

MongoDB

wprowadzenie

Plan

- ▶ Historia
- ▶ Podstawowe pojęcia:
 - ▶ Dokument
 - ▶ Kolekcja
- ▶ Generowanie identyfikatora
- ▶ Model danych
 - ▶ Dokumenty zagnieżdżone
 - ▶ Dokumenty z referencjami
- ▶ ACID
- ▶ Język zapytań
- ▶ Framework agregacji

Wiadomości ogólne

- ▶ MongoDB to baza dokumentów
 - ▶ przechowuje pary: identyfikator dokumentu + dokument
 - ▶ dokument może być złożony
- ▶ <http://www.mongodb.com>
- ▶ Pierwsza wersja w 2007 r.
- ▶ Nazwa nawiązuje do słowa humongous - potężny
- ▶ Open source (GNU AGPL), C++, wiele platform
- ▶ Główny cel to wydajność i niezawodność - replikacja
 - ▶ mniejsza kontrola spójności
 - ▶ brak transakcji
- ▶ Najbardziej znane wdrożenia: eBay, LinkedIn, Foursquare

Model danych

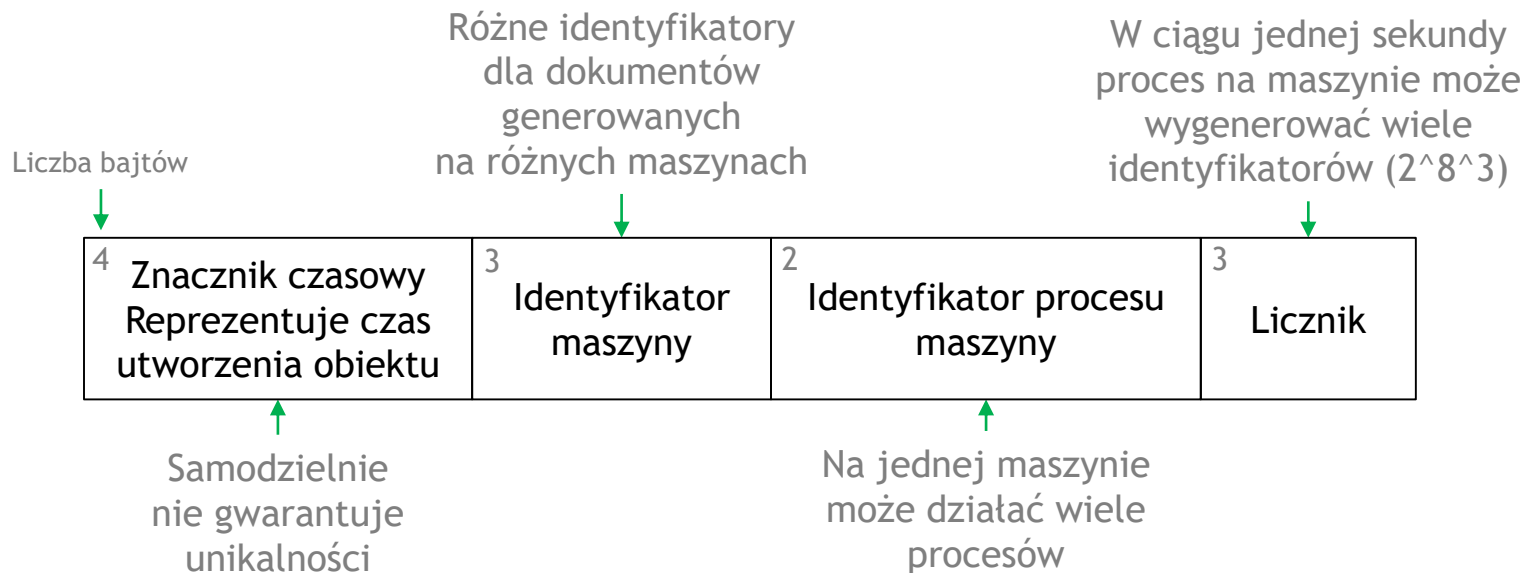
- ▶ Baza danych
 - ▶ składa się z kolekcji
 - ▶ jeden serwer MongoDB może obsługiwać wiele baz danych
- ▶ Kolekcja
 - ▶ przechowuje dokumenty
 - ▶ każdy dokument w ramach kolekcji może mieć inną strukturę
- ▶ Dokument
 - ▶ podstawowa jednostka danych
 - ▶ przechowywany w formacie JSON (fizycznie BSON - Binary JSON)
- ▶ Model zależy przede wszystkim od:
 - ▶ sposobu wykorzystywania danych
 - ▶ intensywności modyfikacji

Dokument

- ▶ Sekwencja par <klucz:wartość>
- ▶ Klucze
 - ▶ są ciągami znaków
 - ▶ nie mogą się powtarzać
- ▶ Każdy dokument ma identyfikator - klucz o nazwie "_id"
- ▶ Wartości
 - ▶ są określonego typu (np. string, date, number, boolean etc.)
 - ▶ mogą być kolejnymi parami <klucz:wartość>
- ▶ Wielkość liter ma znaczenie

Identyfikator dokumentu

- ▶ Automatycznie tworzony po stronie klienta
- ▶ Decentralizacja
- ▶ Wysoka efektywność
- ▶ Schemat identyfikatora dla unikania kolizji
- ▶ 12 bajtów -> 24 znakowa liczba heksadecymalna



Przykłady dokumentów

- prosty dokument

```
{
  _id: ObjectId(...),
  tytuł: "NoSQL w jeden dzień"
}
```

- dokument z tablicą

```
{
  _id: ObjectId(...),
  tytuł: "NoSQL w jeden dzień",
  opis: "Najlepsza książka na rynku",
  tematyka: ["nosql", "bazy danych"]
}
```

- dokument z tablicą zagnieżdżonych dokumentów

```
{
  _id: ObjectId(...),
  tytuł: "NoSQL w jeden dzień",
  opis: "Najlepsza książka na rynku",
  tematyka: ["nosql", "bazy danych"],
  wypożyczenia: [
    {
      osoba: "Artur",
      data: new Date(2017,12,21,8,00)
    },
    {
      osoba: "Julia",
      data: new Date(2017,12,18,12,00)
    }
  ]
}
```

Związki pomiędzy danymi reprezentowane przez zagnieżdżanie dokumentów

Referencje

```
{
  _id: <ObjectID1>
  nazwisko: "WEGLARZ",
  placza_pod: 1730
  id_zesp: <ObjectID3>
}
```

```
{
  _id: <ObjectID3>
  nazwa: "X",
  adres: "PIOTROWO 3A"
}
```

```
{
  _id: <ObjectID1>
  nazwisko: "WEGLARZ",
  placza_pod: 1730
}

{
  _id: <ObjectID2>
  nazwisko: "MORZY",
  placza_pod: 1340
}
```

```
{
  _id: <ObjectID3>
  nazwa: "X",
  adres: "PIOTROWO 3A",
  pracownicy: [<ObjectID1>, <ObjectID2>]
}
```

DO WERSJI 3.2 BRAK OPERACJI POŁĄCZENIA W BAZIE DANYCH
POŁĄCZENIE MOŻLIWE TYLKO NA POZIOMIE APLIKACJI

Związki pomiędzy danymi reprezentowane przez referencje

Dokumenty zagnieżdżone vs referencje

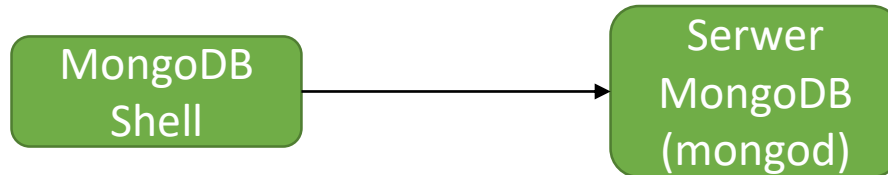
- ▶ Dokumenty zagnieżdżone
 - ▶ dobre dla związków 1:N
 - ▶ rozmiary dokumenty mogą z czasem intensywnie rosnać (ograniczenie to 16MB)
 - ▶ cały dokument jest blokowany do zapisu
 - ▶ w pamięci przechowywane są całe dokumenty
- ▶ Dokumenty z referencjami
 - ▶ podobieństwo do relacyjnych baz danych (normalizacja)
 - ▶ często trudniejsze w przeprowadzaniu operacji bazodanowych
 - ▶ zalecane w sytuacjach:
 - ▶ gdy zagnieżdżanie powoduje zbyt duże powielanie danych
 - ▶ gdy zagnieżdżanie nie przynosi przyrostu wydajności
 - ▶ dla reprezentacji związków M:N

ACID

- ▶ Atomicity
 - ▶ atomowe są operacje na poziomie pojedynczego dokumentu, nawet jeżeli operacja zapisu dotyczy wielu dokumentów
- ▶ Consistency
 - ▶ tylko dla pojedynczego serwera bazy danych i dokumentu
 - ▶ czytający nie zobaczy częściowo zmodyfikowanego dokumentu
 - ▶ przy replikach brak gwarancji spójności (dostępność >> spójność)
- ▶ Isolation
 - ▶ operacje zapisu są automatycznie "zatwierdzone"
 - ▶ widoczne natychmiast dla innych zapytań/połączeń
- ▶ Durability
 - ▶ brak gwarancji, po awarii ostatnie zapisy mogą być utracone.

Brak transakcji

MongoDB Shell



```
mongo --port <port> --host <host> -u <user> -p <password>
```

```
student@lab-ai-sbd:~$ mongo
MongoDB shell version v3.4.10
connecting to: mongodb://127.0.0.1:27017
MongoDB server version: 3.4.10
> show dbs
admin          0.000GB
ksiegarnia    0.000GB
local         0.000GB
> use ksiegarnia
switched to db ksiegarnia
> show collections
ksiazki
wydawnictwa
```

Język dostępu do danych

- ▶ Własny język zapytań (brak SQL)
- ▶ Oparty na notacji JavaScript i JSON
- ▶ Możliwość użycia kodu JavaScript

```
SELECT NAZWISKO,  
PLACA_POD, PLACA_DOD  
FROM PRACOWICY  
WHERE PLACA_POD +  
NVL(PLACA_DOD,0) > 1000
```

```
db.pracownicy.find(  
  {$where:  
    "this.placa_pod +  
      (this.placa_dod != null? this.placa_dod:0) > 1000"},  
  {"nazwisko":1, "placa_pod":1, "placa_dod":1}  
)
```

Framework agregacji

- ▶ Nie musi agregować danych
- ▶ Może mieć charakter przetwarzania
 - ▶ potokowego
 - ▶ MapReduce

Przetwarzanie potokowe



```
db.kolekcja.aggregate(  
[ {$match:...},  
  {$project:...},  
  {$group: ...}  
  {$match:...},  
  {$skip : ...},  
  {$limit : ..}  
])
```

Podsumowanie

- ▶ Dokumentowa baza danych
- ▶ Dokumenty powiązane poprzez
 - ▶ zagnieżdżenia
 - ▶ referencje
- ▶ Brak transakcji
- ▶ Rozwiązania w celu uzyskania wysokiej
 - ▶ wydajności
 - ▶ dostępności
- ▶ Dedykowany, oparty na notacji JavaScript, język dostępu do danych
- ▶ Przetwarzanie potokowe