artan məlumat həcminə dərhal uyğunlaşmaq üçün verilənlər bazasının üfüqi miqyaslaşmasını tələb edir.

Əlaqəli verilənlər bazasında yuxarıda göstərilən tələblərə nail olmaq çox çətindir və çox vaxt qeyri-mümkündür. Birincisi, əlaqə sxeminin sərbəstliyi verilənlərin transformasiyasına maneədir və buna görə də dinamik və ya tez-tez dəyişən böyük verilənlərin işlənməsi ssenariləri olan tətbiqlərin tələblərinə cavab verə bilməz. İkincisi, əlaqəli verilənlər bazasını miqyaslaşdırmaq mövcud olanları əvəz etmək üçün yeni, daha güclü müstəqil serverlərin quraşdırılmasını tələb edir. Bu, hər dəfə yeni serverə köçürüldükdə sistemin əlçatmazlığı və paylanmış normallaşdırılmış məlumatların uzlaşdırılmasında çətinliklərlə nəticələnən bahalı prosesdir. Bundan əlavə, əlaqəli verilənlər bazasında dəstəklənən ACID əməliyyatları səbəbindən mövcudluq və məhsuldarlıqla bağlı problemlər olur. Bu ƏVB problemlərinin həlli məlumatların əlçatanlığını və miqyasını yaxşılaşdırmaq məqsədi daşıyan NoSQL adlanan qeyri-relyasion (qeyri-əlaqəli) verilənlər bazası adlanan yeni nəsil verilənlər bazalarının inkişafına gətirib çıxardı.

NoSQL, əlaqəli verilənlər bazasına xas olan məlumatların saxlanması və istifadəsində məhdudiyyətlərdən xilas olmağa imkan verən məlumatların təşkilinə yanaşmadır. Strukturlaşdırılmamış yanaşmadan istifadə edən NoSQL verilənlər bazaları verilənləri emal etmək üçün bir çox səmərəli yollar təklif edir.

*SQL və NoSQL-in müqayisəsi:*

- ✓ məlumatların saxlanması strukturunu və metodu: əlaqəli verilənlər bazası əvvəlcədən müəyyən edilmiş məlumat saxlama strukturunu tələb edir, NoSQL verilənlər bazası belə məhdudiyyətlər qoymur;
- ✓ sorğular: lisenziyadan asılı olmayaraq, əlaqəli VBİS-lər vahid SQL dilindən istifadə edir, hər bir NoSQL VBİS verilənlərlə işləmək üçün öz üsulunu həyata keçirir;
- ✓ miqyaslılıq: əlaqəli və NoSQL verilənlər bazaları asanlıqla şaquli olaraq miqyaslanır (məsələn, sistem resurslarını artırmaqla). Bununla belə, NoSQL VBİS-ləri üfüqi miqyasda daha asan yollar təqdim edir (məsələn, bir neçə maşından ibarət klaster yaratmaq).
- ✓ etibarlılıq: əlaqəli VBİS-lər ACID xassələrini dəstəkləyir, buna görə də NoSQL-dən daha etibarlı və daha yaxşı məlumatların bütövlüyünü təmin edirlər. Digər tərəfdən, NoSQL VBİS-lər, ACID xüsusiyyətlərindən imtina edərək, səmərəliliyi və miqyaslılıq artırır;
- ✓ texniki dəstək: əlaqəli verilənlər bazası üçün istehsalçılardan və ya istifadəçi qrupundan dəstək almaq asandır, çünki əlaqəli verilənlər bazaları daha uzun inkişaf tarixinə və daha çox istifadəçi əhatəsinə malikdir;

✓ saxlanma və mürəkkəb verilənlər strukturlarına giriş: əlaqəli verilənlər bazaları əvvəlcə mürəkkəb çox əlaqəli verilənlərin emalı üçün nəzərdə tutulmuşdu və buna görə də bu baxımdan NoSQL həllərindən üstündür;

✓ funksionallıq: əlaqəli verilənlər bazaları çoxlu sayda müxtəlif funksiyaları yerinə yetirir və məlumatların ciddi ardıcılığını təmin edir. Bununla belə, bir çox tətbiqlər üçün bu cür funksiyalar dəsti, eləcə də ACID tələblərinə cavab vermək lazım deyil.

NoSQL təsnifatı onların istifadə etdikləri məlumat saxlama mexanizminin növünə əsaslanır. Bu baxımdan NoSQL verilənlər bazalarının dörd əsas növü mövcuddur: açar-dəyər, açar-sənəd, sütun ailəsi və qrafik verilənlər bazası.

NoSQL açar-dəyər verilənlər bazası NoSQL verilənlər bazalarının ən sadə növlərindən biridir, çünki verilənlər sadəcə olaraq açar-dəyər cütləri ilə təmsil olunur, burada unikal açarlar verilənlər bazasında müəyyən bir dəyərə işarə edir. Bu struktur heç-cədvəlinə bənzəyir. Əsas məşhur NoSQL açar-dəyər verilənlər bazalarına Riak, Redis, Berkeley DB daxildir.

NoSQL açar-sənəd verilənlər bazası açar-dəyər strukturuna əsaslanır, lakin məlumatları JSON, BSON, XML və YAML kimi sənədlərdə saxlayır. Bu tip verilənlər bazasında sənədlər əsas anlayışdır. Bu sənədlərdə çoxlu müxtəlif növ məlumat və digər subsənədlər ola bilər. MongoDB, CouchDB, TerraStore, OrientDB, RavenDB ən məşhur açar-sənəd NoSQL verilənlər bazalarıdır.

Sütun ailəsi NoSQL verilənlər bazaları: hər bir sətirdə bir çox sütundan ibarət olan və hər sütunun açar-dəyər cütü olduğu məlumat modelindən istifadə edin. Hər sətirdəki sütunların sayı fərqli ola bilər [1, s.83-89]. Sütunlu verilənlər bazalarına misal olaraq Cassandra, HBase, Hypertable və Amazon DynamoDB daxildir.

NoSQL qrafik verilənlər bazaları qrafik modelindən istifadə edir, burada kənarlar və qovşaqlar verilənlər bazasındakı obyektlər arasındakı əlaqələri təmsil edir [5]. Belə verilənlər bazalarına nümunələr Neo4j, Infinite Graph, OrientDB və ya FlockDB-dir.

### **NewSQL verilənlər bazası**

NewSQL, NoSQL-dən yeni paylanmış sistem yanaşmalarını götürən və əlaqəli məlumat təqdim etmə modelini və SQL sorğu dilini geridə qoyan verilənlər bazasıdır. Başqa sözlə desək, NewSQL SQL və NoSQL-in üstünlüklərinin effektiv birləşməsidir [6, s.45-55].

NewSQL Xüsusiyyətləri:

- ✓ SQL verilənlərlə işləmək üçün əsas mexanizm kimi;
- ✓ əməliyyatların etibarlılığı;
- ✓ real vaxtda bloklama yoxdur;
- ✓ ənənəvi əlaqəli verilənlər bazası idarəetmə sistemi;
- ✓ paylanmış arxitektura;
- ✓ RAM-da verilənlərlə işləmək;
- ✓ “Paylaşılan-heç bir şey” arxitekturası: Bu, hər bir yeniləmə sorğusunun bir qovşaq üzərində yerinə yetirildiyi paylanmış hesablama arxitekturasıdır. Bu, müxtəlif sorğulardan istifadə edərək eyni məlumatları eyni vaxtda yeniləməyə imkan verir.

*NewSQL verilənlər bazası təsnifatı:*

Mütəxəssislər NewSQL kateqoriyasında informasiya verilənlərinin saxlanması və emalı sahəsində bütün effektiv həlləri bir neçə sinfə bölürlər.

1. NewSQL kiçik ölçülü klasterlərdə standart SQL verilənlər bazalarından - Oracle RAC, MySQL Cluster və s.-də istifadə prosesini sadələşdirən təhlükəsiz və etibarlı İT üsullarıdır. Bu cür həllər ara proqramda işləyir - sorğuların verilənlər bazası arasında arakəsmələr üzərində paylanmasına cavabdeh olan unikal ara qatdır.

2. NewSQL hazırda istifadə olunan verilənlər bazası üçün yaradılmış informasiya məlumatlarının emalı və saxlanması üçün mexanizmlərin innovativ modelidir. Bu kateqoriyada ən populyar mexanizm fraktal iyerarxiya əlaqəli indekslərlə işləyən, konfigurasiyası asan və yüksək məhsuldarlığa malik olan TokudB-dir.

3. NewSQL, SQL ilə asanlıqla qarşılıqlı əlaqədə ola bilən, demək olar ki, sıfırdan yaradılmış VBİS-dir. Fakt budur ki, bir çox müasir NewSQL verilənlər bazası yaddaş-daxili verilənlər bazalarıdır, onların əsas məlumat saxlama yeri RAM-da yerləşir.

Vacib iş məlumatlarının diskdə saxlanması bütün mövcud əlaqəli verilənlər bazalarında bir qüsurdur. Sürətli məlumat emalı yalnız bütün məlumatlar RAM-da saxlandıqda mümkündür. Məsələndə RAM miqdarı nisbətən kiçik olsa belə, məlumatların işlənməsi sürətinin itirilməsi bir neçə yüz qovşaqdan ibarət klasterin yığılması ilə asanlıqla kompensasiya edilə bilər.

Qeyd etmək lazımdır ki, məlumatların əməli yaddaşda saxlanması enerjinin gözlənilmədən söndürülməsi dəyərli məlumatların itirilməsinə səbəb olmur, çünki bütün verilənlər bazaları yerinə yetirilən əməliyyatları qeyd edir və məlumatların anlıq görüntülərini mütəmadi olaraq diskdə saxlayır.

Bu yaxınlarda bir çox NewSQL tərtibatçıları söylərini buludlarda və virtual mühitlərdə VBİS-lərin yerləşdirilməsi və idarə edilməsi prosesinin sadələşdirilməsinə yönəlməyə başladılar. Məsələn, NuoDB klaster verilənlər bazası idarəetmə sistemlərinə nəzarət və monitoring üçün nəzərdə tutulmuş xüsusi web interfeysdir.

Strukturlaşdırılmış məlumatlarla işləmək üçün universal platformaların yaradılmasına çox diqqət yetirilir. Burada biz xüsusilə FoundationDB, çeşidlənmiş açar-dəyər cütlərini saxlayan sərbəst paylanmış NoSQL VBİS-ni vurğulamalıyıq.

Beləliklə, yalnız həm NoSQL, həm də SQL ola bilən VBİS-i NewSQL həlli adlandırmaq olar. Digər variantlar yalnız müvəqqəti əvəzedicilər kimi nəzərə alınmalıdır.

NewSQL verilənlər bazası nümunələri NuoDB, ClustrixDB, VoltDB, Xeround, MemSQL-dir.

#### *Əlaqəli verilənlər bazalarının fəaliyyətinin optimallaşdırılmasına yanaşmalar*

ƏVB-nin optimal işləməsi adətən minimum resurs xərcləri ilə maksimum məhsuldarlığı təmin edən məlumatların əldə edilməsi və manipulyasiyası üçün belə şəraitin yaradılması kimi başa düşülür. Verilənlər bazası tərtibatçıları tez-tez server məhsuldarlığı kimi mühüm resursa təsir göstərə bilmirlər. ƏVB-də məlumat əldə etmə sürətinə nəzarət etmək üçün aşağıdakı sahələri vurğulamaq mümkündür:

#### *Əlaqəli verilənlər bazası sxeminin optimallaşdırılması*

ƏVB sxeminin modifikasiyası yolu ilə verilənlərə çıxış sürəti baxımından ƏVB ilə işin optimallaşdırılması iki yolla həyata keçirilə bilər:

1) verilənlər bazası sxeminin normallaşdırılması;

2) verilənlər bazası sxeminin denormallaşdırılması (denormallaşdırılma-verilənlər bazası strukturunun qəsdən normallaşdırma meyarlarına cavab verməyən vəziyyətə gətirilməsidir, adətən lazımsız verilənlər əlavə etməklə verilənlər bazasından oxu əməliyyatlarını sürətləndirmək üçün həyata keçirilir).

#### *Normallaşdırma*

Normallaşdırma, verilənlərin artıqlığı (təkrarlanması) anomaliyasını və əlavə etmə, yeniləmə və silmə anomaliyaları kimi digər arzuolunmaz xüsusiyyətləri minimuma endirəcək şəkildə cədvəllərin formalaşdırılmasına yanaşmadır. Normallaşdırma əlaqəli verilənlər bazasının strukturunu optimallaşdırmaq üçün standart yanaşmadır.

Normallaşdırma əsasən iki məqsəd üçün istifadə olunur:

1. artıq (dublikat) məlumatların aradan qaldırılması;

2. məlumatların istinad bütövlüyünün təmin edilməsi.

Cədvəl normallaşdırılmayıbsa, o, nəinki əlavə yaddaş sahəsi sərf edir, həm də anomaliyaların daxil edilməsi, yenilənməsi və silinməsi səbəbindən məlumatları itirmədən verilənlər bazasının işlənməsini və yenilənməsini çətinləşdirir.

Verilənlər bazası sxeminin normallaşdırılması əlaqə atributlarının funksional asılılığı və parçalanma anlayışlarına əsaslanır.

Ən çox istifadə edilən normal formalar:

- ✓ Birinci normal forma (1NF): Əgər bütün atribut qiymətləri atomardırsa, əlaqə sxemi  $R$  ilk normal formada olur.
- ✓ İkinci normal forma (2NF):  $R$  münasibət sxemi 2NF-də o zaman və yalnız 1NF-də olarsa və əsas hissədən asılı olan qeyri-əsas atributlara malik olmadıqda olur.
- ✓ Üçüncü normal forma (3NF):  $R$  münasibət sxemi 2NF-ni təmin etdikdə 3NF-ni təmin edir və əlaqənin keçid asılılığı olmur.
- ✓ Boyc-Codd Normal Forması (BCNF): Aşağıdakı şərtlər yerinə yetirildikdə, əlaqə sxemi BCNF-də olur:  $R$  3NF-ni təmin edir və hər bir funksional asılılıq üçün  $R \rightarrow Y, X$  super açardır, burada  $X$  və  $Y$   $R$ -ə məxsus atributlar dəstidir.
- ✓ Dördüncü Normal Forma (4NF):  $R$  BCNF-ni təmin etdikdə və əlaqədə atributlar arasında çoxqiymətli asılılıqlar olmadığı zaman əlaqə sxemi dördüncü normal formada olur.

Beləliklə, normallaşdırmanın məqsədi anomaliyaların baş verməsini minimuma endirmək və məlumatların mövcudluğunu maksimum dərəcədə artırmaqdır. Normallaşdırma adətən ƏVB strukturunun yaradılması üçün standart qayda kimi qəbul edilsə də, bəzi hallarda məhsuldarlığı və verilənlər bazasından istifadənin asanlıqını yaxşılaşdırmaq üçün denormallaşdırma prosesindən istifadə etmək mümkündür [8, s.126].

### *Denormallaşdırma*

Denormallaşdırma sorğunun işlənməsi məhsuldarlığını yaxşılaşdırmaq üçün verilənlər bazası sxemini normallaşdırmaqdan uzaqlaşma prosesidir. Denormallaşdırmanın əsas məqsədlərindən biri tələb olunan məlumatları əldə etmək üçün daxil edilməli olan fiziki cədvəllərin sayını azaltmaq, həmçinin sorğuya cavab almaq üçün tələb olunan əlaqə sayını azaltmaqdır.

Verilənlər bazasında nisbətən az yeniləmə əməliyyatları və çoxlu sayda cədvəl birləşmə əməliyyatlarını əhatə edən çoxlu axtarış sorğuları olduqda da normallaşdırma səmərəsiz ola bilər. Digər tərəfdən, denormallaşdırma sorğu sürətini artırarkən məlumatların bütövlüyünü azalda bilər.

Bu gün bir çox ƏVB tərtibatçıları yerinə yetirilən sorğuları nəzərə alaraq verilənlər bazası strukturunun optimallaşdırılması problemlərini həll etmək üçün denormallaşdırmadan istifadə edirlər. Hanus M. verilənlər bazasının istismarı zamanı verilənlərin necə istifadə olunacağını nəzərə alaraq normallaşdırma və denormallaşdırma proseslərinin növlərinin siyahısını işləyib hazırlamışlar. Lakin denormallaşdırmaya bu yanaşma kifayət qədər tam təsvir edilməmişdir və onun üstünlüklərini və mənfi cəhətlərini qiymətləndirmək çətindir [3]. Rodgers U. ƏVB dizaynında bəzi ümumi vəziyyətlərdə denormallaşdırmaya ümumi yanaşmaları təsvir etdilər. Məsələn, müəlliflər əsas olmayan atributları olan birə bir və ya çoxdan çox əlaqəsi olan iki cədvəl olduqda denormallaşdırmanın faydalı olduğunu iddia edir [7, s.46-53]. C.E.Dabrowski, D.K.Jefferson, J.V.Carlis, S.T.March məlumatların klasterləşdirilməsi konsepsiyasına əsaslanaraq denormallaşdırma prosesini minimuma endirmək üçün bir yanaşma təqdim etdilər [4, s.71-86].

Məlumatların klasterləşdirilməsindən istifadə ilə bağlı problemlərdən biri onun verilənlərin müəyyən bir fiziki təşkili növü ilə məhdudlaşmasıdır. Coleman G. əlaqəli verilənlər bazası idarəetmə sisteminin çevikliyi və məhsuldarlığı arasında uyğunluğu nəzərdən keçirirlər [2, s.63-64].

Onlar denormallaşdırma zamanı məlumatların yenilənməsi tezliyi, verilənlər bazası idarəetmə sistemi, əməliyyat sistemi və avadanlıqların birlikdə necə işləməsi haqqında məlumat kimi amilləri nəzərə almağı təklif edirlər.

### **ƏDƏBİYYAT SİYAHISI:**

1. Apte T., Ingle M., Goyal A.K. Column-Store Databases: Approaches and Optimization Techniques // International Journal of Data Mining and Knowledge Management Process. -2013. -Vol.3. -Pp.83-89.
2. Coleman G. Normalizing not only way // Computerworld. -1989. -Vol.12. -Pp. 63-64.
3. Hanus M. To Normalize or Denormalize, That is the Question // Int. CMG Conference. -1993
4. Integrating a knowledge-based component into a physical database design system / C.E. Dabrowski, D.K. Jefferson, J.V. Carlis, S.T. March // Information & Management. -1989. -Vol.17, no.2. -Pp.71-86.
5. Mosin S.V., Zykin S.V. Truth space method for caching database queries // Modelirovanie i Analiz Informatsionnykh Sistem. -2015. -Vol.22, no.2. -Pp.248-258.

6. Pavlo A., Aslett M. What's Really New with NewSQL? // SIGMOD Rec. -2016. -Vol.45, no.2. -Pp.45-55.
7. Rodgers U. Denormalization: Why, What, and How? // Database Programming & Design. -1989. -Vol.12. -Pp.46-53.
8. William H. I. Information Engineering for the Practitioner: Putting Theory into Practice (Yourden Press Computing Series). -Englewood Cliffs, New Jersey: Prentice Hall, 1988. -294 pp.

## SUMMARY

### RELATED DATABASES

**Ali Sabzaliyev**

Databases are used to store and manage data in modern computer systems. Databases are used in almost all areas of our lives: in educational institutions, business or production management, banking, solving scientific, engineering and medical problems. Databases arose from the need to manage large volumes of data in an organized and effective manner.

Metadata contains the properties of the data and a set of relationships that connect the data. For example, metadata stores the name, value type (number, date, or text) of each data element. Thus, metadata provides a more complete description of the data in the database. Given the definition of metadata, a database can be defined as a "set of self-describing data."

The article discusses the development of a set of formalized methods for converting data from a database to a NoSQL database format, and examines methods for integrating a database, optimizing its schema and queries to speed up data processing and storage.

**Keywords:** base, schema, relation, structure, information, query

## РЕЗЮМЕ

### СВЯЗАННЫЕ БАЗЫ ДАННЫХ

**Али Сабзалиев**

Базы данных используются для хранения данных и управления ими в современных компьютерных системах. Базы данных задействованы практически во всех сферах нашей жизни: в образовательных учреждениях, управлении бизнесом или производством, банковском деле, решении научных, инженерных и медицинских задач. Базы данных возникли из-за необходимости организованно и эффективно управлять большими объемами данных.

Метаданные содержат свойства данных и набор отношений, связывающих данные. Например, метаданные хранят имя, тип значения (число, дата или текст) каждого элемента данных. Таким образом, метаданные обеспечивают более полное описание данных в базе данных. Учитывая определение метаданных, базу данных можно определить как «набор самоописывающих данных».

В статье рассматривается разработка комплекса формализованных методов преобразования данных из базы данных в формат базы данных типа NoSQL, исследуются методы интеграции базы данных, оптимизация ее схемы и запросов для ускорения обработки и хранения данных.

**Ключевые слова:** база, схема, отношение, структура, информация, запрос