

# Методы работы с большими данными

Киреев Василий Сергеевич,  
к.т.н., доцент

Москва, 2020

# NoSQL

**NoSQL** - это общее обозначение принципов, направленные на воплощение механизмов управления базами данных, которые имеют ощутимые отличия от привычных моделей с доступом к информации посредством языка SQL. NoSQL характеризуется гибким состоянием, которое может меняться с течением времени и базовой доступностью для каждого запроса.

К особенностям NoSQL можно отнести:

- Использование любых типов хранилищ
- Допускается разрабатывать БД без применения схемы
- Масштабируемость в линейном формате - чем больше процессоров, тем выше производительность
- Универсальность - большие возможности для хранения и аналитики данных

**BIG  
DATA**

# NoSQL СУБД

Attribute	Relational DB	NoSQL DB
Data Modeling	Perceptive	Descriptive
ACID Transaction	Full Support	Not Always with Full Support
ETL	Required	May not be Required
Scalability	Not Scalable	Scalable by Design
High Availability	Local Cluster	Designed for Distributed Environment
Secondary Index	Full Support	Partial Support
Security	High	Not at Granular Level
SQL Support	Full Support	Not Always with SQL support
Sharding	Forced to Do when Scaling	Native
Multiple Data Center Replication	Limited Support	Mostly Supported
Asynchronous Query	Limited Support	Supported

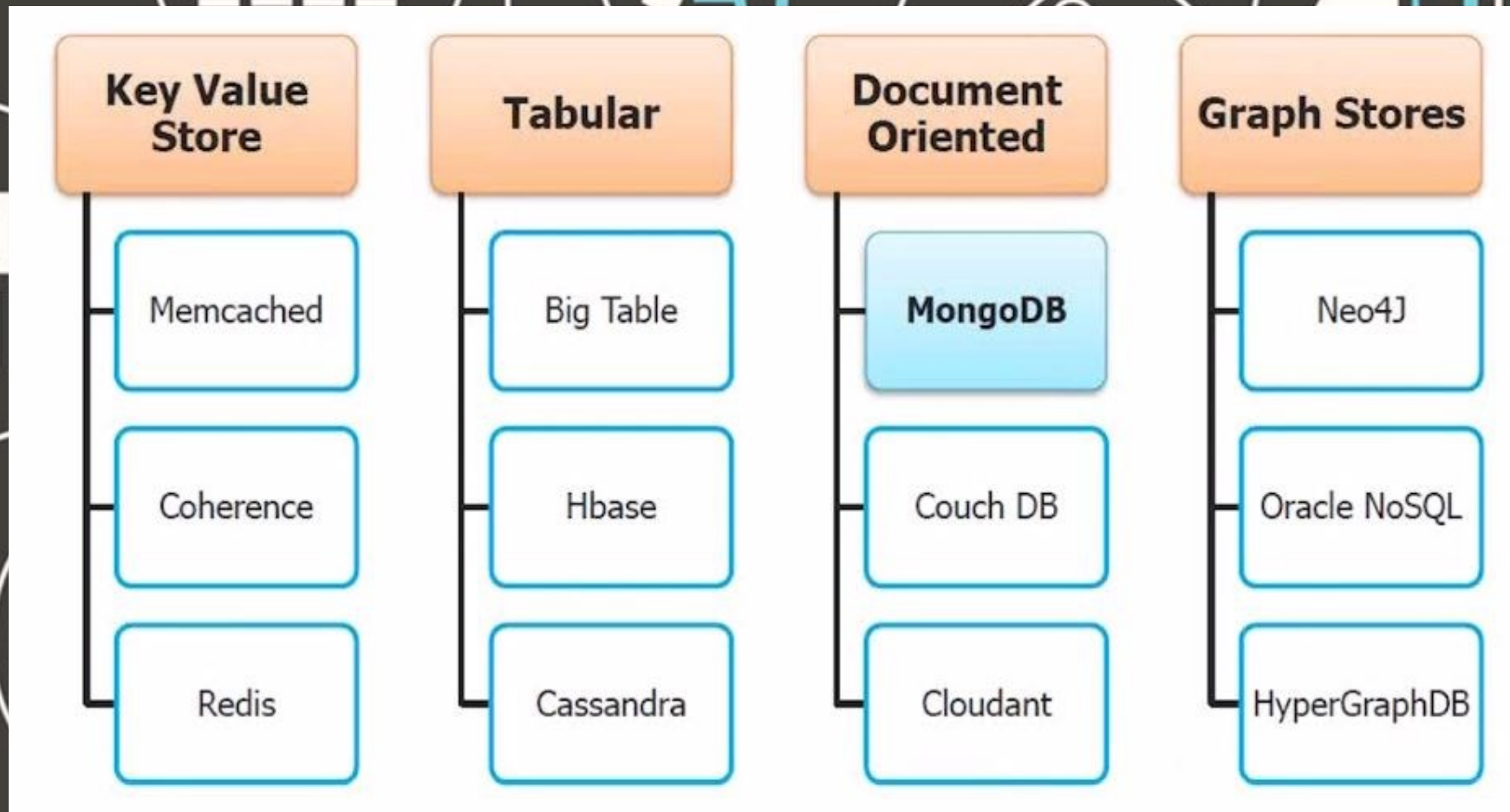


# ACID

**ACID** свойства гарантируют надежность выполнения транзакций.

- ➔ **Atomicity** (атомарность) – гарантия того, что выполнятся все операции транзакции, либо не выполнится ни одна.
- ➔ **Consistency** (непротиворечивость) – состояние базы данных находится в непротиворечивом состоянии до и после выполнения транзакции.
- ➔ **Isolation** (изоляция) – во время выполнения транзакции другие процессы не должны видеть данные в промежуточном состоянии.
- ➔ **Durability** (долговечность) – гарантия того, что если получено уведомление об успешном выполнении транзакции, можно быть уверенным, что изменения не будут отменены из-за кого-либо сбоя.

# NoSQL СУБД



# СУБД ключ-значение

- СУБД “ключ-значение” представляют собой простейший вид базы данных, являясь, по сути, ассоциативным массивом – каждому значению сопоставляется свой уникальный ключ.
- Не требуется никаких схем построения базы данных, нет никакой связи между значениями, по сути количество элементов ассоциативного массива ограничено лишь вычислительными мощностями. Именно потому данный вид хранилищ интересен в первую очередь компаниям, предоставляющим услуги облачного хостинга.

**BIG  
DATA**



# Документные СУБД

- Документные СУБД представляют собой систему хранения иерархических структур данных (документов), имеющую структуру дерева или леса. Структура дерева начинается с корневого узла и может иметь несколько внутренних и листовых узлов. Листовые узлы содержат конечные данные, которые при добавлении заносятся в индексы базы, благодаря которым можно осуществлять быстрый поиск даже при достаточно сложной общей структуре хранилища.
- Механизмы поиска позволяют находить как документы целиком, так и части документов, а древовидная структура позволяет организовывать отдельные коллекции документов одного типа или схожей тематики..

**BIG  
DATA**

# Графовые СУБД

- Графовая СУБД представляет собой обобщение сетевой модели данных и отличается сильными связями между узлами. Графовые базы данных лучше всего подходят для реализации проектов, предполагающих естественную графовую структуру данных - в первую очередь социальных сетей, а так же для создания семантических паутин.
- В подобных задачах они сильно опережают реляционные БД по производительности, простоте внесения изменений и наглядности представления информации. У некоторых баз данных существуют механизмы специальной оптимизации для работы с SSD-накопителями. Для работы с достаточно большими графами используются алгоритмы, предполагающие частичное помещение графа в оперативную память.



# Колоночные СУБД

Колоночные СУБД – это СУБД, в которой данные группируются не по строкам, а по столбцам. В ней «соседними» являются не данные из двух столбцов одной и той же строки, а данные из одного и того же столбца, но из разных строк. Как и традиционные строчные СУБД, колоночные могут использовать языки запросов к базам данных, такие как SQL.



**BIG  
DATA**



# MongoDB

MongoDB — это кроссплатформенная, ориентированная на документы база данных, которая обеспечивает высокую производительность, высокую доступность и простоту масштабирования.



# MongoDB

MongoDB — это кроссплатформенная, ориентированная на документы база данных, которая обеспечивает высокую производительность, высокую доступность и простоту масштабирования.

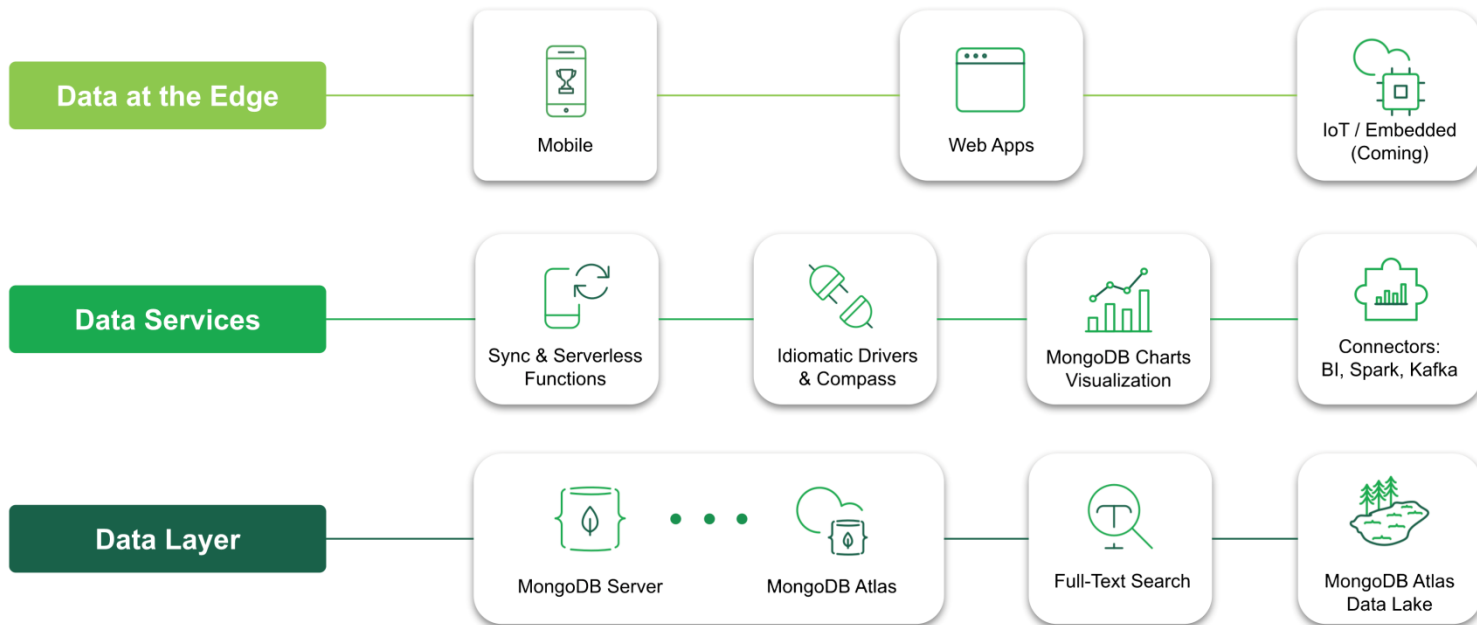
# Модель данных MongoDB

RDBMS	MongoDB
База данных	База данных
Таблица	Коллекция
Кортеж / Ряд	Документ
колонка	поле
Присоединение к таблице	Вложенные документы
Основной ключ	Первичный ключ (ключ по умолчанию <code>_id</code> , предоставленный самой mongodb)
Сервер базы данных и клиент	
Oracle	mongod
MySQL / SQLPLUS	монго



# MongoDB

## MongoDB Data Platform



# Структура данных MongoDB

user document

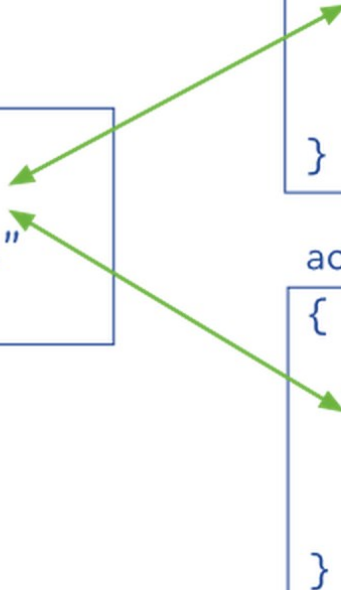
```
{  
  _id: <ObjectId1>,  
  username: "123xyz"  
}
```

contact document

```
{  
  _id: <ObjectId2>,  
  user_id: <ObjectId1>,  
  phone: "123-456-7890",  
  email: "xyz@example.com"  
}
```

access document

```
{  
  _id: <ObjectId3>,  
  user_id: <ObjectId1>,  
  level: 5,  
  group: "dev"  
}
```



# Cassandra

**Apache Cassandra** — это нереляционная отказоустойчивая распределенная СУБД, рассчитанная на создание высокомасштабируемых и надёжных хранилищ огромных массивов данных, представленных в виде хэша. Проект был разработан на языке Java в корпорации Facebook в 2008 году, и передан фонду Apache Software Foundation в 2009 .



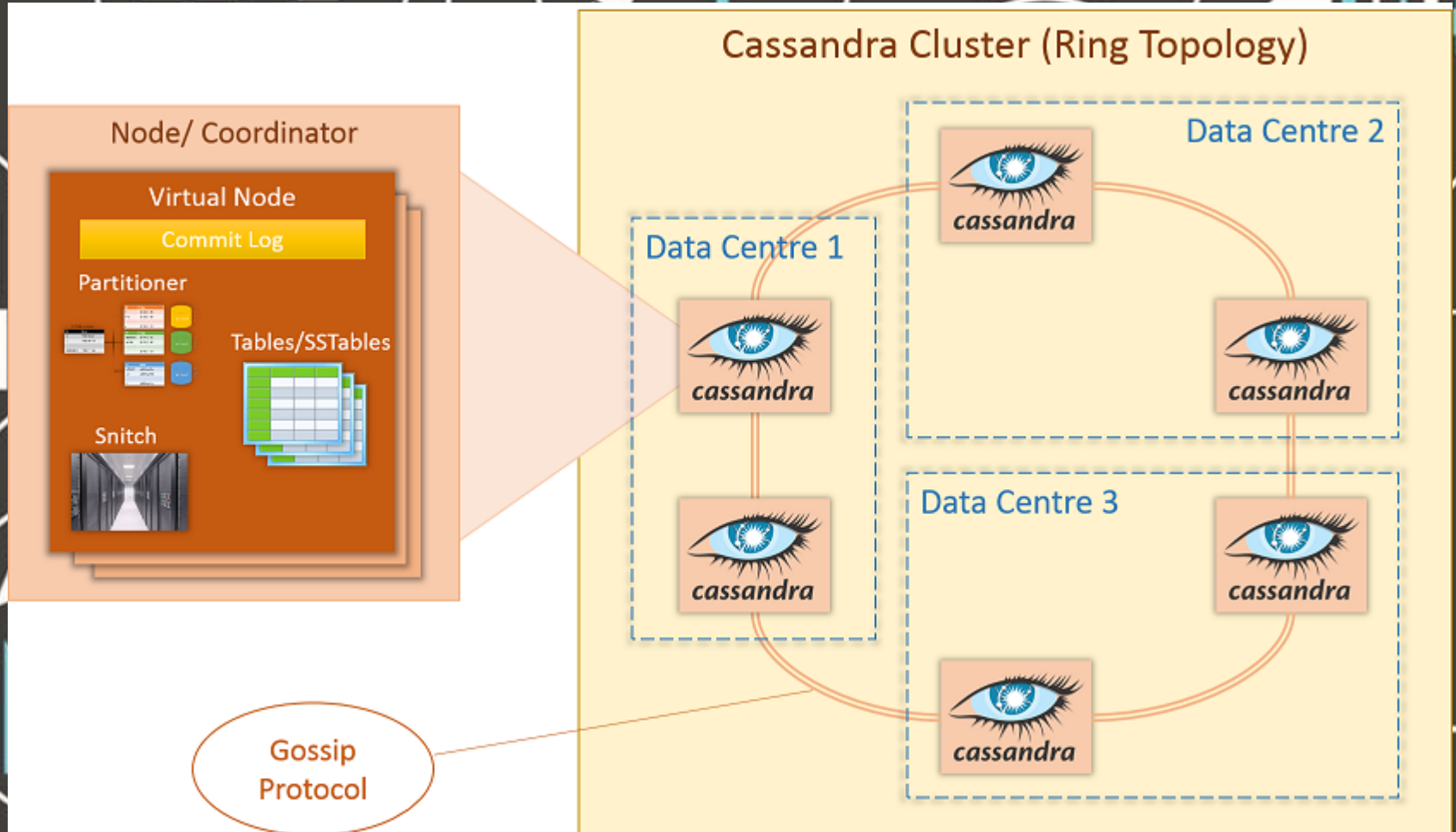


# Модель данных Cassandra

- **столбец или колонка (column)** – ячейка с данными, включающая 3 части – имя (column name) в виде массива байтов, метку времени (timestamp) и само значение (value) также в виде байтового массива. С каждым значением связана **метка времени** – задаваемое пользователем 64-битное число, которое используется для разрешения конфликтов во время записи: чем оно больше, тем новее считается столбец. Это учитывается при удалении старых колонок.
- **строка или запись (row)** – именованная коллекция столбцов;
- **семейство столбцов (column family)** – именованная коллекция строк;
- **пространство ключей (keyspace)** – группа из нескольких семейств столбцов, собранных вместе. Оно логически группирует семейства столбцов и обеспечивает изолированные области имен.

**BIG  
DATA**

# Cassandra



# HBase

**HBase** - это масштабируемая распределенная база данных, поддерживающая структурированное хранение данных для больших таблиц. Так же, как Bigtable использует распределенное хранилище данных, предоставляемое файловой системой Google, Apache HBase предоставляет возможности Bigtable-like поверх Hadoop и HDFS.



## BIG DATA

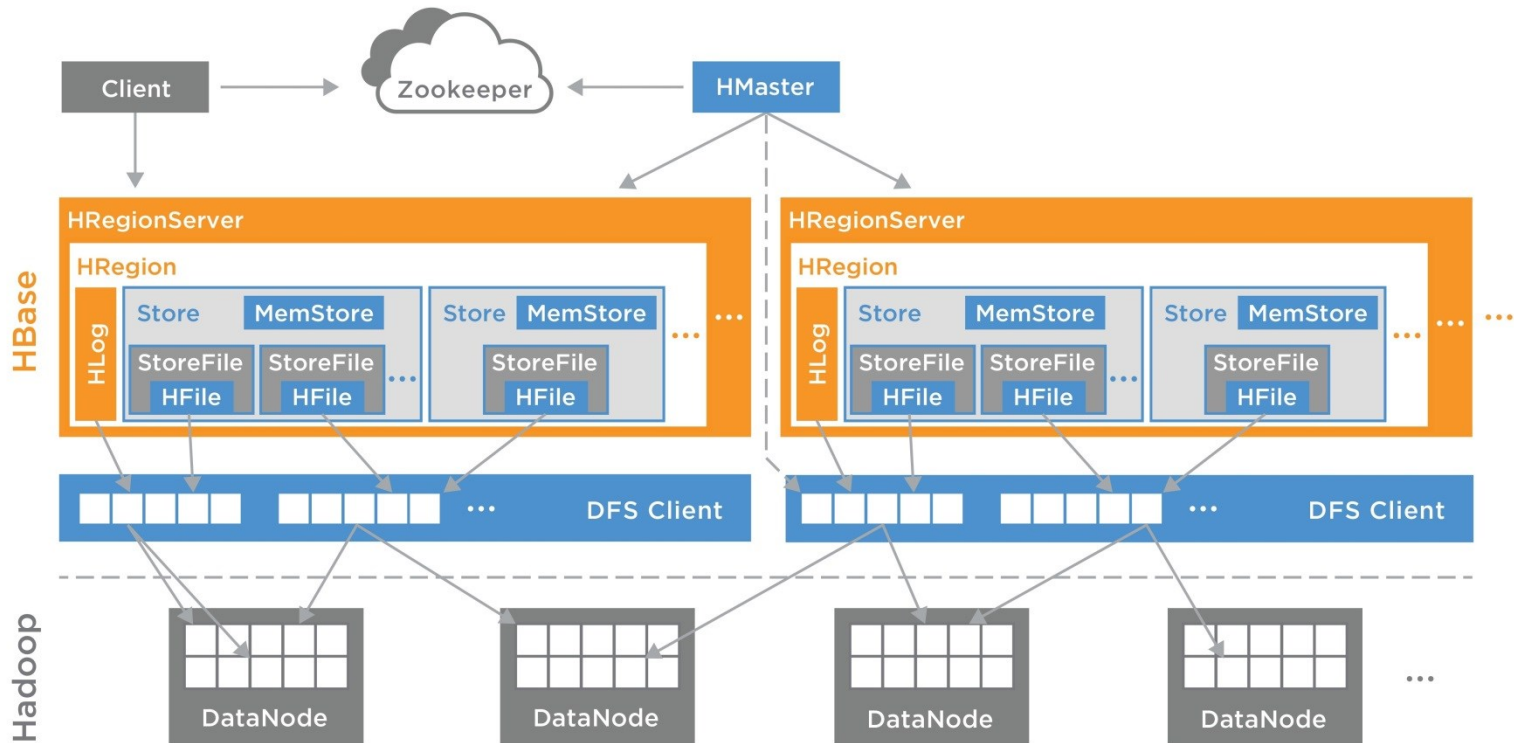


# Модель данных HBase

- Данные организованы в таблицы, проиндексированные первичным ключом **RowKey**.
- Для каждого RowKey ключа может храниться неограниченный набор атрибутов (или колонок).
- Колонки организованы в группы колонок, называемые **Column Family**. Как правило в одну Column Family объединяют колонки, для которых одинаковы паттерн использования и хранения.
- Для каждого атрибута может храниться несколько различных версий. Разные версии имеют разный **timestamp**.

**BIG  
DATA**

# HBase



# Neo4j

**Neo4j** — графовая система управления базами данных с открытым исходным кодом, реализованная на Java. По состоянию на 2017 год считается самой распространённой графовой СУБД. Разработчик — американская компания Neo Technology, разработка ведётся с 2003 года.



**BIG  
DATA**



# Модель данных Neo4j

RDBMS	База данных графиков
таблицы	диаграммы
Ряды	Вершины
Столбцы и данные	Свойства и его значения
Ограничения	Отношения
присоединяется	пересечение

# Neo4j

### Database Information

#### Node Labels

\*(198)

Book

Database

Fifth

First

Fourth

Hundred

Int

Message

Million

Movie

Person

Second

Sixth

Tens

Third

Thousand

#### Relationship Types

\*(277)

ACTED\_IN

D

DIRECTED

EMAILS

FOLLOWS

M

MEETS\_WITH

PRODUCED

R

REVIEWED

SAYS

WROTE

#### Property Keys

age

born

city

comment

date

id

isbn

location

name

num

prop1

rating

released

roles

strength

summary

tagline

title

#### Connected as

Username: neo4j

Roles: admin

Admin: :server user add

#### Database

Version: 3.4.7

Edition: Enterprise

Name: graph.db

Size: 4.10 MiB

Information: :sysinfo

Query List: :queries

\$

\$ MATCH p()->->() RETURN p LIMIT 25

Graph

Table

Text

Code

\*(26)

Database(1)

Message(1)

Person(6)

Sixth(1)

Tens(9)

First(1)

Second(1)

\*(24)

SAYS(1)

MEETS\_WITH(5)

EMAILS(1)

M(9)

R(5)

D(3)

Displaying 26 nodes, 24 relationships.

\$ MATCH (n) WHERE EXISTS(n.born) RETURN DISTINCT "node" as ent...

Table

Text

Code

entity	born
"node"	1964
"node"	1967
"node"	1961
"node"	1960
"node"	1965
"node"	1952
"node"	1978
"node"	1975
"node"	1940
"node"	1944

# Бенчмарки

	single read	single write	single write sync	aggregation (ad-hoc query)	shortest path	neighbors* distinct 1st+2nd deg	neighbors with profile data	memory usage
<b>ArangoDB</b>	100.00 %	100.00 %	100.00 %	100.00 %	100.00 %	100.00 %	100.00 %	100.00 %
2.7.0 RC2	16.962 sec.	20.530 sec.	91.816 sec.	1.250 sec.	0.061 sec.	0.464 sec.	4.327 sec.	13.142 GB
<b>MongoDB</b>	<b>86.76 %</b>	143.11 %	352.83 %	246.08 %		242.26 %	146.32 %	<b>49.20 %</b>
3.0.6 - WiredTiger	<b>14.716</b>	29.380	323.953	3.077		1.124	6.331	<b>6.466 GB</b>
<b>Neo4j</b>	522.31 %		173.02 %	225.37 %	694.41 %	511.43 %	152.59 %	<b>68.47 %</b>
2.3 M3	88.593		158.860	2.818	0.422	2.372	6.602	<b>8.998 GB</b>
<b>OrientDB</b>	168.55 %	116.96 %		1860.12 %	2188.82 %	1119.4 %	905.72 %	102.06 %
2.2 alpha	28.590	24.011		23.259	1.331	5.192	39.187	13.413 GB
<b>PostgreSQL (json)</b>	111.07 %	106.92 %	<b>75.12 %</b>	1401.22 %		461.91 %	<b>66.21 %</b>	<b>85.21 %</b>
9.4.4 - json	18.840	21.951	<b>68.972</b>	17.521		2.142	<b>2.865</b>	<b>11.199 GB</b>
<b>PostgreSQL (tab.)</b>	267.93 %	153.27 %	<b>76.87 %</b>	<b>48.77 %</b>		425.08 %	<b>43.34 %</b>	<b>77.40 %</b>
9.4.4 - tabular	45.445	31.465	<b>70.581</b>	<b>0.61</b>		1.972	<b>1.875</b>	<b>10.172 GB</b>

\*) distance 1 and the distance 2 neighbors, each of them once.

Weinberger 2015-10-13 (r207)