



2

Bazy danych

- MySQL, Oracle, SQL Server...
- Przechowują zorganizowane dane wraz z dużą ilością metadanych
 - Skończony zestaw predefiniowanych rekordów
 - Wiele informacji księgowych (statystyki tabeli, indeksy na jedną lub więcej kolumn, ograniczenia integralności danych ...)
 - Naprawdę fajne struktury danych (np. B-Drzewa)
- Za: SZYBKIE zapytania wybranych typów
 - Metadane mogą być dostosowane do pewnych zapytań lepiej
- Przeciw: Metadane - koszty tworzenia, utrzymania (kontrola schematu informacji).

3

Przykładowa tabela



```
mysql> use corp;  
Database changed
```

```
mysql> describe employees;
```

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
firstname	varchar(32)	YES		NULL	
lastname	varchar(32)	YES		NULL	
jobtitle	varchar(64)	YES		NULL	
start_date	date	YES		NULL	
dept_id	int(11)	YES		NULL	

4

Bazy danych są dobre w ...



- Korzystaniu z indeksu do wyszukiwania konkretnego wiersza
- Grupowaniu wierszy z uwzględnieniem wspólnego klucza
 - funkcje agregujące (SUM, AVG, STDDEV...)
- Egzekwowanie jakości danych
 - np.: brak duplikatów; zgodność typów; inne ograniczenia logiki biznesowej

5

Problemy ...



- Twardy dysk zapisuje z prędkością 50—60 MB/sec.
 - ... tylko przy liniowym zapisie
 - Utrzymanie zindeksowanych danych może powodować spadek 10x lub więcej
 - Zapis buforowany / opóźniony może pomóc
- Wykonywanie równoległych operacji bazodanowych jest skomplikowane i skalowalność jest trudna do poprawnego wdrożenia
- Bazy danych mogą przechowywać max 10 TB; zapytania mogą wyszukiwać około 10%

6

Więcej danych, lepsze przetwarzanie



- Jak można gromadzić 1000x więcej danych?
- W jaki sposób przetwarzać dane, gdy nie wiemy z góry schematu?
- W jaki sposób wykonywać bardziej skomplikowane przetwarzanie?
 - NLP przetwarzanie języka naturalnego, uczenie maszynowe, przetwarzanie obrazów, eksploracja zasobów web
- Jak to zrobić z prędkością przesyłu danych rzędu TB/godz?

7

Co potrzebujemy



- Skutecznego sposobu podzielenia zadania na równoległe części
- Sposobu odczytu i zapisu danych równoległe
- Sposobu do zminimalizowania zużycie pasma
- Niezawodnego sposobu na przeprowadzenie obliczeń

8

Wymagania dotyczące niezawodności



- Obsługa częściowych awarii
 - Cały system musi obsługiwać duży spadek wydajności aplikacji, a nie pełnego zatrzymania

9

Wymagania dotyczące niezawodności



- Odzyskiwanie danych
 - Jeśli elementy zawiodą, ich obciążenie musi być odebrane przez wciąż funkcjonujące jednostki

10

Wymagania dotyczące niezawodności



- Indywidualne możliwości odzyskiwania
 - Węzły, które zawiodą i się zrestartują muszą być w stanie wrócić do grupy bez pełnej konieczności restartu grupy

11

Wymagania dotyczące niezawodności



- Konsystencja
 - Jednoczesne operacje lub częściowe wewnętrzne niepowodzenia nie powinny powodować widocznych zewnętrznych niedeterminizmów.

12

Wymagania dotyczące niezawodności



- Skalowalność
 - Dodawanie zwiększonego obciążenia do systemu nie powinno spowodować natychmiastowej awarii, ale znaczący spadek wydajności
 - Zwiększenie zasobów powinno skutkować proporcjonalnym wzrostem wydajności

13

Lokalność



- Zarządca dzieli zadania na podstawie lokalizacji danych: stara się mieć przydzielić zadania na tym samym komputerze co fizycznych dane, lub co najmniej tej samej szafie
- Przydzielone zadania podzielone są na 64-128 MB bloki: taki sam jak rozmiar klastrów systemu plików
 - Przetwarzanie elementów pojedynczego pliku równolegle

14

Tolerancja błędów



- Zadania przeznaczone do niezależnego procesowania
- Zarządca wykrywa awarie wątków pracowniczych
- Zarządca ponownie wykonuje zadania, które ulegną uszkodzeniu gdy są w toku
- Zrestartowanie jednego zadania nie wymaga komunikacji z innymi
- Dane są replikowane do zwiększenia dostępności, trwałości

15