

Chapter 12

AWS Database Design

Episode 12.01

Database Types

Database Types

- Hosted services
 - Relational
 - Non-relational (NoSQL)
- Custom instance installs
 - BYOL

Hosted Services

- AWS Relational Database Service (RDS)
 - Aurora MySQL
 - Aurora PostgreSQL
 - Oracle
 - SQL Server
 - MySQL
 - PostgreSQL
 - MariaDB

Custom Instance

1. Start the instance with the required OS
 - AMI
2. Install the database service
 - ISO image
3. Create the database

Flat File vs. Relational

- Flat file databases
 - Have one line per record
 - Doesn't contain multiple tables
- Relational databases
 - Store portions of the data in designated tables
 - Tables are related based on unique identifier

NoSQL

- Not based on SQL or relational design theory
- Design supports fast transactions
- DynamoDB is a NoSQL service
 - Create
 - Query
 - Read/write/modify

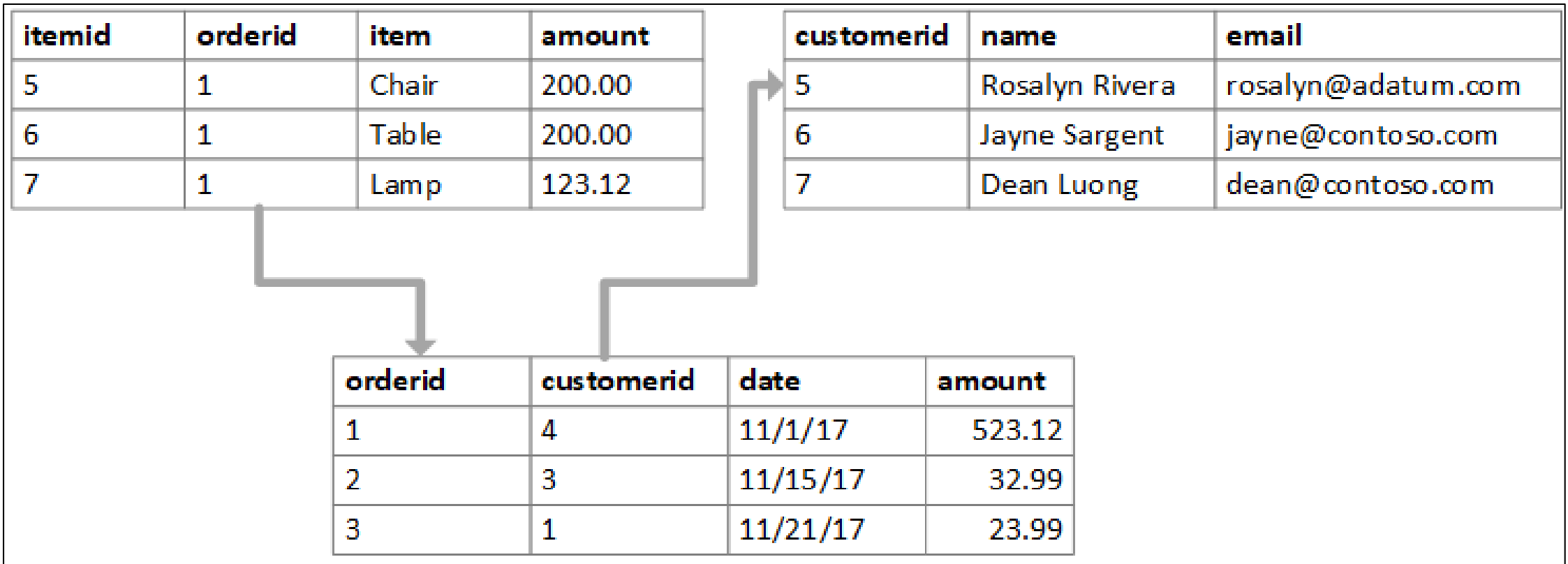
Data Warehouses

- Large, central repository for data
- Data aggregated from one or more sources
- Used for Online Analytical Processing (OLAP)
 - Redshift

Episode 12.02

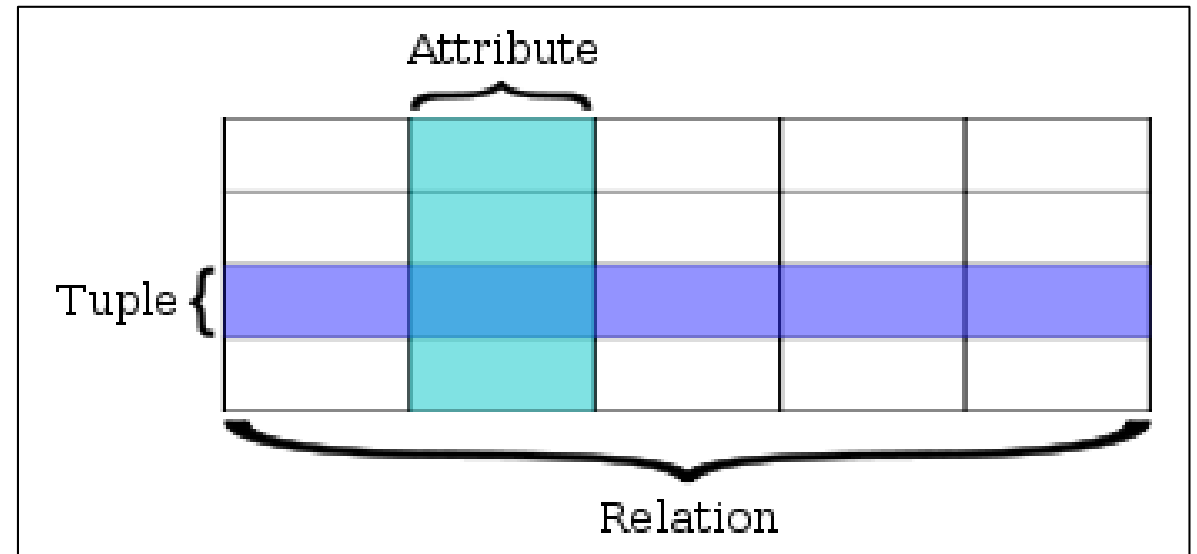
Relational Databases

Relational Databases (RDBs)



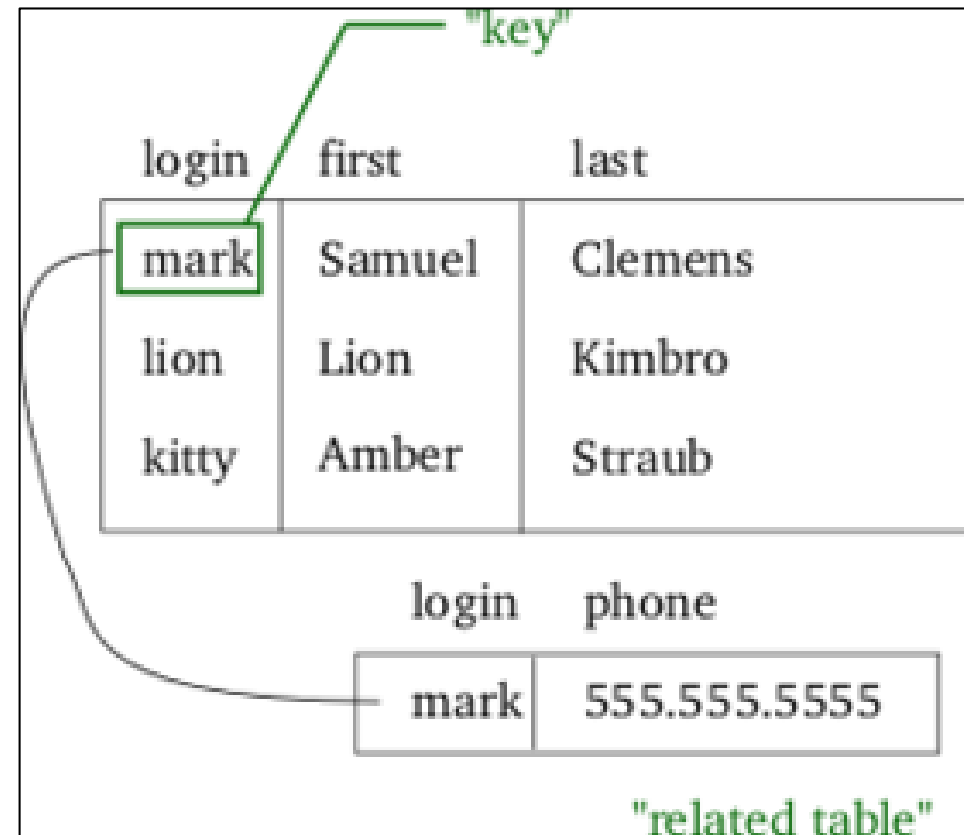
RDB Terminology

- Rows – tuples
- Columns – attributes
- Columns – properties
- Tables – relations
- Tables – entities
- Tables – objects
- Views and results



Relationships

- Primary key
- Foreign key
- Join



Normalization

- Process for evaluating and correcting structures
 - Determines the best assignments of attributes to entities
- Normalization provides micro view of entities
 - Focuses on qualities of specific entities
 - May result in additional entities
- Works through a series of stages called normal forms
 - 1NF → 2NF → 3NF → 4NF (optional)
- Higher the normal form (closer to 4NF), slower the database response
 - More joins are required to answer end-user queries
- Higher the normal form, faster writes

Episode 12.03

Database Hosting Methods

EC2 Instance-Based

1. Launch an instance
2. Install the database service
3. Open appropriate ports in security groups
4. Connect to the database

AWS Service-Based

1. Create the database
2. Connect to the database

Instance-Based Considerations	Service-Based Considerations
Complete control	Less control
Manual performance management	Automatic performance management
Manual updates	Automatic updates

Episode 12.04

High Availability Solutions

Clustering

- Multiple servers (instances)
- One database with replication
- Increases availability
- Automatic failover
- Increased cost

Standby Instances

- Multiple servers (instances)
- One database with replication
- Increases recoverability
- No automatic failover
- Reduced costs

Single AZ Deployment

- One instance
- One AZ
- One region

Multiple AZ Deployment

- Multiple instances
- Multiple AZs
- One region
- Replicated storage
 - Increased availability
 - Increased performance
- Cost

Episode 12.05

Scalability Solutions

Scalability

- Can increase capacity
 - Storage
 - Processing
 - Network operations
 - Throughput

Scaling the Instance

- Changes the type/class
- Auto Scaling is not supported in RDS
 - Can be scripted with CLI commands

Read Replica

- Read-only copy of the database
- Offloads read-only traffic from the main database
- Multiple instances can be in different regions

Episode 12.06

Database Security

Encryption

- RDS databases support “at rest” encryption
- Must be enabled at creation time
- Can be enabled on recovery (manually)

Permissions

- Administration access based on IAM
- Data access based on database capabilities
 - CRUD
 - DB admin

DEMO

- IAM database access account

Episode 12.07

Aurora

Aurora

- Relational DB
- Optimized for Online Transaction Processing (OLTP)
 - Very fast writes
- MySQL-compatible database system
- Increased performance over MySQL

Scaling Aurora

- Initially 10 GB, scaling in 10 GB increments
 - Max 64 TB
- Compute resources
 - Max 32 CPUs
 - Max 244 GiB RAM
- <https://aws.amazon.com/blogs/database/reduce-resource-consumption-by-consolidating-your-sharded-system-into-aurora/>

Aurora Availability

- Availability defaults:
 - 2 DB copies in each AZ
 - Minimum of 3 AZs
- Write capability
 - Continues with up to two copies lost
- Read capability
 - Continues with up to three copies lost

Aurora Replicas

- Up to 15 Aurora replicas
 - Automatic failover
- Up to 5 MySQL read replicas
 - No automatic failover

Episode 12.08

Redshift

Redshift

- Data warehouse database
- Optimized for Online Analytical Processing (OLAP)
- AWS managed
- Pricing
 - Entry point of \$0.25/hr
 - \$1,000 per TB/yr

Redshift

- Single node
 - 160 GB
- Multiple node
 - Leader node
 - Connections and queries
 - Compute node
 - Store data and execute queries and calculations

Redshift Speed

- Columnar data stores
 - Sequential reads
 - Very fast reads
- Data compression
- Massively Parallel Processing (MPP)

Redshift Security

- SSL transit encryption
- AES-256 storage encryption
- Keys managed through AWS Key Management

Redshift Availability

- Operates in one AZ
- Snapshots can be restored to new AZs

Episode 12.09

DynamoDB

DynamoDB

- NoSQL database service
- Provides special features
 - Millisecond latency at any scale
 - Very VERY fast read/write
 - Stored on SSD
 - Spread across 3 distinct data centers

DynamoDB

- Read consistency types
 - Eventual consistent reads
 - Strongly consistent reads

DynamoDB Pricing

- Storage
 - \$0.25/GB per month
- Throughput
 - Write: billed per hour for every 10 units
 - Read: billed per hour for every 50 units
 - 1 unit equals 1 write per second