GOING SCHEMA-LESS:
HOW TO MIGRATE A RELATIONAL DATABASE TO A NOSQL DATABASE


10
THAT®
CONFERENCE

THANK YOU, THAT CONFERENCE PARTNERS!

CUNA MUTUAL GROUP

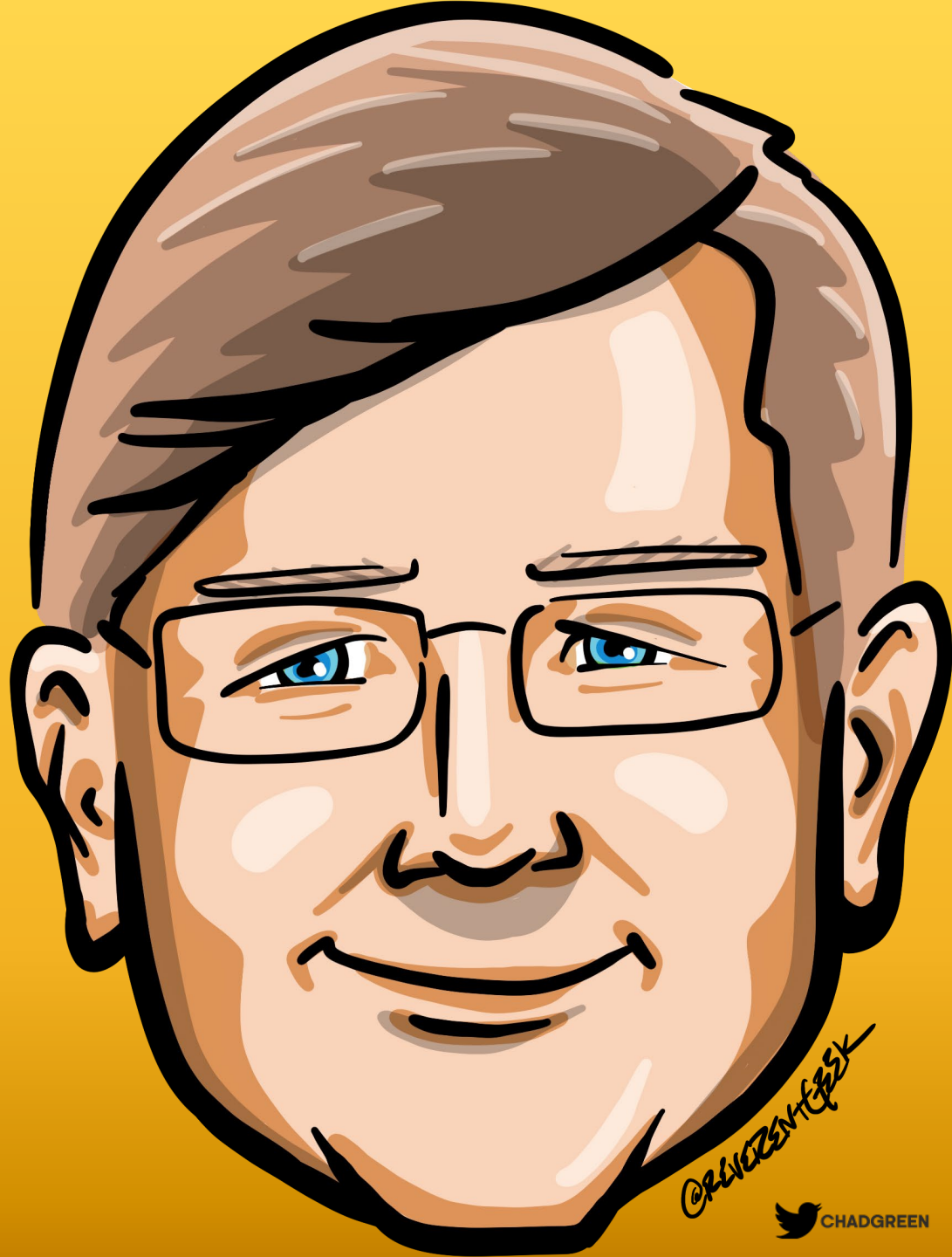
Unspecified SOFTWARE CO

NVISIA

KBX

Who is Chad Green

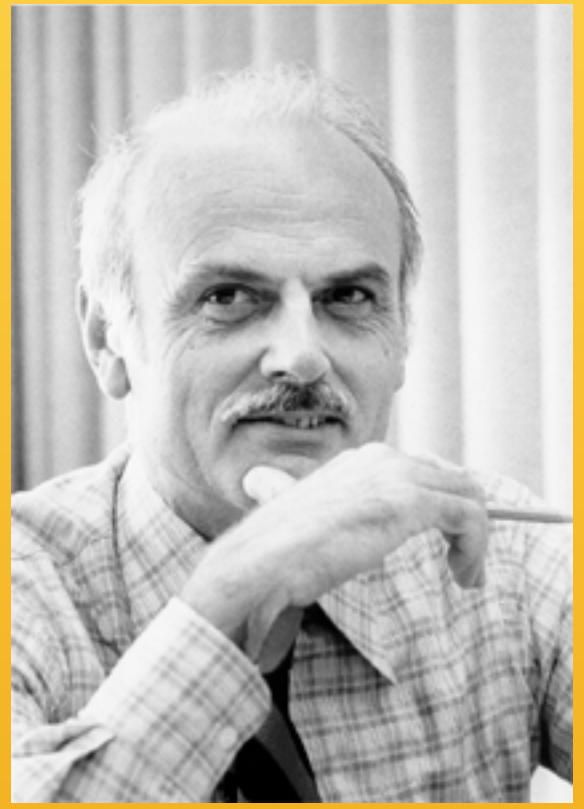
- chadgreen@chadgreen.com
- TaleLearnCode
- ChadGreen.com
- ChadGreen & TaleLearnCode
- ChadwickEGreen




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

Codd's 12 Rules

0: Foundation Rule

1: Information Rule

Codd's 12 Rules

0: Foundation Rule

1: Information Rule

2: Guaranteed Access

Codd's 12 Rules

0: Foundation Rule

1: Information Rule

2: Guaranteed Access

3: Systematic treatment of null values

Codd's 12 Rules

0: Foundation Rule

1: Information Rule

2: Guaranteed Access

3: Systematic treatment of null values

4: Dynamic online catalog based on relational model

Codd's 12 Rules

0: Foundation Rule

1: Information Rule

2: Guaranteed Access

3: Systematic treatment of null values

4: Dynamic catalog based on relational model

5: Comprehensive data sublanguage

Codd's 12 Rules

- 0: Foundation Rule
- 1: Information Rule
- 2: Guaranteed Access
- 3: Systematic treatment of null values
- 4: Dynamic online catalog based on relational model
- 5: Comprehensive data sublanguage
- 6: View Updating

Codd's 12 Rules

- 0: Foundation Rule
- 1: Information Rule
- 2: Guaranteed Access
- 3: Systematic treatment of null values
- 4: Dynamic online catalog based on relational model
- 5: Comprehensive data sublanguage
- 6: View Updating
- 7: Possible for high-level insert, update, and delete

Codd's 12 Rules

- 0: Foundation Rule
- 1: Information Rule
- 2: Guaranteed Access
- 3: Systematic treatment of null values
- 4: Dynamic online catalog based on relational model
- 5: Comprehensive data sublanguage
- 6: View Updating
- 7: Possible for high-level insert, update, and delete
- 8: Physical data independence

Codd's 12 Rules

- 0: Foundation Rule
- 1: Information Rule
- 2: Guaranteed Access
- 3: Systematic treatment of null values
- 4: Dynamic online catalog based on relational model
- 5: Comprehensive data sublanguage
- 6: View Updating
- 7: Possible for high-level insert, update, and delete
- 8: Physical data independence
- 9: Logic data independence

Codd's 12 Rules

0: Foundation Rule

1: Information Rule

2: Guaranteed Access

3: Systematic treatment of null values

4: Dynamic online catalog based on relational model

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

Codd's 12 Rules

- 0: Foundation Rule
- 1: Information Rule
- 2: Guaranteed Access
- 3: Systematic treatment of null values
- 4: Dynamic online catalog based on relational model
- 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

Codd's 12 Rules

0: Foundation Rule

1: Information Rule

2: Guaranteed Access

3: Systematic treatment of null values

4: Dynamic online catalog based on relational model

5: Comprehensive data sublanguage

6: View Updating

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

8: Physical data independence

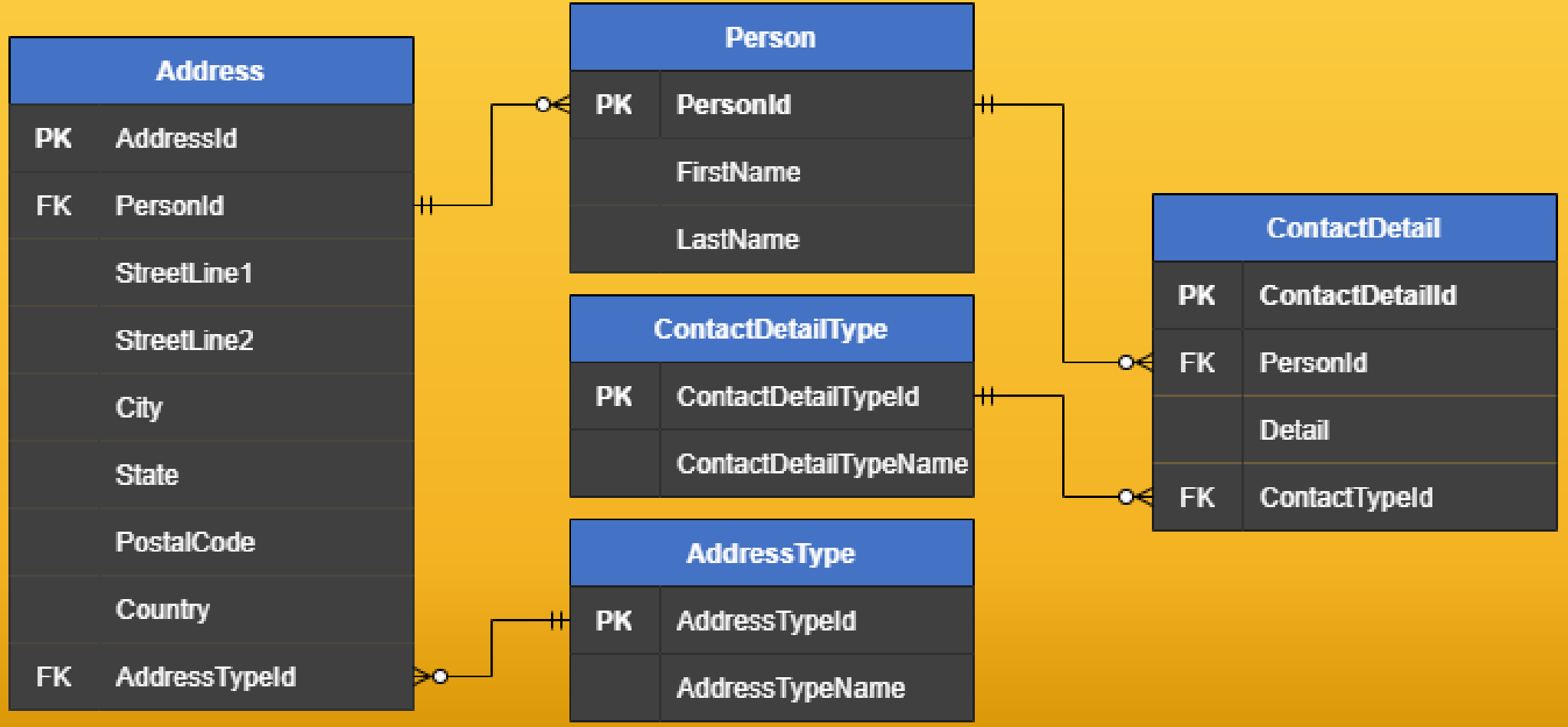
9: Logic data independence

10: Integrity Independence

11: Distribution Independence

12: Nonsubversion Rule

Typical Relational Model



True star of Relational Databases

Structured Query Language

SEQUEL

True star of Relational Databases

SQ

Structured

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

Big Names in Relational Databases

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

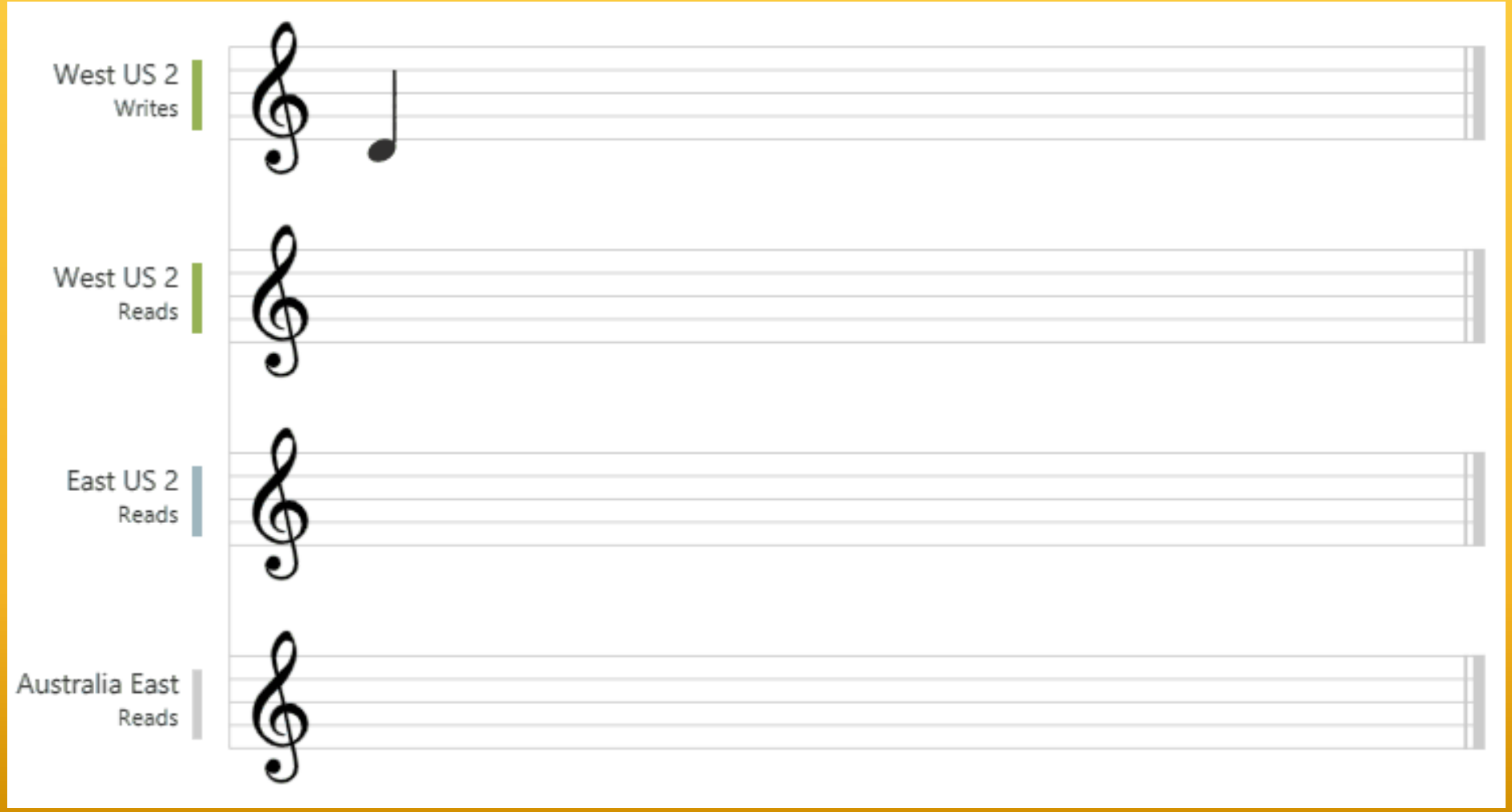
- Simplicity of Design
- Simpler Horizontal Scaling
- Finer Control over Availability
- Limiting Object-Relational Impedance

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

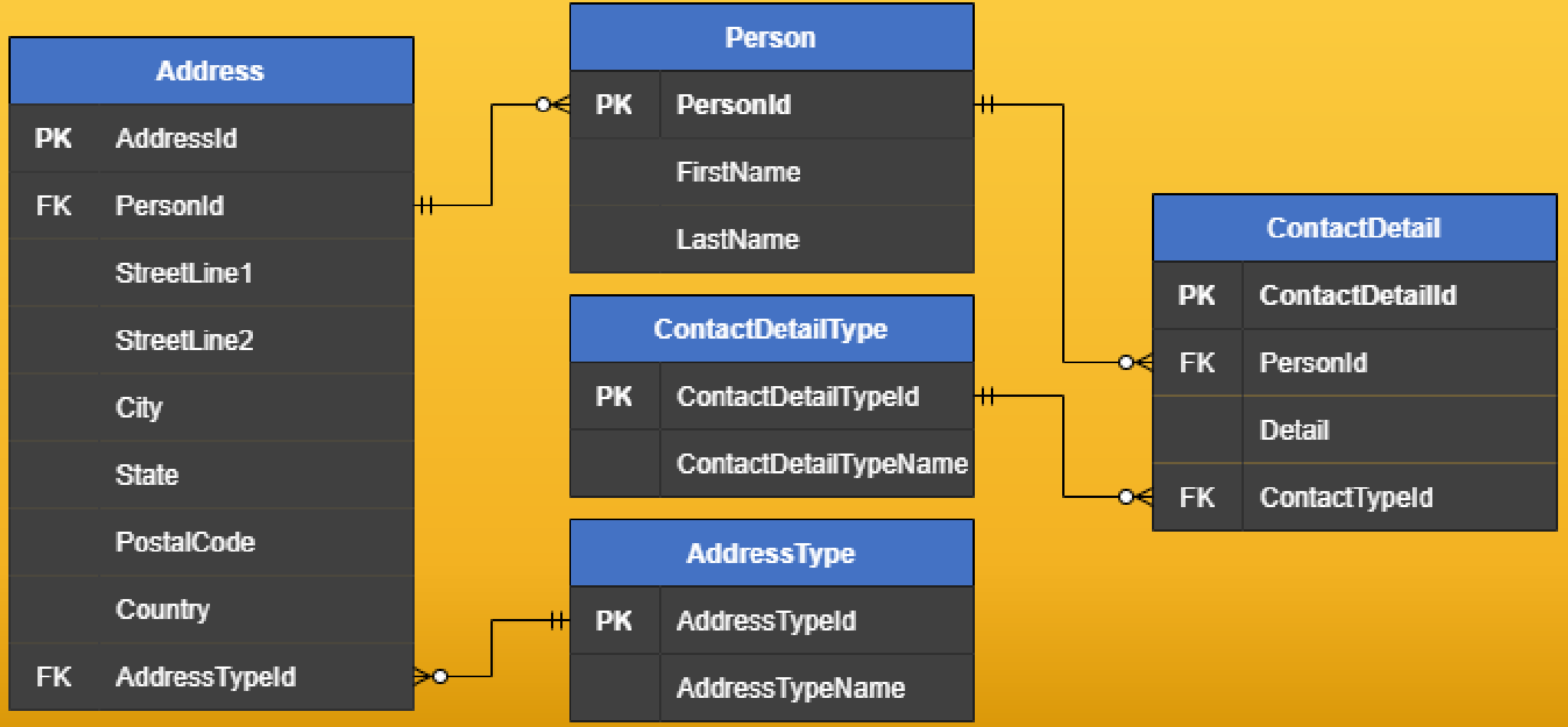
Couchbase

Document

CouchDB

ArangoDB

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

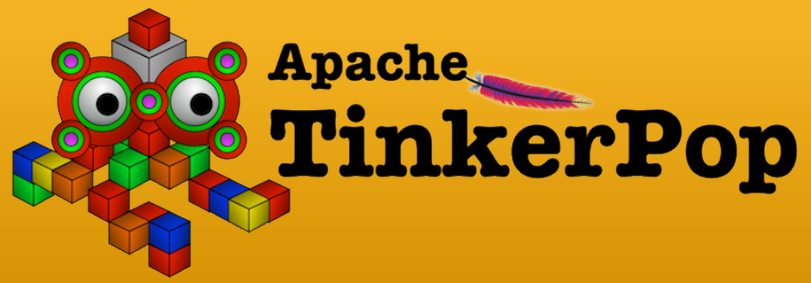
Key-Value

Many types of NoSQL databases

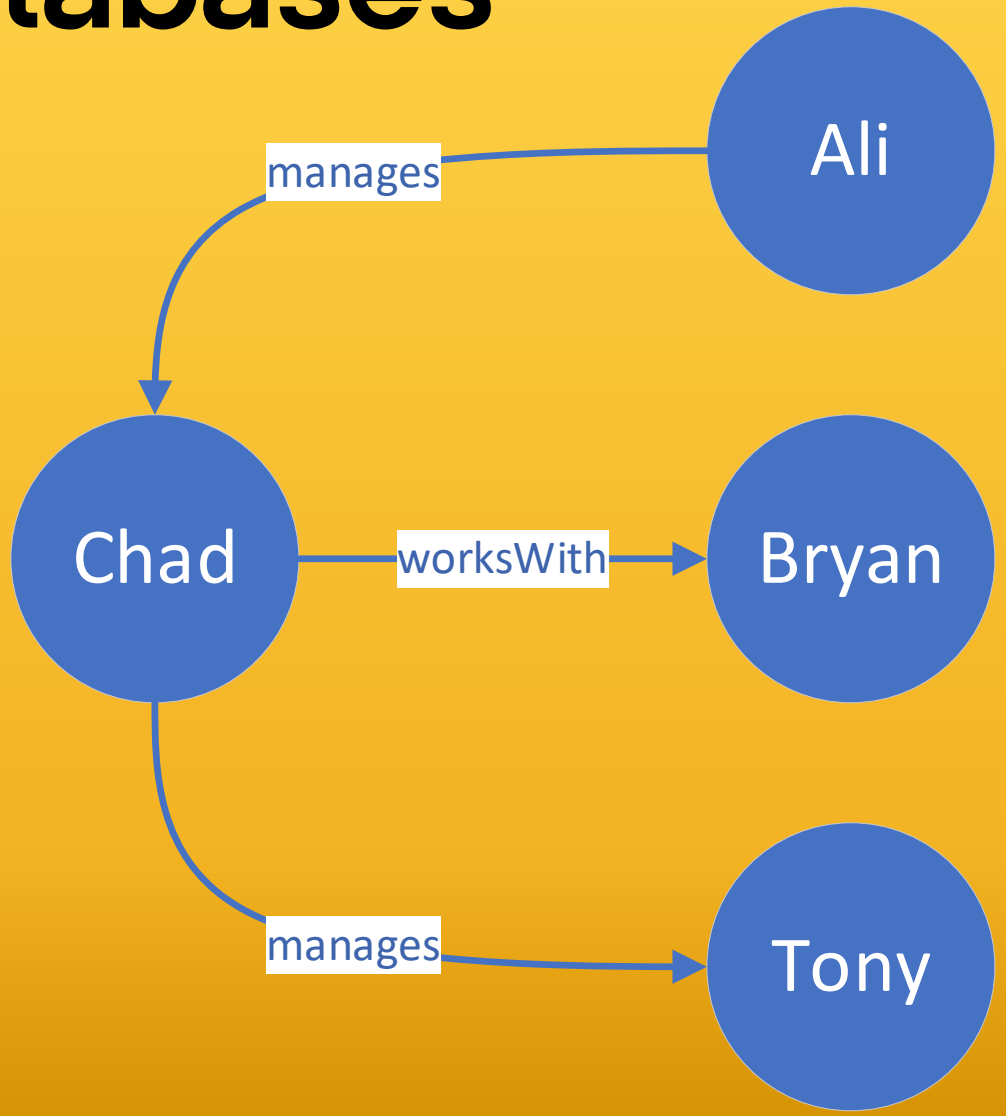
Wide Column

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

Things to think about

Skillset

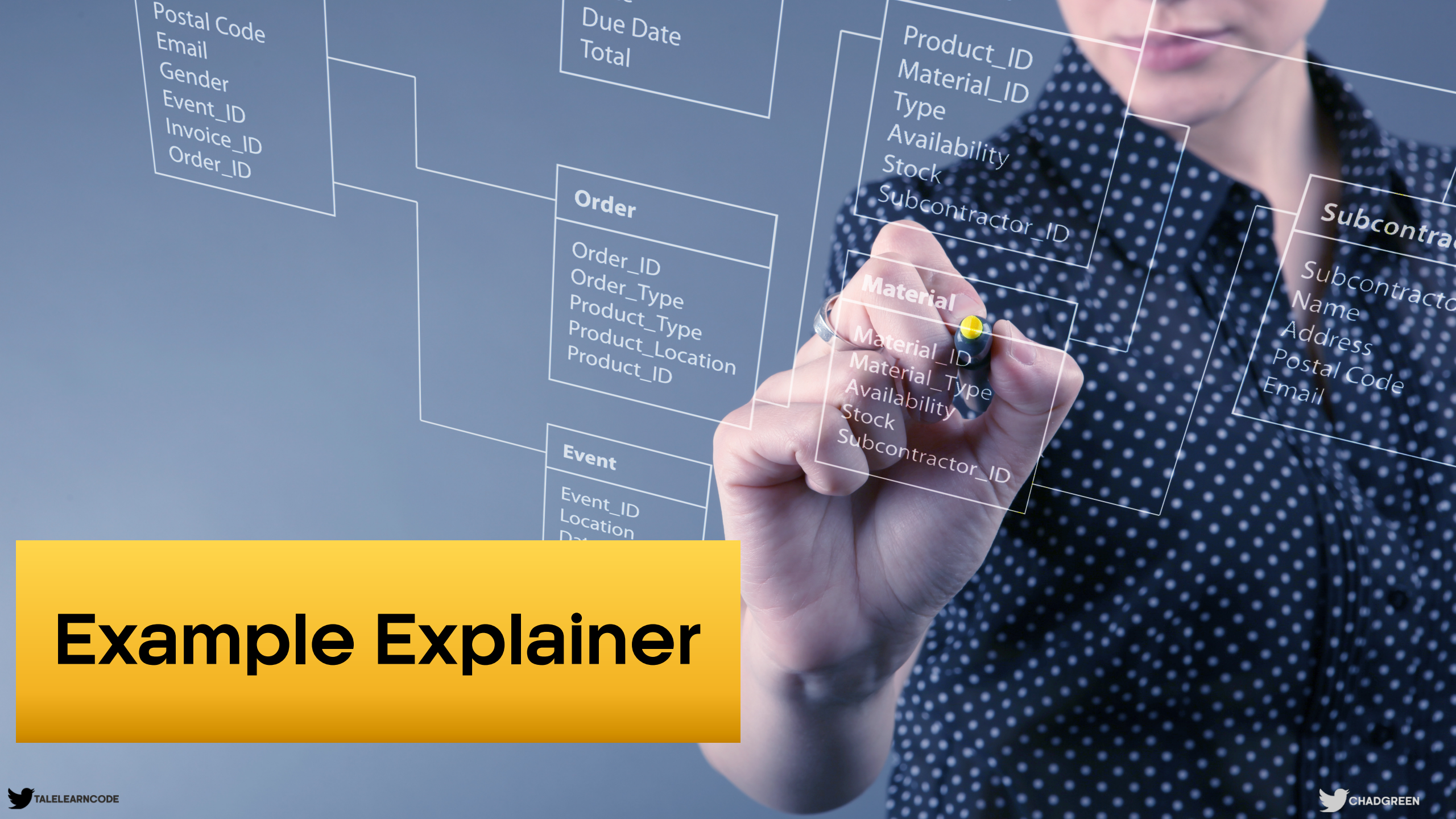
Time to Market

Known Data Structure

Scalability

Don’t forget

Hybrid



Example Explainer

Based on Real-World Project

Product & Pricing Management (PPM)

Vacation Rental Listing

- Allow property owners to list their vacation rentals
- Allow vacationers the ability to search for vacation rentals
- Provide vacationers with details of the properties
- Allow for configurable property/room attributes
- Localized versions of all the information

Data Model

User Accounts

Attributes

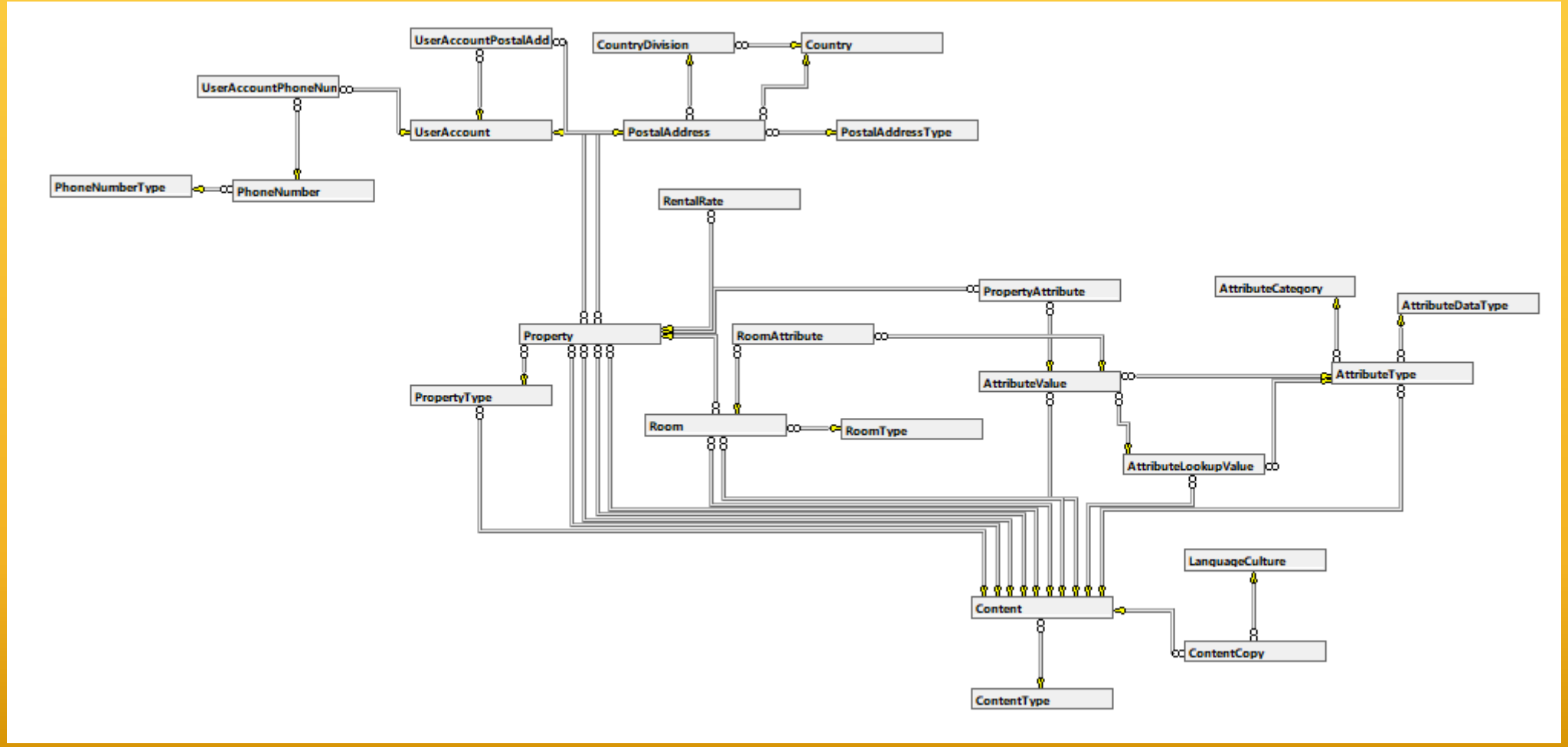
Properties

Reference Types

Content

Rooms

Relational Data Model



Real World: Why Relational

Skillset

Time to Market

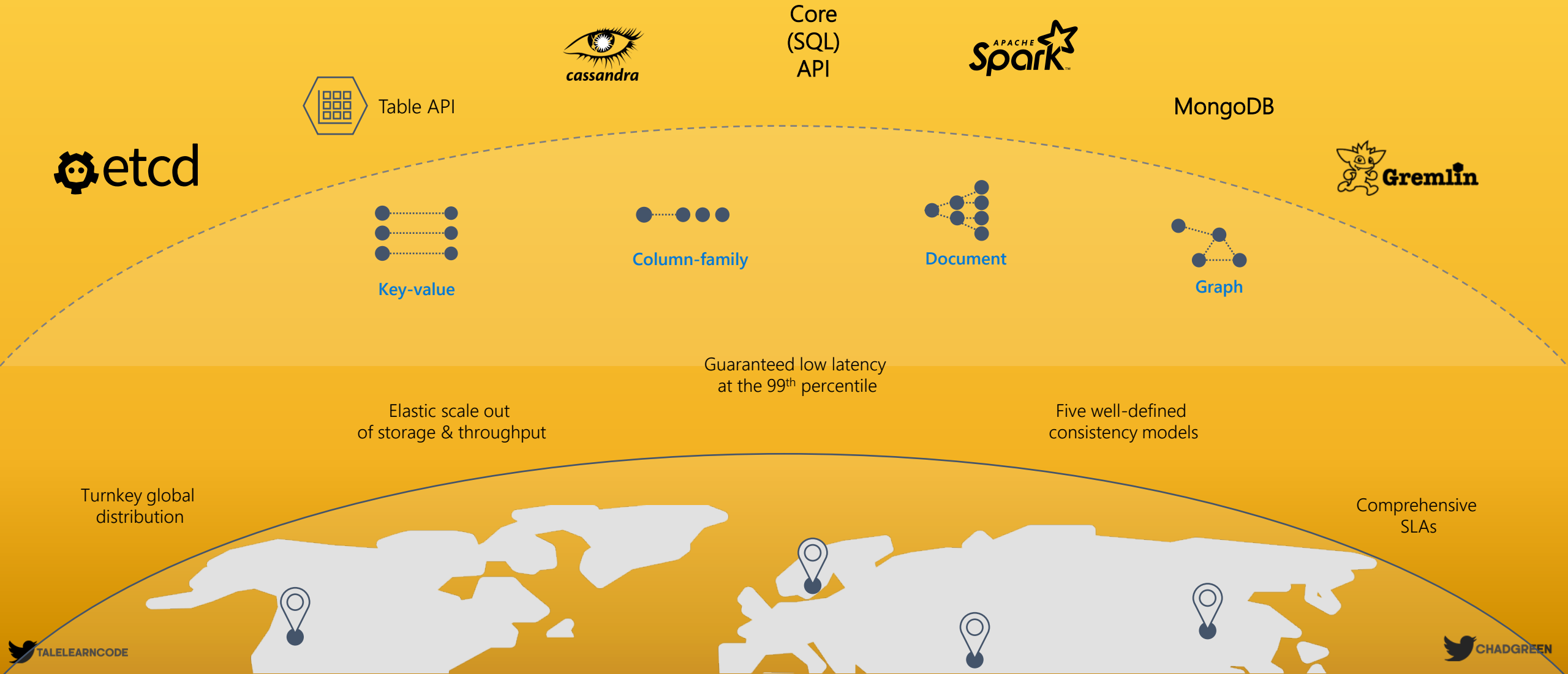
Other Products

Issues Found in Real-World Project

- Searching against the attributes is difficult
- Navigation is deep

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

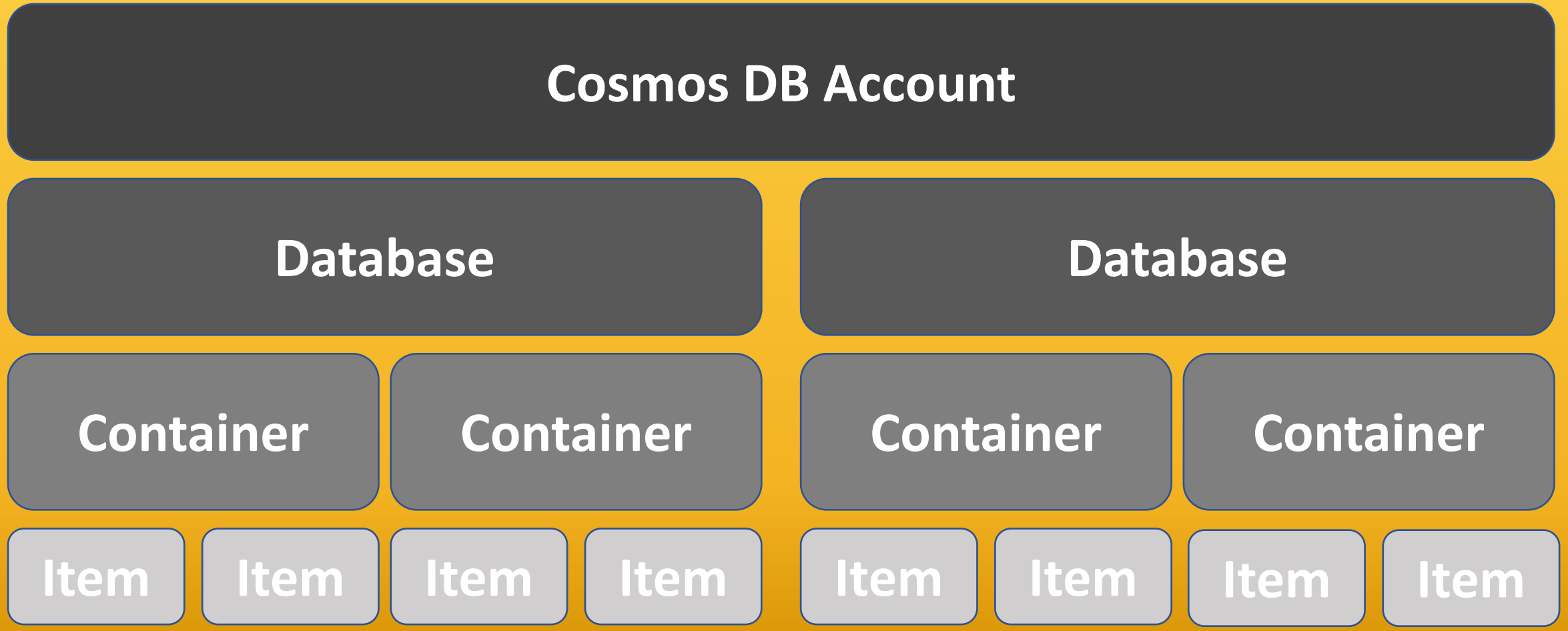
Which Azure Cosmos DB Data API?

Core (SQL) API

Cassandra

Migrating to NoSQL

Document Database Structure



Vacation Rentals Data Model

User Accounts

Attributes

Properties

Reference Types

Content

Rooms

Vacation Rentals Data Model

User Accounts
userAccountId

Attributes
attributeTypeId

Properties
propertyId

referenceTypeName
Reference Types

Content

Rooms

Vacation Rentals Data Model

User Accounts
userAccountId

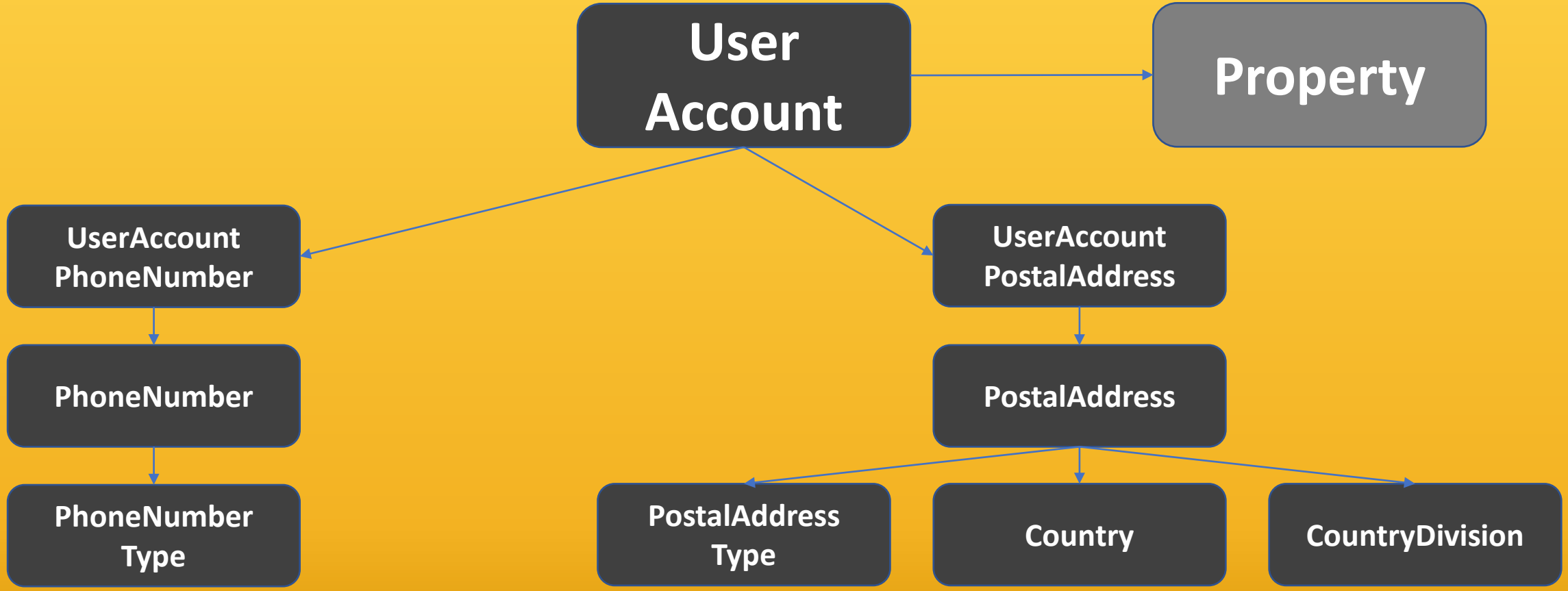
Attributes
attributeTypeId

Reference Types
referenceTypeName

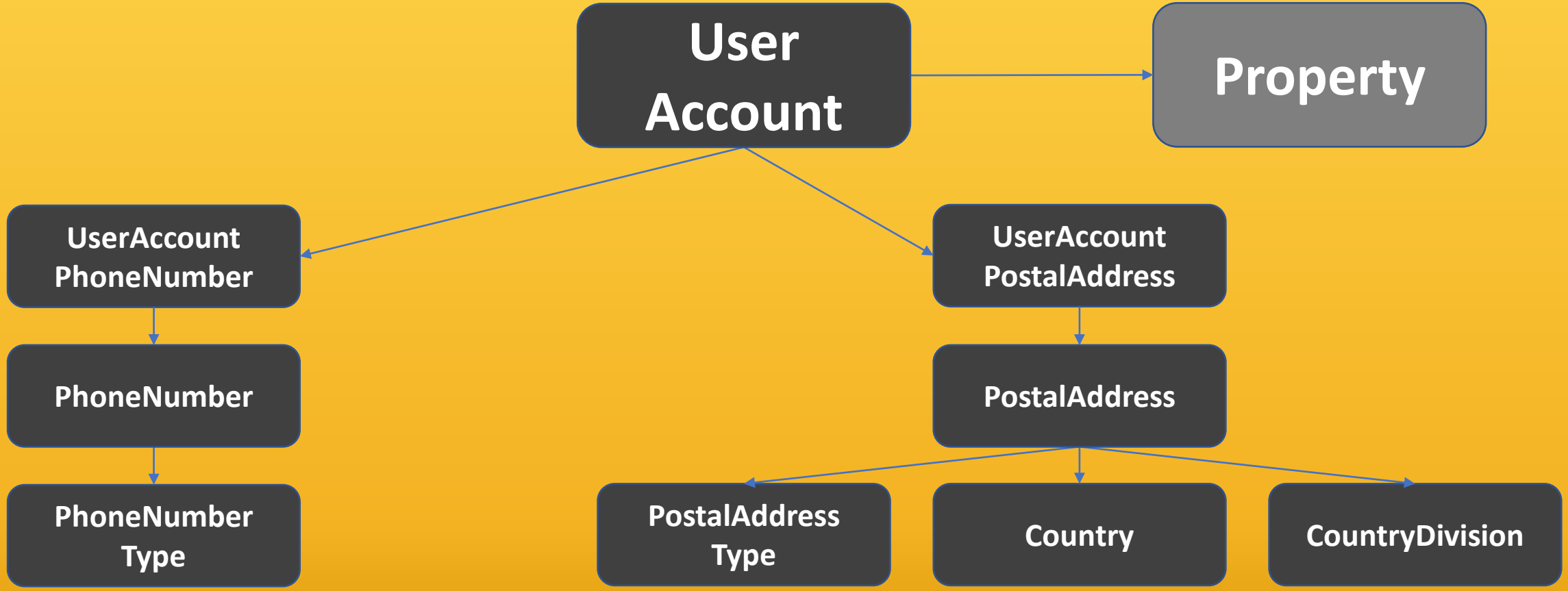
Properties
propertyId

Properties by Location
locationId

User Account Migration

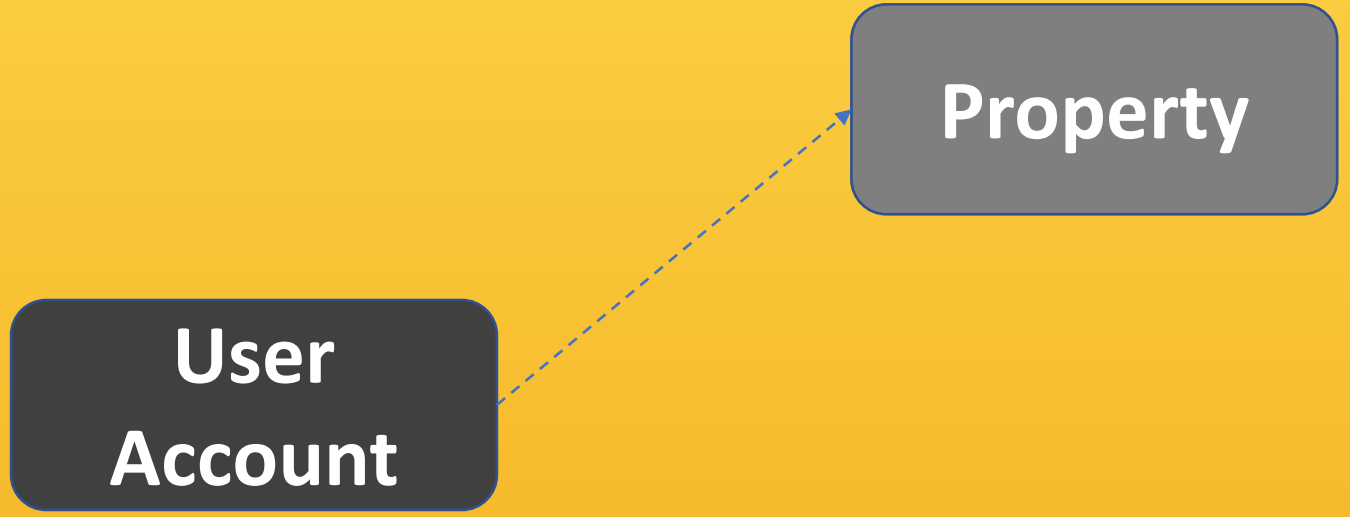


User Account Migration

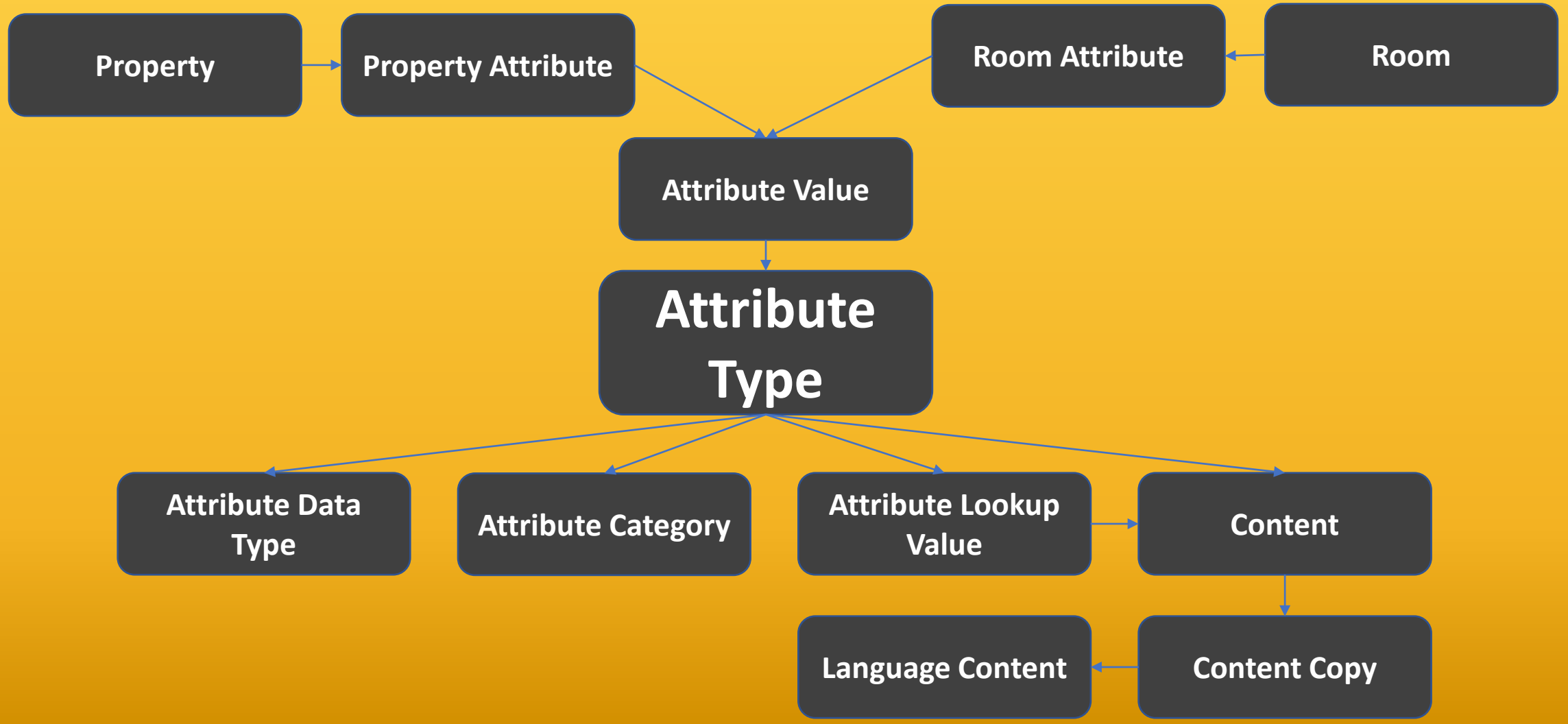


User Account Migration

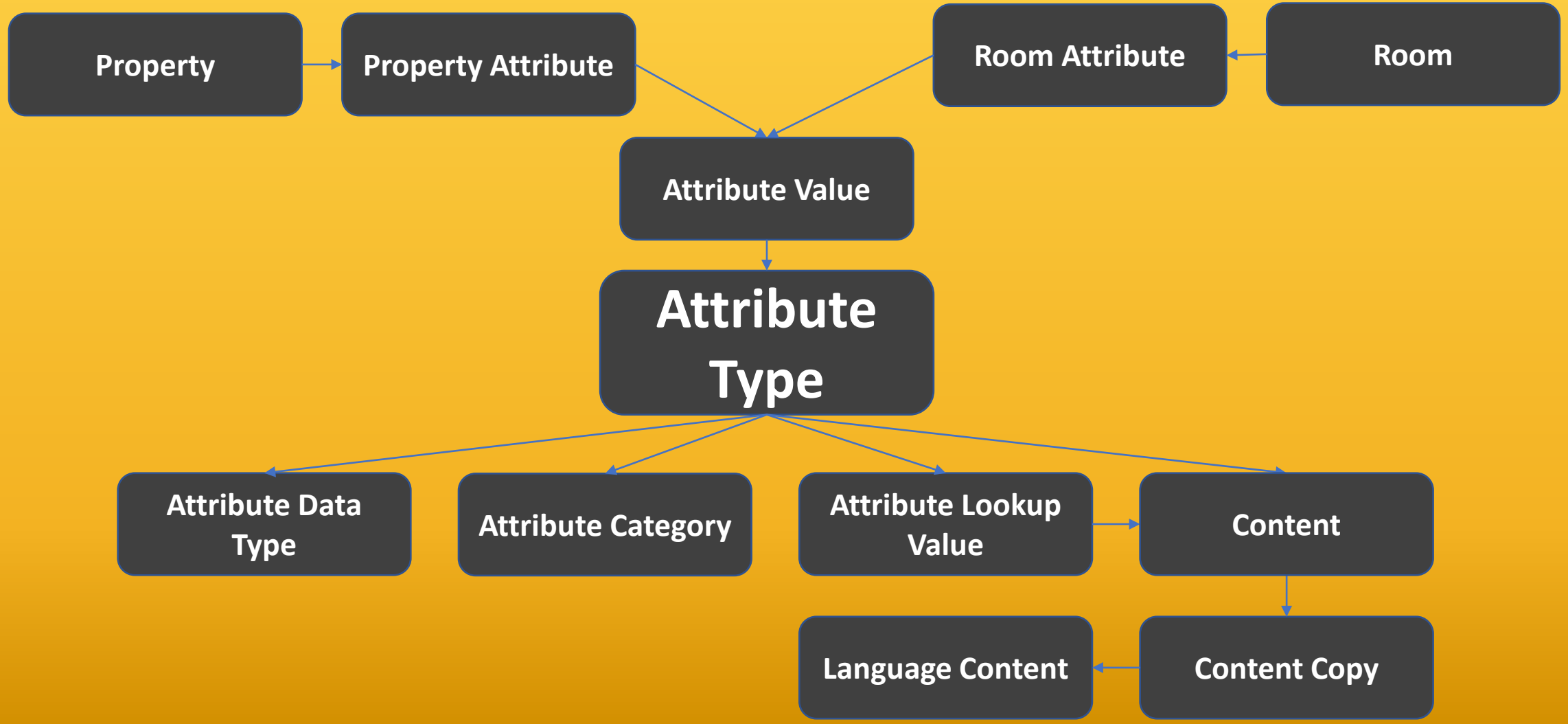
User Account Migration



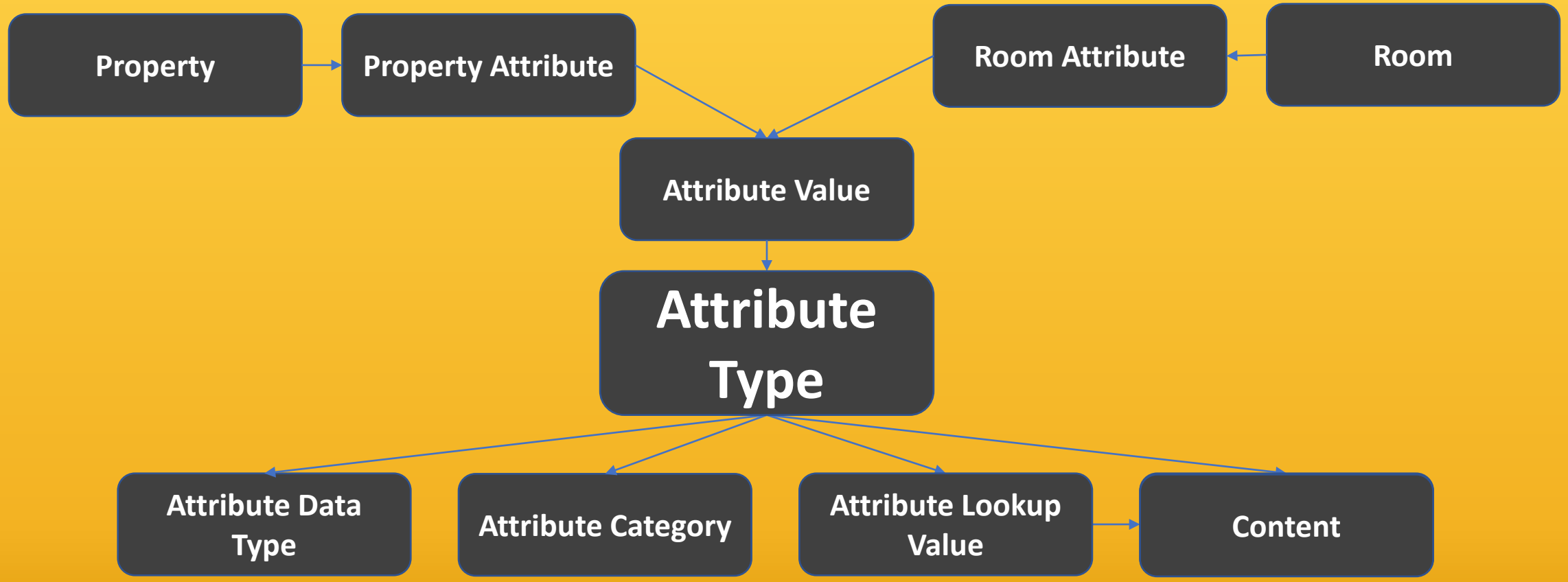
Attribute Migration



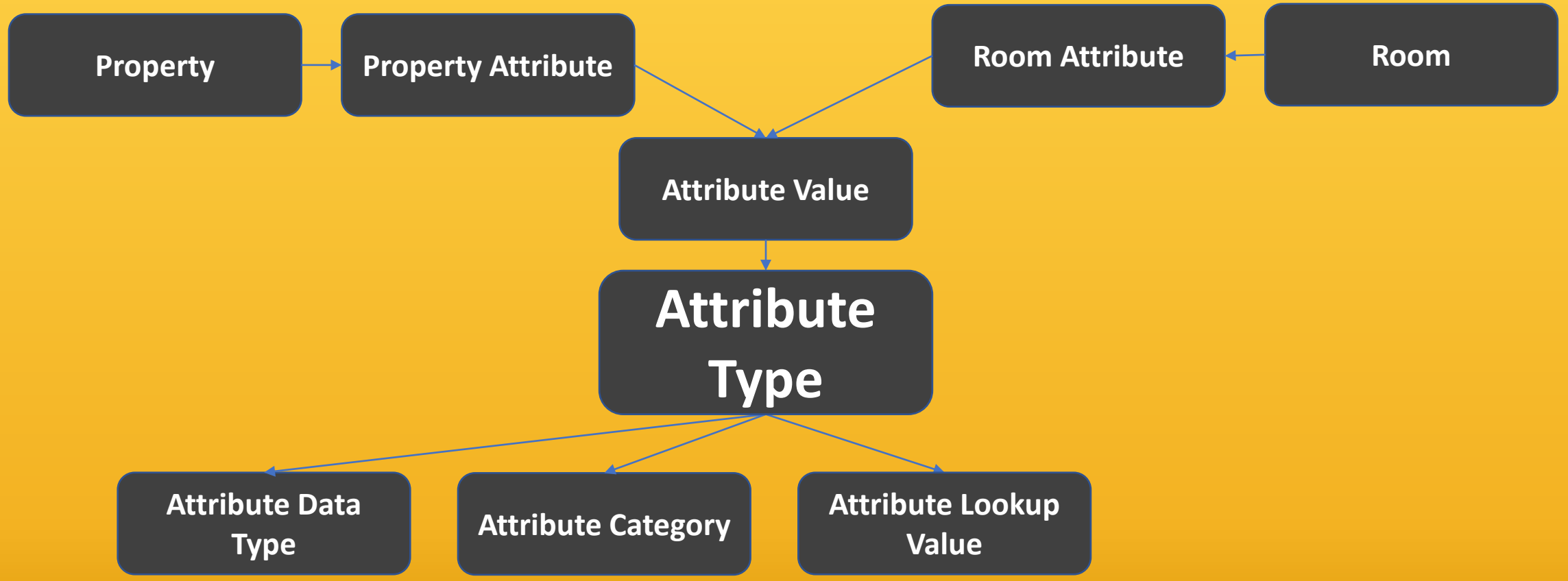
Attribute Migration



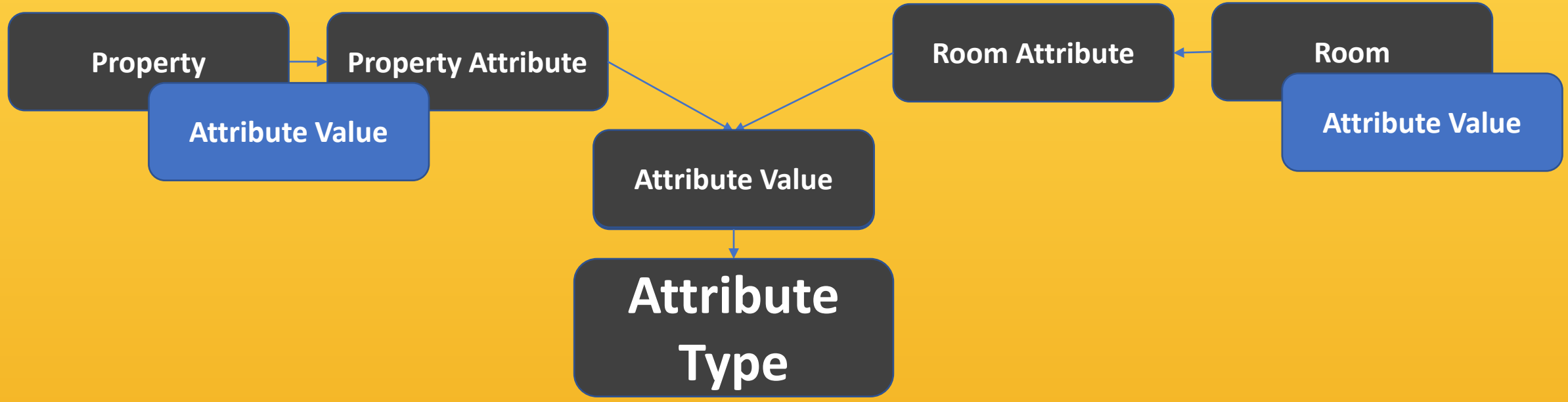
Attribute Migration



Attribute Migration

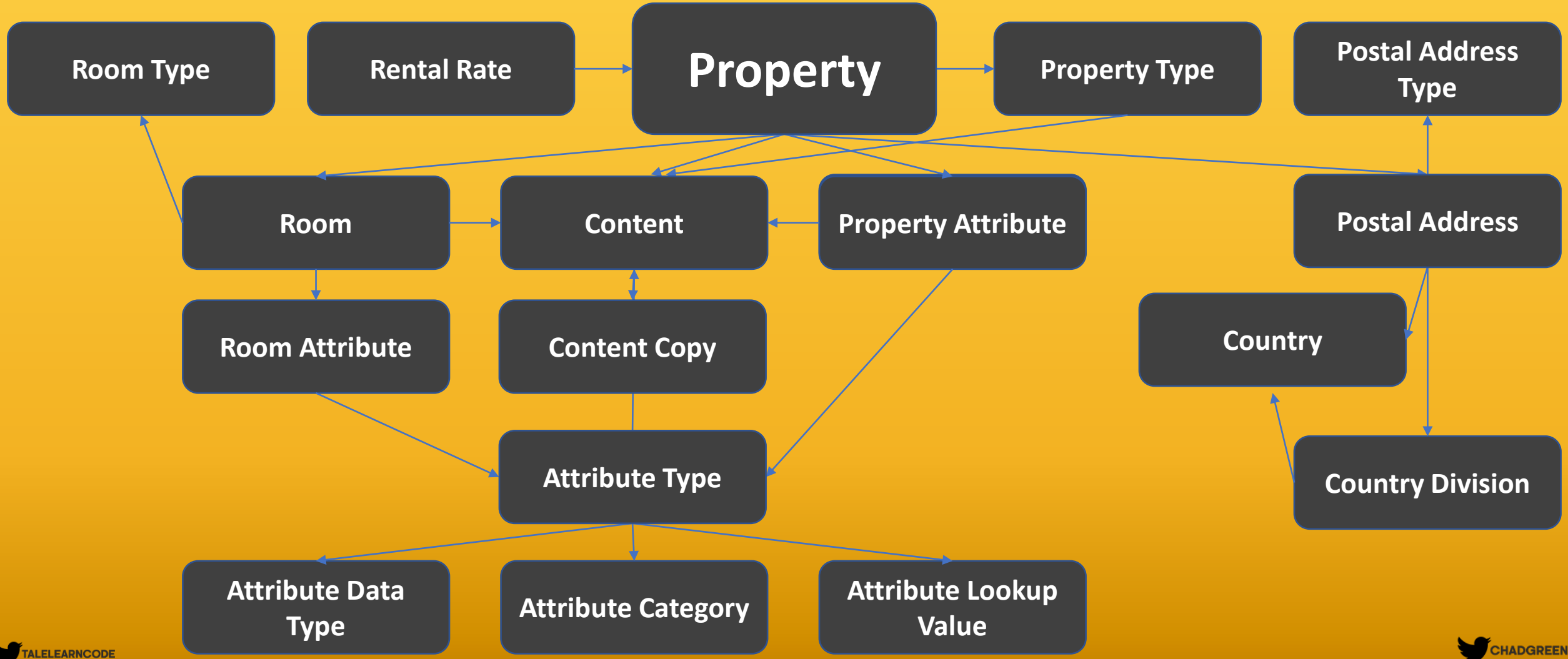


Attribute Migration

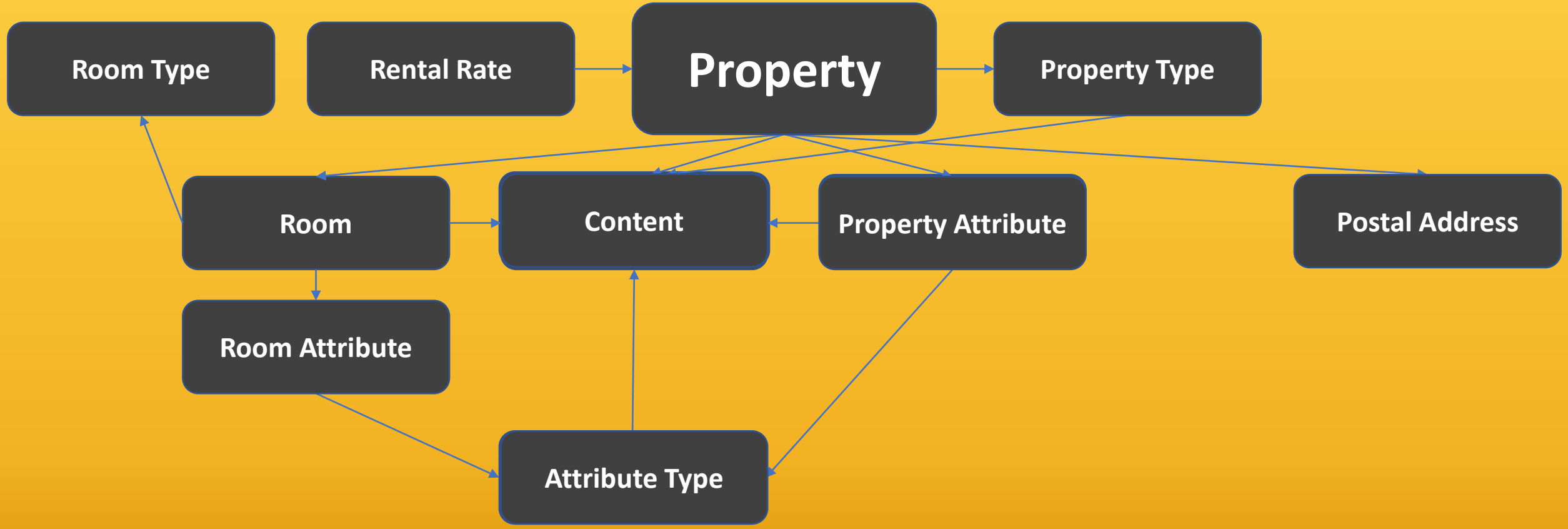


Attribute Migration

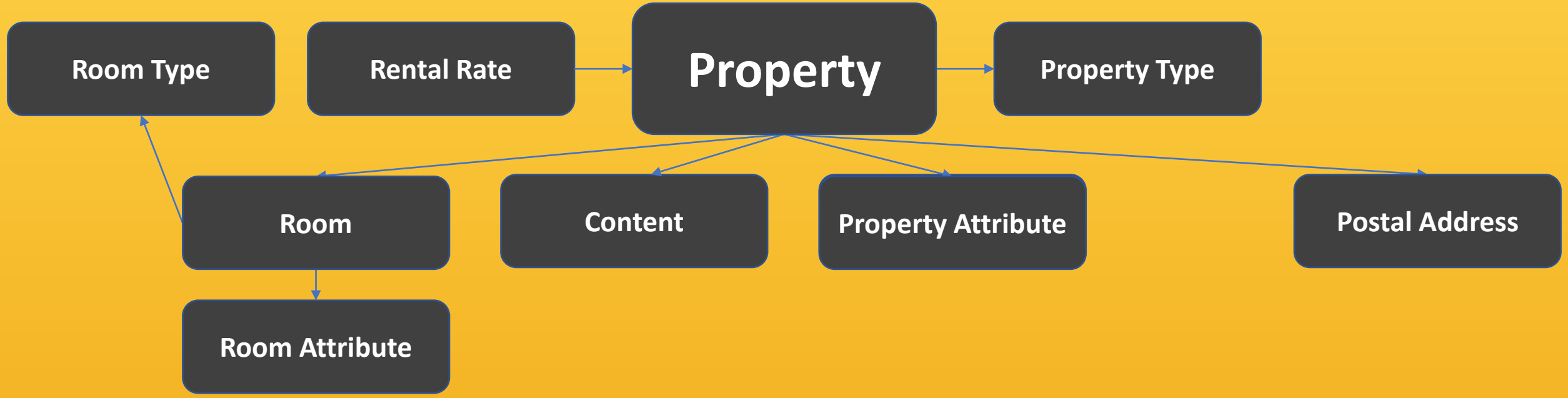
Property Migration



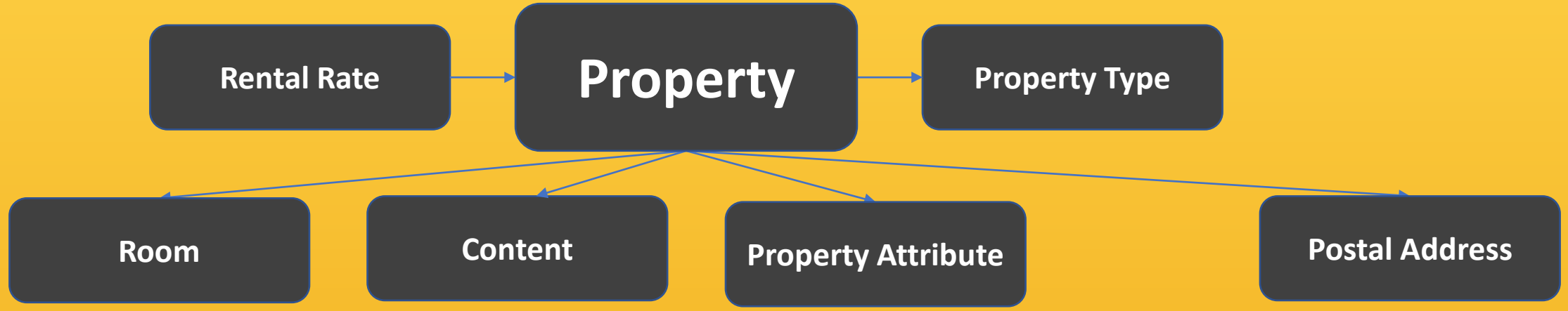
Property Migration



Property Migration



Property Migration



Property Migration

Property Migration

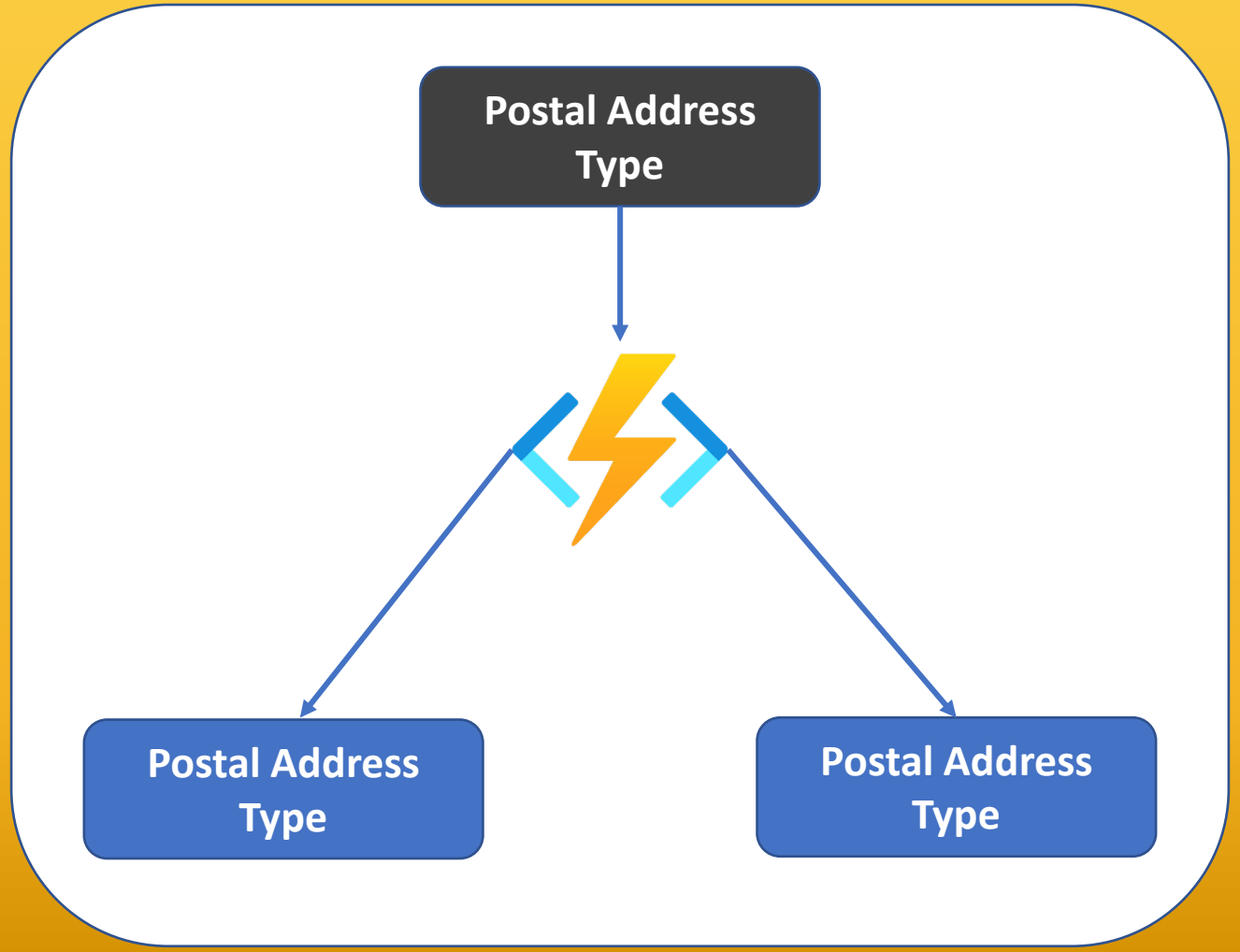
Reference Types

- Country
- Country Division
- Language/Culture
- Phone Number Type
- Postal Address Type
- Property Type
- Room Type
- Attribute Data Type
- Attribute Category

Reference Types

- Country
- Country Division
- Language/Culture
- Phone Number Type
- Postal Address Type
- Property Type
- Room Type
- Attribute Data Type
- Attribute Category

Reference Types



Best Tool(s) for the Job

Thank You

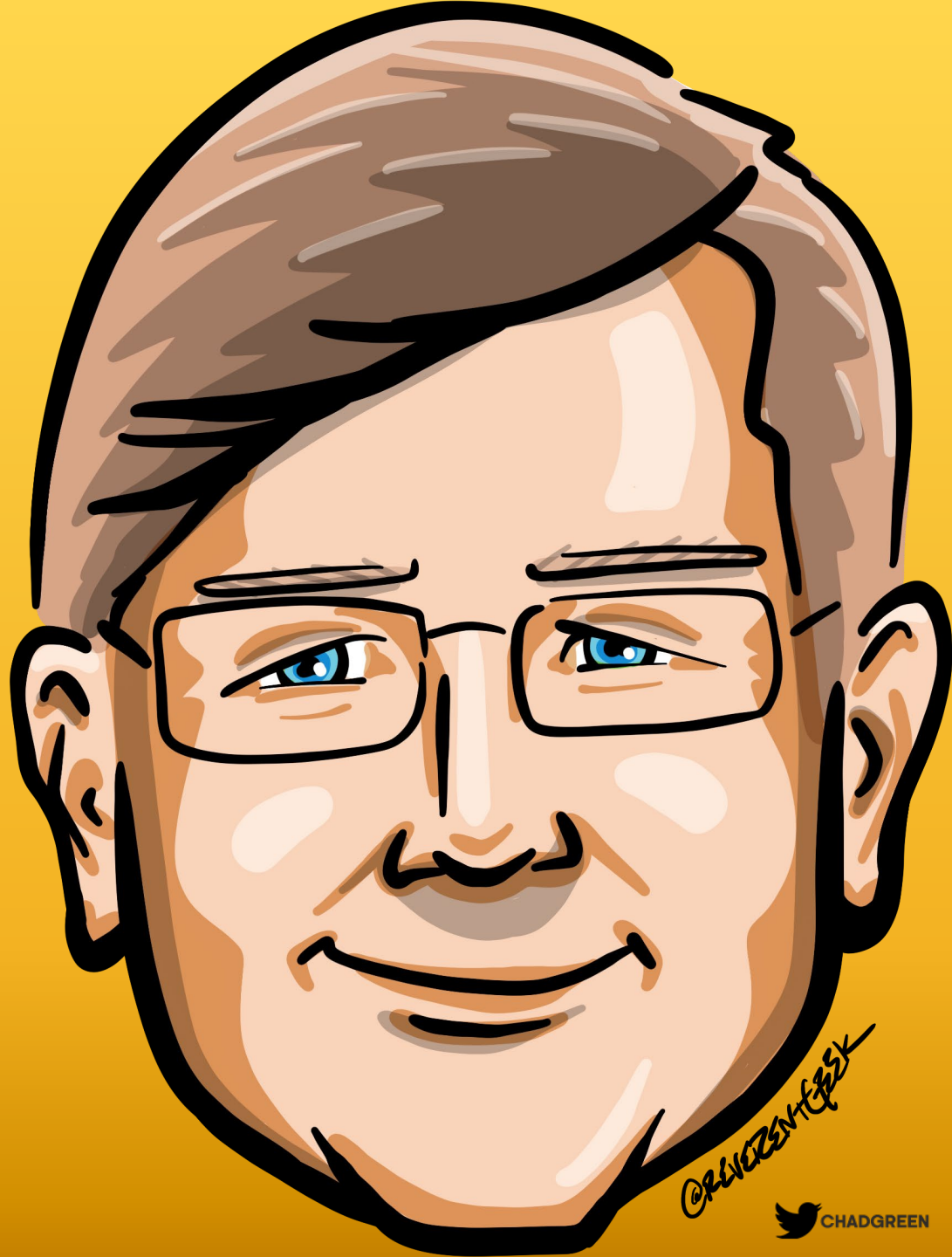
chadgreen@chadgreen.com

TaleLearnCode

ChadGreen.com

ChadGreen & TaleLearnCode

ChadwickEGreen




THAT®.US
JOIN US
DAILY