The Goal:

Basic understanding of Cassandra data modeling

- **Fundamental Cassandra storage constructs**
- **How data modeling differs between Cassandra and relational**
- **Learn a Cassandra data modeling process**

Relational Expectations…

Given this table

Column Name	Column Type
userid	UUID
created_date	TIMESTAMP
email	TEXT
firstname	TEXT
lastname	TEXT

You might expect to execute this query

```
SELECT * FROM killrvideo.users WHERE email = 'me_email@gmail.com';
```

Relational Expectations…

Column Name	Column Type
userid	UUID
created_date	TIMESTAMP
email	TEXT
firstname	TEXT
lastname	TEXT

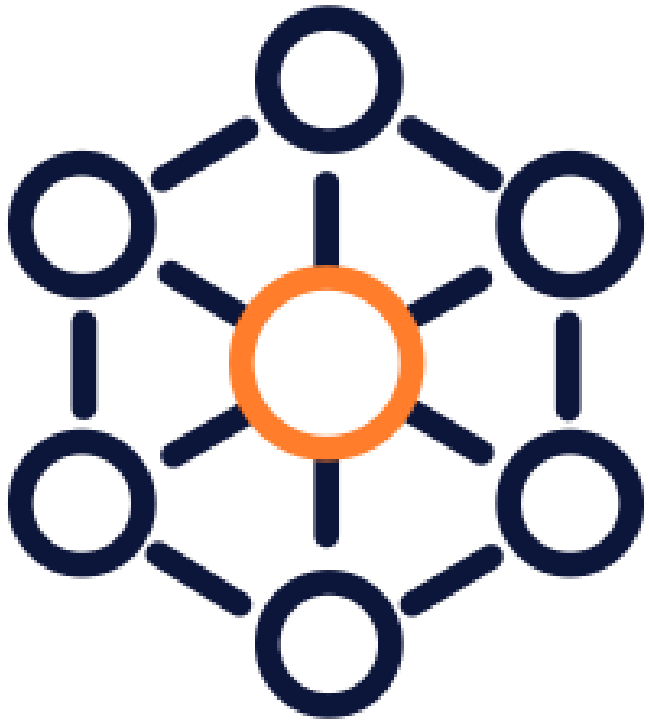
```
SELECT * FROM killrvideo.users WHERE email = 'me_email@gmail.com';
```

Execution Error

An error occurred on line 1:
Cannot execute this query as it might involve data filtering and thus may have unpredictable performance. If you want to execute this query despite the performance unpredictability, use ALLOW FILTERING
(Hint: Ctrl-L to toggle line numbers.)

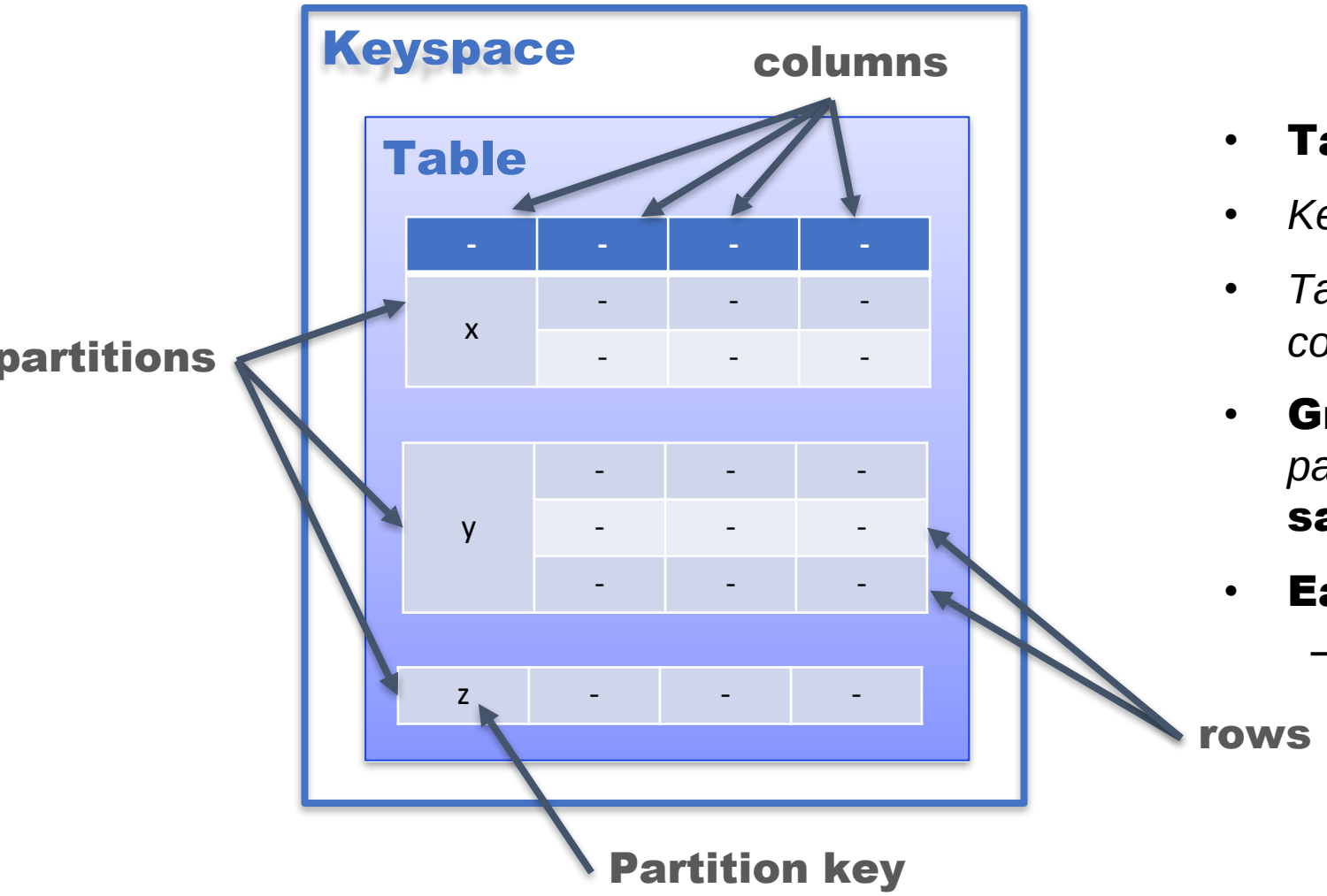
But it doesn't work!

Cassandra Tables Work Differently!



Let's see why...

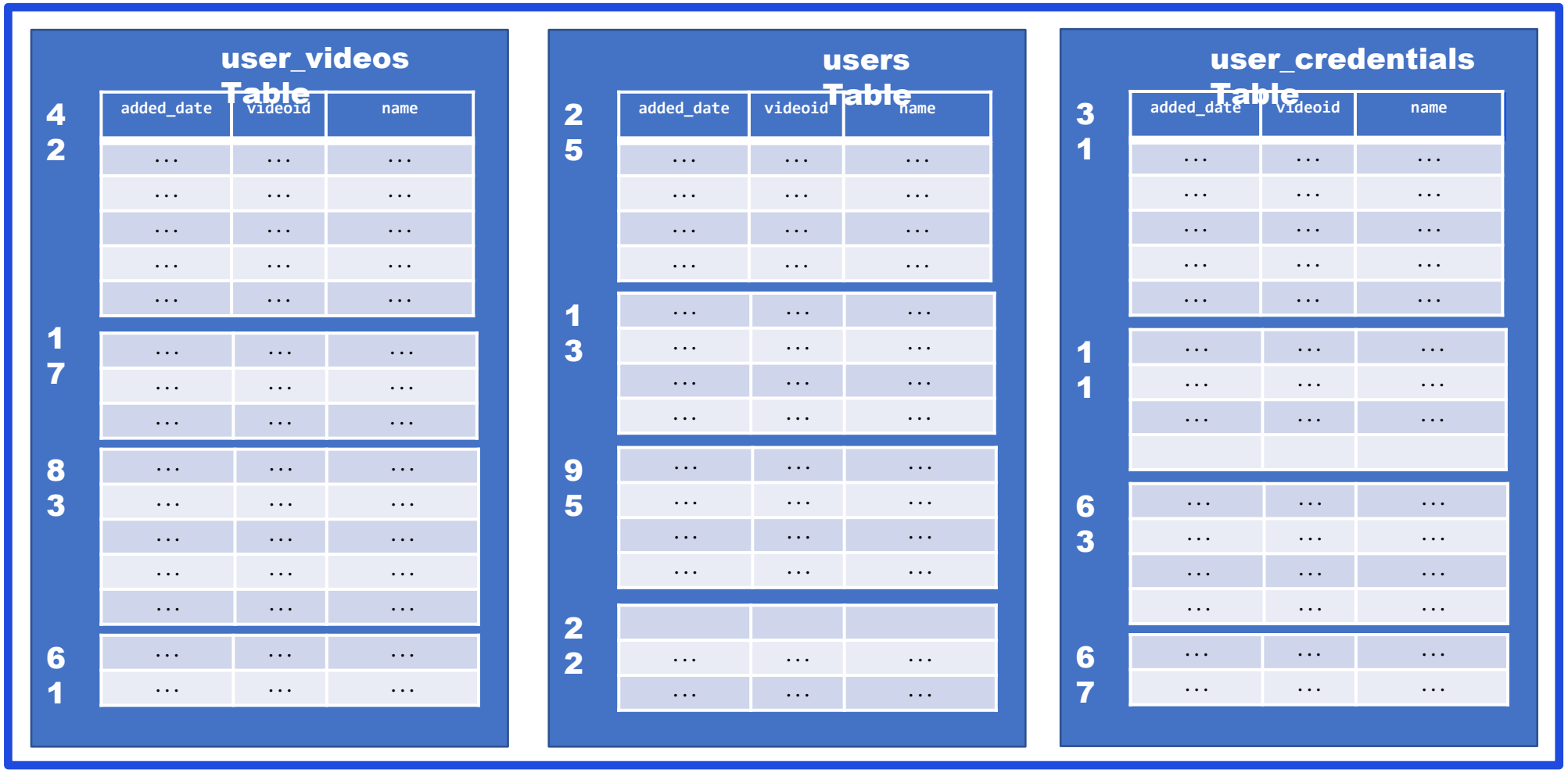
How does Cassandra structure data?



- **Tabular data model, with one twist**
- *Keyspaces* **contain** *tables*
- *Tables* **are organized in** *rows* **and** *columns*
- **Groups of related rows, called** *partitions,* **are stored together on the same node (or nodes)**
- **Each row contains a** *partition key*
  - **One or more columns that are hashed to determine which node(s) store that data**

Cassandra Data Modeling

killrvideo Keyspace



Cassandra Data Modeling

```
CREATE TABLE user_videos (
    userid uuid,
    added_date timestamp,
    videoid uuid,
    name text,
    PRIMARY KEY ((userid), added_date, videoid)
);
```

Cassandra Data Modeling

```
CREATE TABLE user_videos (
    userid uuid,
    added_date timestamp,
    videoid uuid,
    name text,
    PRIMARY KEY ((userid), added_date, videoid)
);
```

Primary Key

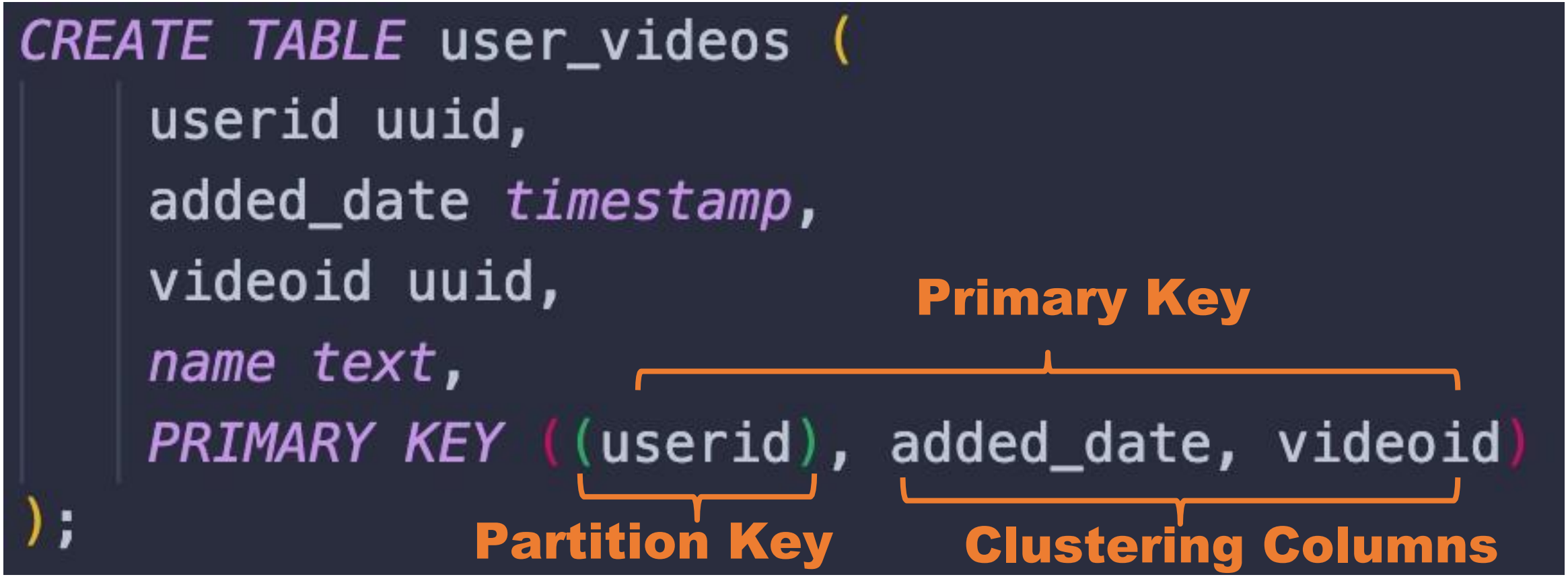
Cassandra Data Modeling

```
CREATE TABLE user_videos (
    userid uuid,
    added_date timestamp,
    videoid uuid,
    name text,
    PRIMARY KEY ((userid), added_date, videoid)
);
```

Primary Key

Partition Key

Cassandra Data Modeling



Cassandra Data modeling

- Partition - the fundamental unit of access

- **A Partition has**
  - **A partition key**
  - **Associated rows with**
    - **Clustering columns**
    - **Data columns**

Partition

userid	added_date	videoid	name
749b2843-…	2019-09-18 10:38:42	2370b72f-…	Cat videos
749b2843-…	2019-09-18 10:38:42	e860f31a-…	Dog videos
749b2843-…	2019-09-19 11:15:27	89e11f78-…	Parakeet videos
749b2843-…	2019-09-19 21:13:03	115ee279-…	Turtle videos
749b2843-…	2019-09-20 12:17:11	49ef2219-…	Moose videos

Cassandra Data modeling

- Partition - the fundamental unit of access

- **A Partition has**
  - **A partition key**
  - **Associated rows with**
    - **Clustering columns**
    - **Data columns**

userid	added_date	videoid	name
749b2843-…	2019-09-18 10:38:42	2370b72f-…	Cat videos
749b2843-…	2019-09-18 10:38:42	e860f31a-…	Dog videos
749b2843-…	2019-09-19 11:15:27	89e11f78-…	Parakeet videos
749b2843-…	2019-09-19 21:13:03	115ee279-…	Turtle videos
749b2843-…	2019-09-20 12:17:11	49ef2219-…	Moose videos

Partition Key:

749b2843-…

Cassandra Data modeling

- Partition - the fundamental unit of access

- **A Partition has**
  - **A partition key**
  - **Associated rows with**
    - **Clustering columns**
    - **Data columns**

userid	added_date	videoid	name
749b2843-…	2019-09-18 10:38:42	2370b72f-…	Cat videos
749b2843-…	2019-09-18 10:38:42	e860f31a-…	Dog videos
749b2843-…	2019-09-19 11:15:27	89e11f78-…	Parakeet videos
749b2843-…	2019-09-19 21:13:03	115ee279-…	Turtle videos
749b2843-…	2019-09-20 12:17:11	49ef2219-…	Moose videos

Partition Key:

749b2843-…

Cassandra Data modeling

- Partition - the fundamental unit of access

- **A Partition has**
  - **A partition key**
  - **Associated rows with**
    - **Clustering columns**
    - **Data columns**

userid	added_date	videoid	name
749b2843-…	2019-09-18 10:38:42	2370b72f-…	Cat videos
749b2843-…	2019-09-18 10:38:42	e860f31a-…	Dog videos
749b2843-…	2019-09-19 11:15:27	89e11f78-…	Parakeet videos
749b2843-…	2019-09-19 21:13:03	115ee279-…	Turtle videos
749b2843-…	2019-09-20 12:17:11	49ef2219-…	Moose videos

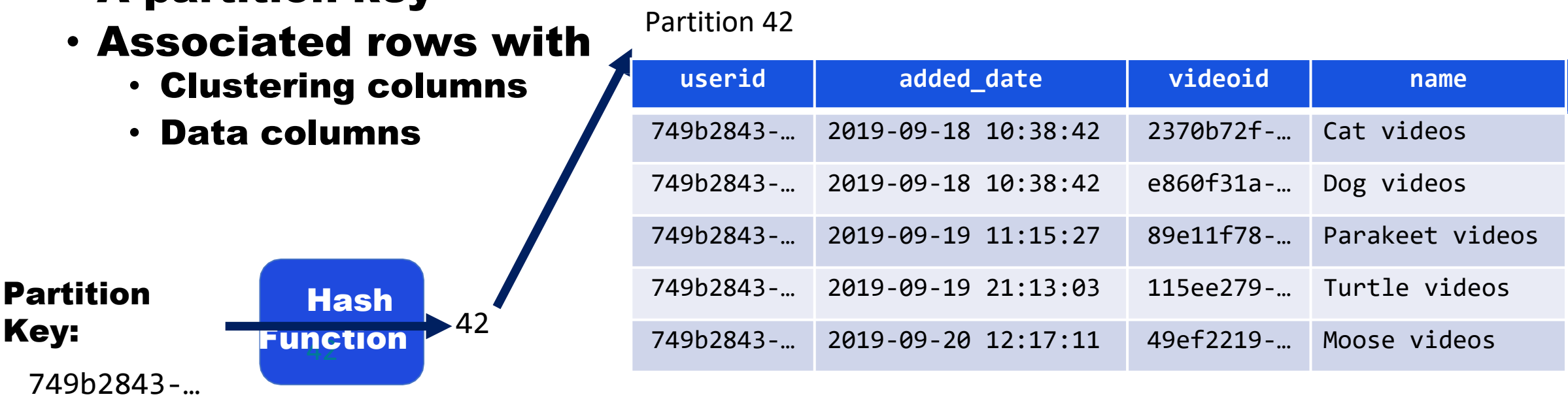
Partition Key:

749b2843-…

Cassandra Data modeling

- Partition - the fundamental unit of access

- **A Partition has**
  - **A partition key**
  - **Associated rows with**
    - **Clustering columns**
    - **Data columns**



Cassandra Data modeling

- Partition - the fundamental unit of access

- **A Partition has**
  - **A partition key**
  - **Associated rows with**
    - **Clustering columns**
    - **Data columns**

Clustering Columns

Partition 42

userid	added_date	videoid	name
749b2843-…	2019-09-18 10:38:42	2370b72f-…	Cat videos
749b2843-…	2019-09-18 10:38:42	e860f31a-…	Dog videos
749b2843-…	2019-09-19 11:15:27	89e11f78-…	Parakeet videos
749b2843-…	2019-09-19 21:13:03	115ee279-…	Turtle videos
749b2843-…	2019-09-20 12:17:11	49ef2219-…	Moose videos

Cassandra Data modeling

- Partition - the fundamental unit of access

- **A Partition has**
  - **A partition key**
  - **Associated rows with**
    - **Clustering columns**
    - **Data columns**

Partition 42

Data Column

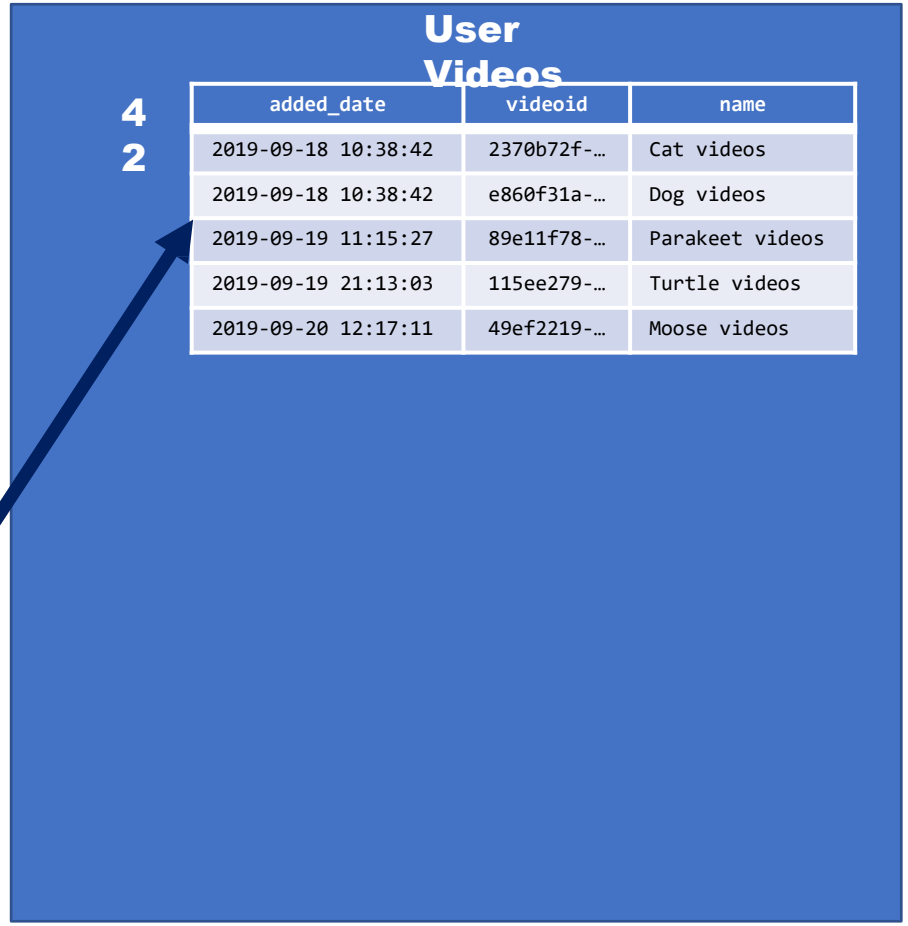
userid	added_date	videoid	name
749b2843-…	2019-09-18 10:38:42	2370b72f-…	Cat videos
749b2843-…	2019-09-18 10:38:42	e860f31a-…	Dog videos
749b2843-…	2019-09-19 11:15:27	89e11f78-…	Parakeet videos
749b2843-…	2019-09-19 21:13:03	115ee279-…	Turtle videos
749b2843-…	2019-09-20 12:17:11	49ef2219-…	Moose videos

Cassandra Data Modeling

Tables hold many partitions

Partition 749b2843-…

added_date	videoid	name
2019-09-18 10:38:42	2370b72f-…	Cat videos
2019-09-18 10:38:42	e860f31a-…	Dog videos
2019-09-19 11:15:27	89e11f78-…	Parakeet videos
2019-09-19 21:13:03	115ee279-…	Turtle videos
2019-09-20 12:17:11	49ef2219-…	Moose videos

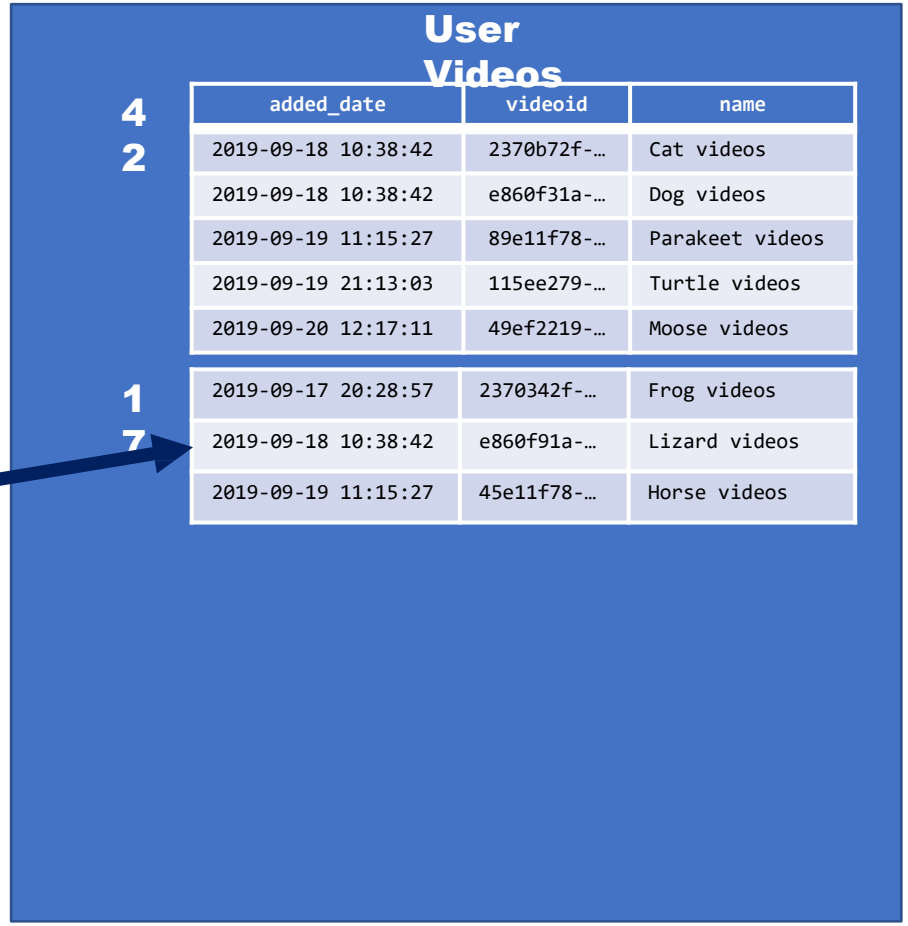


Cassandra Data Modeling

Tables hold many partitions

Partition

added_date	videoid	name
2019-09-17 20:28:57	2370342f-…	Frog videos
2019-09-18 10:38:42	e860f91a-…	Lizard videos
2019-09-19 11:15:27	45e11f78-…	Horse videos

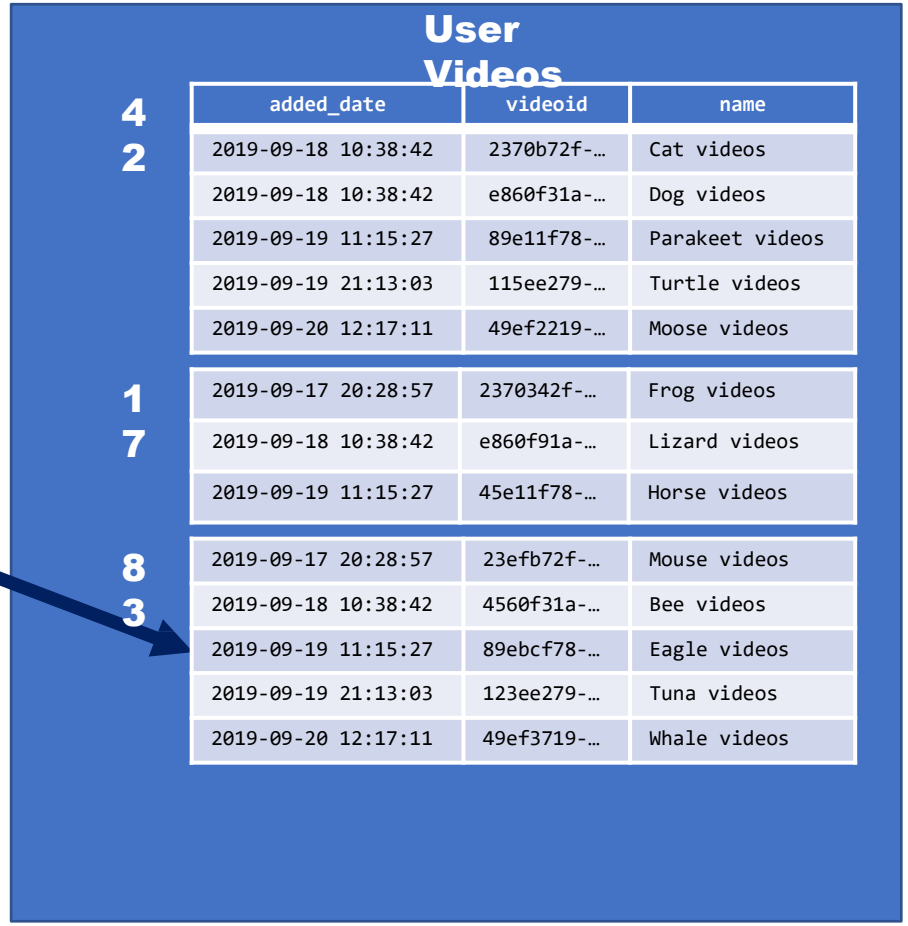


Cassandra Data Modeling

Tables hold many partitions

Partition

added_date	videoid	name
2019-09-17 20:28:57	23efb72f-…	Mouse videos
2019-09-18 10:38:42	4560f31a-…	Bee videos
2019-09-19 11:15:27	89ebcf78-…	Eagle videos
2019-09-19 21:13:03	123ee279-…	Tuna videos
2019-09-20 12:17:11	49ef3719-…	Whale videos

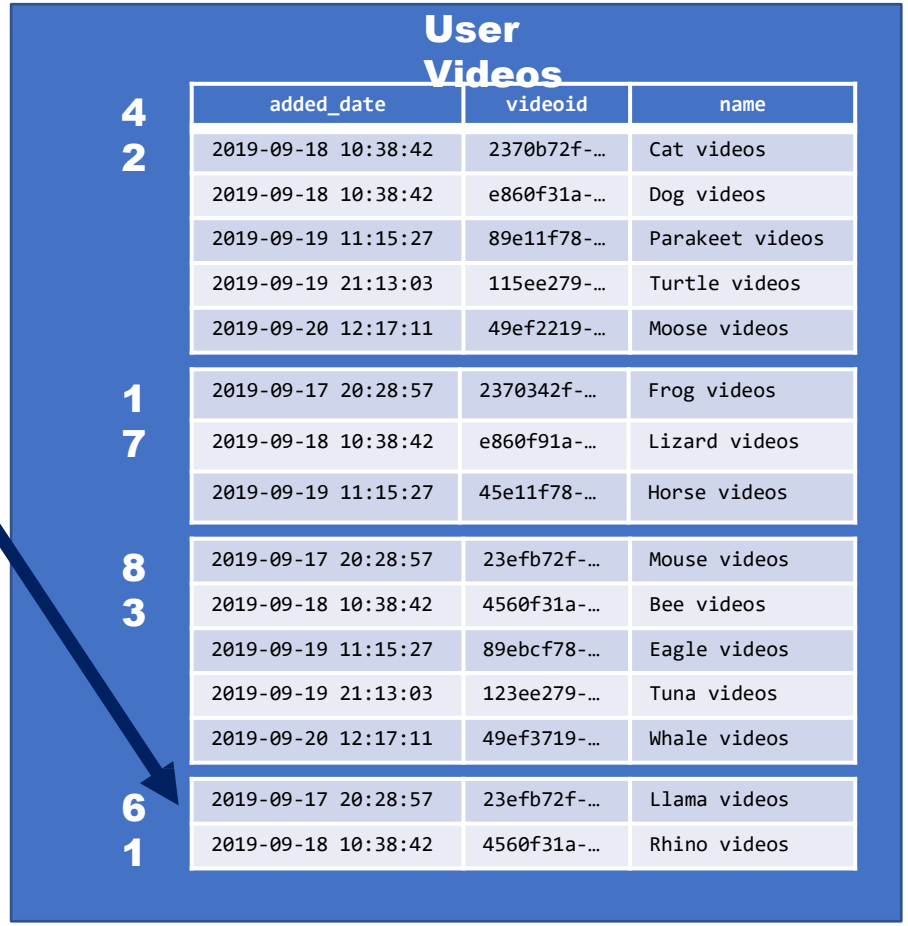


Cassandra Data Modeling

Tables hold many partitions

Partition

added_date	videoid	name
2019-09-17 20:28:57	23efb72f-…	Llama videos
2019-09-18 10:38:42	4560f31a-…	Rhino videos



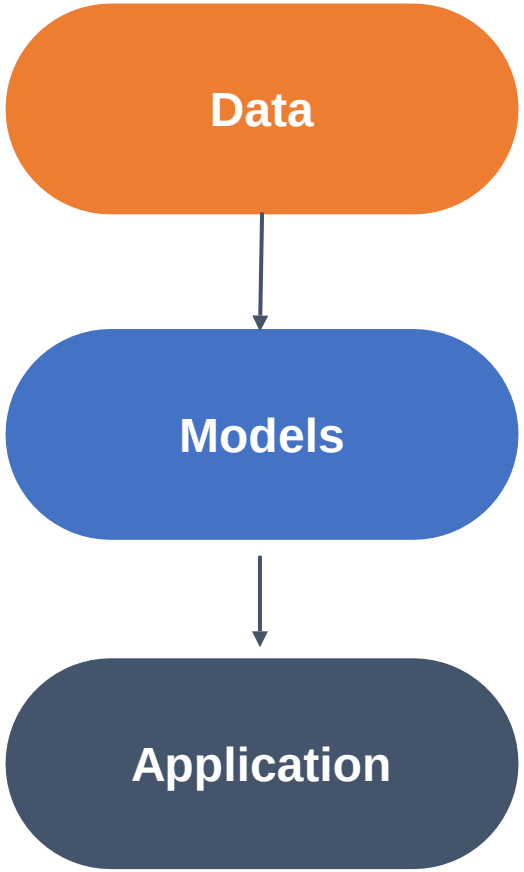
Cassandra Data Modeling

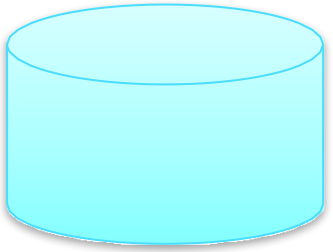
“Partitions are pre-computed queries”

Cassandra Data Modeling

“How do I figure out my schema?”

Relational Modeling





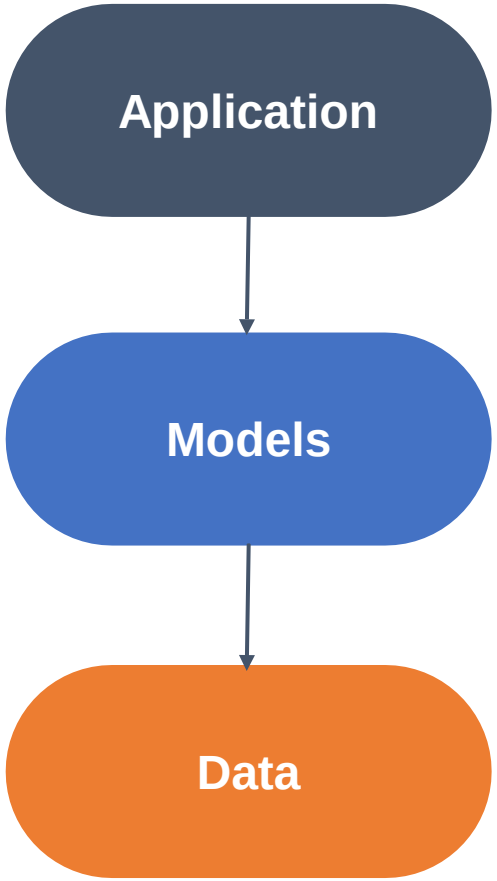
Employees

userId	firstName	lastName
1	Edgar	Codd
2	Raymond	Boyce

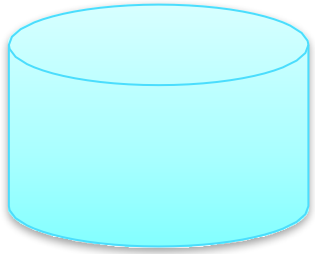
Department

departmentId	department
1	Engineering
2	Math

Cassandra Modeling



id	firstName	lastName	department
1	Edgar	Codd	Engineering
2	Raymond	Boyce	Math

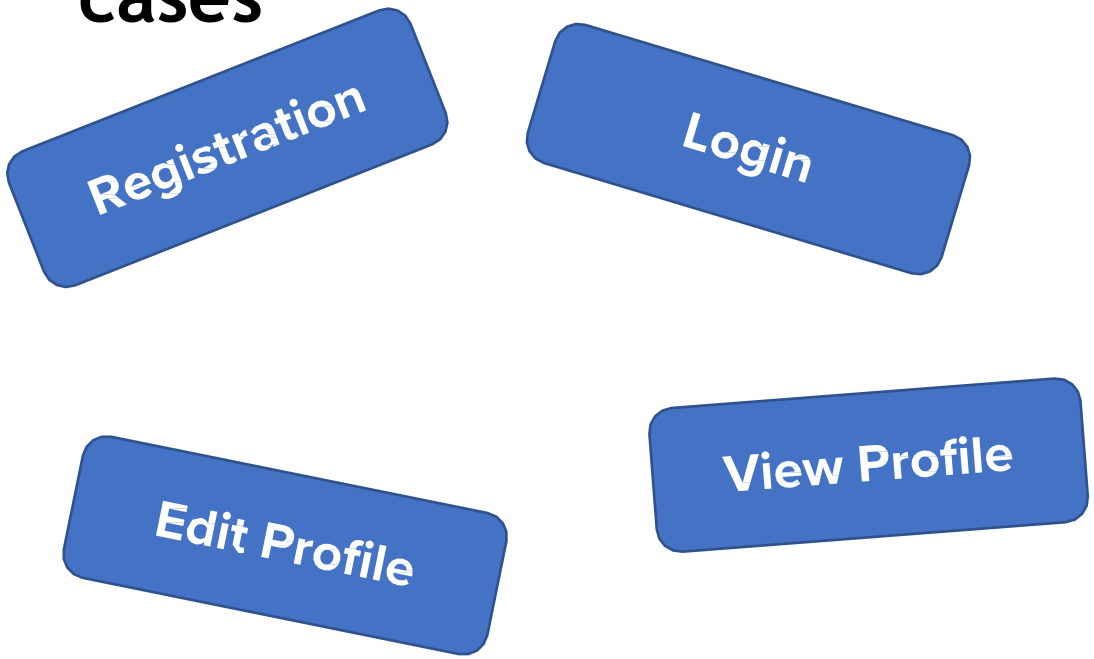


Cassandra Data Modeling Process

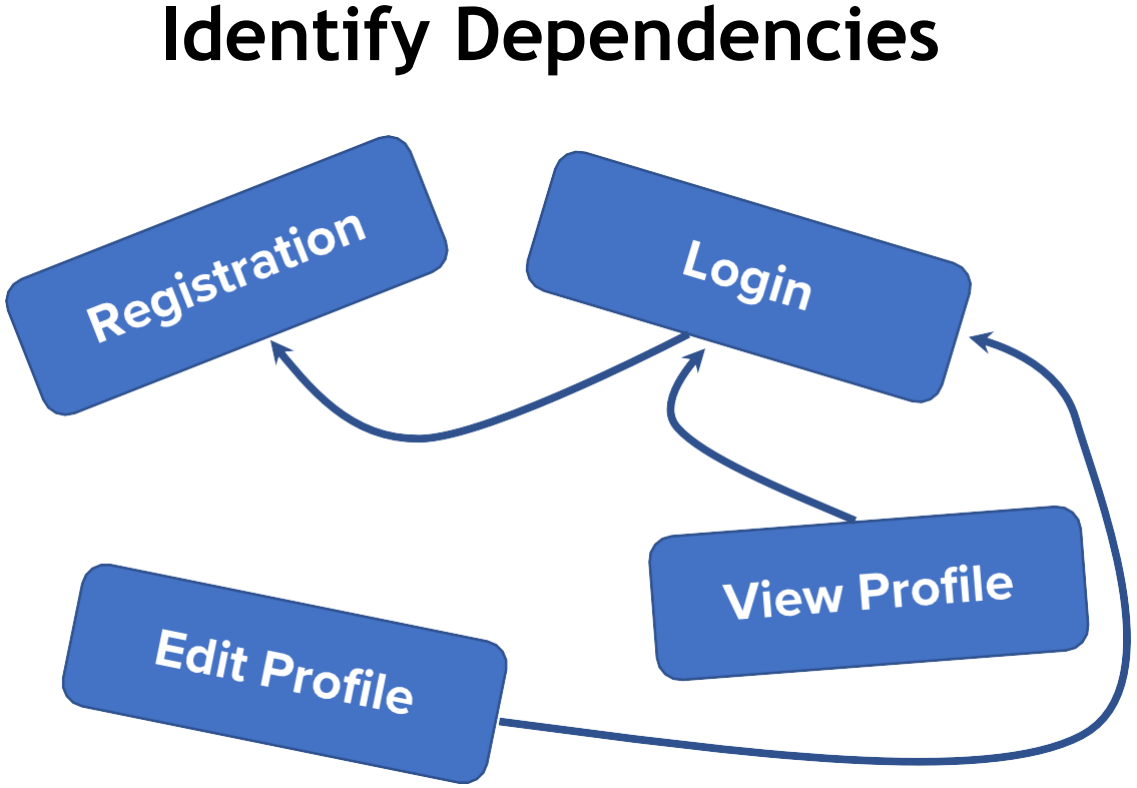
Use-cases

Example: User Management Use-cases

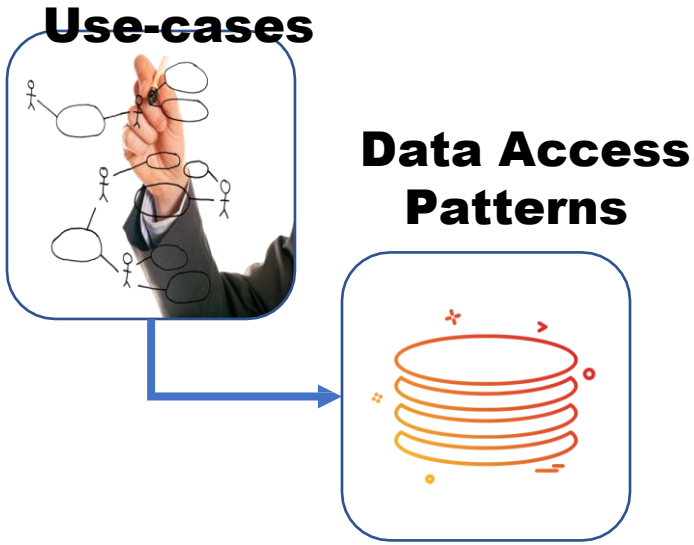
Brainstorm/List Use-cases



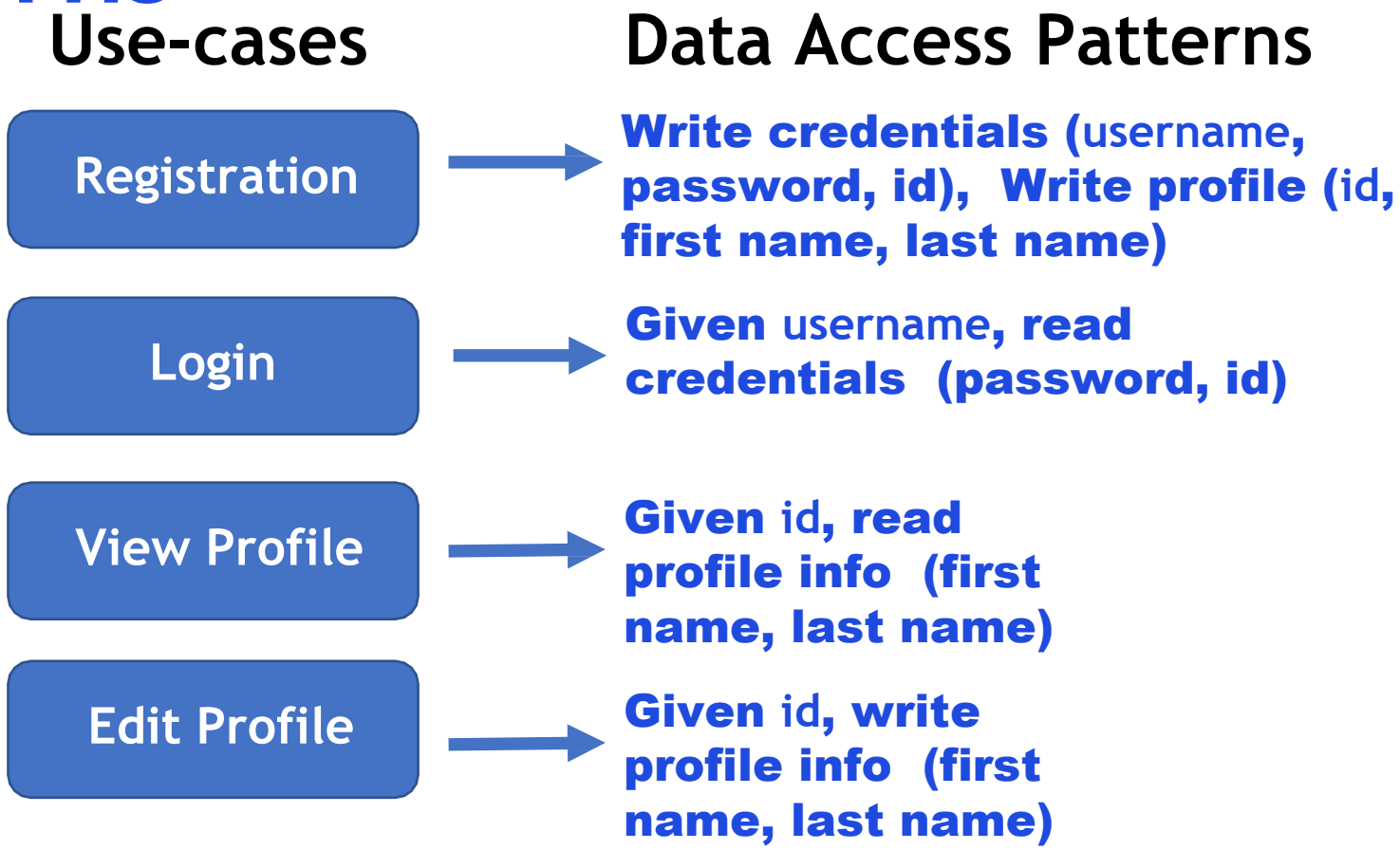
Example: User Management Use-cases



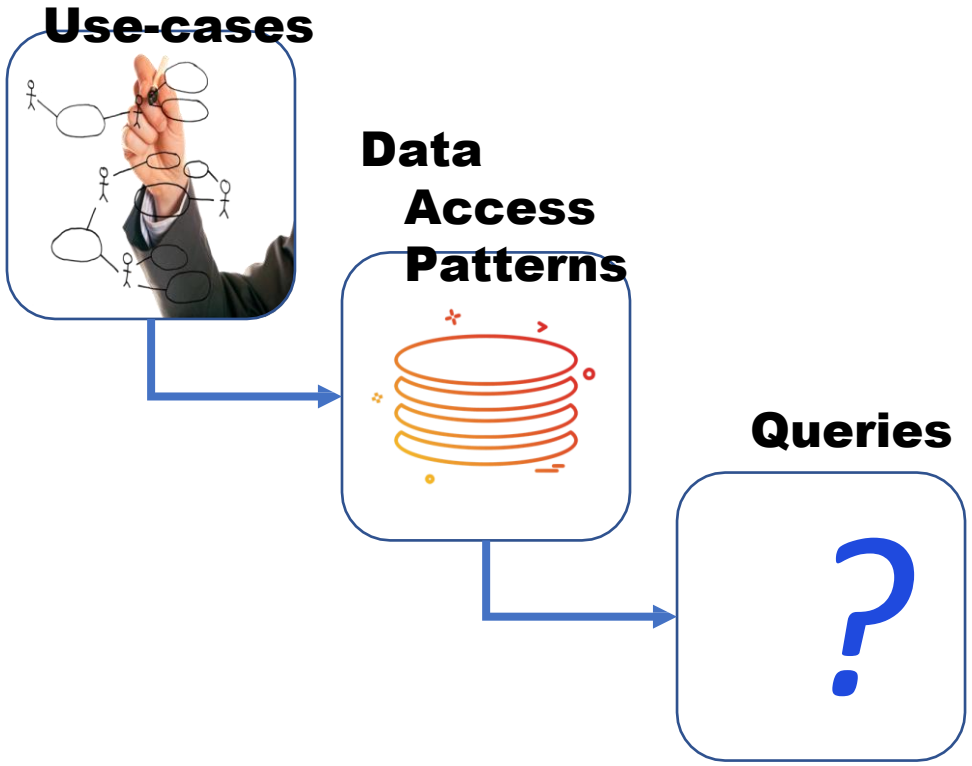
Cassandra Data Modeling Process



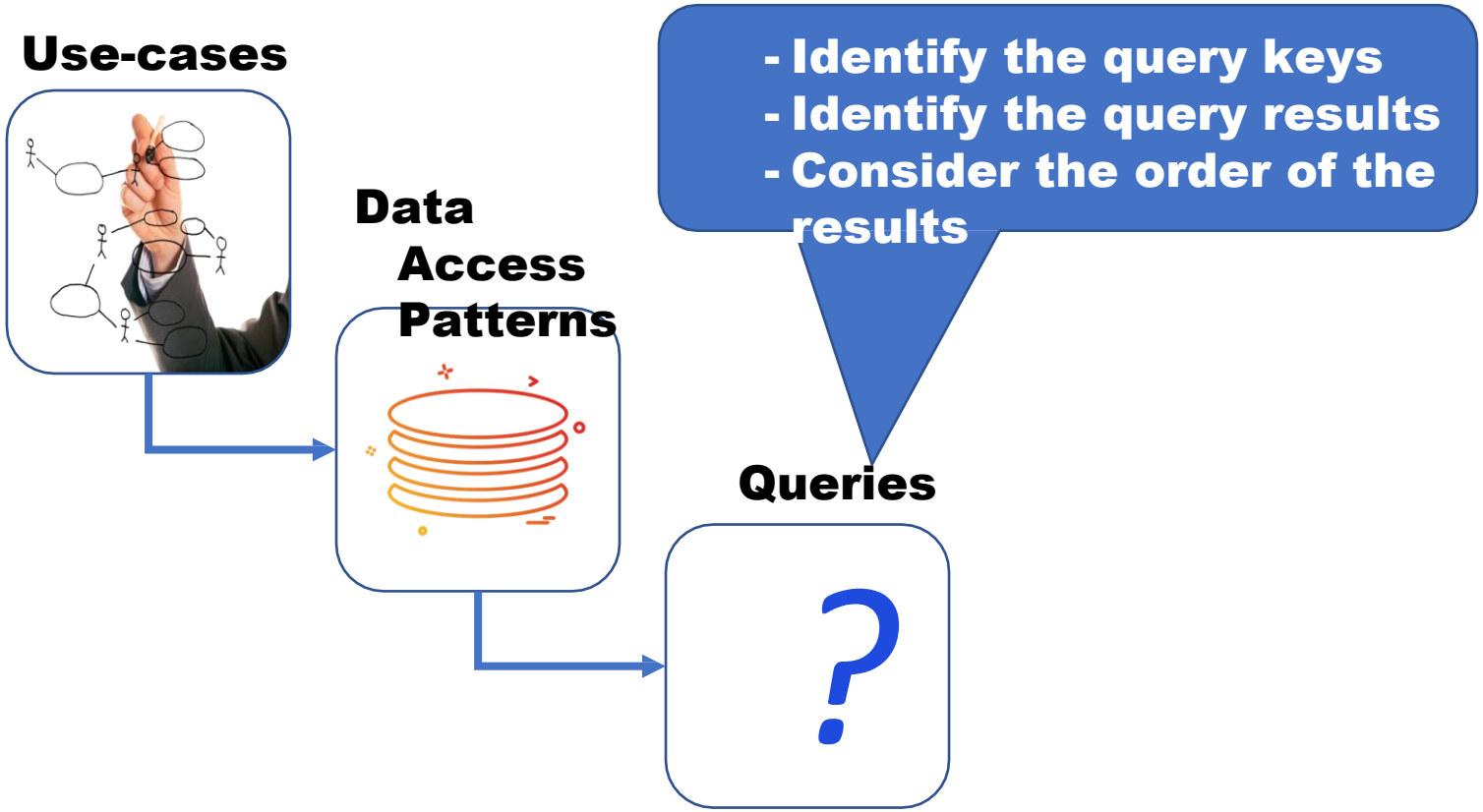
Example: User Management Access Patterns



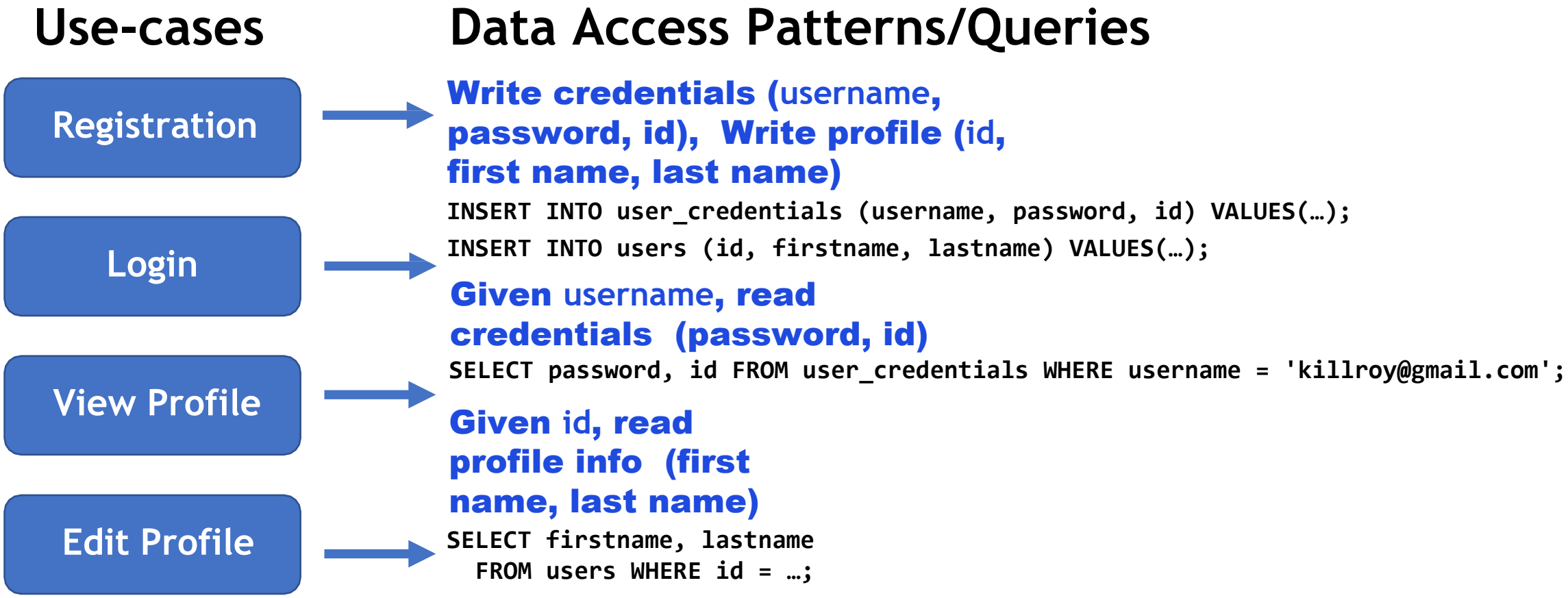
Cassandra Data Modeling Process



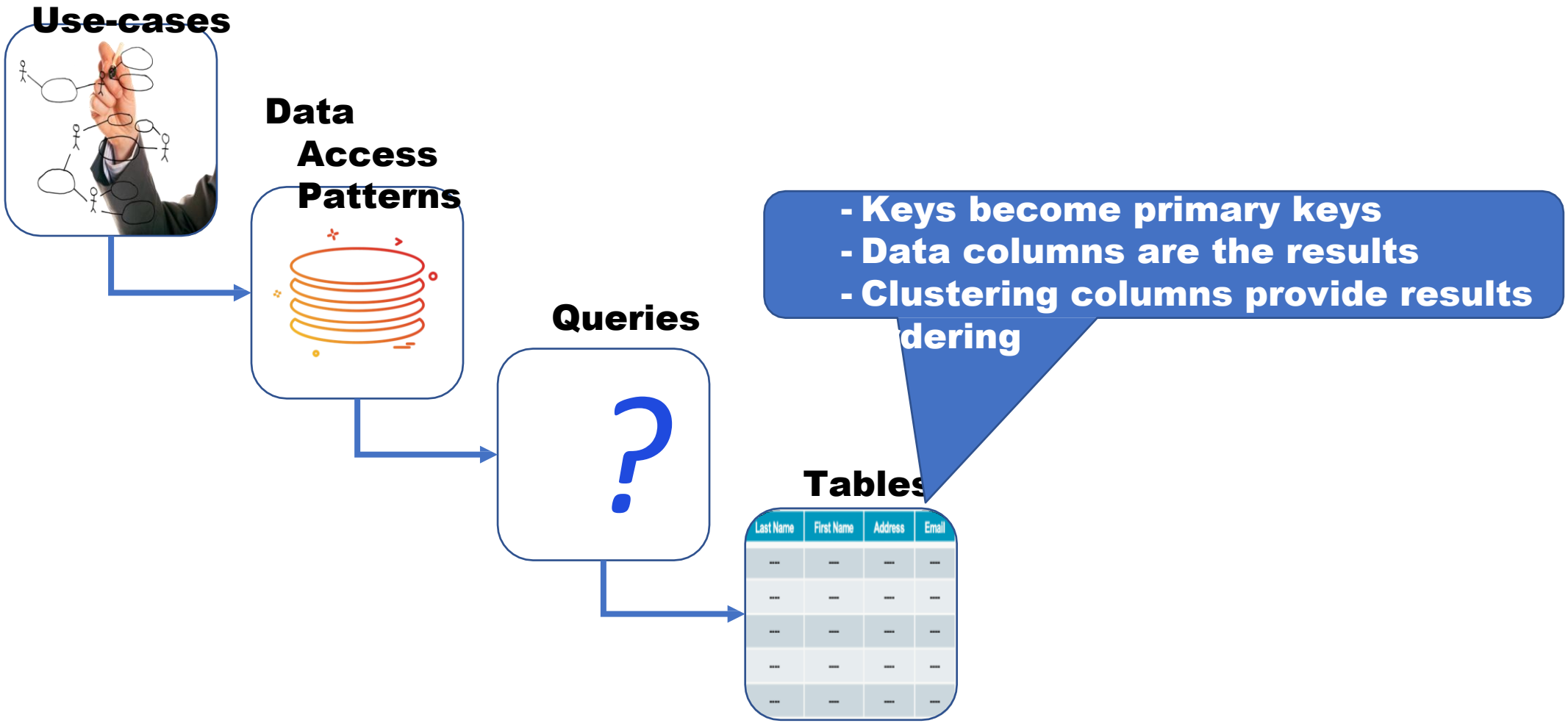
Cassandra Data Modeling Process



Example: User Management Queries



Cassandra Data Modeling Process



Example: User Management Tables

user_credentials Table

Name	Type
username	text
password	text
id	uuid

users Table

Name	Type
id	uuid
lastname	text
firstname	text

Cassandra Data Modeling

“Are we talking denormalization?”

Cassandra Data Modeling

“Denormalization, oh no!”

We *are* Talking *Denormalization! And it's OK!*

We *are* Talking *Denormalization! And it's OK!*

Cassandra scales using denormalization

We *are* Talking *Denormalization! And it's OK!*

Cassandra scales using denormalization

And it works great!

Cassandra Data Modeling

Denormalization example:

index	userid	created_date	email	firstname	lastname
0	7a992284-eff5-420f-8f76-c55e0f8e926c	2019-04-24T22:58:43.120+0000	jking@datastax.com	Jamie	King
1	b0ed10eb-413c-4d77-9a67-412e6ed7e148	2019-04-24T22:58:43.124+0000	moo@you.com	Moo	You
2	48fda2de-c65b-49fb-aa3e-8c5ca60111fd	2019-04-24T22:58:43.117+0000	shalladay@datastax.com	Steve	Halladay

index	email	password	userid
0	shalladay@datastax.com	St3v3s_P@ssw0rd	48fda2de-c65b-49fb-aa3e-8c5ca60111fd
1	moo@you.com	moo	b0ed10eb-413c-4d77-9a67-412e6ed7e148
2	jking@datastax.com	J@m13s_P@ssw0rd	7a992284-eff5-420f-8f76-c55e0f8e926c

Upserts - Mutations with Side-effects

- No read-before-write - because it's fast!

Upserts – Mutations with Side-effects

- No read-before-write – because it's fast!
- Side-effects:
  - Inserts can update
  - Updates can insert

Summary

Learned how Cassandra is different

- **Queries require partition key**
- **Upserts**

Practice Cassandra's data modeling process

- **List use-cases and dependencies**
- **Identify data-access patterns**
- **Formulate queries**
- **Define tables**

Created the KillrVideo `user_credentials` table

Now You Know…

Cassandra data modeling

- **Fundamental Cassandra storage constructs**
- **How data modeling differs between Cassandra and relational**
- **The Cassandra data modeling process**