NoSQLdossier Project DB3: Health-E

Project DB 3: Health-E

Maxim Vandebroeck & Lukas Hanot

Inhoud

[Wat is Nosql? 2](#__RefHeading___Toc678_360408335)

[Mongodb datasharding 2](#__RefHeading___Toc692_360408335)

[Mongodb sharding setup 3](#__RefHeading___Toc694_360408335)

[Sharding setup 3](#__RefHeading___Toc696_360408335)

[Bronnen 5](#__RefHeading___Toc698_360408335)

# Wat is Nosql?

NOSQL is een omvattende naam voor alle database systemen die niet alleen met SQL werken. NoSQL is niet relationeel van structuur, het is dus minder of niet gebaseerd op het concept van tabellen, rijen, kolommen en joins. Het is ook vaak schema-less de databank legt dus geen constraints op aan de structuur en relatie van de data. Een belangerijk voordeel van NoSQL is dat het eenvoudig is om te clusteren. Dit verhoogt de performantie en de beschikbaarheid. De NoSQL waar wij gebruik van maken maakt gebruik van document stores. De data wordt bijgehouden in een document (set van gegevens die bij elkaar horen) en deze worden gebundeld in een collectie.

<http://dev.mysql.com/doc/refman/5.7/en/partitioning-limitations.html>

# Mongodb datasharding

Sharding maakt het mogelijk om data van een MongoDB database server te spreiden over meerdere shards (al dan niet meerdere fysieke server) waardoor bij het opzoeken van data niet de hele database moet worden doorzocht. Door de data te spreiden over meerdere fysieke server wordt de lees en schrijf snelheid verhoogt omdat dit parallel over meerder instanties kan gebeuren.

De voordelen van sharding zijn:

* loadbalancing: De load verdelen over vershillende servers en dus shards om performance te verhogen. Met behulp van een mongos router wordt de data verdeeld over de verschillende shards ook regelt deze automatisch de wijzigingen die aan de data gebeuren. Het is ook mogelijk om queries uit te voeren op een specifieke shard/ shard set waardoor de efficientie van het hele systeem omhoog gaat.
* Data Capacity: Sharding verdeeld de data over verschillende shards hierdoor is het mogelijk om een hoger opslag volume te bereiken en als er noodzaak is aan meer volume kunnen er gewoon nieuwe shards aan toegevoegd worden al dan niet op nieuwe servers.
* Availability: Een gesharde cluster kan blijven werken zelfs met maar gedeeltelijke lees/schrijf bewerkingen. Moest een shard onbeschikbaar worden in het systeem kan de rest van de data nog gelezen en weggeschreven worden. Config servers kunnen ook ingesteld worden als replica sets waardoor het mogelijk is dat deze het werk van de shards overnemen zolang er genoeg overblijft van de replica set.

De nadelen van sharding:

* De shard key heeft een grote impact op de performantie van de sharded cluster een slechte shard key kan leiden tot tragere queries dan op een non-sharded systeem.
* Sharding leid tot een complexere infrastructuur en vraagt meer maintenance om in te zetten.

Conclusie:

Sharding is een intresante technologie die de mogelijkheid bied om een veel groter systeem uit te bouwen. Sharding brangt ook een aantal nuttige voordelen mee zoals de mogelijkheid om te blijven werken (al zei het met minder data) wanneer een shard zou uitvallen. Anderzijds zorgt de toename in complexiteit ervoor dat het opzetten van sharding voor een kleine databank vaak meer werk is dan de winst die euit gehaald kan worden.

*Referentie*: <https://docs.mongodb.com/manual/sharding/>

https://docs.mongodb.com/v3.2/faq/sharding/

## Mongodb sharding setup

* Windows (7 en hoger)
* Poorten: 27001, 27002, 27003, 27010
* MongoDB moet op de standaard locatie zijn geïnstalleerd (C:\Program Files\MongoDB)
* MongoDB 3.2

### Sharding setup

Maak 3 folders aan om de serverdata in op te slaan. Open een commandprompt en gebruik de commando’s

|  |
| --- |
| Mkdir C:\MongodbServers\shard1  mkdir C:\MongodbServers\shard2  mkdir C:\MongodbServers\configserver |

Verander de current directory naar de bin folder van de mongodb installatie

|  |
| --- |
| cd "C:\Program Files\MongoDB\Server\3.2\bin" |

Start de shards, configuratie server, de router en de mongo terminal

|  |
| --- |
| start cmd /k mongod --dbpath C:\MongodbServers\shard1 --port 27001  start cmd /k mongod --dbpath C:\MongodbServers\shard2 --port 27002  start cmd /k mongod --dbpath C:\MongodbServers\configserver --port 27003 --configsvr  start cmd /k mongos --port 27010 --configdb localhost:27003  start cmd /k mongo --host localhost --port 27010 |

Configureer de shards in de mongo terminal.

|  |
| --- |
| sh.addShard("localhost:27001")  sh.addShard("localhost:27002") |

Aangezien we nog met een kleine database werken kunnen we best de chunksize aanpassen zodat mongo gebruik maakt van de shards i.p.v. alles op 1 shard te houden omdat de data te klein is. Gebruik volgende commando’s in de mongo terminal om dit te bereiken

|  |
| --- |
| use config  db.settings.save({ \_id:"chunksize", value: 1}) |

Activeer sharding via de mongo terminal

|  |
| --- |
| use HealthE  sh.enableSharding("HealthE") |

We hebben data nodig om onze shards te vullen. Open de originele command prompt (cd = C:\Program Files\MongoDB\Server\3.2\bin) en gebruik hier mongoimport om de in de database te importeren. !!Stuur deze data naar de mongo router!!

|  |
| --- |
| mongoimport --port 27010 --db HealthE --collection metingen --type json --file C:\temp\mongoTestData.json |

Duid m.b.v. de mongo terminal de index sleutel aan op de nieuwe collection metingen (ensureIndex)

|  |
| --- |
| db.metingen.ensureIndex({\_id:1}) |

Start het sharding proces via de mongo terminal (shardCollection)

|  |
| --- |
| sh.shardCollection("HealthE.metingen", {\_id:1}) |

# Bronnen

* <https://www.digitalocean.com/community/tutorials/how-to-measure-mysql-query-performance-with-mysqlslap>
* <https://www.opsdash.com/blog/mysql-replication-howto.html>
* <http://dev.mysql.com/doc/refman/5.7/en/mysqlslap.html>
* <https://dev.mysql.com/doc/refman/5.5/en/custom-benchmarks.html>
* <https://docs.mongodb.com/manual/sharding/>
* <https://docs.mongodb.com/v3.2/faq/sharding/>
* Databanken 3 cursus
* <http://dev.mysql.com/doc/refman/5.7/en/optimization.html>
* <http://dev.mysql.com/doc/refman/5.7/en/data-types.html>
* <http://dev.mysql.com/doc/refman/5.7/en/innodb-storage-engine.html>
* <http://dev.mysql.com/doc/refman/5.7/en/replication.html>
* <http://dev.mysql.com/doc/refman/5.7/en/partitioning.html>
* <http://dev.mysql.com/doc/refman/5.7/en/indexes.html>
* <http://dev.mysql.com/doc/refman/5.7/en/multiple-windows-services.html>