

NoSQL бази от данни – възможности и приложение

Веселин Николов

Темата

1. Какво е NoSQL
2. Проблеми пред RDBMS
3. Някои NoSQL DB
4. MySQL като NoSQL DB
5. Малко практика

Какво е NoSQL

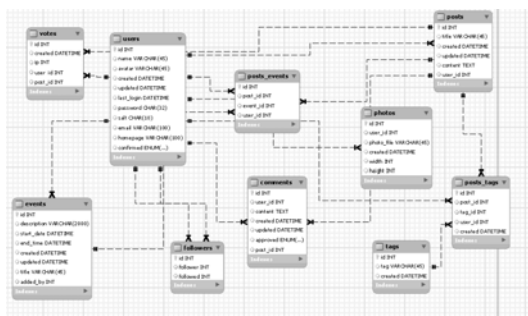


Нерелационни бази от данни,
които евентуално не поддържат SQL

Проблеми пред RDBMS?

- ✓ С прост модел
- ✓ Добре познати
- ✓ Надеждни
- ✓ Таблиците се възприемат лесно
- ✓ Учат се в училище
- ✓ Стандартизиран SQL

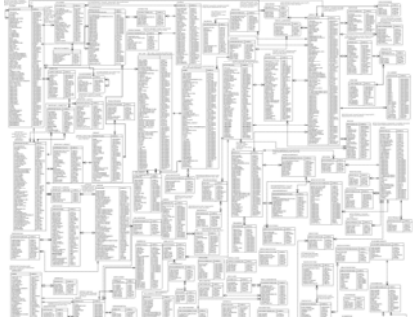
Примерна схема



Примерна заявка

```
SELECT u.id, max(u.username), count(*)  
FROM users u  
JOIN followers f ON u.id = f.followed  
GROUP BY u.id  
ORDER BY count(*) DESC  
LIMIT 10;
```

Истинска схема



govnokod.ru

```
01 SELECT
02 CASE WHEN Subtask.ParentTask_ID = 1
03 THEN 0 ELSE CASE WHEN
04 (
05 SELECT SUM(Percent) AS SumOfPercent
06 FROM Reports GROUP BY Issue_ID
07 HAVING (Issue_ID = SubTask.SubTask_ID) IS NULL
08 THEN 0 ELSE (SELECT SUM(Percent)) AS SumOfPercent
09 FROM Reports GROUP BY Issue_ID
10 HAVING (Issue_ID = SubTask.SubTask_ID)
11 ) END
12 AS SumOfPercent,
13 CASE WHEN Subtask.IsContinued < 1
14 THEN ((person_1.Base / 0.25) * 4
15 CASE WHEN Subtask.dataEnding IS NULL
16 THEN CAST(SubTask.SubTask_EndDate - DATETIME(0, GETDATE()), 0) AS integer)
17 ELSE CAST(SubTask.SubTask_EndDate - SubTask.dataEnding AS integer)
18 END -
19 DATEDIFF(mw, CASE WHEN Subtask.dataEnding IS NOT NULL THEN Subtask.dataEnding ELSE getdate() END,
20 SubTask.SubTask_EndDate) * 2) / 8 * CAST(Priority.Priority_Name AS numeric) / 1000)
21 ELSE CASE WHEN ((person_1.Base / 0.25) * (CASE WHEN Subtask.dataEnding IS NULL THEN
22 CAST(SubTask.SubTask_EndDate - DATETIME(0, GETDATE()), 0) AS integer)
23 ELSE CAST(SubTask.SubTask_EndDate - SubTask.dataEnding AS integer)END -
24 DATEDIFF(mw, CASE WHEN Subtask.dataEnding IS NOT NULL THEN Subtask.dataEnding ELSE getdate() END,
25 SubTask.SubTask_EndDate) * 2) / 8 * CAST(Priority.Priority_Name AS numeric) / 1000) > 0 THEN 0
26 ELSE ((person_1.Base / 0.25) * (CASE WHEN Subtask.dataEnding IS NULL
27 THEN CAST(SubTask.SubTask_EndDate - DATETIME(0, GETDATE()), 0) AS integer)
28 ELSE CAST(SubTask.SubTask_EndDate - SubTask.dataEnding AS integer) END -
29 DATEDIFF(mw, CASE WHEN Subtask.dataEnding IS NOT NULL THEN Subtask.dataEnding ELSE getdate() END,
```

Компромиси

Промени в схемата

```
ALTER TABLE users
ADD COLUMN followers_count INT DEFAULT 0;
```

И кеширане извън БД:
Memcache, Redis

...и проблеми

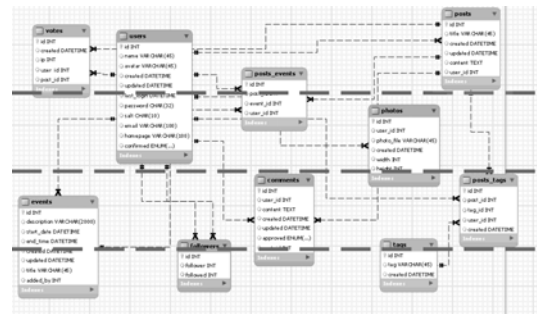
- Размер на БД > RAM
- Размер на индексите > RAM
- Размер на БД > дисковото пространство
- Locks
- Недостиг на CPU

=> #FAIL

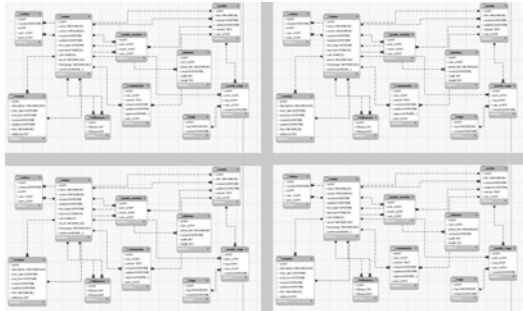
Ограничени ресурси



Проблемни места



Репликация



Скалируемост



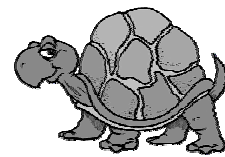
Partitioning + Sharding =

- JOIN #FAIL
- Трудни обобщения
- Много логика във вашето приложение



Проблеми пред RDBMS

- Скалируемост
- Partitioning
- Sharding
- Кеширане
- Денормализация
- Промени в схемата



Теоремата CAP

Консистентност (C):

Всички клиенти на базата от данни виждат една и съща информация, даже при конкурентно обновяване.

Наличност (A):

Всички клиенти на базата от данни могат да достъпват някоя версия на информацията.

Възможност за разделяне върху много сървъри (P):

Базата от данни може да се разделя върху множество сървъри.

Изберете две.

Ерик Брюър, 2000 г.

NoSQL решения

- Управление на компромисите
- Евентуална консистентност
- Отказ от фиксирана схема
- JavaScript, JSON, REST
- MapReduce
- GFS, HDFS
- Отказ от SQL

Мащаби на бедствието

- BigTable, Dynamo
- Cassandra, Riak
- CouchDB, MongoDB
- Redis, MemcacheDB
- Hadoop, Hbase, RavenDB, Kyoto Cabinet, Sherpa, Neo4j, FlockDB, Hypertable и др.



Google BigTable

- Многомерен масив
- Колони и семейства от колони
- SSTable
- Компресия
- GFS
- 2004 г.

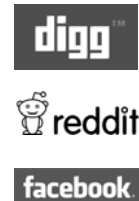


Amazon Dynamo

- Хоризонтално скалируема
- Евентуално консистентна
- Равнопоставени сървъри
- Key/value БД
- put/get
- 2007 г.

Cassandra

- BigTable + Dynamo = Cassandra
- Настройваема консистентност
- Колони, семейства, суперколони
- Cassandra-cli shell
- Вместо индекси, нови семейства
- Gossip
- Java, Thrift



Колони

```
{
  "name": "City",
  "value": "Sofia",
  "timestamp": "1290196015"
},
{
  "name": "Code",
  "value": "1784",
  "timestamp": "1290196013"
}
```

Семейства колони

```
{
  "Veselin Nikolov":{
    "Users":{
      {"emailAddress":{"name":"emailAddress","value":"dzver@bar.com"},
       "webSite":{"name":"webSite","value":"http://bar.com"}},
      "Lectures":{"Openfest":{"name":"lectureName","value":"NoSQL DB"}}
    },
    "Stefan Kanev":{
      "Users":{
        {"emailAddress":{"name":"emailAddress","value":"skanev@foo.br"},
         "twitter":{"name":"twitter","value":"skanev"}}
      },
      "Lectures":{
        {"Openfest":{"name":"lectureName","value":"MongoDB Rulez"}}
      }
    }
  }
}
```

Суперколони

```
Friend_Diggs { // Column Family – гласовете на приятелите
  12345 : { // story_id as Row key
    user_id: { // SuperColumns are User's IDs
      friend_id1: true,
      friend_id2: true,
      friend_id3: true
    }
  }
}
```

Суперколони = k/v двойка, в която стойността е набор от колони.

<https://nosqleast.com/2009/slides/sarkissian-cassandra.pdf>

Някои проблеми с Cassandra

1. Digg #FAIL, Reddit #WIN
2. Индекс = Ново семейство колони
3. Ново семейство колони = restart
4. WTF is a SuperColumn?

CouchDB

- Документи
- REST, JavaScript, JSON
- Материални изгледи
- MapReduce
- MVCC – ревизии на документите



Документ

```
{
  "_id": "163271278f438b6ccb9c87879d0001c7",
  "_rev": "3-29830772003eb7f6e49f55834d28c640",
  "url": "http://dzver.com/blog/?p=2000",
  "author": "Eric Brewer",
  "title": "The CAP Theorem",
  "comments": [
    {
      "author": "Veselin Nikolov",
      "published": "2010-08-17 15:00:00",
      "body": "Cool Dude!"
    },
    {
      "author": "Mark Twain",
      "published": "2010-08-17 15:01:00",
      "body": "Вафла трепач"
    }
  ]
}
```

Още документи

```
{
  "_id": "60206fa17eb65874b38594abd57ded49",
  "_rev": "3-29830772003eb7f6e49f55834d28c640",
  "doc_type": "Post",
  "url": "http://dzver.com/blog/?p=2000",
  "author": "Eric Brewer",
  "title": "The CAP Theorem",
}

{
  "_id": "60206fa17eb65874b38594abd57dcafd",
  "_rev": "1-5e9d312e407a13661fa69fb626c78466",
  "body": "Hello Dolly",
  "doc_type": "Comment",
  "author": "Veselin",
  "id_post": "60206fa17eb65874b38594abd57ded49",
}
```

Design document

The screenshot shows a web browser interface for editing a CouchDB design document. The address bar shows 'localhost:5984/_utils/document.html?couch3test/_design/all'. The document content is as follows:

```
{
  "_id": "_design/all",
  "_rev": "2-a1fb964b024789c060efc1b778fc48496",
  "views": {
    "by_post_id": {
      "map": "function(doc) { if (doc.doc_type=='Comment') emit(doc.id_post, 1) }",
      "reduce": "function(key, values, reduce) { return sum(values); }"
    }
  }
}
```

At the bottom, it indicates 'Showing revision 2 of 2'.

Материален изглед

Key	Grouping	exact	Value
"dc37970848c450bddb53e910930165a"			2
"dc37970848c450bddb53e91092fba5"			4
"dc37970848c450bddb53e91092f7be5"			5
"dc37970848c450bddb53e91092e6678"			3
"dc37970848c450bddb53e91092e2f9a"			7
"dc37970848c450bddb53e91092d627e"			4
"dc37970848c450bddb53e91092cf991"			2
"dc37970848c450bddb53e91092cb00e"			6
"dc37970848c450bddb53e91092c096e"			3
"dc37970848c450bddb53e91092c04dc"			5
"dc37970848c450bddb53e91092b749e"			9
"dc37970848c450bddb53e91092b296c"			5
"dc37970848c450bddb53e91092a8810"			3
"dc37970848c450bddb53e910929b3fd"			8

Някои проблеми с CouchDB

1. Бавен запис
2. Бавно първоначално създаване на изгледи
3. GROUP BY .. ORDER BY count – само с hack.

MongoDB

- Документи
- Индекси
- Mongo shell
- JSON, JavaScript, MapReduce
- ReplicaSet, auto sharding*



Разлики с CouchDB

- No single server durability (fixed?)
- MapReduce работи подобно на SQL
- Работи бързо

Hadoop

- HDFS
- MapReduce
- HBase
- Pig
- Hive

Още няколко важни NoSQL DB

- Redis & MemcacheDB
- FlockDB за friends & followers
- Neo4j за графи
- Riak – Динамо аналог с MapReduce

MySQL NoSQL

1. FriendFeed
2. Yoshinori Matsunobu

MySQL във FriendFeed

1. Сериализиран масив в blob
2. Нова таблица за всеки индекс

<http://bret.appspot.com/entry/how-friendfeed-uses-mysql>

MySQL без SQL

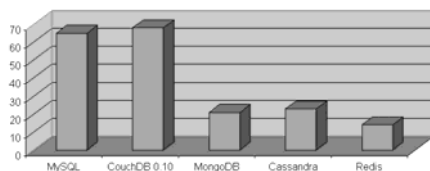
1. NDB Cluster + NDBAPI
2. HandlerSocket Plugin
3. Заявки по PK, без SQL

<http://yoshinorimatsunobu.blogspot.com/2010/10/using-mysql-as-nosql-story-for.html>

Малко практика

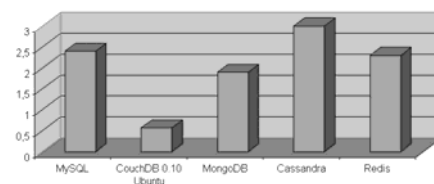
Тестовите са спекулативни,
ако не се провеждат в разпределена среда

Запис на данни



5000 документа в 50 нишки
MySQL: 65 sec, MongoDB 21 sec, Cassandra: 23 sec

Извличане на данни



1000 изчитания на статия с нейните коментари, конкурентно в 10 нишки
MySQL: 2.4 sec, CouchDB: 0.57 sec.

Обобщение

- Прости и бързи
- Подходящи за специфични задачи
- Нужни са много тестове

Перспективи за развитие

- MongoDB single server durability
- Cassandra – конфигурация в реално време
- CouchDB – поддредане на MapReduce

- Hadoop PIG, Hadoop Hive QL
- MapReduce в RDBMS