UNIUNEA EUROPEANĂ

GUVERNUL ROMÂNIEI

Instrumente Structurale
2007-2013

Platformă de e-learning și curriculă e-content pentru învățământul superior tehnic

Baze de date 1

14. Baze de date non SQL

Baze de date non SQL

In acest modul vom prezenta:

- Caracteristici pentru baze de date non SQL (numite in literatura si NoSQL - not only SQL)
- Operatii cu baze de date semistructurate
- Analiza performante framework Hadoop, MongoDB.

Caracteristici

- Exista actualmente o serie de sisteme de gestiune care se deosebesc de cele clasic relationale prin una sau ambele din caracteristicile de mai jos:
  - Folosirea modelului relational al datelor
  - Folosirea limbajului de interogare SQL

Caracteristici

- Astfel de sisteme se caracterizeaza in plus prin unele din elementele de mai jos:
  - Sunt proiectate pentru medii distribuite sau paralele
  - Sunt folosite pentru gestiunea documentelor si mai putin a datelor atomice
  - Ofera garantii mai slabe de consistenta

Caracteristici

- continuare
  - Unele sunt proiectate pentru a oferi consistenta la citire, specifica in sistemele clasice
  - In cazul arhitecturilor distribuite datele sunt prezente redundant in mai multe noduri. In felul acesta este realizata scalabilitatea si protectia impotriva caderilor accidentale ale unor noduri

Limbaje folosite

- Sistemele de tip NoSQL pot folosi si folosesc si ele limbaje de cereri, ca de exemplu:
  - UnSQL
  - Xquery

UnSQL

- UnSQL = Unstructured Query Language
- Exemple (wikipedia):

```
INSERT INTO abc VALUE { type: "nested", content:
  { content: "nested object", x: 1, y: {str:
  "hi", str2: "there"}, z: true } };

SELECT { x:abc.type, y:abc.content.x,
  z:abc.content.x+50 } FROM abc;
```

XQuery

- Folosit pentru cereri pe colectii de date XML
- Exemplu: Fie documentul STUD.XML

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<studenti>
   <student spec="CALC">
     <nume>Ionescu Ion</nume>
     <an>3</an>
     <medie>9.30</medie>
   </student>
<student spec="AII">
     <nume>Popescu Vasile</nume>
     <an>2</an>
     <medie>9.30</medie>
   </student>
</studenti>
```

XQuery

- Atunci cererea:

```
doc("STUD.XML")/studenti/student[an=3]
```

Va returna:

```
<student spec="CALC">
     <nume>Ionescu Ion</nume>
     <an>3</an>
     <medie>9.30</medie>
   </student>
```

- Iar cererea:

```
doc("STUD.XML")/studenti/student/nume
```

Va returna:

```
<nume>Ionescu Ion</nume>
<nume>Popescu Vasile</nume>
```

Date semistructurate

- Datele semistructurate nu sunt organizate conform modelului relational
- Ele contin totusi marcaje pentru a separa elementele din care se compun (care se numesc campuri sau atribute - ca in cazul relational)
- Nu exista o schema a datelor, ele insa contin in ele descrierea
- Pot exista ierarhii de elemente
- Ordinea campurilor nu este semnificativa
- Elemente diferite din aceeasi clasa pot avea atribute diferite

Exemple de date semistructurale

- Documentele XML sau alte documente similare
- Documentele HTML
- Documente de tip EDI (ex.: standardul UN/EDIFACT)
- OEM (Object Interchange Model)

LORE si LOREL

- La Stanford a fost dezvoltat un sistem pentru gestiunea datelor semistructurate: LORE
- Acesta a fost adaptat ulterior pentru gestiunea fisierelor XML
- LORE folosea un limbaj inspirat din SQL: LOREL

Exemple de cereri

(Exemplele sunt preluate din [4])

```
select Guide.restaurant.name,
Guide.restaurant(.address)?.zipcode
where Guide.restaurant.% grep "cheap";

select X.address
from Guide.restaurant X, X.zipcode Y
where Y = 92310;

select X.name
from John.child X
where exists JN in John.name :
exists XN in X.name : JN == XN;
```

MongoDB - sistem Non SQL

- Un exemplu de sistem non SQL este MongoDB.
- Acesta are ca si caracteristici:
  - Este orientat obiect
  - Performante foarte bune (si disponibilitate f. buna)
  - Scalabil

Din ce se compune MongoDB

In [1] este prezentata urmatoarea descriere pentru MongoDB:

- 1. Unul sau mai multe "shards" (cioburi), fiecare ciob deţinând o parte din datele totale (gestionate în mod automat). Citirile şi scrierile sunt automat rutate catre ciobul-cioburile corespunzătoare. Fiecare ciob este susţinut de o mulţime-copie (replica set) care deţine doar datele acelui pentru ciob.
- O mulţime-copie este formata din unul sau mai multe servere, fiecare continand copii ale aceloraşi date. La un moment dat unul este primar, iar restul sunt secundare. În cazul în care copia primara cade una din cele secundare devine automat primara. Toate scrierile şi citirile consistente merg la copia primara şi toate eventualele citiri consistente sunt distribuite între cele secundare.

Din ce se compune MongoDB

- continuare

- 2. Mai multe servere de configuratie (config), fiecare dintre ele deţine o copie a metadatelor care indica ce se gaseste pe fiecare ciob.
- 3. Unul sau mai multe routere, fiecare acţionează ca un server pentru unul sau mai multi clienti. Clientii trimit interogări / actualizări la un router, iar routerul le ruteaza catre ciobul corespunzător în timp ce consultarea serverelede configurare.

Din ce se compune MongoDB

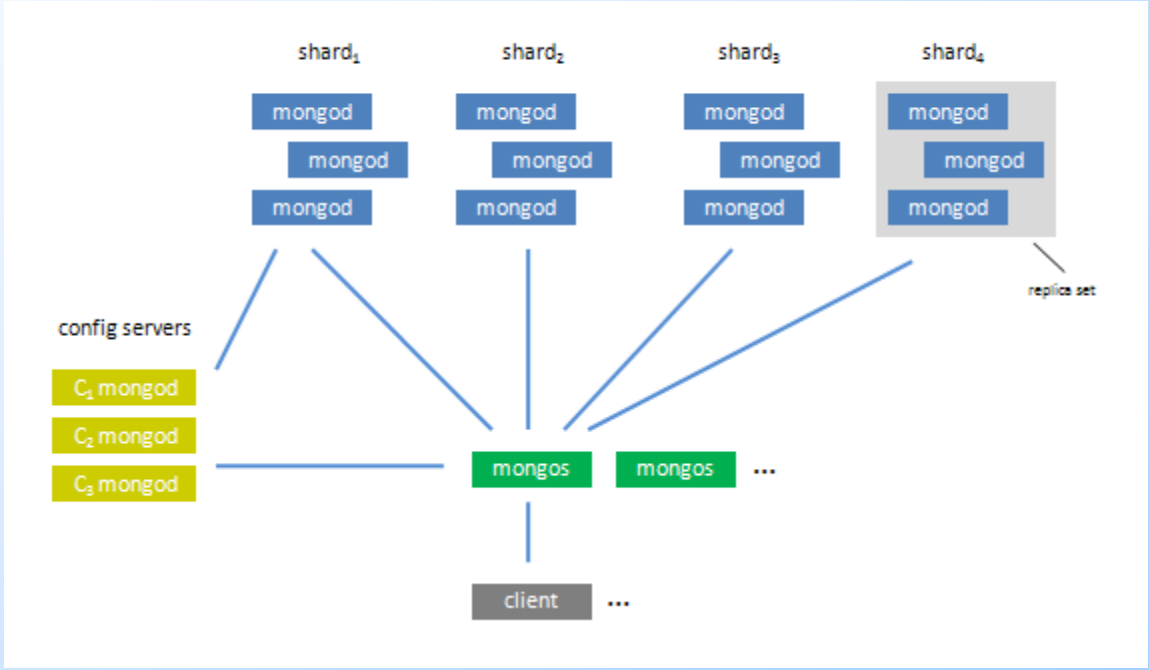
- Continuare

4. Unul sau mai multi clienti, fiecare dintre ele este (parte din) cererea utilizatorului şi probleme de comenzi pentruun router, prin intermediul bibliotecii client Mongo (driver) pentru limba ei.

- **mongod** este programul de server (de date sau config) - echivalent mysqld
- **mongo** este programul client - echivalent mysql
- **mongos** este programul router.

Exemplu de arhitectura

(din [1]):



Bibliografie

- 1. MongoDB Introduction, http://www.mongodb.org/display/DOCS/Introduction
- 2. MongiDB Handoop, http://www.slideshare.net/mdirolf/mongodb-hadoop-dc
- 3. Wikipedia: NoSQL http://en.wikipedia.org/wiki/NoSQL
- 4. S. Abiteboul, D. Quass, J. McHugh, J. Widom, and J. Wiener. The Lorel Query Language for Semistructured Data. International Journal on Digital Libraries, 1(1):68-88, April 1997