

NoSQL

Big Data Systems

Dr. Rubi Boim

NoSQL

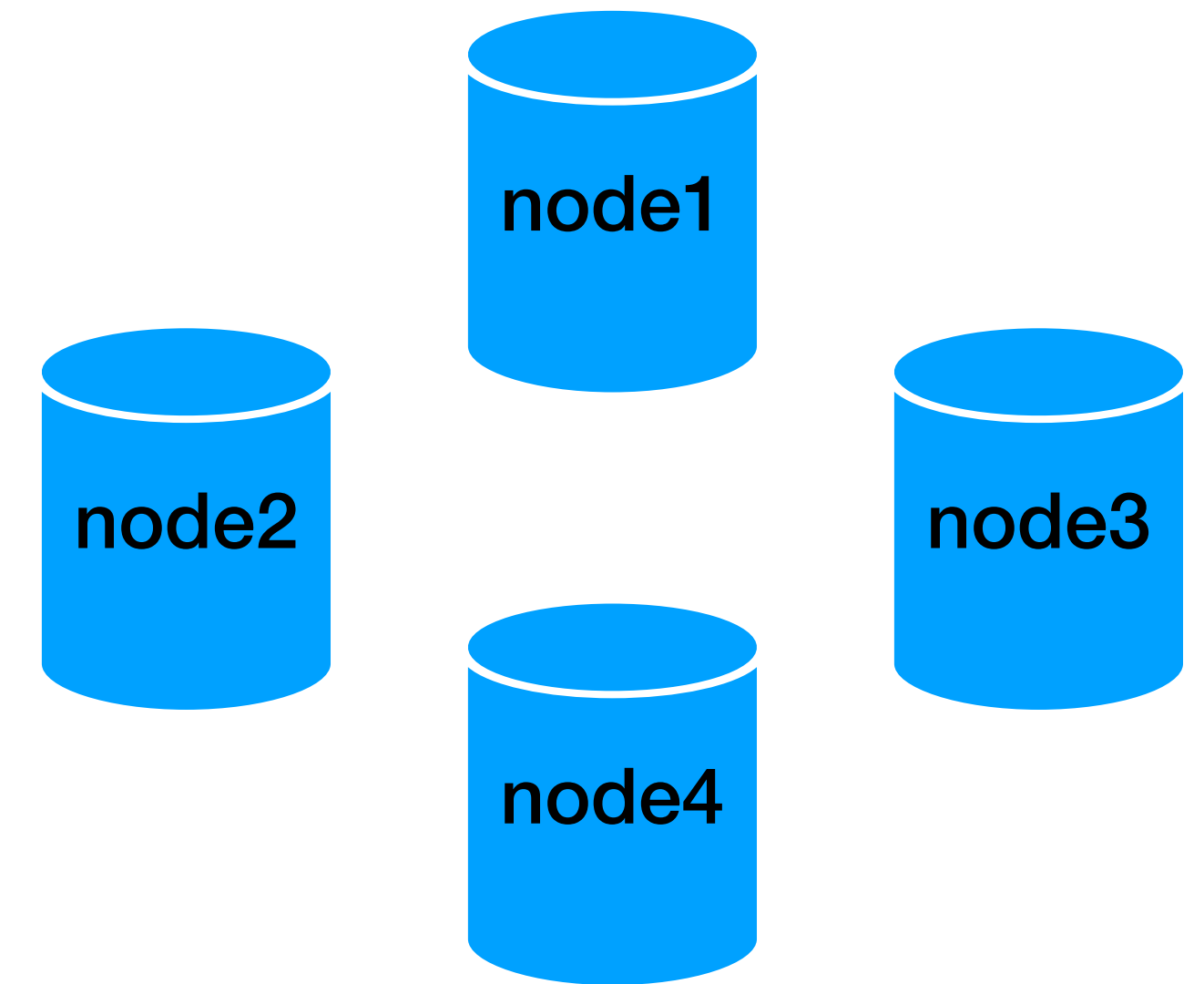
- Non SQL
- Not Only SQL
- Non schema SQL
- Non Relational SQL
- Most (distributed) Big Data storage / database systems



SQL is an API

NoSQL - motivation

- Scaling issues of RDBMS
- Variety of data (schema-less)
- Read / write performance
- Product needs of Google / Amazon / Facebook...



NoSQL - popular properties

- “Simpler” design / API
- Availability over Consistency
- **Performance over ACID transactions**
extremely fast read/write at scale
- Distributed - scale wide (commodity hardware)
- Schema less
- Multi data center

NoSQL - main types

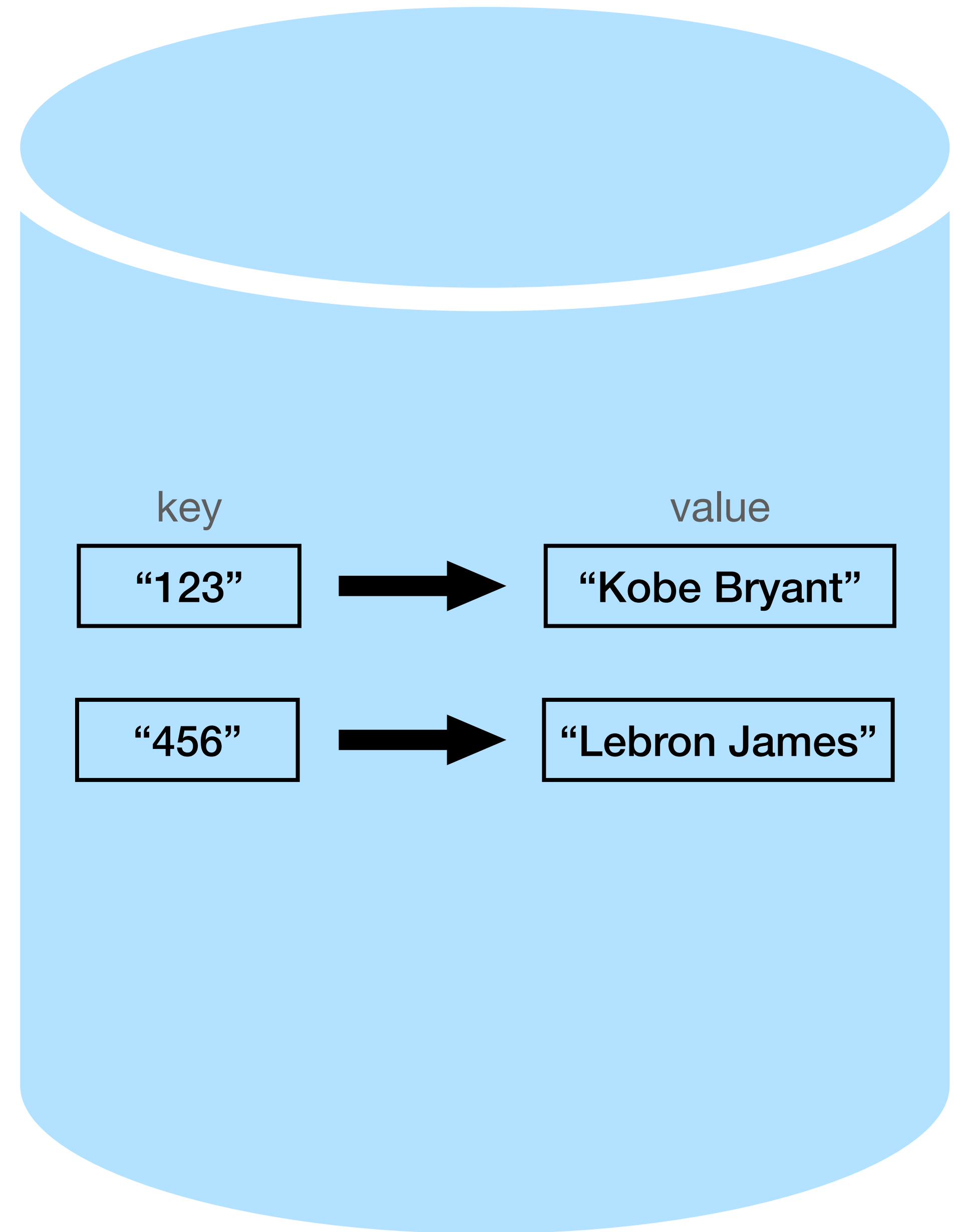
- Key Value
- Document
- Graph
- **Wide column**



by their internal
data model / API

NoSQL - Key Value

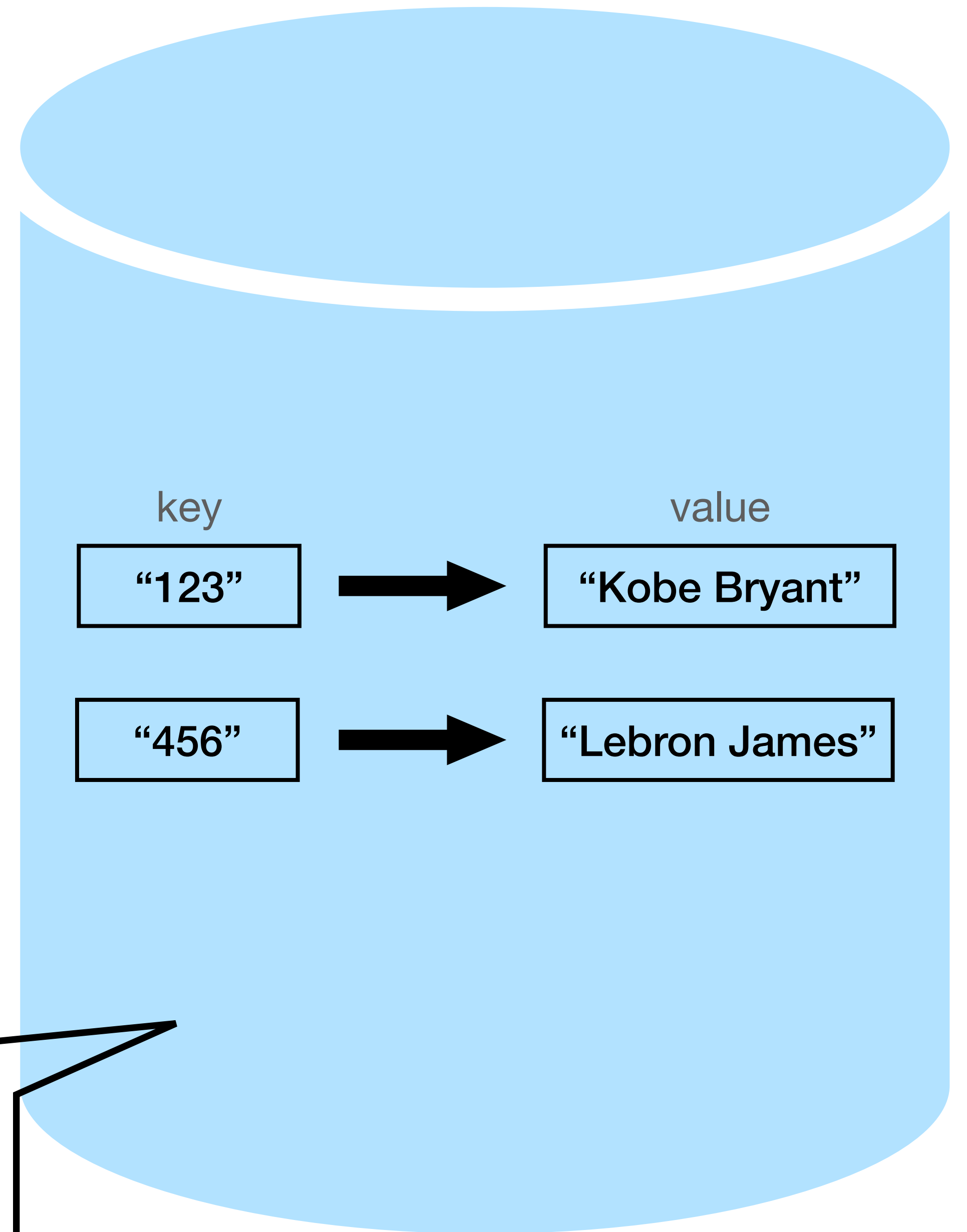
- “Simplest” NoSQL database
- Hash table / dictionary
key-value pairs
- **CRUD API**
`create (key/value)`
`read (key)`
`update (key/value)`
`delete (key)`



NoSQL - Key Value

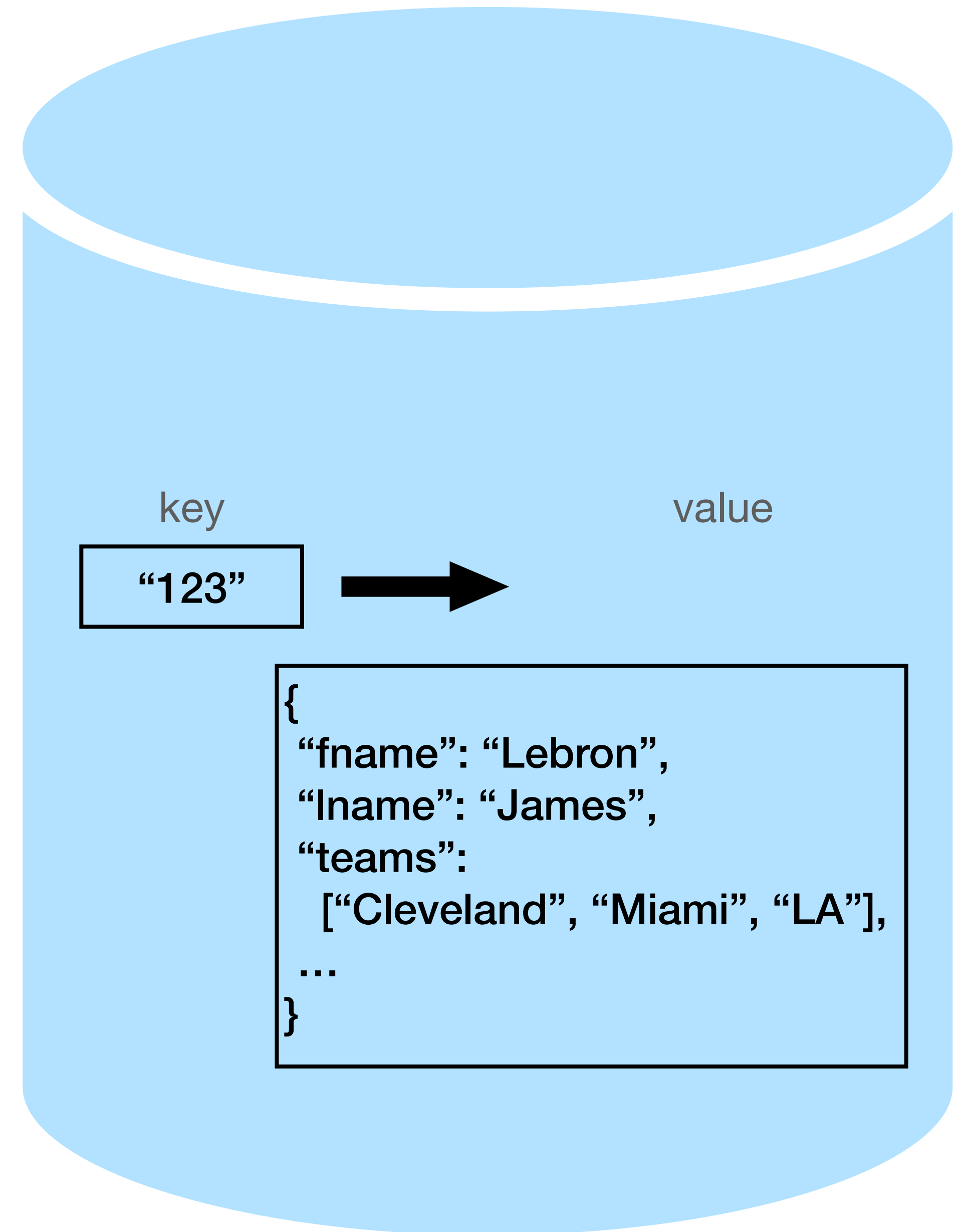
- “Simplest” NoSQL database
- Hash table / dictionary
key-value pairs
- **CRUD API**
`create(key/value)`
`read(key)`
`update(key/value)`
`delete(key)`

popular systems
dynamo, levelDB,
redis, memcached



NoSQL - Document

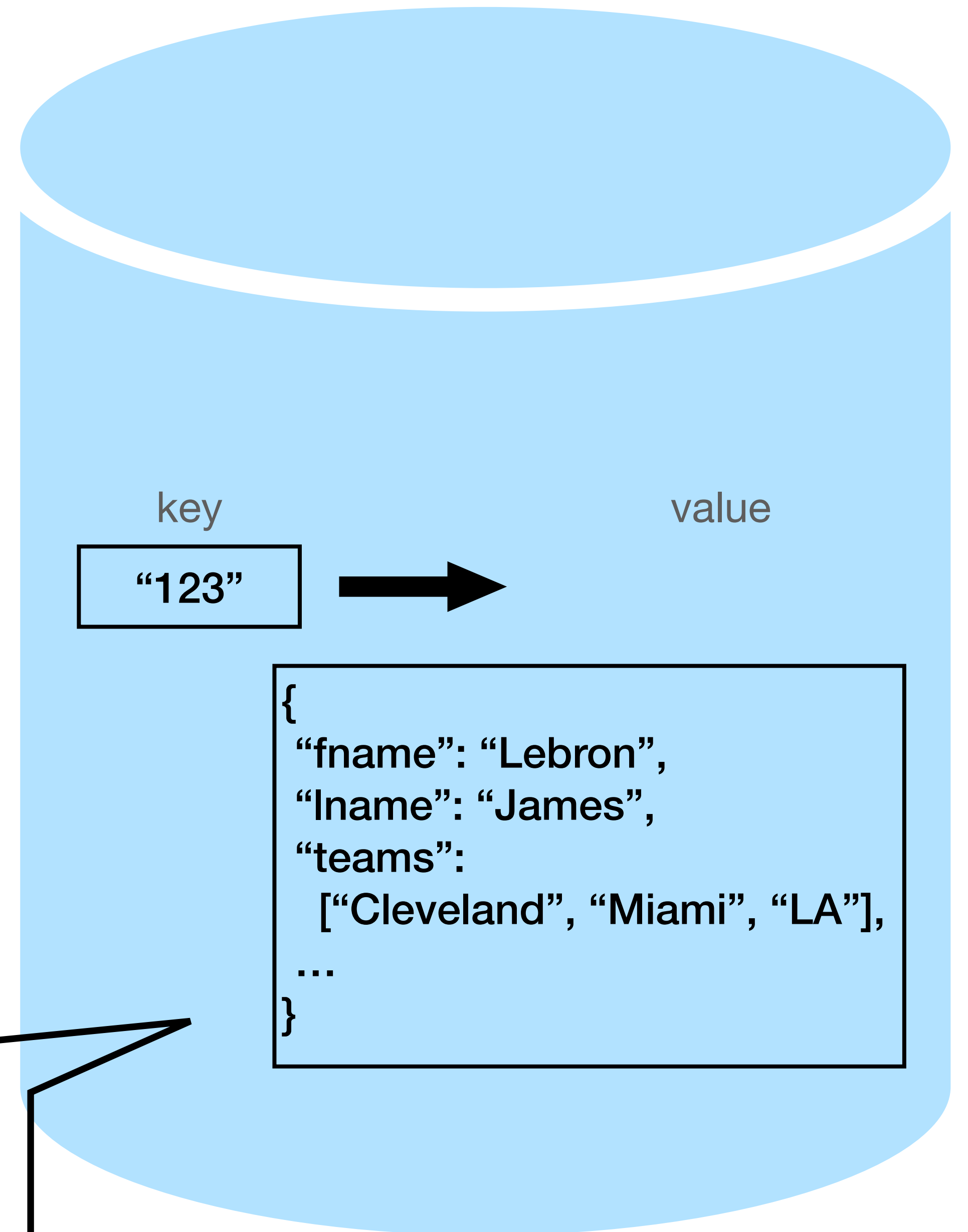
- Scheme less “document”
accessible by unique key
- Document encoding
json / xml / yaml / bson
- Stores all information of a
given object in a single place
- CRUD API
but not only
(find, sort...)



NoSQL - Document

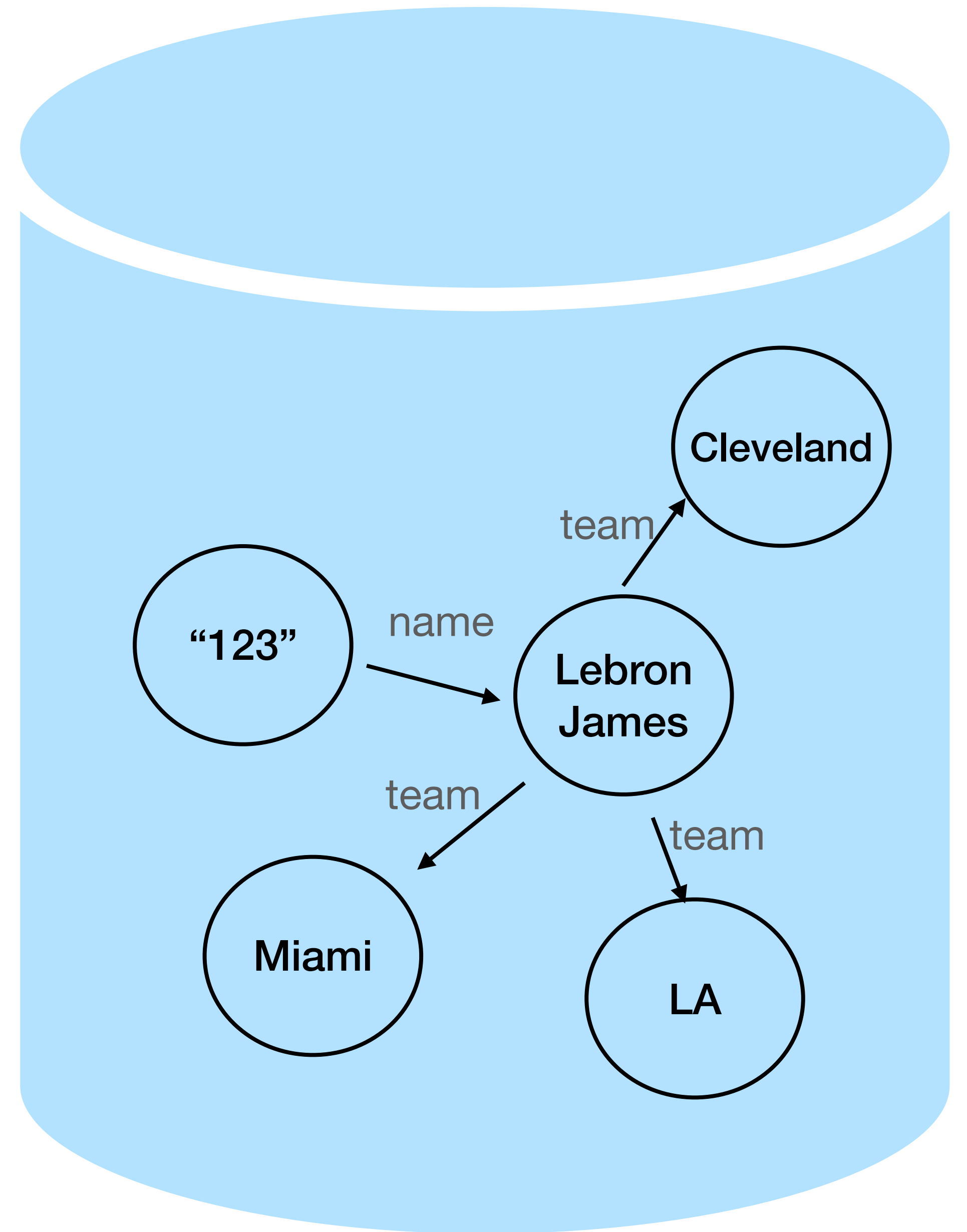
- Scheme less “document”
accessible by unique key
- Document encoding
json / xml / yaml / bson
- Stores all information of a
given object in a single place
- CRUD API
but not only
(find, sort...)

popular systems
mongoDB, CouchDB



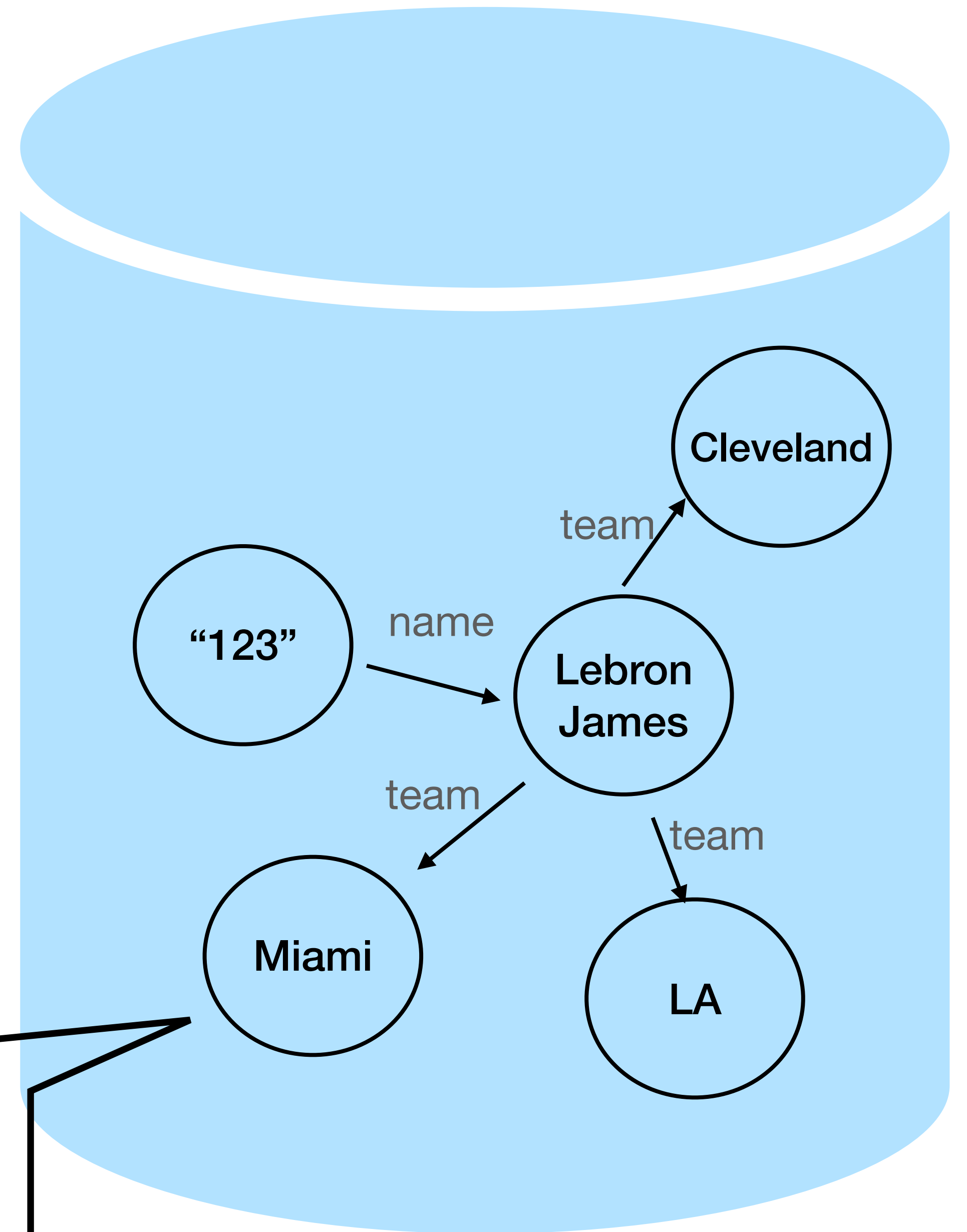
NoSQL - Graph

- Based on graph theory
nodes, edges, properties, weights
- ACID / Transactions
not always
- Built in graph algorithms



NoSQL - Graph

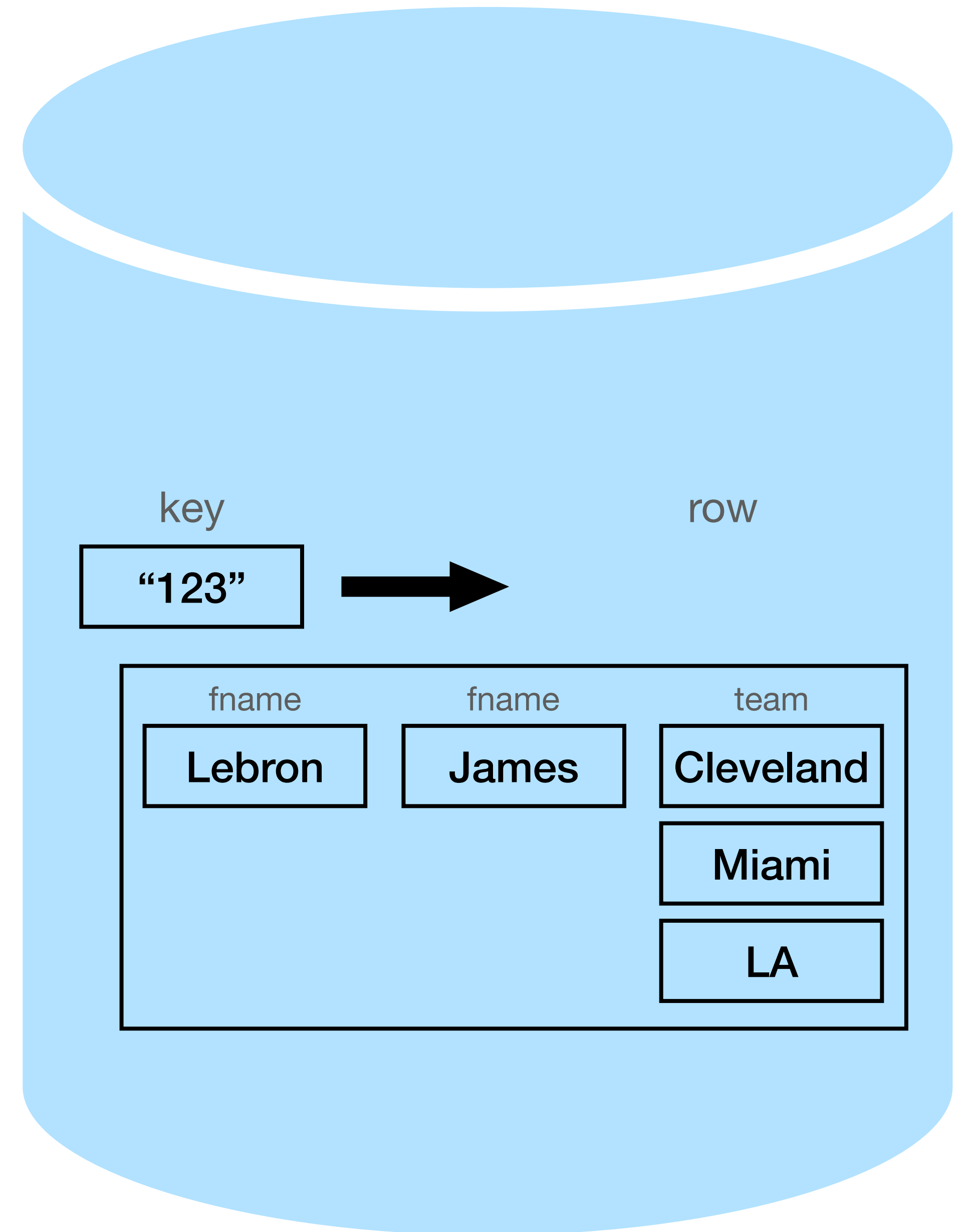
- Based on graph theory
nodes, edges, properties, weights
- ACID / Transactions
not always
- Built in graph algorithms



popular systems
neo4j, DataStax
enterprise graph

NoSQL - Wide column

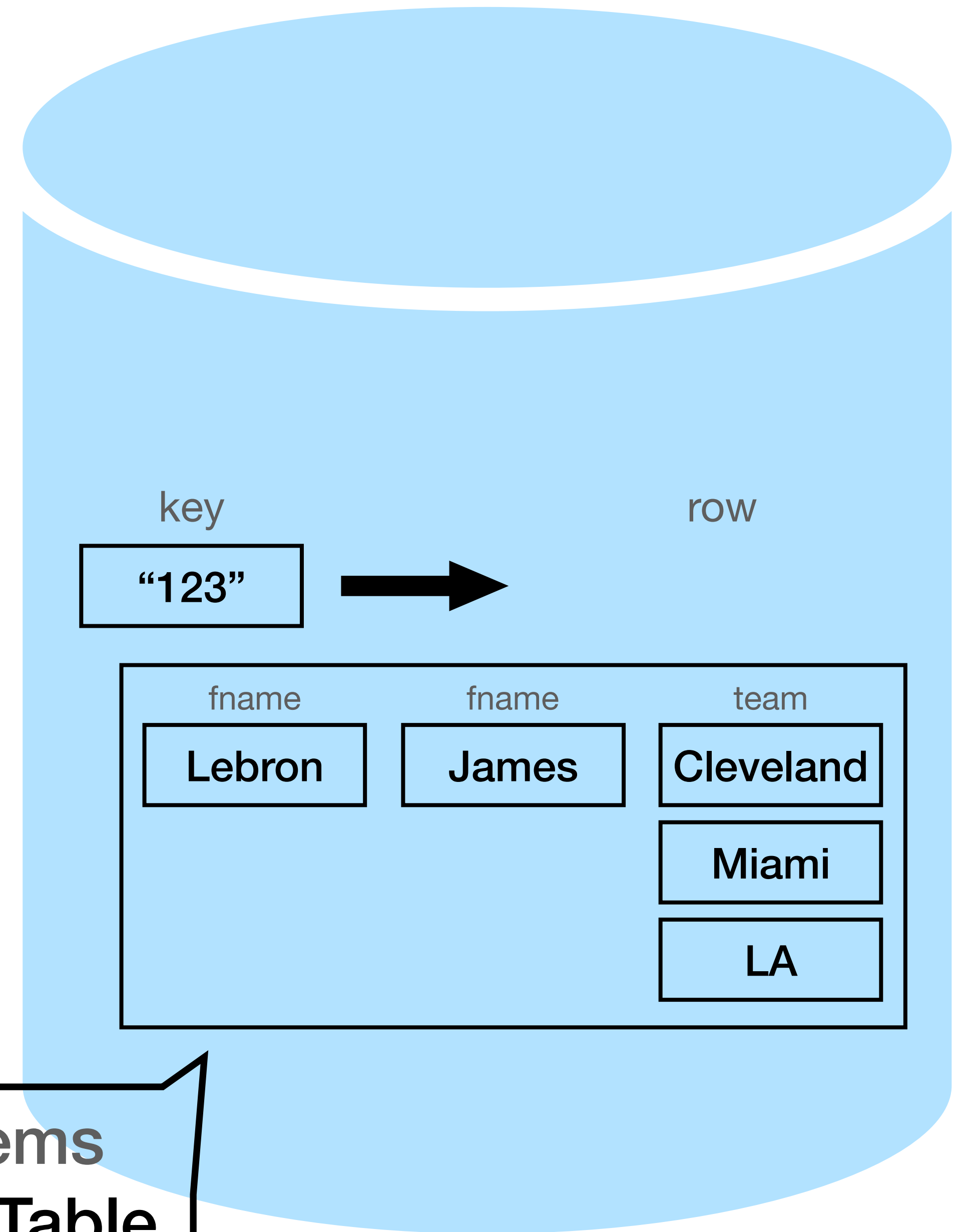
- Tables, rows, columns
can vary from row to row within a table
- Can be viewed as
2-dimensional key-value store
- Custom API
with some SQL sometimes (CQL)
- NOT a columnar database
but can be sometimes



NoSQL - Wide column

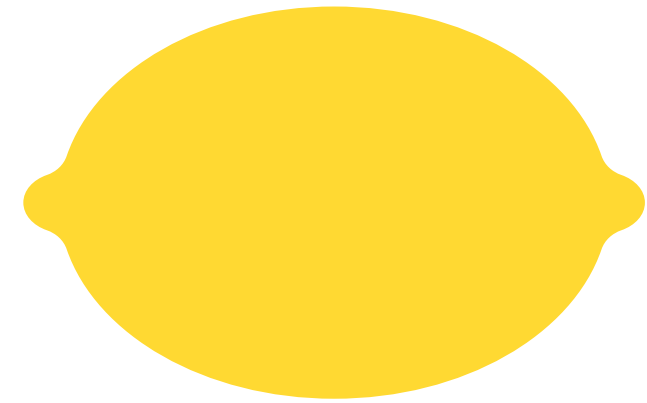
- Tables, rows, columns
can vary from row to row within a table
- Can be viewed as
2-dimensional key-value store
- Custom API
with some SQL sometimes (CQL)
- NOT a columnar database
but can be sometimes

popular systems
Cassandra, BigTable,
DynamoDB, HBase



NoSQL - Wide column

- A lot more on these systems
- Stay tuned :)



ACID

vs



BASE

Basically Available

Soft state

Eventual consistency

ACID (reminder)

- **Atomicity**
- **Consistency**
correctness / referential integrity (foreign key) - **NOT like in CAP**
- **Isolation**
- **Durability**

BASE

- **Basically Available**

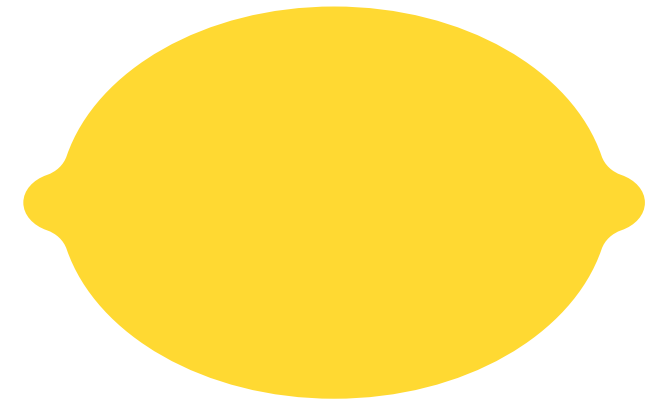
There will be a response even if node fails (response=fail)

- **Soft state**

State can change even if no read/write are performed
(the system is aiming towards consistent)

- **Eventual consistency**

reads may be inconsistent, but over time will be consistent



ACID

vs

BASE



Most relational DBs
(Oracle, MySQL...)

Most NoSQL systems
(C*, BigTable, MongoDB,...)

NoSQL and CAP

