



Mobile Application

PERTEMUAN 5

Firestore = NoSQL

- ▶ Dalam menyimpan datanya, Firestore menggunakan prinsip NoSQL
- ▶ SQL = Terstruktur (Menggunakan Relational Table)
- ▶ NoSQL = Not Only SQL = Menggunakan berbagai cara selain tabel untuk menyimpan data

SQL vs NoSQL

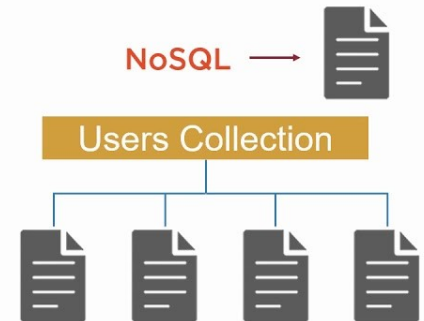
SQL

Table: Users

	id	name	createdAt	updatedAt
Filter	Filter	Filter	Filter	Filter
1	1	Ashton	2018-08-12 1...	2018-08-12 1...
2	2	Jakeem	2018-08-12 1...	2018-08-12 1...
3	3	Uma	2018-08-12 1...	2018-08-12 1...

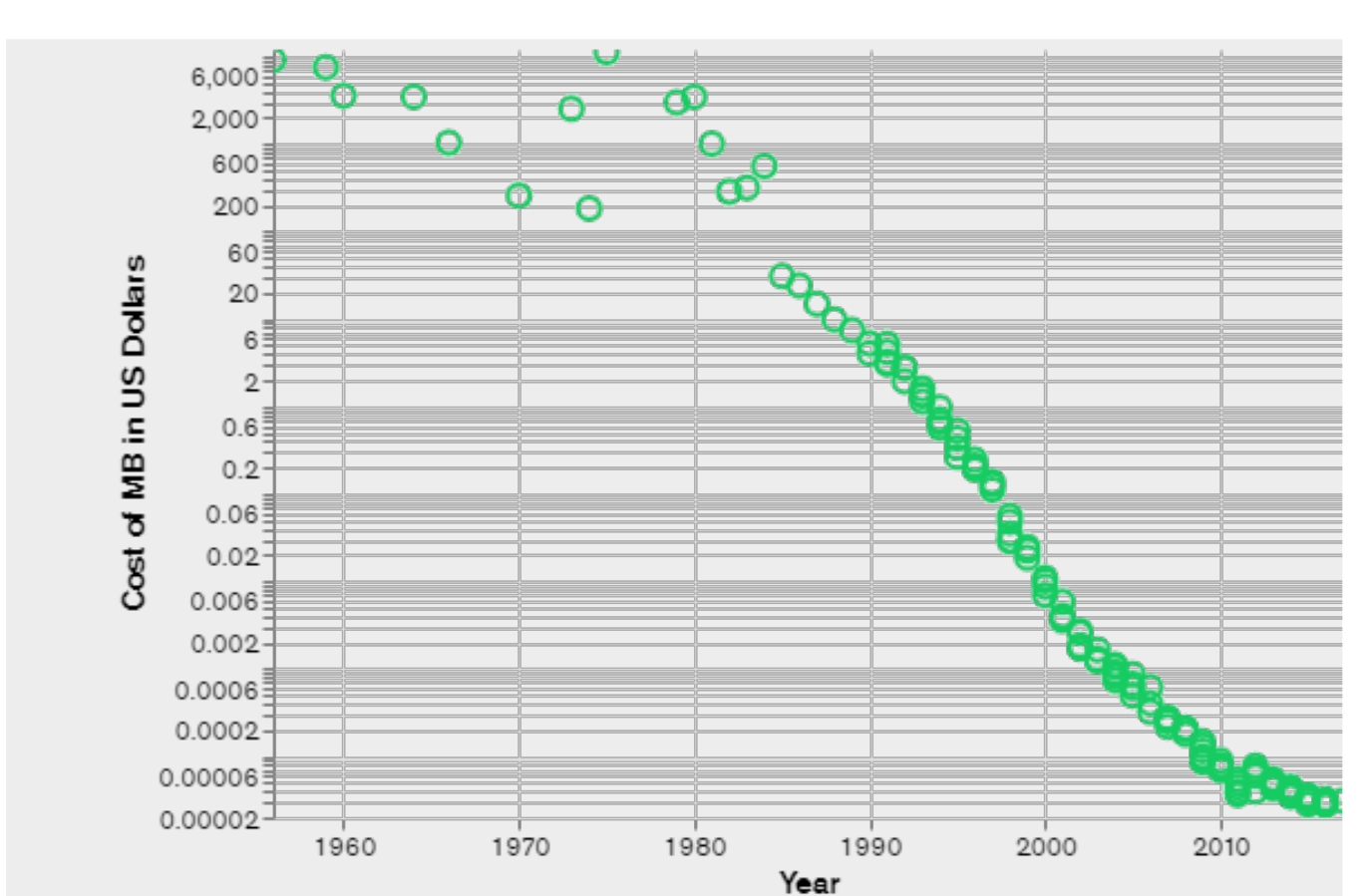
Relationships with JOINS

NoSQL



Mengapa NoSQL Muncul?

- ▶ Menurunnya biaya penyimpanan dari tahun ke tahun

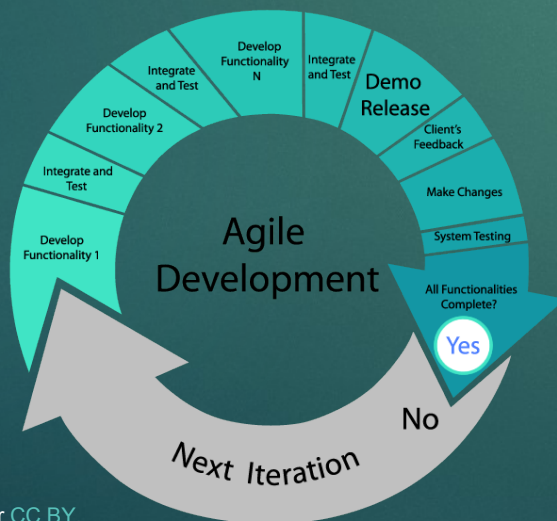


Mengapa NoSQL Muncul?

- ▶ Karena biaya penyimpanan menurun dengan cepat, jumlah data yang dibutuhkan aplikasi untuk disimpan dan kueri meningkat.
- ▶ Data datang dalam berbagai bentuk dan ukuran — terstruktur, semi-terstruktur, dan polimorfik.
- ▶ Sehingga untuk mendefinisikan skema menjadi hampir mustahil.
- ▶ Basis data NoSQL memungkinkan pengembang untuk menyimpan sejumlah besar data tidak terstruktur, memberi mereka banyak **fleksibilitas**.

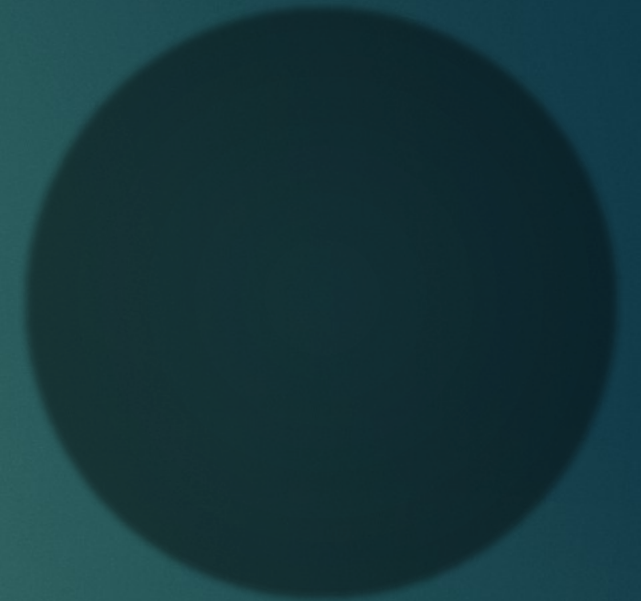
Mengapa NoSQL Muncul?

- ▶ Karena munculnya model SDLC (Agile) dengan **Agile Manifesto** nya. Maka spesifikasi pembuatan software sangat mudah berubah
- ▶ Untuk mengadaptasi perubahan inilah NoSQL digunakan
- ▶ Fitur Cloud Computing yang semakin populer, dan para developer ingin menyebar data mereka ke server-server berbeda dunia



Fitur NoSQL

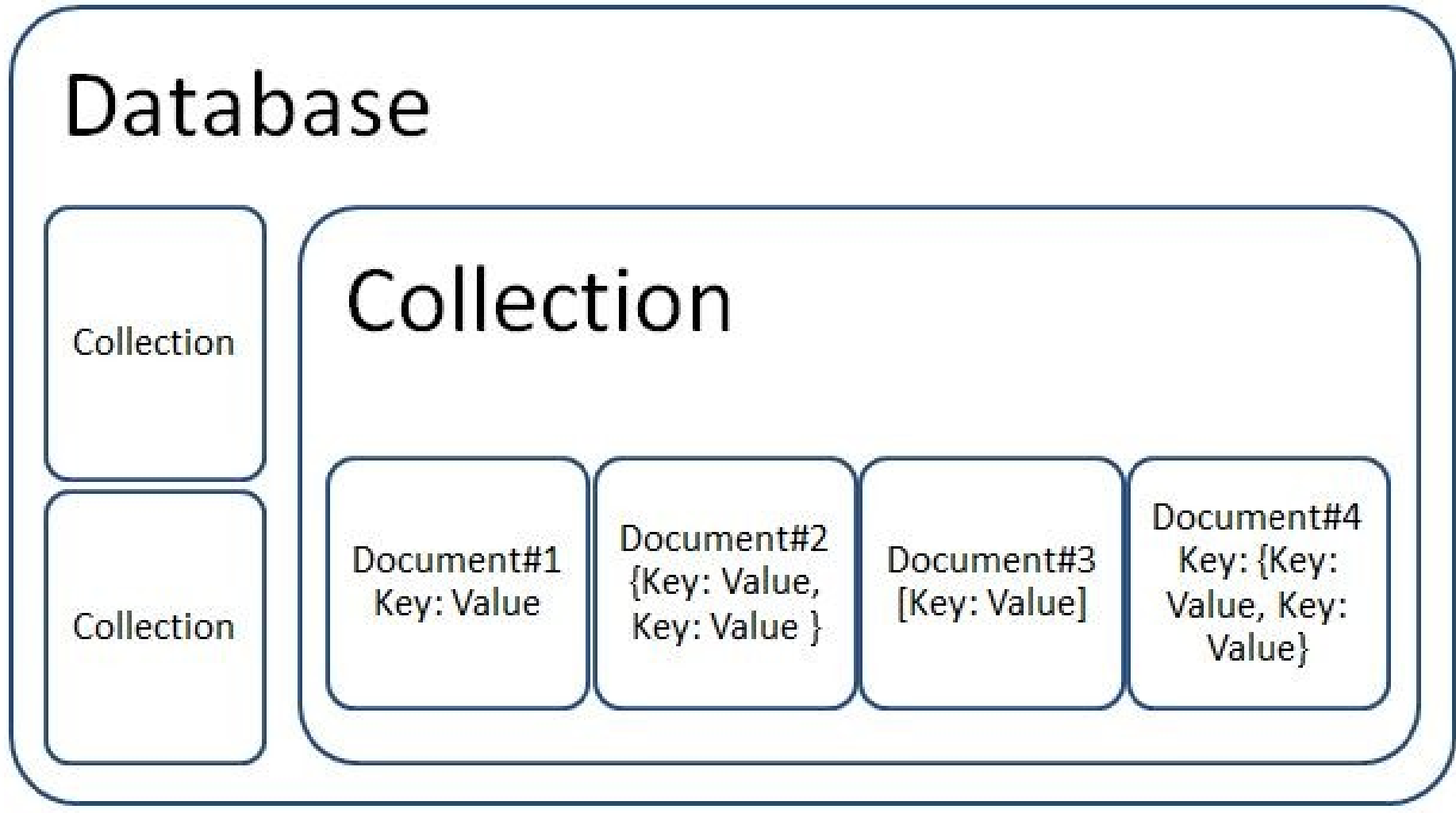
- ▶ Skema database yang fleksibel
- ▶ Penskalaan yang melebar (banyak server)
- ▶ Kueri yang cepat karena model data
- ▶ Penggunaan yang mudah untuk pengembang



Jenis Database NoSQL

- ▶ Document Database
 - ▶ Menyimpan data menyerupai JSON. Masing-masing dokumen memiliki pasangan **field** dan **value**. Firebase menggunakan jenis ini
- ▶ Key-value Database
 - ▶ Menyimpan data secara lebih sederhana di mana tiap item memiliki **keys** dan **values**
- ▶ Wide-column Database (Mirip SQL standar)
 - ▶ Menyimpan data dalam bentuk tabel, baris, dan kolom dinamis
- ▶ Graph Database
 - ▶ Menyimpan data dalam bentuk titik (**nodes**) dan sudut (**edges**). Biasanya digunakan untuk menyimpan informasi kontak, tempat, dan barang

Document Database



1 Database berisikan berbagai macam koleksi

1 Koleksi bisa berisikan beberapa dokumen

1 Dokumen bisa berisikan berbagai macam data

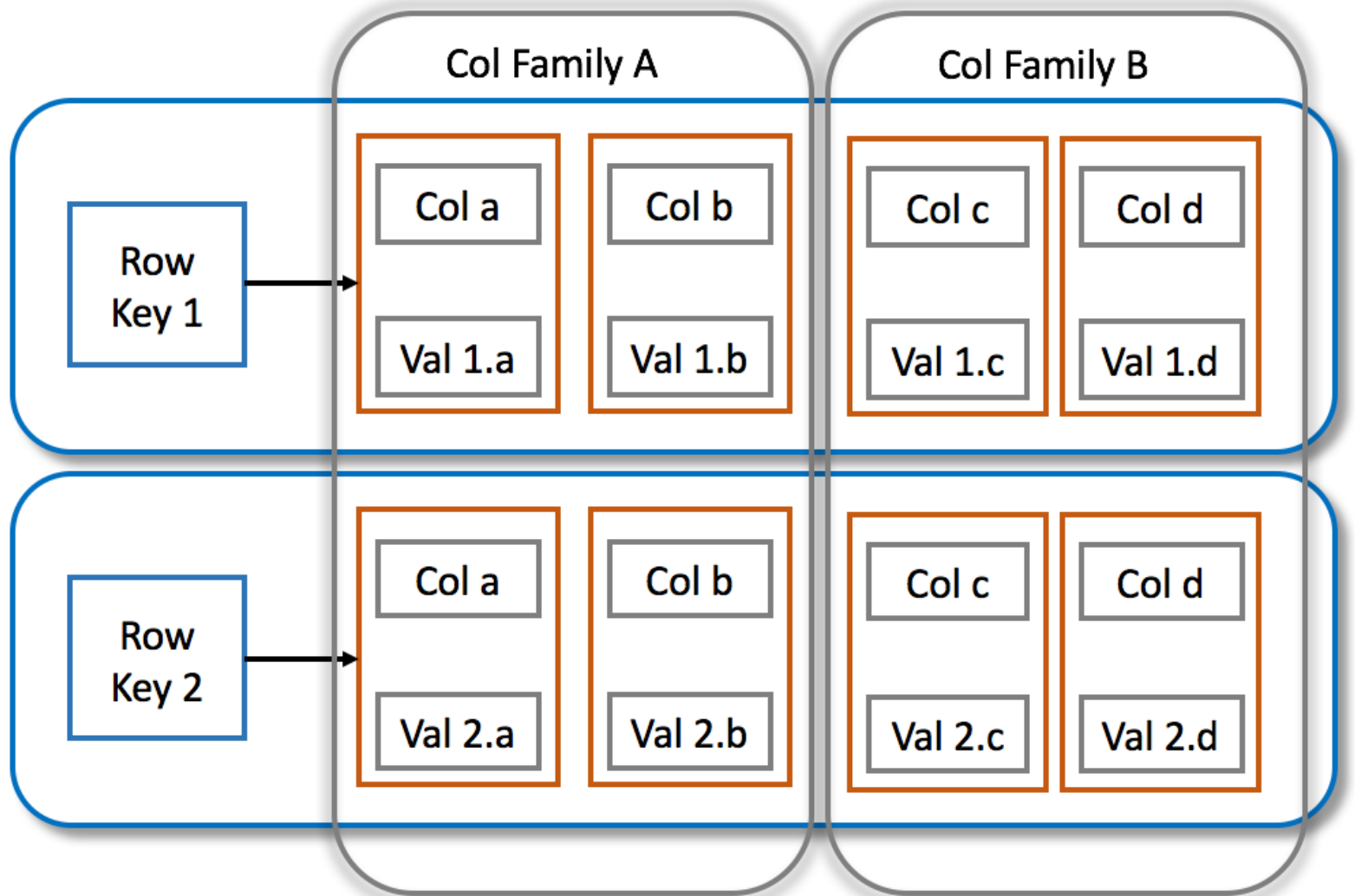
Key-Value Database

Key	Value
K1	AAA,BBB,CCC
K2	AAA,BBB
K3	AAA,DDD
K4	AAA,2,01/01/2015
K5	3,ZZZ,5623

Jenis ini sangat sederhana dibandingkan yang lainnya

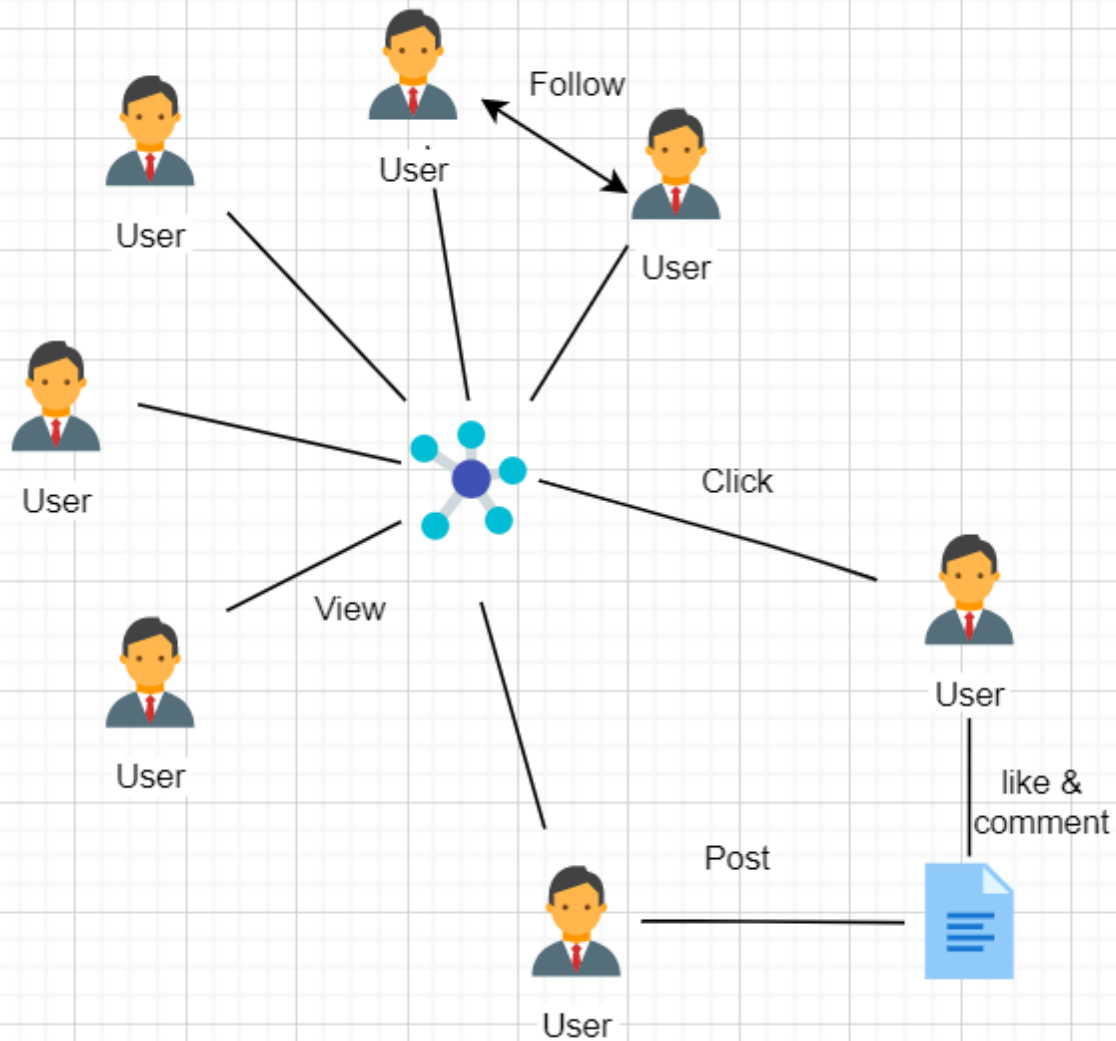
Namun tidak bisa mengatur data kompleks

Wide-column Database



Memiliki kemiripan dengan RDBMS/SQL biasa, namun lebih fleksible

Graph Database



Berbentuk seperti jaring laba-laba. Sehingga sangat cocok digunakan untuk kontak, peta, dll

Perbedaan RDBMS - NoSQL

► Data Model

- Dengan RDBMS, developer harus membuat 2 tabel untuk dapat

Users

ID	first_name	last_name	cell	city
1	Leslie	Yepp	8125552344	Pawnee

Hobbies

ID	user_id	hobby
10	1	scrapbooking
11	1	eating waffles
12	1	working

```
{
  "_id": 1,
  "first_name": "Leslie",
  "last_name": "Yepp",
  "cell": "8125552344",
  "city": "Pawnee",
  "hobbies": ["scrapbooking", "eating waffles", "working"]
}
```

Kapan NoSQL dibutuhkan?

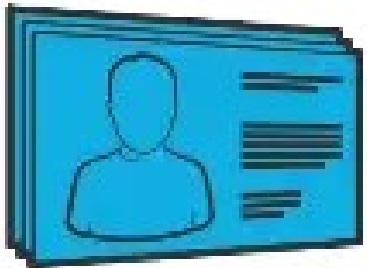
- ▶ Saat memutuskan database mana yang akan digunakan, pengambil keputusan biasanya menemukan satu atau lebih dari faktor berikut yang mengarahkan mereka untuk memilih database NoSQL:
- ▶ Pengembangan Agile yang serba cepat
- ▶ Penyimpanan data terstruktur dan semi terstruktur
- ▶ Volume data yang besar
- ▶ Persyaratan untuk arsitektur scale-out
- ▶ Paradigma aplikasi modern seperti layanan mikro dan streaming realtime

Siapa yang menggunakan NoSQL?



Motif NoSQL

Profile
Management



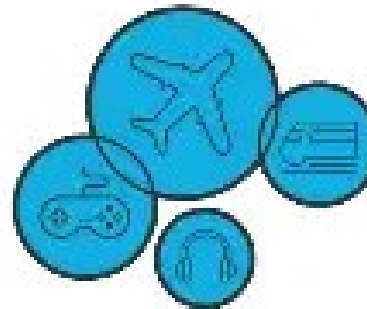
Caching



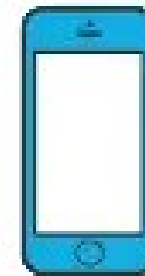
360 Degree
Customer View



Internet of
Things



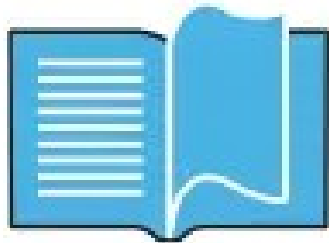
Mobile
Applications



Content
Management



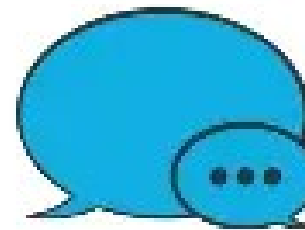
Catalog



Real Time
Big Data



Digital
Communication



Fraud
Detection

