

# Praca Dyplomowa Magisterska

## Mechanizm modelowania danych i mapowania obiektowego dla Apache Cassandra

Jakub Turek  
J.Turek@stud.elka.pw.edu.pl

9 października 2014r.

# Cel pracy

System mapowania obiektowego dla bazy Apache Cassandra:

- ▶ Zachowanie różnicy w wydajności pomiędzy relacyjnymi bazami danych a Cassandra.
- ▶ Możliwość stosowania wzorców modelowania do optymalizacji.
- ▶ Zachowanie zgodności z istniejącymi mechanizmami mapowania obiektowo-relacyjnego.

# Kundera

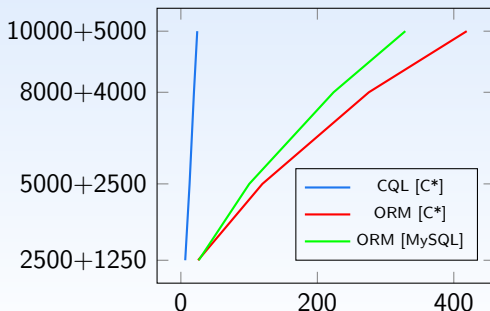
## Kundera

Implementacja Java Persistence API dla baz danych NoSQL.

Wspierane silniki:

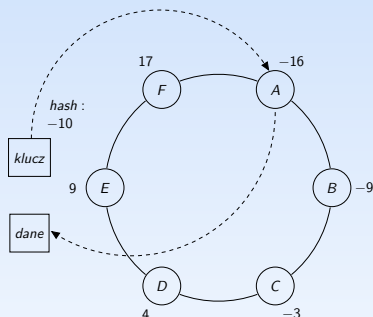
- ▶ Cassandra
- ▶ HBase
- ▶ MongoDB
- ▶ Redis
- ▶ Oracle NoSQL
- ▶ Neo4j
- ▶ Couchdb
- ▶ Elastic Search

Porównanie czasu wstawiania rekordów:



# Analiza problemu

Modelowanie relacji wiele-do-wielu przez bibliotekę Kundera



**Użytkownik    Wiek**

jkowski	24
mnowak	45

**Przedmiot    Cena**

laptop	2276.99
audiobook	42.40

Klucz	Użytkownik	Przedmiot 1	Przedmiot 2
1	jkowski	laptop	audiobook
2	mnowak	audiobook	(null)

# Modelowanie zależności (1/2)

Zależność znormalizowana zwrotna

Użytkownik	Wiek
jkowalski	24
mnowak	45

Przedmiot	Cena
laptop	2276.99
audiobook	42.40

Użytkownik	Przedmiot 1	Przedmiot 2
jkowalski	laptop	audiobook
mnowak	audiobook	(null)

Przedmiot	Użytkownik 1	Użytkownik 2
audiobook	jkowalski	mnowak
laptop	jkowalski	(null)

# Modelowanie zależności (2/2)

Zależność zdenormalizowana bezzwrotna

Użytkownik	Wiek	P. 1	C. 1	P. 2	C. 2
jkowalski	24	audiobook	42.40	laptop	2276.99
mnowak	45	audiobook	42.40	(null)	(null)

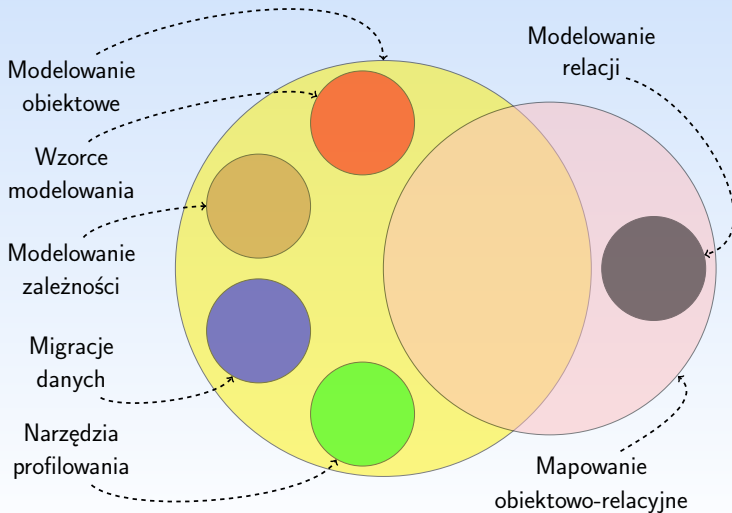
## Zalety:

- ▶ Bardzo wysoka wydajność:
  - ▶ Pełna informacja w jednym odwołaniu.
  - ▶ Cassandra została zaprojektowana do wielokrotnych wstawień.

## Wady:

- ▶ Problem z aktualizacją danych. Dwie alternatywy:
  - ▶ Czasochłonna aktualizacja wymagająca dodatkowych indeksów.
  - ▶ Niestójność danych.

# Modelowanie obiektowe



# Wzorce modelowania



# Koniec

**Dziękuję za uwagę!**