AWS re:Invent

NOV. 28 – DEC. 2, 2022 | LAS VEGAS, NV

DAT201-R

Data modeling best practices with Amazon DocumentDB

Douglas Bonser

Sr. DocumentDB Specialist SA
AWS

Cody Allen

Sr. DocumentDB Specialist SA
AWS

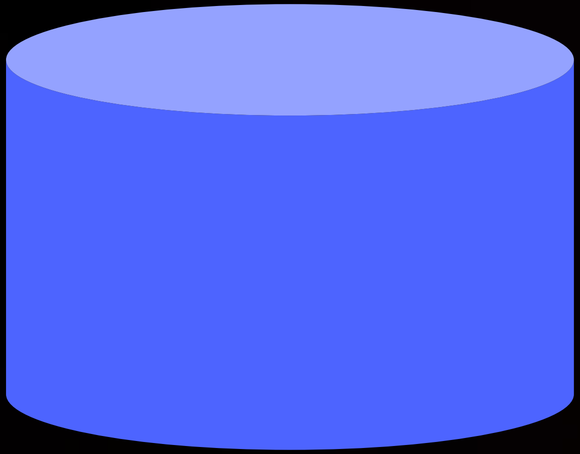
Agenda

Relational vs. NoSQL terminology/concepts

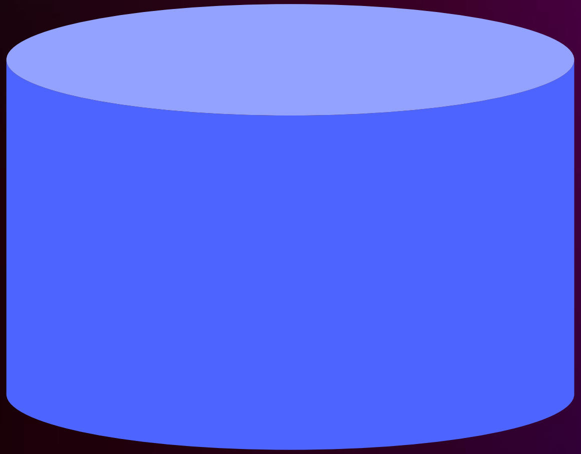
NoSQL data modeling approaches

Whiteboarding and open discussion

Relational vs. NoSQL terminology/concepts



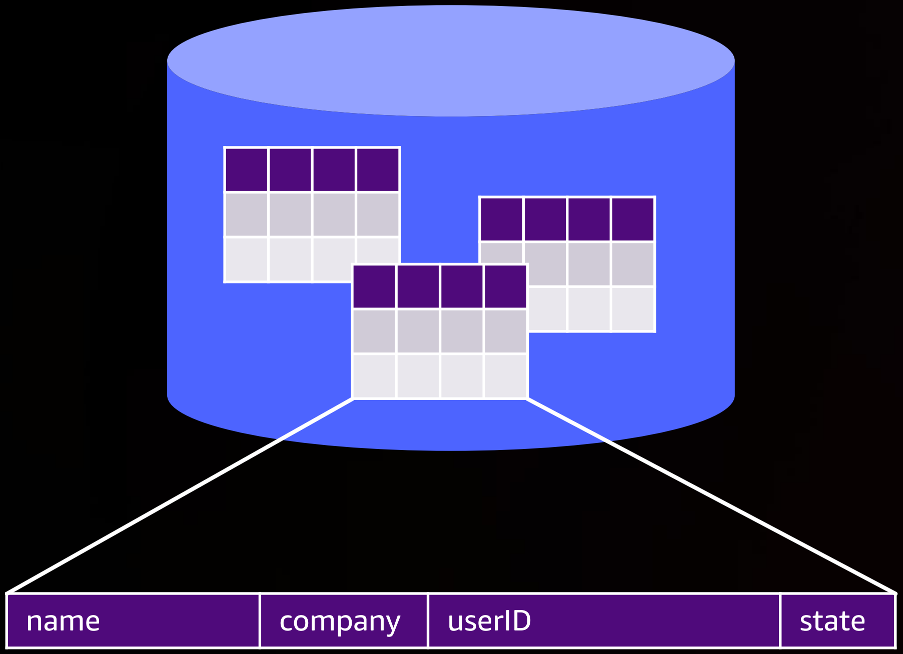
Database – Database



Relational database

Document database

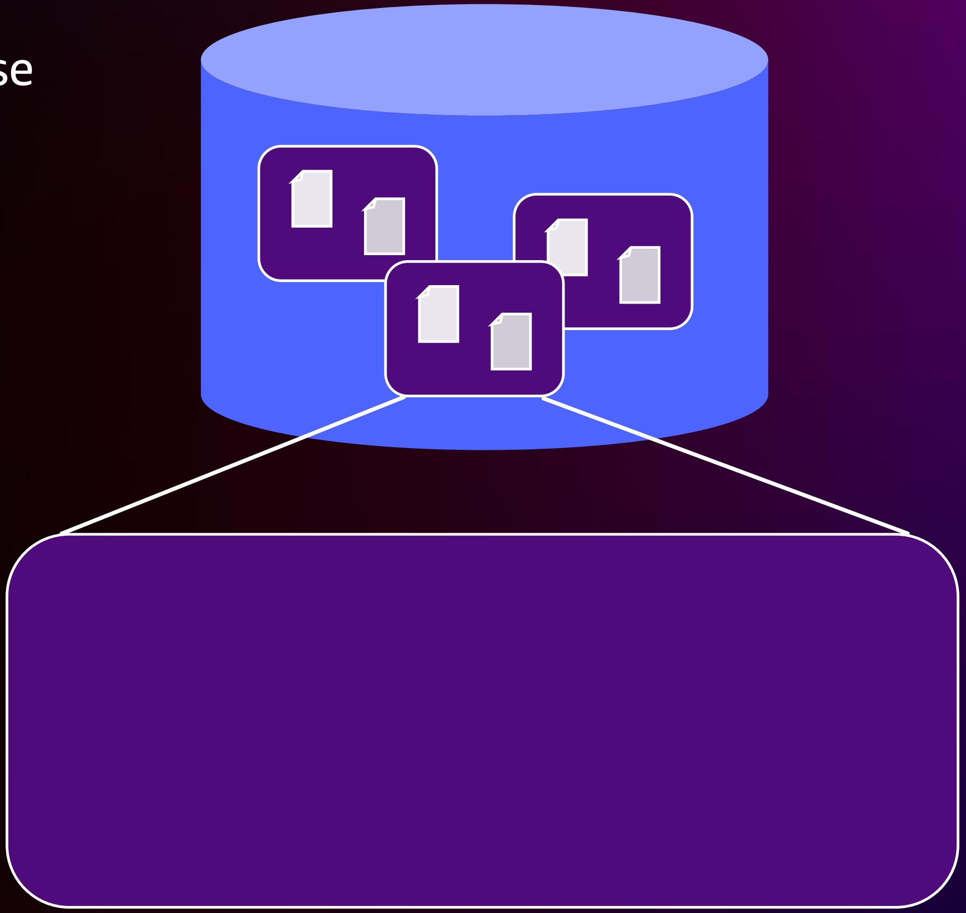
Relational vs. NoSQL terminology/concepts



name	company	userID	state

Database – Database

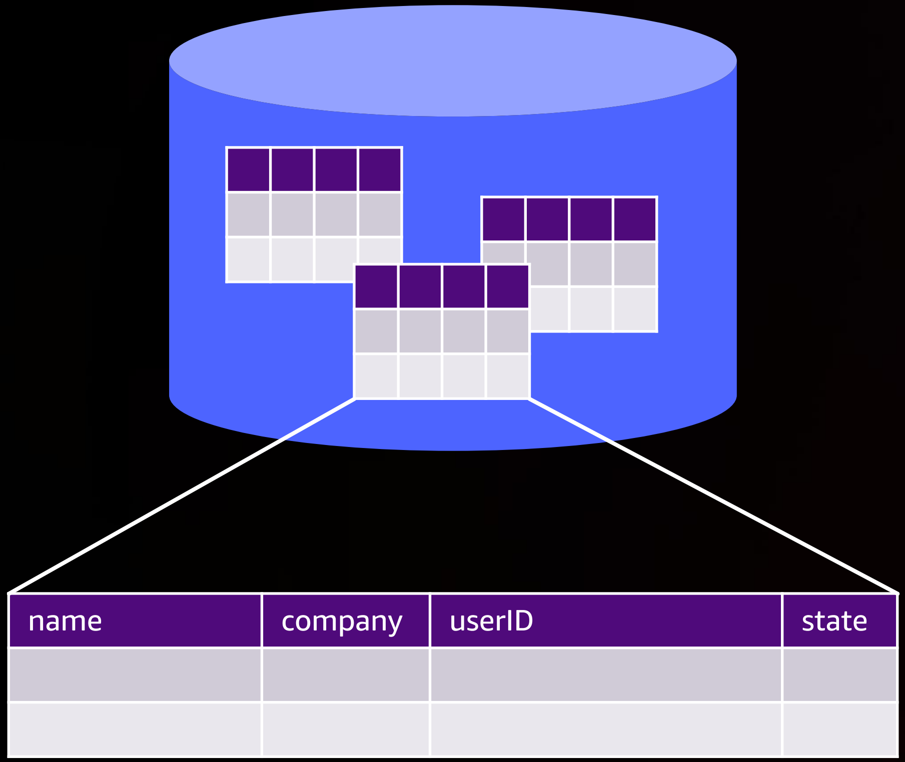
Table – Collection



Relational database

Document database

Relational vs. NoSQL terminology/concepts

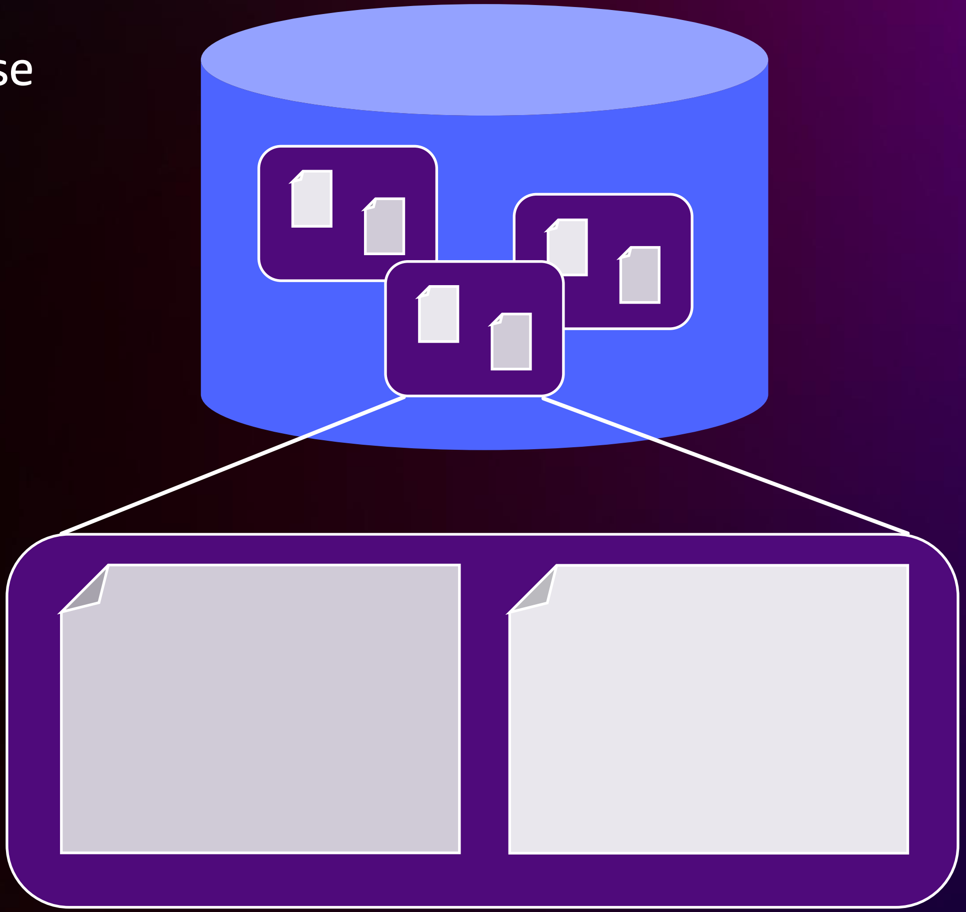


name	company	userID	state

Database – Database

Table – Collection

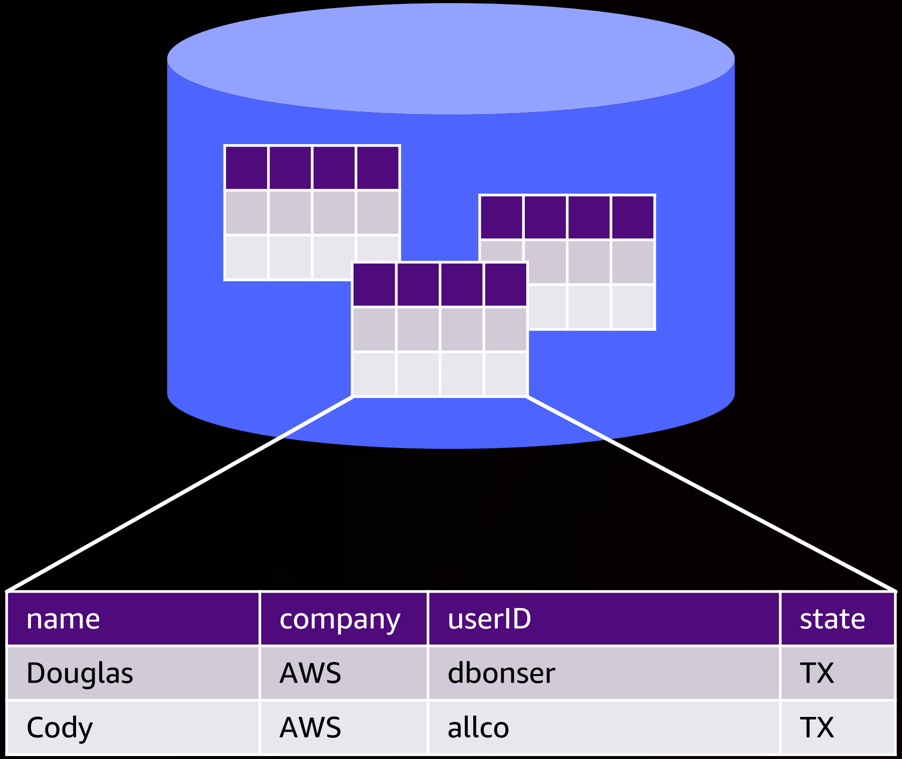
Row – Document



Relational database

Document database

Relational vs. NoSQL terminology/concepts



name	company	userID	state
Douglas	AWS	dbonser	TX
Cody	AWS	allco	TX

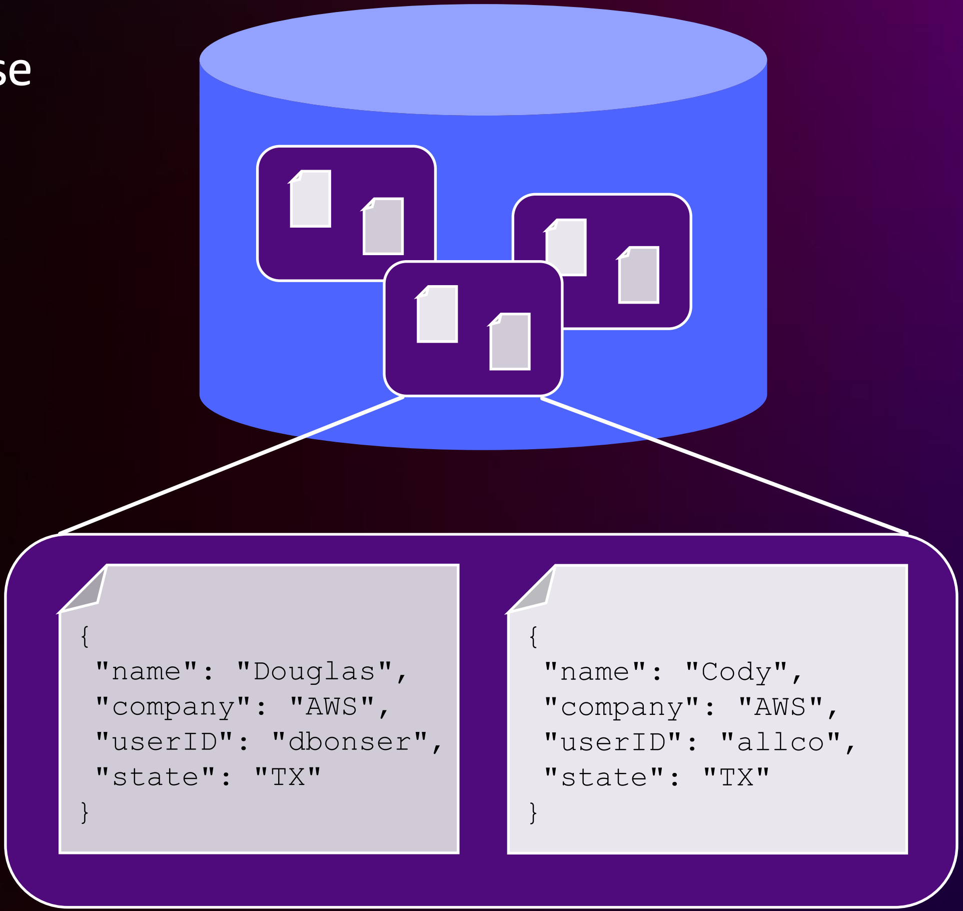
Relational database

Database – Database

Table – Collection

Row – Document

Column – Field



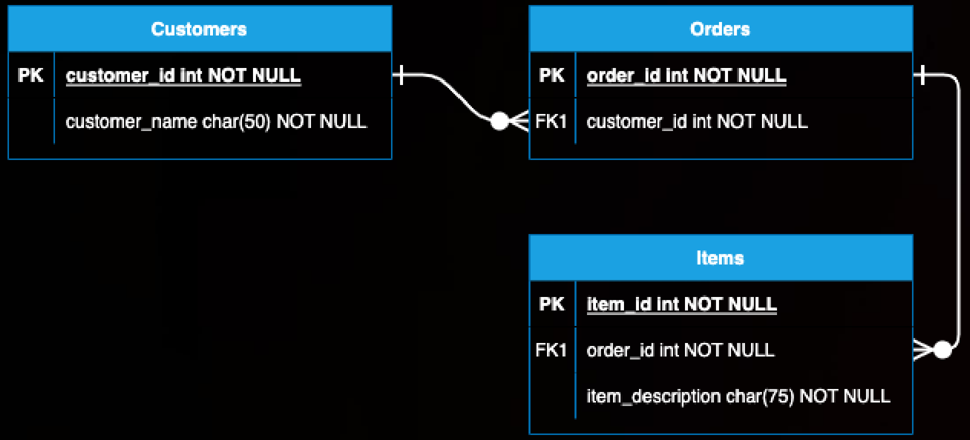
```
{
 "name": "Douglas",
 "company": "AWS",
 "userID": "dbonser",
 "state": "TX"
}
```

```
{
 "name": "Cody",
 "company": "AWS",
 "userID": "allco",
 "state": "TX"
}
```

Document database

Why is a different approach needed?

1. Define the data model



Customers	
PK	customer_id int NOT NULL
	customer_name char(50) NOT NULL

Orders	
PK	order_id int NOT NULL
FK1	customer_id int NOT NULL

Items	
PK	item_id int NOT NULL
FK1	order_id int NOT NULL
	item_description char(75) NOT NULL

2. Write data

```
INSERT INTO Customers(customer_id, customer_name)
  VALUES("181276", "Mary Major")
INSERT INTO Orders(order_id, customer_id)
  VALUES("87652", "181276")
INSERT INTO Items(item_id, order_id, item_description)
  VALUES("00157-383", "87652", "chair")
INSERT INTO Items(item_id, order_id, item_description)
  VALUES("ad14-3372", "87652", "table")
```

Relational database

Why is a different approach needed?

3. Read data

```
SELECT Customers.customer_name, Items.item_id
FROM Customers
JOIN Orders
  ON Orders.customer_id = Customers.customer_id
JOIN Items
  ON Orders.order_id = Items.order_id
WHERE Customers.customer_id = "181276"
  AND Orders.order_id = "87652"
```

4. Convert result to application model

customer_name	item_id
Mary Major	00157-383
Mary Major	ad14-3372

```
{
  "customer_name": "Mary Major",
  "items": [
    { item_id: "00157-383" },
    { item_id: "ad14-3372" },
  ]
}
```

Relational database

Why is a different approach needed?

1. Write data

```
db.orders.insertOne(
{
  "customer_id": "181276",
  "customer_name": "Mary Major",
  "order_id": "87652",
  "items": [
    {
      "item_id": "00157-383",
      "item_description": "chair"
    },
    {
      "item_id": "ad14-3372",
      "item_description": "table"
    }
  ]
})
```

2. Read data

```
db.orders.find(
  {
    customer_id: "181276",
    order_id: "87652"
  },
  {
    customer_name: 1,
    "items.item_id": 1,
    _id: 0
  }
)
```

3. Use result as is

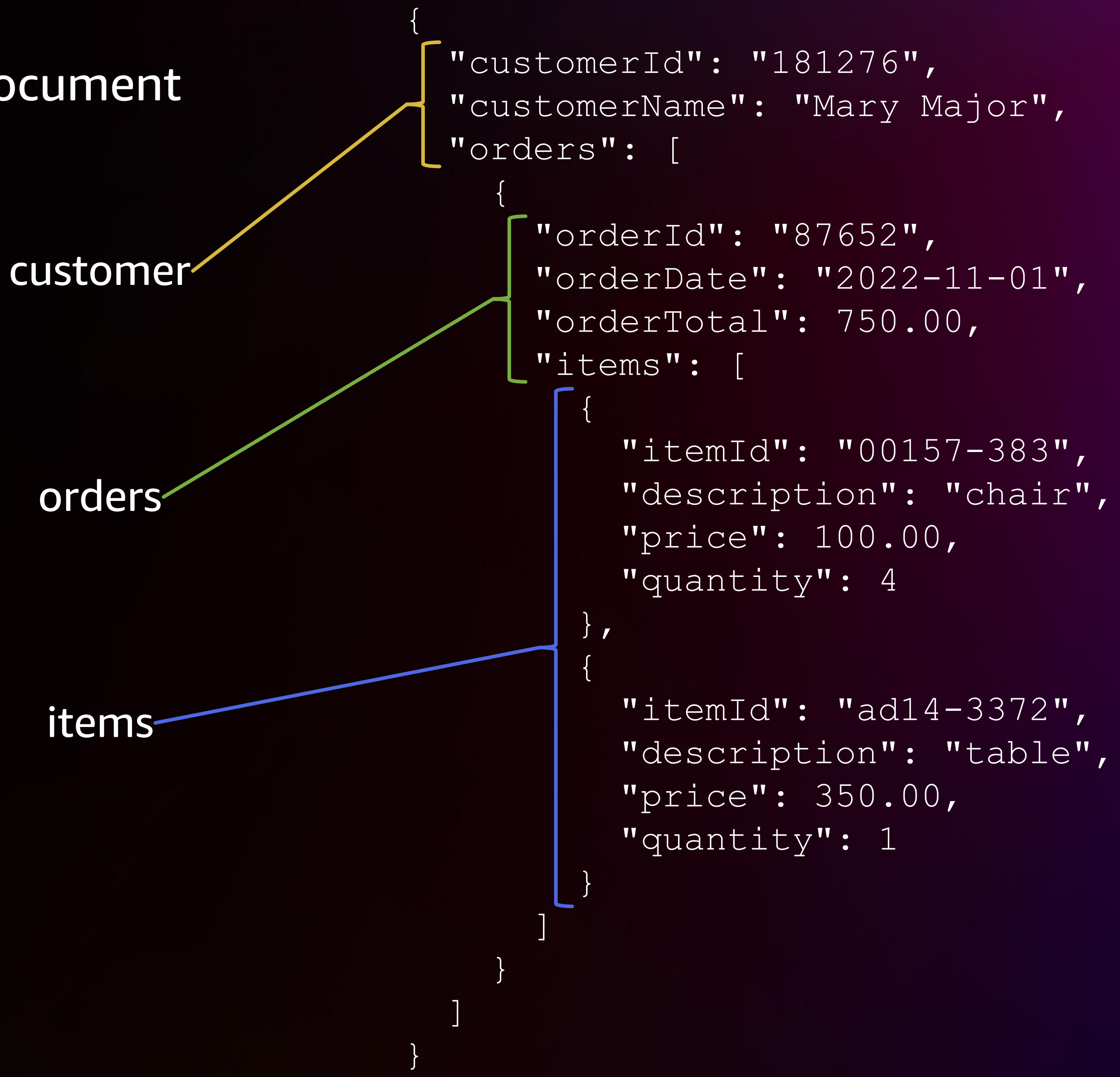
```
{
  "customer_name": "Mary Major",
  "items": [
    { item_id: "00157-383" },
    { item_id: "ad14-3372" },
  ]
}
```

Document database

NoSQL data modeling approaches

Embedding

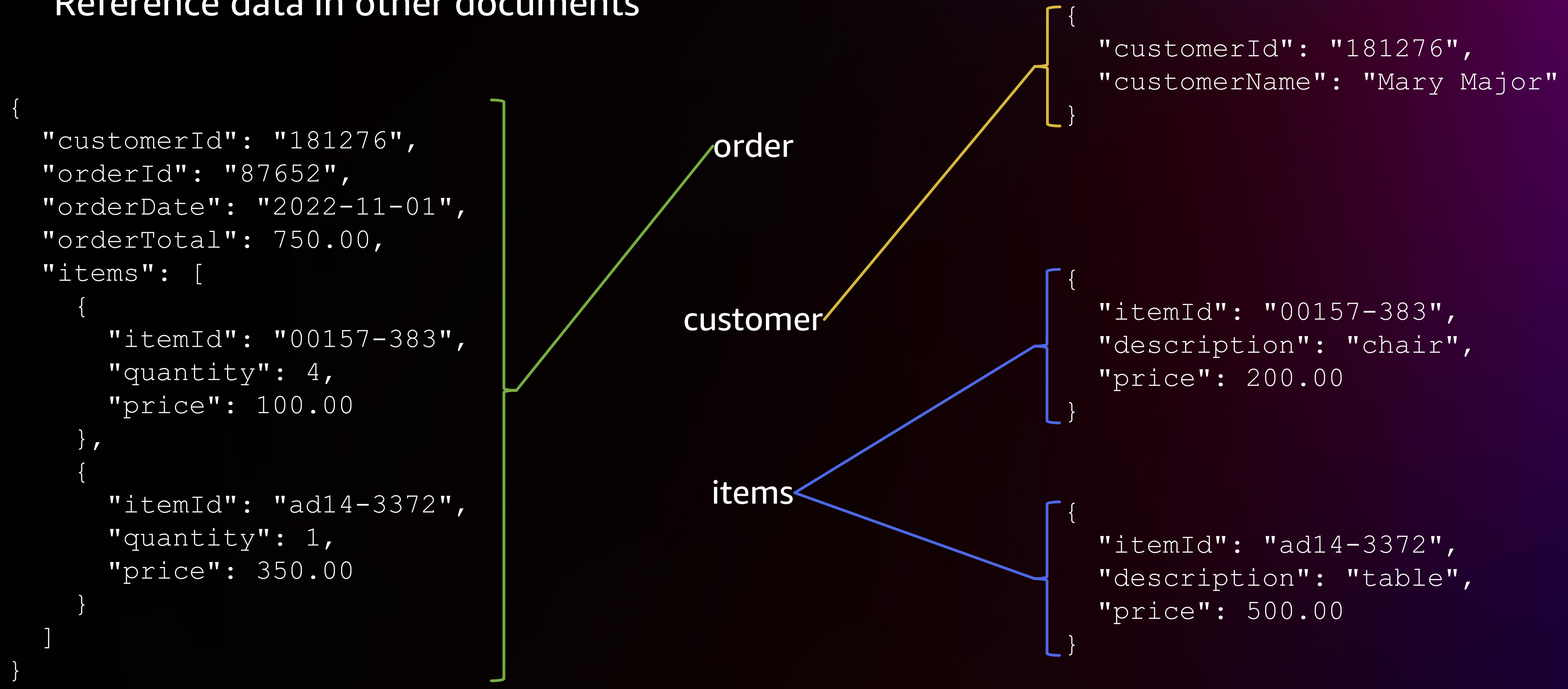
Embed related data in a single document



```json
{
  "customerId": "181276",
  "customerName": "Mary Major",
  "orders": [
    {
      "orderId": "87652",
      "orderDate": "2022-11-01",
      "orderTotal": 750.00,
      "items": [
        {
          "itemId": "00157-383",
          "description": "chair",
          "price": 100.00,
          "quantity": 4
        },
        {
          "itemId": "ad14-3372",
          "description": "table",
          "price": 350.00,
          "quantity": 1
        }
      ]
    }
  ]
}
```

Referencing

Reference data in other documents



```
{
  "customerId": "181276",
  "orderId": "87652",
  "orderDate": "2022-11-01",
  "orderTotal": 750.00,
  "items": [
    {
      "itemId": "00157-383",
      "quantity": 4,
      "price": 100.00
    },
    {
      "itemId": "ad14-3372",
      "quantity": 1,
      "price": 350.00
    }
  ]
}
```

```
{
  "customerId": "181276",
  "customerName": "Mary Major"
}
```

```
{
  "itemId": "00157-383",
  "description": "chair",
  "price": 200.00
}
```

```
{
  "itemId": "ad14-3372",
  "description": "table",
  "price": 500.00
}
```

Summary

- Not bound to rigid, pre-defined schemas
- Data model based on read/write access patterns & relationships
- Mix of embedding & referring
- Different models for different use cases

Whiteboarding and open discussion

Thank you!

Douglas Bonser
dbonser@amazon.com

Cody Allen
allco@amazon.com

Please complete the session survey in the **mobile app**