

GOING SCHEMA-LESS:

HOW TO MIGRATE

A RELATIONAL DATABASE

TO A NOSQL

DATABASE



Who is Chad Green

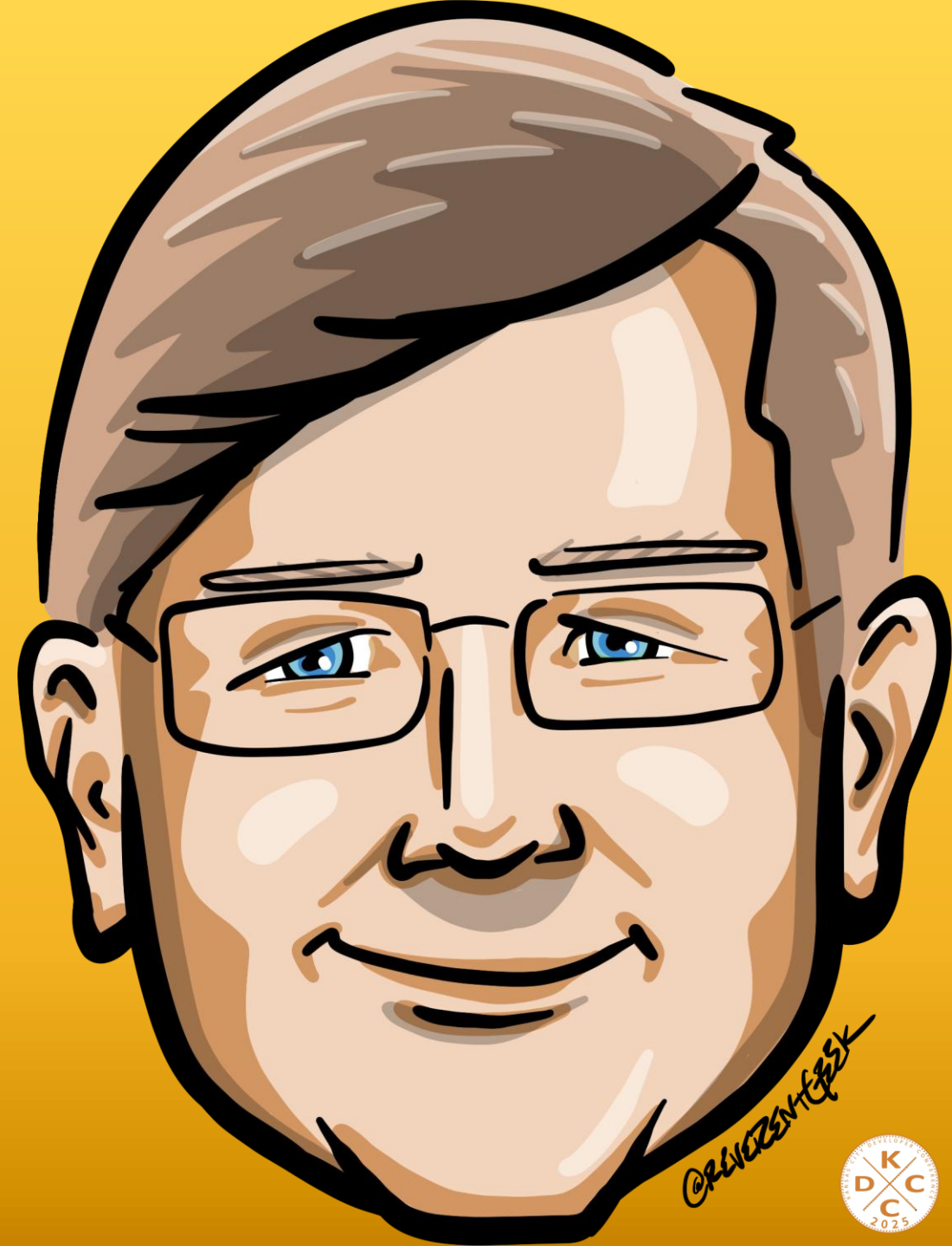
✉ chadgreen@chadgreen.com

💬 TaleLearnCode

🌐 ChadGreen.com

🐦 ChadGreen & TaleLearnCode

📌 ChadwickEGreen

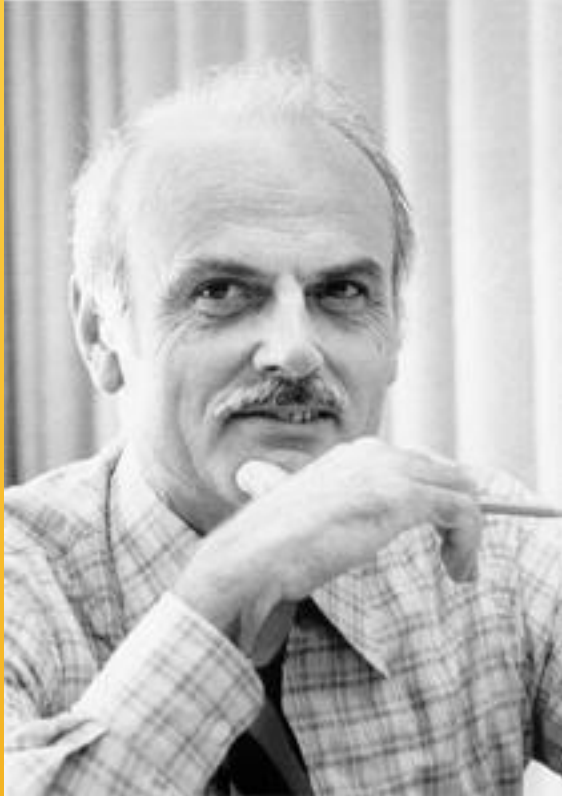




**How did I get started
with NoSQL databases?**

What are Relational Databases

Relational Model



- First-order predicate logic
- Described by Edgar Codd in 1969
- Data represented in terms of tuples
- Purpose is to provide declarative method for specifying data and queries

Codd's 12 Rules

0: Foundation Rule

1: Information Rule

2: Guaranteed Access

3: Systematic treatment of
NULL values

4: Active Online Catalog

5: Comprehensive data
sublanguage

6: View Updating

7: Possible for high-level
insert, update, and delete

8: Physical data
independence

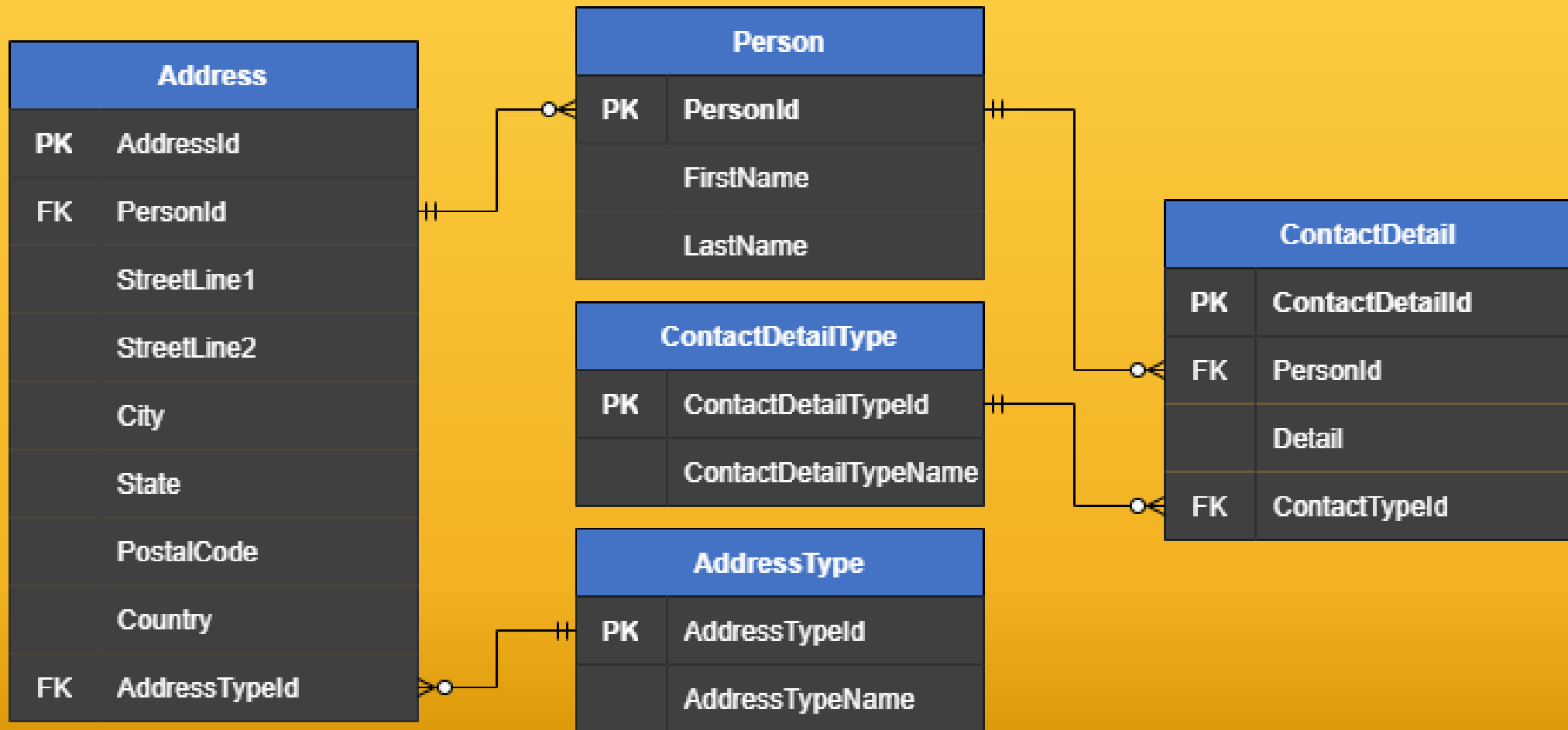
9: Logic data independence

10: Integrity Independence

11: Distribution
Independence

12: Nonsubversion Rule

Typical Relational Model



True star of Relational Databases

SQL

Structured Query Language

SEQUEL

True star of Relational Databases

SQL

Structured



By Saufhn - Own work, CC BY-SA 4.0,
<https://commons.wikimedia.org/w/index.php?curid=87255205>



Big Names in Relational Databases

ORACLE®



What are NoSQL Databases

What are NoSQL Databases

Modeled in means other than tabular relations

Existed since late 1960s

Increasingly used in big data and real-time web applications

NoSQL Motivations

Simplicity of Design

**Simpler Horizontal
Scaling**

**Finer Control over
Availability**

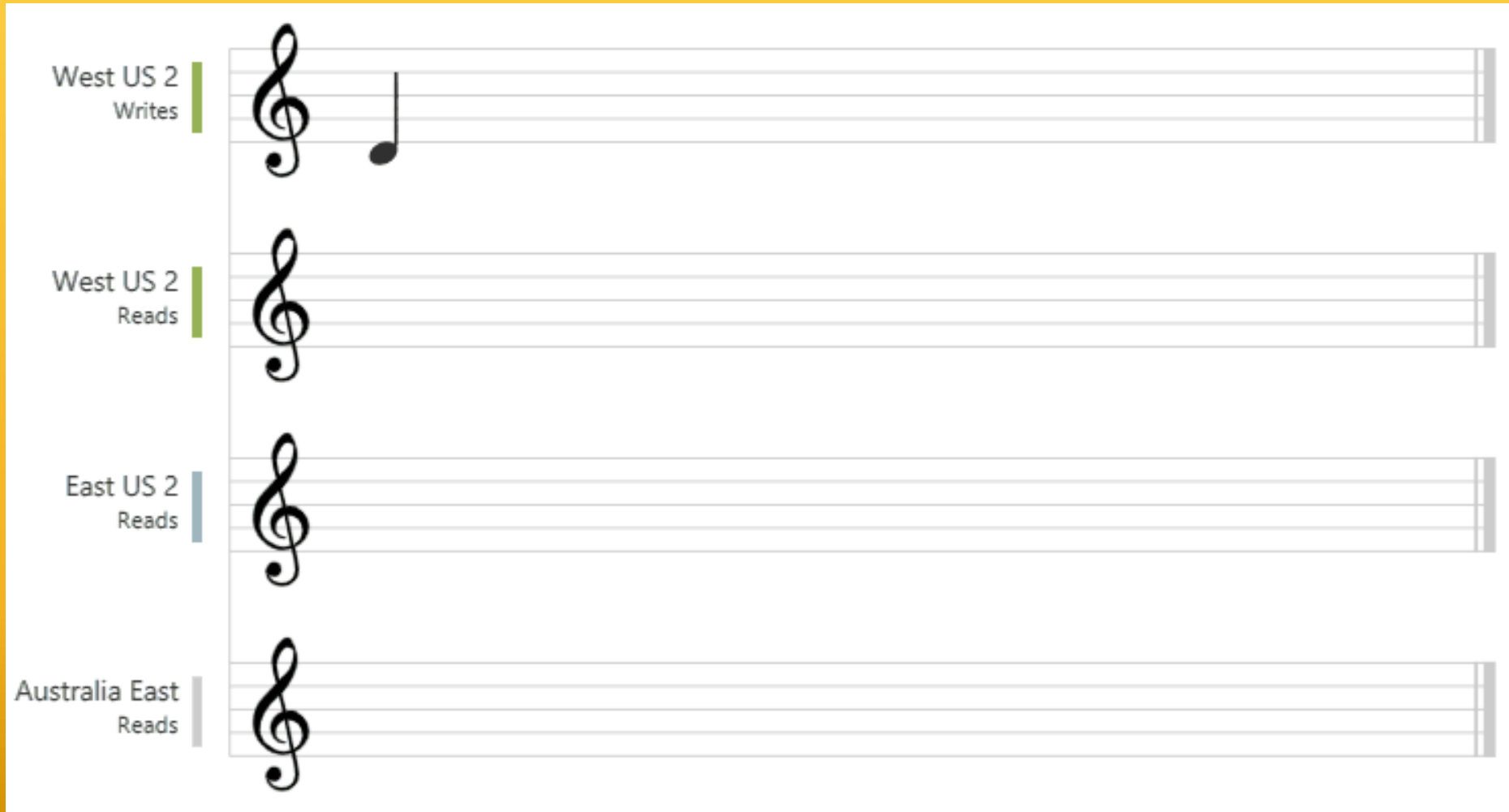
**Limiting Object-
Relational Impedance**

Availability over Consistency

Relational
ACID Transactions

NoSQL
Eventual Consistency

Eventual Consistency



What's in a Name

NoSQL

What's in a Name

NoSQL

What's in a Name

Not only SQL

Non-SQL

Non-Relational

NoSQL

What's in a Name

Not only SQL

Non-SQL

Non-Relational

NoSQL

What's in a Name

No-Schema

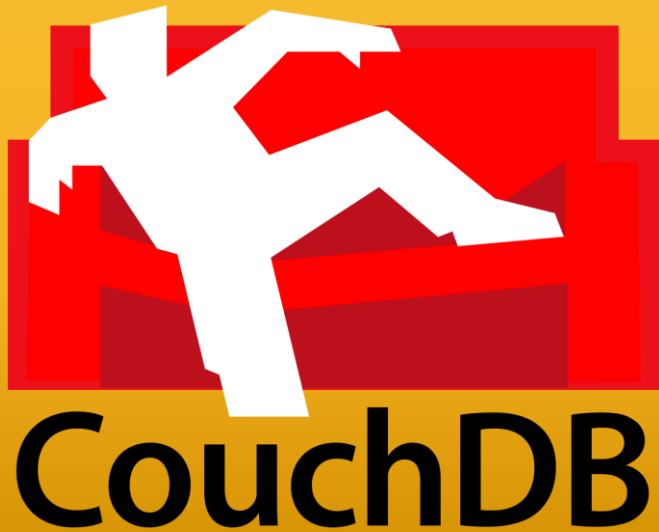
Not only SQL

Non-SQL

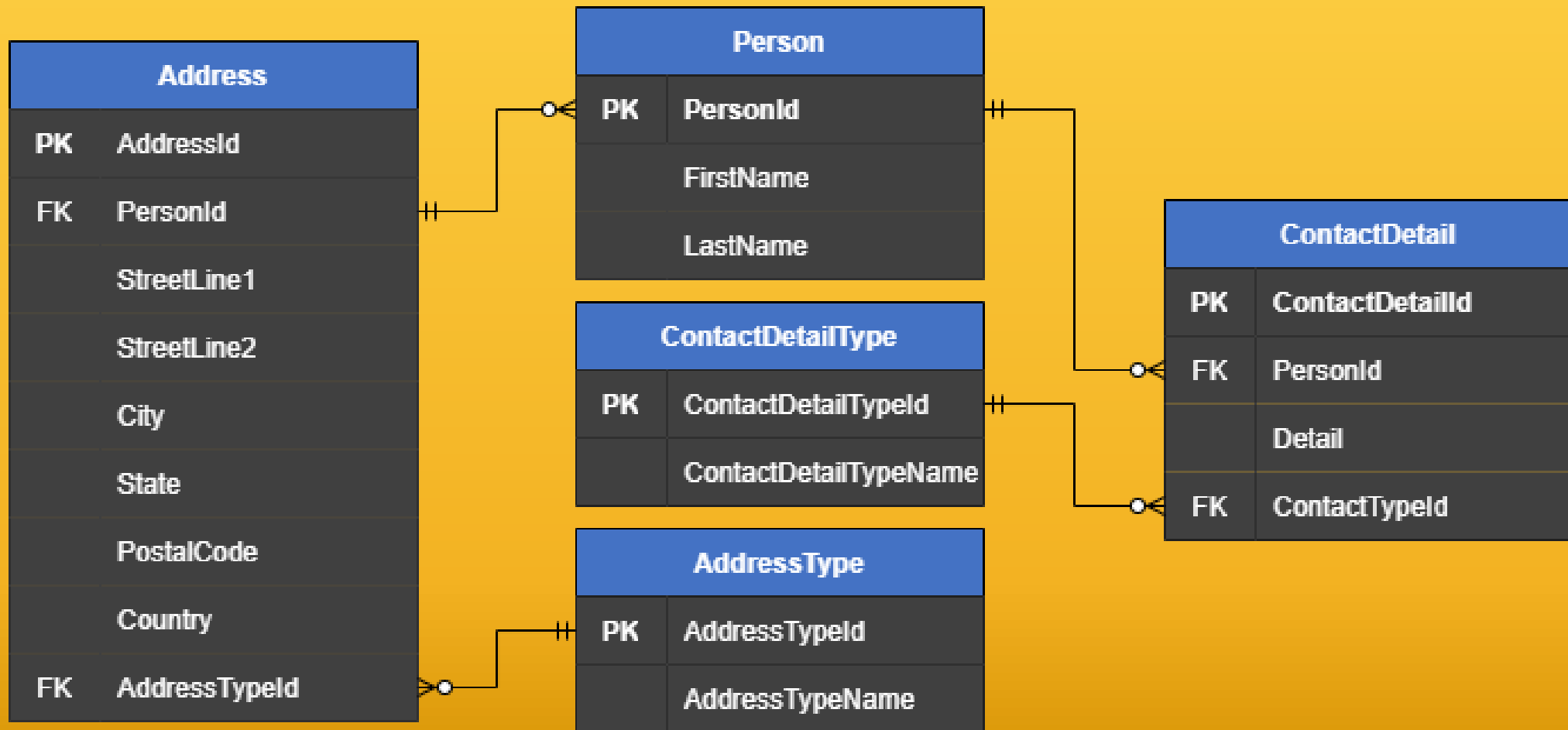
Non-Relational

NoSQL

Many types of NoSQL databases



Typical Relational Model



Same but in a document database

```
{
  "id": "1",
  "firstName": "Thomas",
  "lastName": "Andersen",
  "addresses": [
    {
      "city": "Seattle",
      "state": "WA",
      "type": {
        "name": "Primary"
      }
    }
  ],
  "contactDetails": [
    {
      "detail": "First Detail",
      "type": {
        "name": "A detail type"
      }
    }
  ]
}
```

Many types of NoSQL databases



Key-Value



redis

Many types of NoSQL databases



Wide Column



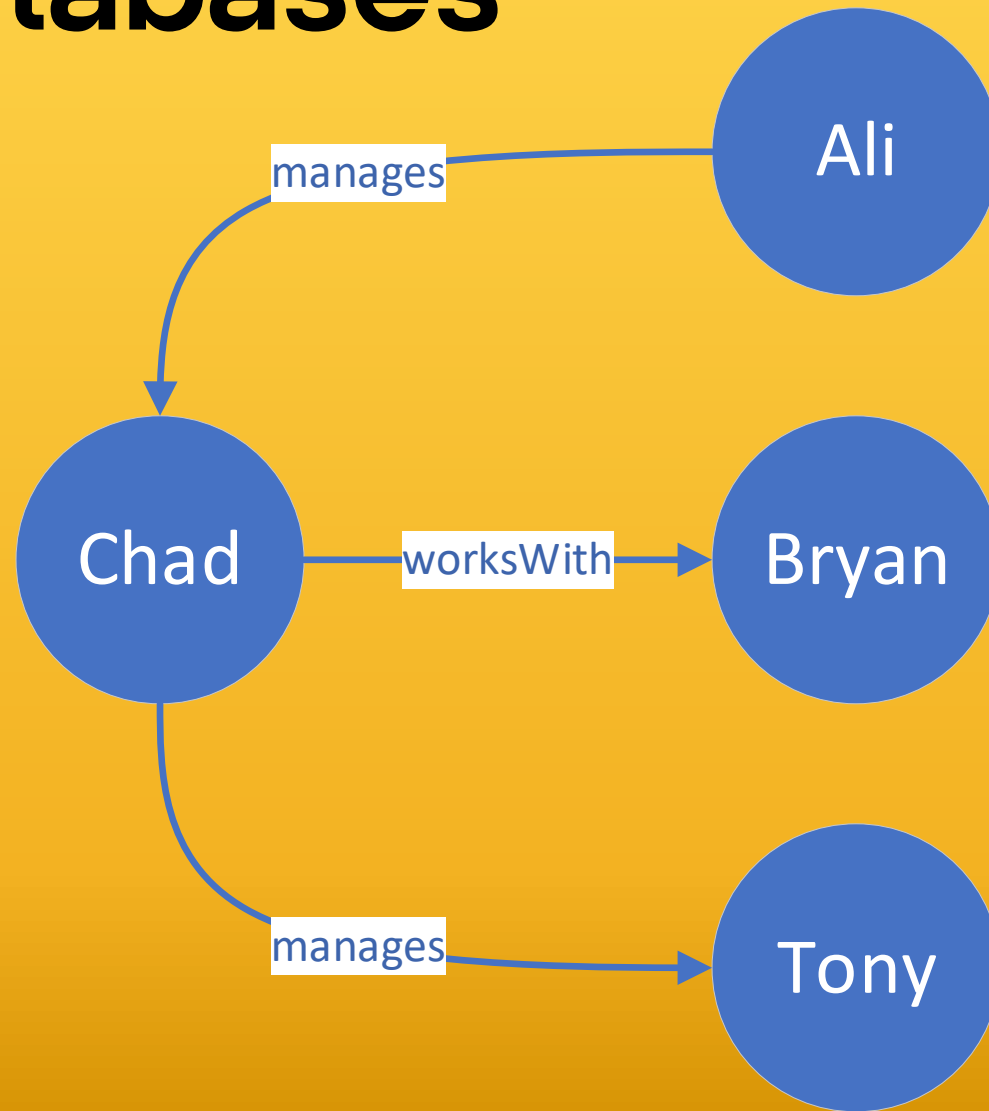
Many types of NoSQL databases



Graph



Graph Databases



Many types of NoSQL databases

Document

Key-Value

Wide Column

Graph

Object

Tabular

Tuple Store

Triple Store

Picking a Data Store

Data Model Comparison

Data Model	Performance	Scalability	Flexibility	Complexity	Functionality
Key-Value Store	High	High	High	None	Variable (None)
Column Store	High	High	Moderate	Low	Minimal
Document Store	High	Variable (High)	High	Low	Variable (Low)
Graph	Variable	Variable	High	High	Graph Theory
Relational	Variable	Variable	Low	Moderate	Relational Algebra

Ben Scofield – NoSQL presentation at CodeMash 2010



Things to think about

Skillset

Time to Market

Known Data Structure

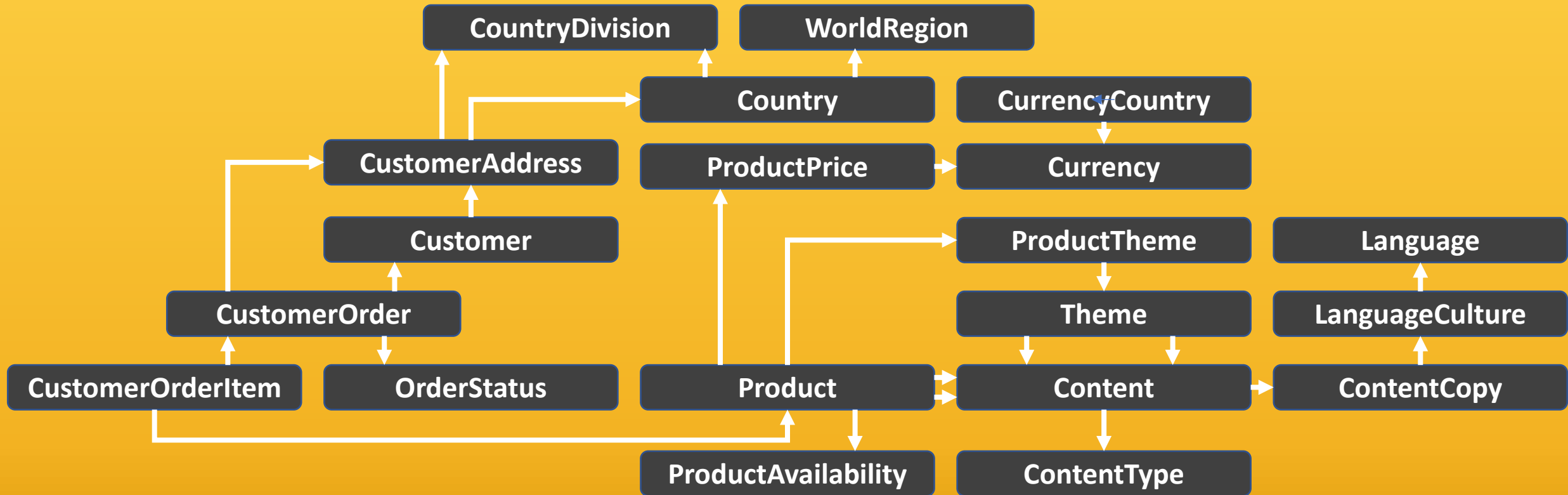
Scalability

Don't forget

Hybrid

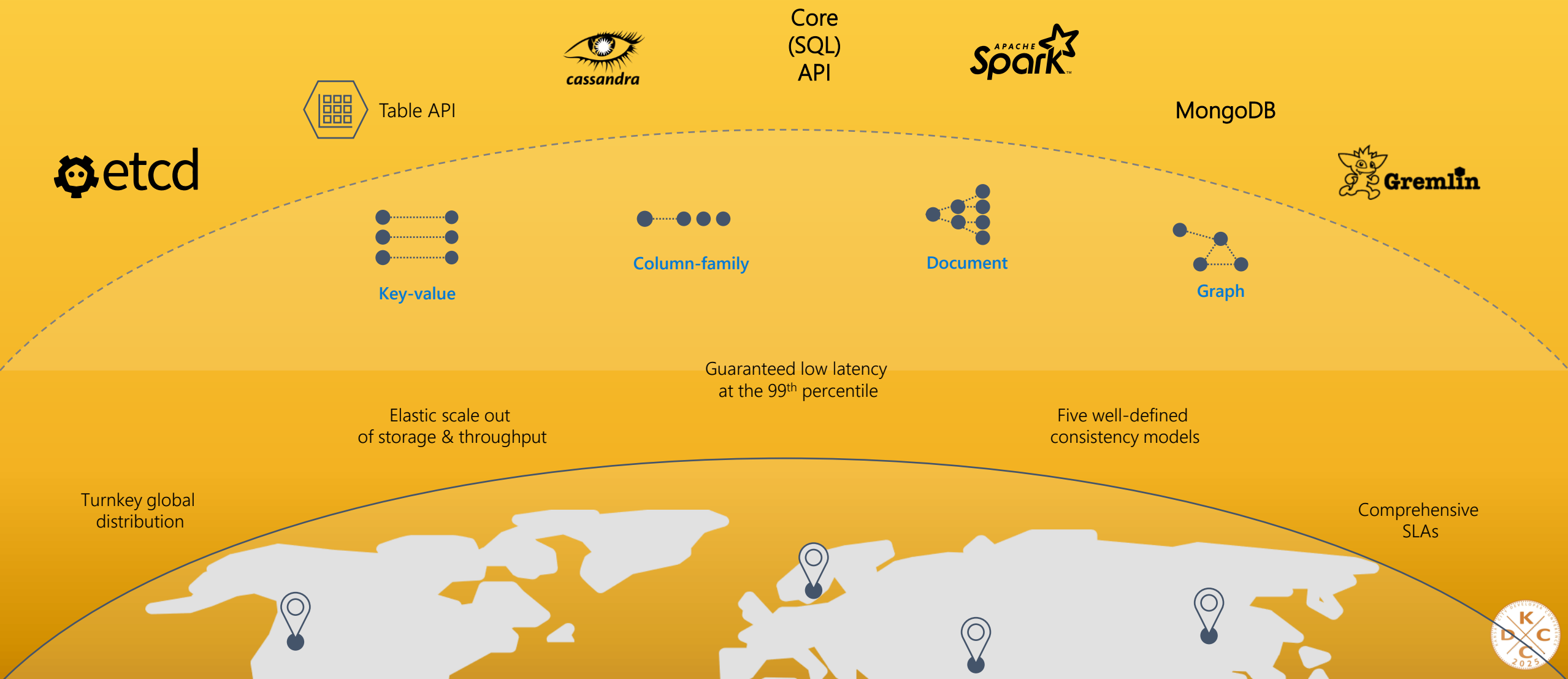
Example Explainer

Data Model



Very Quick Into to Cosmos DB

Azure Cosmos DB



Which Azure Cosmos DB Data API?

Core
(SQL)
API

Core (SQL) API

Which Azure Cosmos DB Data API?

Core
(SQL)
API



MongoDB

Which Azure Cosmos DB Data API?

Core
(SQL)
API



Table Storage

Which Azure Cosmos DB Data API?

Core
(SQL)
API



Gremlin

Which Azure Cosmos DB Data API?

Core
(SQL)
API



Cassandra

Migrating to NoSQL

Database Considerations

- Data Model/API



Database Considerations

- Data Model/API
- Document Structure

Database Considerations

- Data Model/API
- Document Structure
- Partition Key
- Access Patterns
- Even Data Distributions
- Cardinality
- Query Isolation
- Write Patterns
- Data Growth
- Familiarity with Data
- Data Relationship
- Cost Considerations
- Immutable Properties
- Data Size
- Trial and Error



Database Considerations

- Data Model/API
- Document Structure
- Partition Key
- Indexing



Database Considerations

- Data Model/API
- Document Structure
- Partition Key
- Indexing
- Query Performance



Database Considerations

- Data Model/API
- Document Structure
- Partition Key
- Indexing
- Query Performance
- Consistency Level



Database Considerations

- Data Model/API
- Document Structure
- Partition Key
- Indexing
- Query Performance
- Consistency Level
- Time-to-Live (TTL)



Database Considerations

- Data Model/API
- Document Structure
- Partition Key
- Indexing
- Query Performance
- Consistency Level
- Time-to-Live (TTL)
- Data Migration

Database Considerations

- Data Model/API
- Document Structure
- Partition Key
- Indexing
- Query Performance
- Consistency Level
- Time-to-Live (TTL)
- Data Migration
- Versioning and Evolution



Document Database Structure

Cosmos DB Account

Database

Database

Container

Container

Container

Container

Item

Item

Item

Item

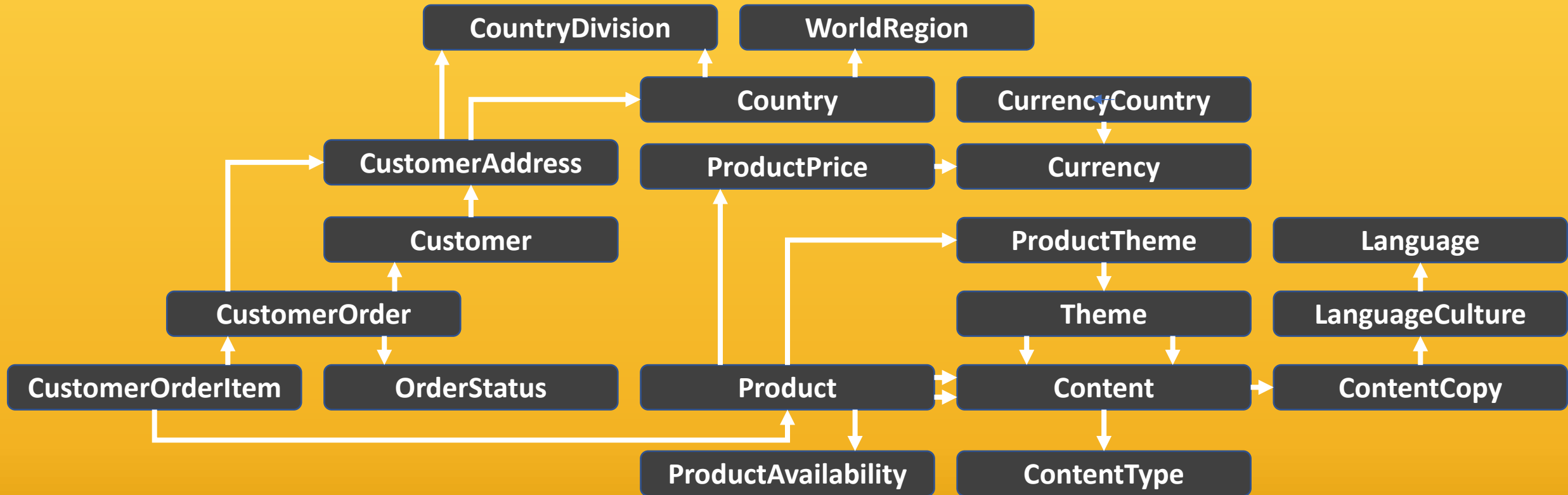
Item

Item

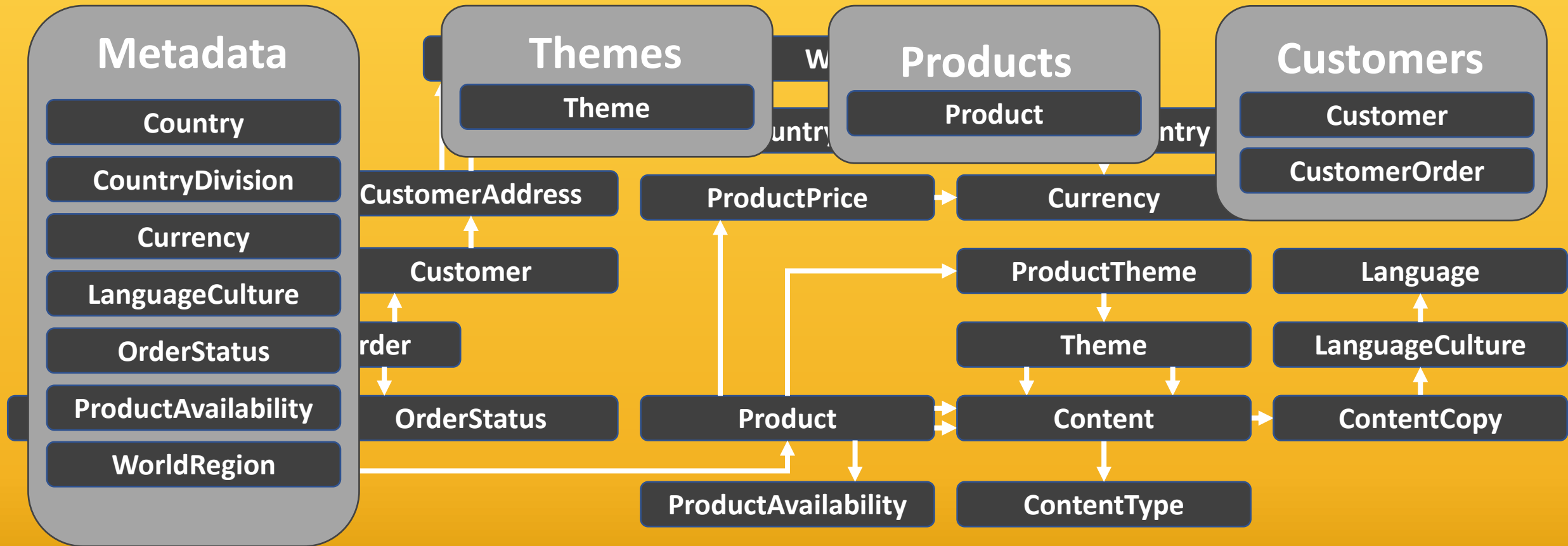
Item

Item

Data Model



Data Model



Data Model

Metadata

Country

CountryDivision

Currency

LanguageCulture

OrderStatus

ProductAvailability

WorldRegion

Themes

Theme

Products

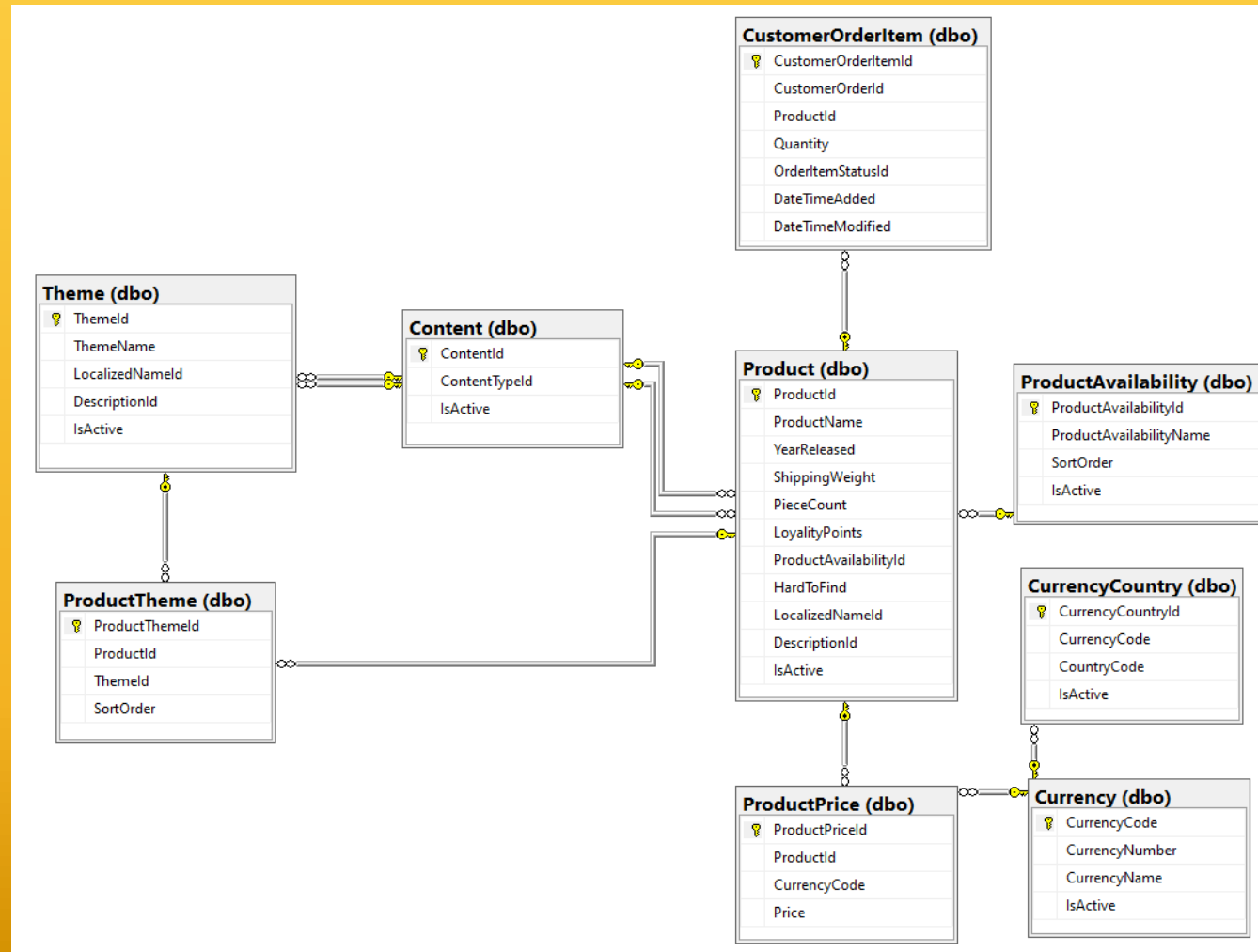
Product

Customers

Customer

CustomerOrder

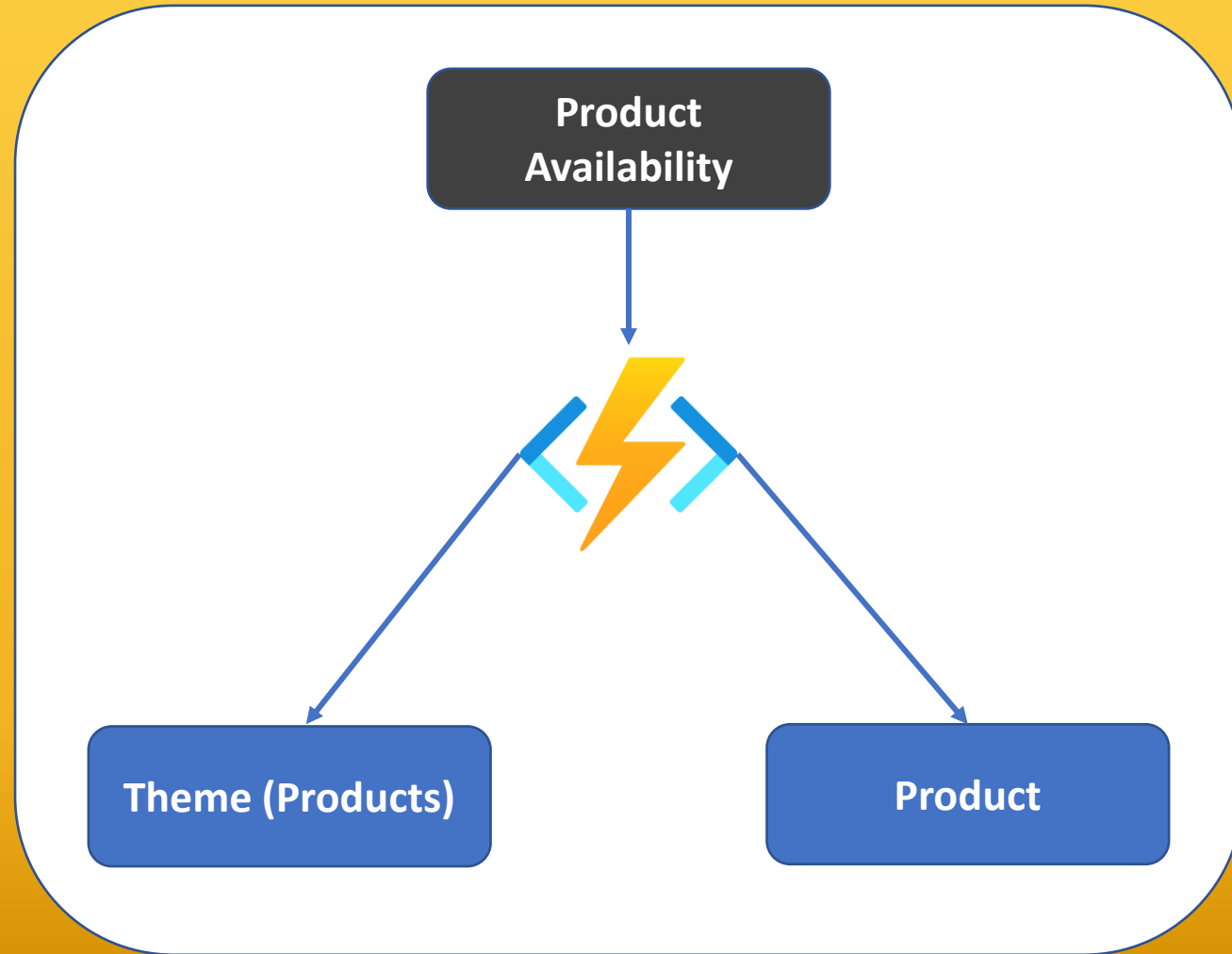
Data Model - Theme



Data Model – Theme

```
{
  "id": "43C2E388-9BB9-463A-B14B-28B538229A03",
  "name": "Architecture",
  "localizedNames": [],
  "descriptions": [],
  "products": [
    {
      "id": "21058",
      "pieceCount": 1476,
      "hardToFind": false,
      "prices": [],
      "localizedNames": [],
      "description": []
    },
    ...
  ]
}
```

Reference Types



Code Examples

Best Tool(s) for the Job

Thank You

✉ chadgreen@chadgreen.com

💬 TaleLearnCode

🌐 ChadGreen.com

🐦 ChadGreen & TaleLearnCode

🌐 ChadwickEGreen

