

# **Bazy danych NoSQL**

# NoSQL

**NoSQL** – jest klasą systemów zarządzania bazą danych nie pasujących do powszechnie stosowanego modelu relacyjnego baz danych:

- - brak języka SQL
- - nierelacyjny model danych
- - nie musi zapewniać ACID
- - rozproszona, odporna na awarie architektura

NoSQL – to klasa systemów bazodanowych zoptymalizowanych pod kątem: dopisywania i odczytu danych. Nie udostępniają mechanizmów relacyjnych, są przeznaczone dla trzymancia dużej ilości danych o przeważnie prostej strukturze (np. klucz->wartość). Nie udostępniają mechanizmu JOIN. Nie gwarantują także właściwości ACID. Dzięki temu wszystkiemu, czas dostępu do danych jest tu znacznie mniejszy niż w przypadku tradycyjnych relacyjnych baz danych.

# Modele danych stosowane w rozwiązaniach NoSQL

- Bazy klucz-wartość
- Bazy kolumnowe
- Bazy dokumentowe
- Bazy grafowe
- Bazy XML
- Bazy obiektowe ???

# Bazy klucz-wartość

- Bazy typu klucz-wartość mają bardzo prostą budowę. W dużym uproszczeniu jest to jednak tabela, która zawiera dwie kolumny tekstowe: klucz oraz wartość. W bardziej rozbudowanych modelach wartością może być liczba, łańcuch znaków, tablica czy lista.
- Model może wydawać się aż zbyt prosty, ale ma jedną niezwykle ważną cechę – jest niezwykle szybki (zarówno jeśli chodzi o zapis jak i o odczyt danych).

# Bazy klucz wartość

- **Redis** – napisany w języku ANSI C, najbardziej odpowiednim opisem jest określenie, że jest to serwer danych strukturalnych.
- Wspiera następujące struktury danych:
  - łańcuchy znaków
  - zbiory
  - posortowane zbiory
  - listy
  - tablice haszujące

# Bazy kolumnowe

- W bazach kolumnowych nastąpiła zmiana sposobu postrzegania danych, tzn. zamiast tradycyjnego zapamiętywania informacji w wierszach, informacje są zapamiętywane w kolumnach.
- W kolumnach przechowywane są dane tego samego typu, więc możliwe jest stosowanie bardziej wydajnych algorytmów kompresji danych.

# Bazy kolumnowe

- **Apache Cassandra** – rozproszona baza danych napisana w języku Java.
- Charakteryzuje się wysoką wydajnością, liniową skalowalnością, decentralizacją, odpornością na awarie.
- Ograniczenia: Brak joinów i podzapytań, klucze wierszy nie mogą przekraczać 64 KiB.



# Bazy dokumentowe

- W bazach tego typu zamiast wierszy występuje pojęcie dokumentu, który zawiera pary klucz-wartość.
- Podejście takie jest niezwykle elastyczne i umożliwia bardzo wierne odtwarzanie rzeczywistych danych w systemach informatycznych

# Bazy dokumentowe

- **Mongo** -to otwarty, nierelacyjny system zarządzania bazą danych napisany w języku C++. Charakteryzuje się dużą skalowalnością, wydajnością oraz brakiem ściśle zdefiniowanej struktury obsługiwanych baz danych. Zamiast tego, dane składowane są jako dokumenty w stylu JSON, co umożliwia aplikacjom bardziej naturalne ich przetwarzanie, przy zachowaniu możliwości tworzenia hierarchii oraz indeksowania.

# Bazy grafowe

- Reprezentacja grafowa opiera się na dwóch pojęciach: encji (inaczej obiektu, wężła) oraz relacji (powiązania, krawędzi).
- Encja reprezentuje pojedynczy byt.
- Relacja jest pewną własnością występującą (lub nie) pomiędzy encjami.
- Encje oraz relacje mogą być opisane przez atrybuty (własności).

# Bazy grafowe

- **Neo4j** – jest grafową bazą danych napisaną w Java.
- Zalety:
  - Możliwe wywoływanie zapytań rekurencyjnych: wyszukiwanie ścieżki, pokazywanie ścieżki, czyli to co niemożliwe lub bardzo trudne w relacyjnej bazie danych
  - Bardzo wyrafinowane języki zapytań: Gremlin, a w szczególności Cypher
  - Dzięki pakietowi „bulbs” możliwość synchronizacji z relacyjną bazą danych

# Bazy XML

- Używają dokumentów XML jako podstawowe jednostki przechowujące.
- Dodatkowo wprowadzono mechanizmy umożliwiające wyszukiwanie czyli XQuery oraz XPath.

# Bazy XML

- **BaseX** - jest natywnym i kompaktowym systemem zarządzania bazą danych XML. Stosowany jest przede wszystkim do zapisu, wyszukiwania i wizualizacji dużych dokumentów i kolekcji XML.
- W przeciwieństwie do innych baz danych bazy danych XML opierają się na standardach jak XPath i XQuery.

# Bazy obiektowe

- Elementarnymi jednostkami modelowania są wartości i obiekty. Obiekty wyróżniają się posiadaniem unikalnych identyfikatorów.
- Obiekty i wartości są wystąpieniami typów. Wszystkie wystąpienia danego typu współdzielą zbiór dopuszczalnych stanów oraz zachowanie.
- Stan obiektu jest reprezentowany przez zbiór wartości jego cech strukturalnych. Cechy mogą być atrybutami obiektu lub związkami z innymi obiektami.

# Bazy obiektowe – cd.

- Zachowanie obiektu jest opisane przez zbiór operacji, które mogą być wykonywane na obiekcie. Z operacjami są związane listy parametrów wejściowych i wyjściowych, każda określonego typu. Każda operacja może również zwrócić wynik określonego typu.
- Baza danych składa się z obiektów umożliwiając ich współdzielenie przez wielu użytkowników. Obiekty składowane w bazie danych są wystąpieniami typów zdefiniowanych za pomocą języka ODL w schemacie bazy danych.



# Bazy obiektowe

- **Db4o** (database for objects) to obiektowa baza danych na platformę Java i .NET.
- Podwójna licencja – GPL oraz komercyjna.