MongoDB vs MySQL

* Basisinformatie:
  + Oorsprong
  + Maker
* Jargon/termen
* Scaling: verticaal vs horizontaal
* BSON vs tables
* Pre-defined schema vs dynamic schemas
  + Dynamic schemas: geen probleem als je meer/andere data krijgt dan verwacht. Je kan arrays en objecten opslagen in een MongoDB Document.
* Query language

## MySQL – basis

MySQL is een managementsysteem voor relationele databases. Ze werd in 1995 ontwikkeld door MySQL AB en later overgenomen door Oracle Corporation. Ze is ontwikkeld met gebruik door internettoepassingen op het oog. MySQL is compatibel met en gratis beschikbaar voor alle gangbare besturingssystemen. De community edition (MySQL CE) wordt gezien als het meest gebruikte databasesysteem ter wereld.

## MongoDB – basis

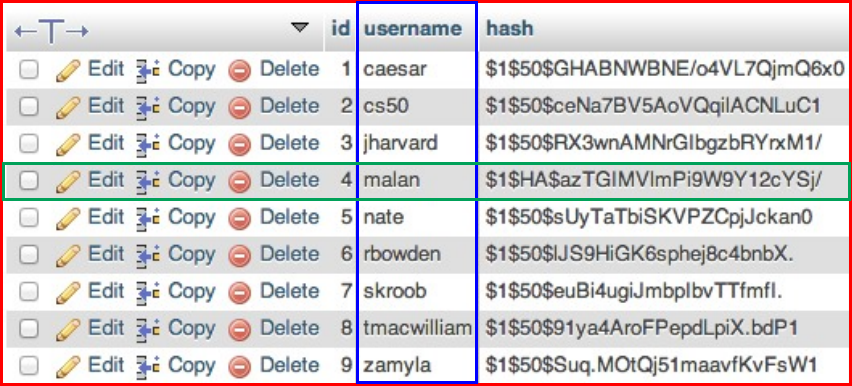
MongoDB is een NoSQL databasesysteem, ontwikkeld door MongoDB, Inc sinds 2007 met hun eerste openbare versie in 2009. Het is volledig open-source beschikbaar op GitHub. MongoDB wordt vooral gebruikt voor big-data en real-time toepassingen. Ook MongoDB is compatibel met alle gangbare besturingssystemen, maar wordt vooral op UNIX-systemen gebruikt.

## Jargon

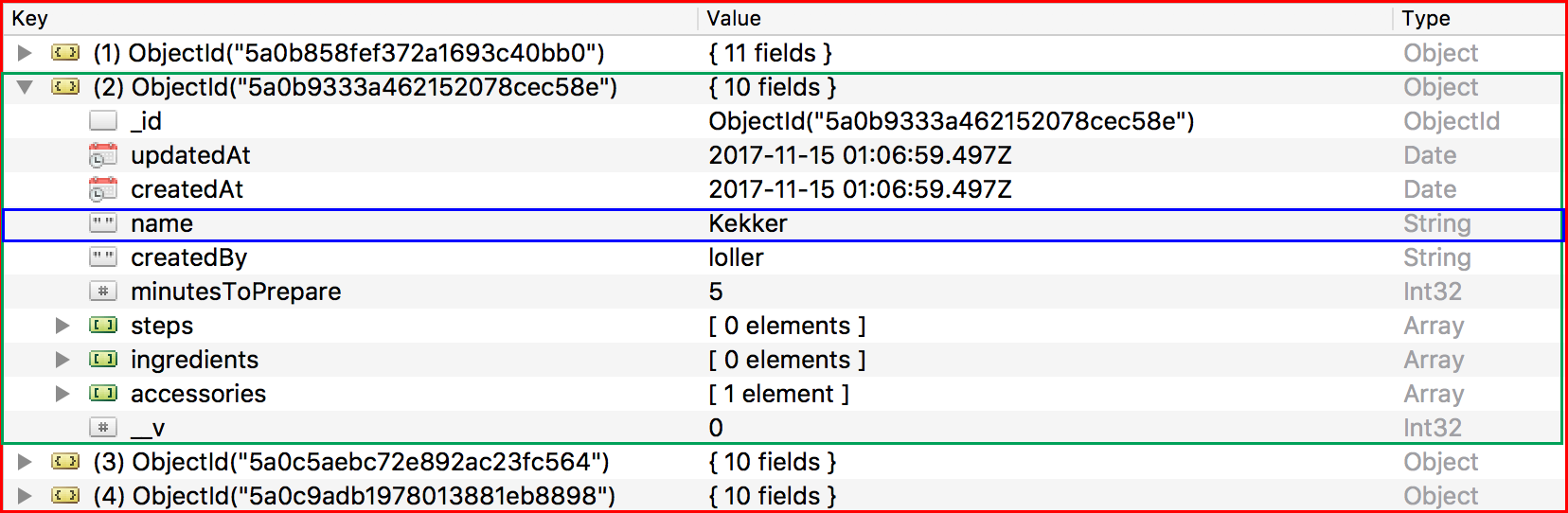
MySQL en MongoDB gebruiken andere terminologie voor gelijkende concepten. De meest voorkomende vindt u hier terug, met nadien een uitleg:

Wat men in MySQL een *Table* of *Tabel* noemt, heet in MongoDB een *Collection* of *Collectie*. Een *Row* of *Rij* wordt een *Document*. Ten laatste krijg de term *Column* of *Kolom* bij MongoDB de noemer *Field* of *Veld*. Hier vindt u nog een korte uitleg over elke term, aan de hand van foto’s:

MySQL



MongoDB



## Schaalbaarheid

MongoDB staat bekend voor zijn gebruik in big-data toepassingen. Dit wilt ook zeggen dat schaalbaarheid redelijk belangrijk is: als de hoeveelheid data de capaciteit van de server dreigt te bereiken, moet hier een oplossing voor zijn. Vaak is de eerste oplossing, zowel bij MySQL als bij MongoDB, vertical scaling. Dit doe je door meer kracht aan de database, of de server waarop die draait, te geven. Zo kan je de server krachtiger maken door er bijvoorbeeld een extra of een betere CPU bij te steken of de hoeveelheid RAM op te waarderen.

Vertical scaling heeft echter een limiet: een server kan maar zo krachtig worden voor hij niet meer verder kan. Wanneer je dit punt bereikt, ga je over op horizontal scaling: meer servers installeren, en de database over deze verschillende servers simultaan laten lopen.

Vertical scaling in MySQL is relatief complex en moet met het oog op het type data gebeuren. Vertical scaling in MongoDB is vrijwel ingebouwd door middel van *sharding*. Bij sharding wordt de database opgesplitst op basis van een index. Als je een database hebt met records met een id van 1-1000, kan deze bijvoorbeeld opgesplitst worden in 2 databases van 1 tot 500, en van 501 tot 1000. Als de applicatie of user document 300 nodig heeft, spreekt hij de eerste database aan enzovoort.