

NoSQL & relax with CouchDB

Adam Dzedzic

PyWaw #23

8 kwiecień 2013

Agenda

- 1 NoSQL - nierelacyjne systemy baz danych
 - Wprowadzenie do NoSQL
 - Rodzaje i porównanie baz NoSQL
 - Polyglot persistence
- 2 Relax with CouchDB
 - CouchDB
 - python-couchdb
 - Projekt w CERN wykorzystujący CouchDB
- 3 Podsumowanie
 - Warsaw NoSQL User Group
 - NoSQL - podsumowanie
 - Specjalna część

NoSQL

nierelacyjne systemy baz danych

NoSQL

NoSQL = **Not only SQL** (nie tylko SQL)



Systemy zarządzania bazami danych, które nie opierają się na tradycyjnym modelu relacyjnym.

Cechy:

- rezygnacja z właściwości ACID i innych cech relacyjnych (np. złączeń)
- łatwe skalowanie horyzontalne
- brak pojedynczego punktu awarii
- elastyczne schematy danych (agregaty)

Dlaczego NoSQL?

Big data - wielkie zbiory danych



amazon
Dynamo



Cassandra

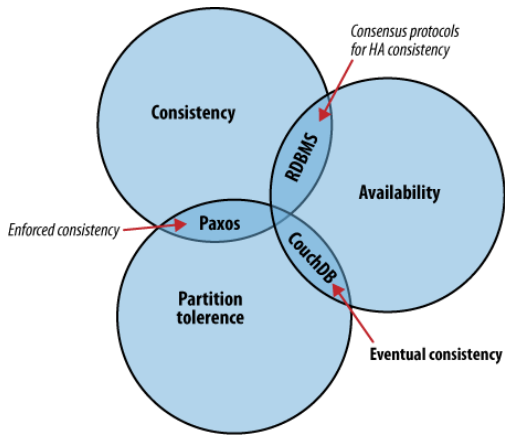
Google
Bigtable



Linked in
Voldemort

Teoria CAP

Consistency (Spójność), Availability (Dostępność), Partition tolerance (Rozproszenie)



Rodzaje baz danych NoSQL (taksonomia)

podstawowe trzy rodzaje

1. Klucz-wartość



2. Bazy dokumentowe



3. Bazy kolumnowe



Inne? (grafowe, obiektowe, hierarchiczne)



Polyglot persistence



Polyglot
Persistence

Why now?

<http://bit.ly/ZHDQ1B>

Polyglot persistence



CouchDB - główne komponenty

RELAX, Ubuntu One, BBC, CERN



Apache CouchDB has started. Time to relax.



CouchDB - główne komponenty

budowa wewnętrzna

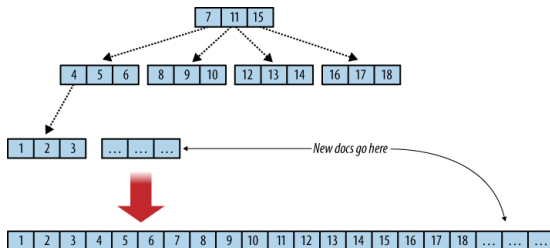


CouchDB - główne komponenty od strony użytkownika



CouchDB - wewnętrzna struktura

drzewo B+, plik append-only



- zapisy nie blokują odczytów
- dowolna liczba odczytów
- trwałe zapis danych
- duża zajętość dysku - baza wymaga kompresji

Konsekwencje pliku append-only

Ciekawostka

- 4 dokumenty w bazie (rozmiar w bajtach - czerwony kolor)
-rw-r--r-- 1 adam adam **16479** 2012-06-04 18:25
adam.couch
- po usunięciu dokumentów: 0 dokumentów
-rw-r--r-- 1 adam adam **36962** 2012-06-04 18:27
adam.couch
- po kompresji (compaction)
-rw-r--r-- 1 adam adam **4188** 2012-06-04 18:29
adam.couch

Futon

administracja, replikacja, kompresja

Overview > pdub >

Save Document Add Field Upload Attachment...
 Delete Document...

Fields Source

Field	Value
<input type="checkbox"/> _id	"108cadb746ad4944e50379159d0a41b2"
<input type="checkbox"/> _rev	"4-681092b03946d2e55d338316891e519e"
<input checked="" type="checkbox"/> added	"2012-04-19T08:24:48Z"
<input checked="" type="checkbox"/> clients	1 "host1" 2 "host2"
<input checked="" type="checkbox"/> doc_type	"pdu"
<input checked="" type="checkbox"/> hostname	"pdu1"
<input checked="" type="checkbox"/> location	"513"
<input checked="" type="checkbox"/> rack_id	"0e1074eedfbb45d41d1006abb94e0c2c"
<input checked="" type="checkbox"/> rackname	"rack01"
<input checked="" type="checkbox"/> slotCount	2

Showing revision 4 of 5 [← Previous Version](#) | [Next Version →](#)

Design documents

MapReduce: liczba slotów PDU-s

```
function(doc) {
  if (doc.doc_type == 'pdu' && doc.slotCount)
  {
    emit(doc._id, doc.slotCount);
  }
}
function(keys, values, rereduce)
{
  return sum(values);
}
```

- incremental views: funkcje MapReduce (~ perspektywy zmaterializowane)
- validation functions: sprawdź wymagane pola
- shows functions: konwertuj dokument do wybranego formatu (np. HTML)

Jak Python rozmawia z CouchDB?

moduły biblioteki python-couchdb

`couchdb.client`

administracja: tworzenie bazy, usuwanie, replikacja,
compaction (usuwanie poprzednich wersji dokumentów)

`couchdb.mapping`

odwzorowanie dokumentów w formacie JSON na obiekty w
języku Python (uwaga na pole `_rev` - artefakt)

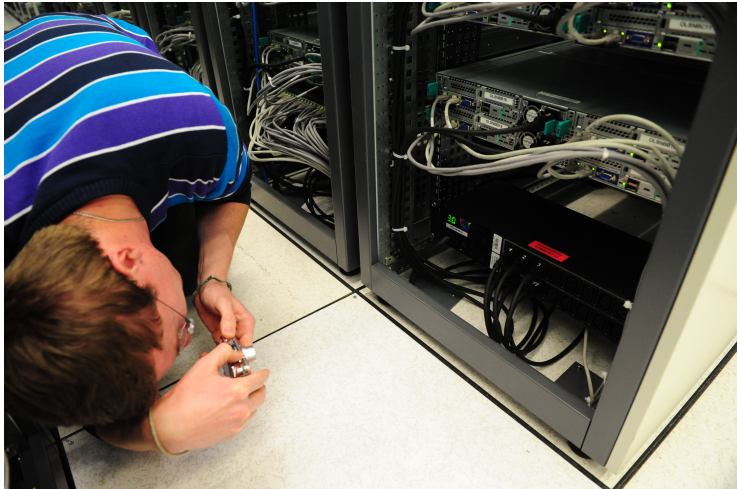
`couchdb.design`

zarządzanie design document (funkcje walidacji, formatowanie)

`couchdb.views`

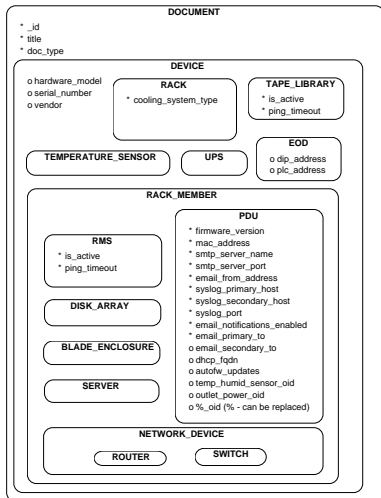
tworzenie zapytań do bazy w formie funkcji MapReduce

Konfiguracja urządzeń: PDU (Power Distribution Unit)

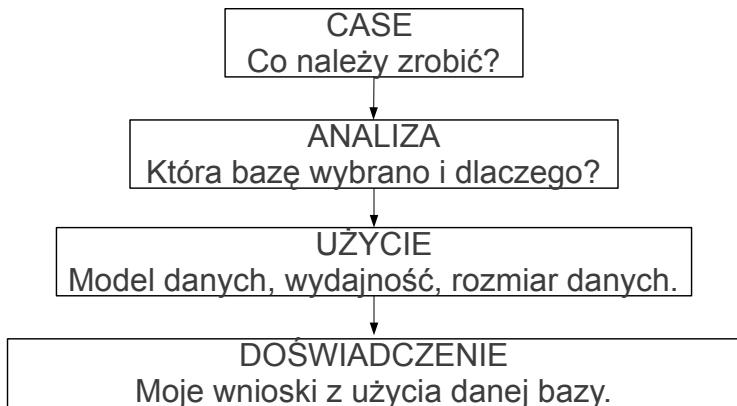


Schemat danych dla CouchDB

konfiguracja urządzenia <-> dokument w bazie CouchDB

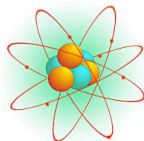


Warsaw NoSQL User Group



NoSQL - podsumowanie

- bazy danych NoSQL dla wielkich zbiorów danych i specjalnych form danych (graf, częsta replikacja)
- teoria CAP i rozproszone bazy danych
- rodzaje baz NoSQL
 - klucz-wartość
 - dokumentowe
 - kolumnowe
- wykorzystanie bazy dokumentowej **CouchDB**
- bazy relacyjne wciąż użyteczne w ok. **90%** przypadkach
- **polyglot persistence** - właściwa baza do danego problemu



Dziękuję za uwagę