Journey from Relational to NoSQL

An approach to DocumentDB

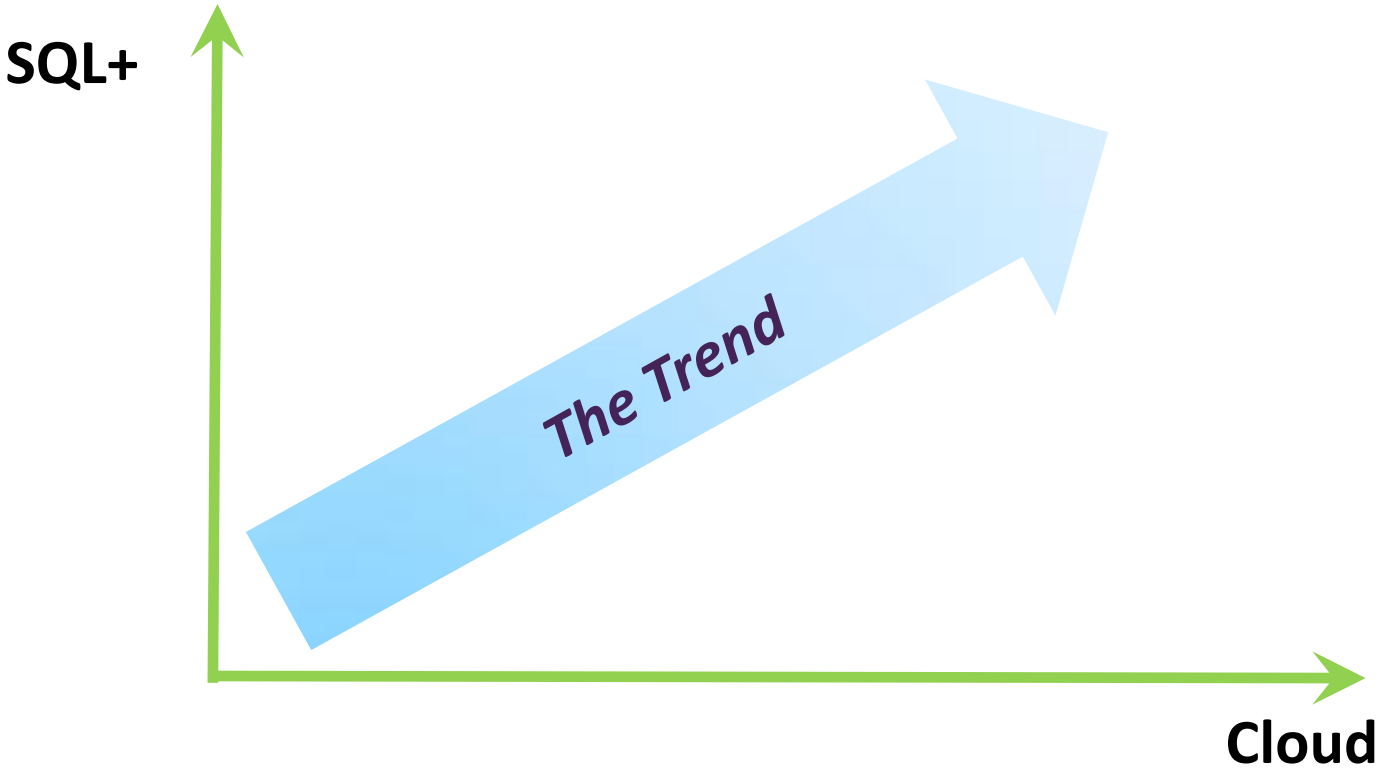
Wriju Ghosh & Rahul Gangwar

Sr. Tech Consultant, Microsoft

Why NoSQL?

- NoSQL technologies are a fundamental to part of cloud platform today
- IoT brings diverse kind of data
- NoSQL does take away the guideline through

Data trending today



NoSQL Pros and Cons

To scale for lots of users and lots of data

Pros: NoSQL technologies can offer more scalability than relational databases

Cons: Often lose some benefits of relational databases, e.g., database-wide transactions

To work better with different data formats, e.g., JSON

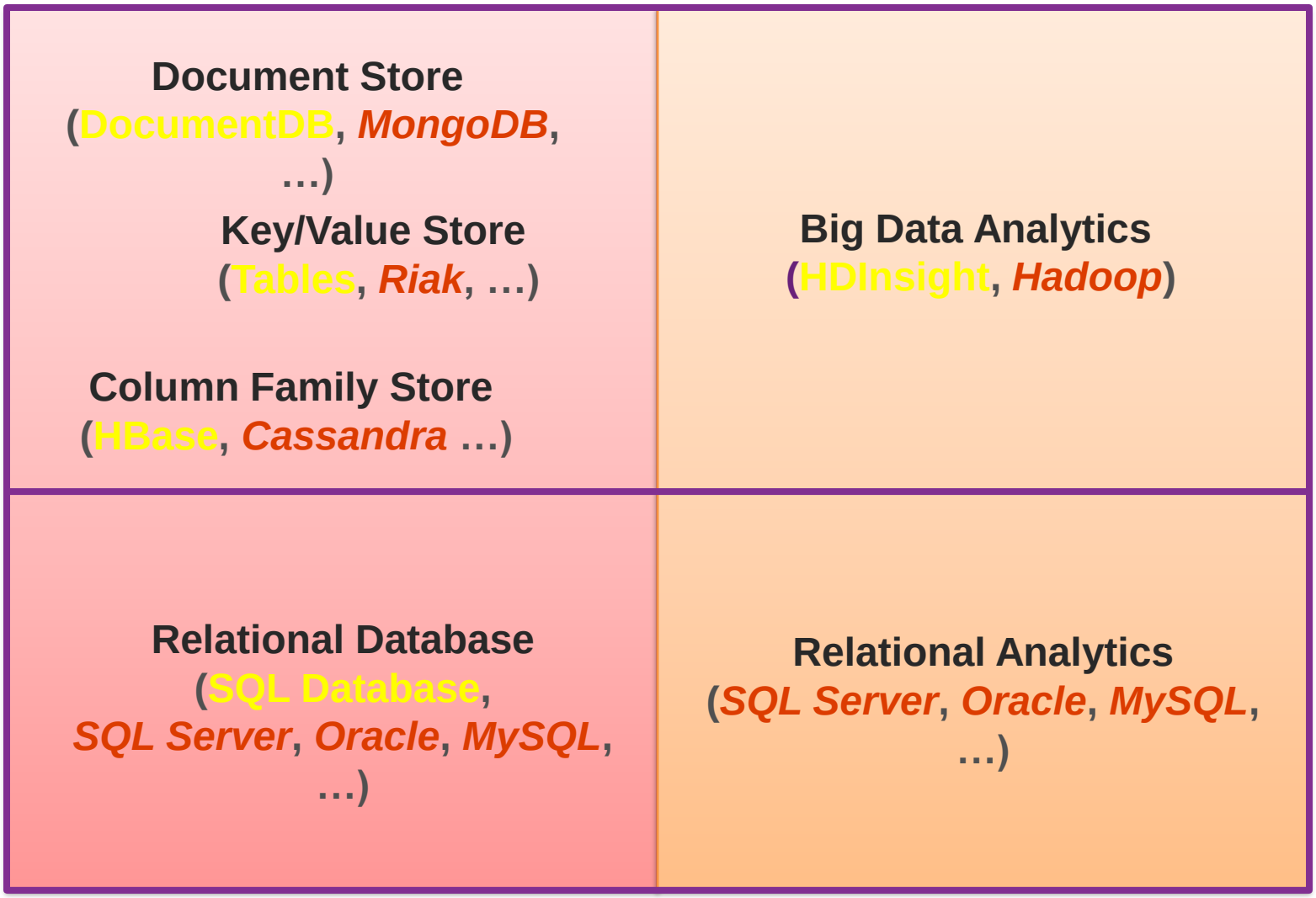
Pros: Avoiding object/relational mapping makes code easier to write

Cons: Limited BI tools; persistent data designed for a single application is harder to share

To work with data in a more flexible way

Pros: NoSQL technologies don't have fixed schemas

Cons: Fixed schemas help prevent errors


Document Store
(DocumentDB, MongoDB, …)
Key/Value Store
(Tables, Riak, …)
Column Family Store
(HBase, Cassandra …)
Big Data Analytics
(HDInsight, Hadoop)
Relational Database
(SQL Database, SQL Server, Oracle, MySQL, …)
Relational Analytics
(SQL Server, Oracle, MySQL, …)

Why NoSQL?

To scale for lots of users and lots of data	To work better with different data formats, e.g., JSON	To work with data in a more flexible way
Pros: NoSQL technologies can offer more scalability than relational databases	**Pros:** Avoiding object/relational mapping makes code easier to write	**Pros:** NoSQL technologies don't have fixed schemas
Cons: Often lose some benefits of relational databases, e.g., database-wide transactions	**Cons:** Limited BI tools; persistent data designed for a single application is harder	**Cons:** Fixed schemas help prevent errors

DocumentDB

Document 1

```
{
  "name": "Rahul",
  "country": "USA",
  "age": 23,
  "lastUse": "March 4, 2014"
}
```

Document 2

```
{
  "name": "Gunter",
  "country": "Germany",
  "age": 25
}
```

Document 3

```
{
  "name": "David",
  "country": "Australia",
  "age": 51,
  "firstUse": "May 8, 2013"
}
```

Document 4

```
{
 "docCount": 3,
 "last": "May 1, 2014"
}
```

DocumentDB

RESTful access methods

For Create/Read/Update/Delete (CRUD) operations

DocumentDB SQL

A query language with SQL-derived syntax

Example:

```
SELECT c.age
FROM customers c
WHERE c.name =
"Lou"
```

Executing logic in the database

Stored procedures

Triggers

User-defined functions (UDFs)

- Allow extending DocumentDB SQL

Thank you!