

Databases

Lernziel

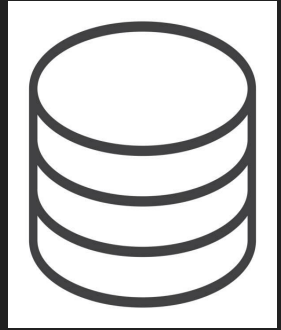
- Warum verwenden wir Datenbanken?
- Welche Optionen bietet AWS für relationale Datenbanken?
- Welche Optionen bietet AWS für nicht-relationale Datenbanken?
- Was sind OLTP (Online Transaction Processing) und OLAP (Online Analytical Processing) ? → Welche Systeme dafür?
- Welche Möglichkeiten zur Analyse bietet AWS?

Warum nicht einfach?

EFS, EBS, EC2 Instance Store, S3 haben Limits

→ Strukturierte Datenspeicherung hat Vorteile:

- man hat einen Index mit dem man effizient anfragen(query) und suchen(search) kann
- wir können Abhängigkeiten zwischen Datensätzen definieren
- Datenbanken sind optimiert für einen Zweck und haben unterschiedliche Eigenschaften, Formen und Grenzen



Wiederholung - Relationale Datenbanken

Tabellendatenbank in der Regel mit SQL

→ Beispiel

Patienten ID	Arzt ID	Vorname	Nachname
1	1	Hans	Müller

Skalieren:

- nur vertikal

Arzt ID	Vorname	Nachname	Spezialisierung
1	Gregor	Maier	Lungenfacharzt

Wiederholung - NoSQL Datenbanken

- nicht-relationale Datenbanken
- flexibles und anpassbares Schema
- Vorteile:
 - Flexibilität
 - Performanz optimiert für spezifische Datenmodelle
 - Hochfunktional: Für das Datenmodell optimierte Typen
 - Skalieren:
 - horizontal
 - vertikal
- Beispiele für Datenbanken sind: Key-value, Dokument, Graph, in-memory

Wiederholung - NoSQL Datenbeispiel: XML

```
1 <breakfast_menu>
2   <food>
3     <name>Belgian Waffles</name>
4     <price>$5.95</price>
5     <description>Two of our famous Belgian Waffles with real maple syrup</description>
6     <calories>650</calories>
7   </food>
8   <food>
9     <name>Strawberry Belgian Waffles</name>
10    <price>$7.95</price>
11    <description>Light Belgian waffles covered with strawberrie</description>
12    <calories>900</calories>
13  </food>
14  <food>
15    <name>Berry-Berry Belgian Waffles</name>
16    <price>$8.95</price>
17    <description>Light Belgian waffles with an assortment of fresh berries</description>
18    <calories>900</calories>
19  </food>
20 </breakfast_menu>
```

AWS Managed Databases - PaaS (Platform as a Service)

Vorteile

- Schnelle Bereitstellung
- Hohe Verfügbarkeit
- Skalieren: vertikal und horizontal
- Automatisiertes Backup & Restore
- System-Patches sind die Verantwortung von AWS
- Monitoring und Alarmierung wird gestellt

Eine Alternative ist:

Datenbank auf EC2 laufen lassen, aber dann haben wir alle Verantwortungen

Das wollen wir in der Regel nicht

RDS - Relational Database Service

- Verwalteter Service (managed Service)
- OLTP → Online Transaction Processing
- kontinuierliche Backups und Wiederherstellungsfunktion (Point in Time Restore)
- Überwachungsdashboards (Monitoring dashboards)
- Replikation der Datenbank für Leseanfragen → Erhöhte Performanz
- Multi AZ Einrichtung für DR (Disaster Recovery)
- Skalierung: vertikal und horizontal
- Beispiel für RDS

Aurora

- Sprachen: PostgreSQL und MySQL
- OLTP → Online Transaction Processing
- Cloudoptimiert 3-5 mal so schnell wie Mysql / Postgres auf RDS
- Aurora Speicherplatz wächst automatisch in 10 GB Schritten
- Maximaler Speicher 128 TB
- Aurora kostet 20% mehr als RDS
- nicht Teil vom AWS FREE TIER

Amazon Aurora Serverless

- Automatisierte Datenbank instanziierung und auto-scaling basierend auf Nutzung
- Sprachen: PostgreSQL und MySQL
- Keine Kapazitätsplanung
- Kein Management
- Zahlen für jede Sekunde (Pay per second) → manchmal günstiger
- Nutzungsszenario:
 - seltene Nutzung
 - unvorhersehbare Arbeitslasten (wenn wir nicht wissen, wann die Datenbank viel benutzt wird)

Für die Prüfung:

- Aurora ohne Verwaltungsaufwand
- Aurora with no management overhead

RDS Optionen

Read Replicas Beispiel

Multi AZ Beispiel

Amazon ElastiCache



- ElastiCache → verwaltete Redis oder Memcached
- Hochperformant (so wie RAM) - *High Performance*
- Schnelle Reaktionszeit / Kurze Verzögerung - *low latency*
- Verringert die Last von Datenbanken mit vielen Lesezugriffen
- Verwaltet von AWS:
 - Wartung / Patching
 - Optimierungen
 - Einrichtung
 - Konfiguration
 - Überwachung
 - Fehlerbehebung
 - Backups

Prüfung:

- in memory database

DynamoDB

- sehr viel verwendeter Service
- Vollständig von AWS verwaltet, hochverfügbar mit Replikation über 3 Verfügbarkeitszonen (3 AZs)
- **NoSQL - nicht-relationale Datenbank**
- Skaliert automatisch nach Bedarf
- **serverless**
- **Einstellige Millisekunden-Latenzzeit - schnelle Datenabrufzeit**
 - Single-digit millisecond latency - low latency retrieval
- Schlüssel / Wert Datenbank

DynamoDB Accelerator - DAX

- in memory Database für DynamoDB
- 10-Fache Geschwindigkeitssteigerung
- DAX wird NUR für DynamoDB verwendet

DynamoDB - Global Tables

- geringe Latenz in mehreren AWS Regionen
 - 2-way aktiv-aktiv Replication
 - read/write in allen Regionen verfügbar

Redshift



- Sprache: PostgreSQL
- NICHT für OLTP (Online Transaction Processing)
- für OLAP → online analytical processing (analytics and data warehousing)
- 10x besser als andere Datenlager
- lädt Daten nur einmal pro Stunde
- Skaliert bis zu Petabytegrößen → sehr groß
- spaltenorientierter Datenspeicher - columnar storage of data
- pay as you go - basierend darauf wie viele Instanzen laufen
- integriert mit BI - Tools wie Quicksight und Tableau
- kann auch serverless laufen als **Redshift Serverless**
 - zahlen nur für das Benutzte
 - für Echtzeitanalyse

QuickSight

- **serverless** durch machine learning gestützter BI service für interaktive dashboards
- schnell, skaliert automatisch, kann man embedden auf anderen Seiten, kostet pro session

- **Nutzung:**

- Geschäftsanalyse
- Visualisierungen erstellen
- spontane analyse
- Geschäftseinblicke durch Daten bekommen



- Integration mit RDS, Aurora, Athena, Redshift, S3 und weiteren

Aufgabe

Erstellt in euren Gruppen kurze Zusammenfassungen folgender Services. Fasst die Informationen die ihr findet auf eine Länge von jeweils max. 2 Folien zusammen und bereitet euch auf eine kurze Präsentation davon vor.

1. DokumentDB
2. Neptune
3. Timestream
4. QLDB
5. Managed Blockchain
6. Glue

Bei Fragen jederzeit die Hand im Call heben. :)

DokumentDB

- DokumentDB ist das für MongoDB, was Aurora für MySQL ist
- MongoDB speichert JSON daten
- ähnlicher Service wie Aurora
- fully managed Service mit Replikation über 3 AZs
- Speicher steigt automatisch in 10 GB Schritten
- Skaliert automatisch für millionen von Zugriffen pro Sekunde

Neptune

- Voll verwaltete Graphdatenbank (fully managed)
- Beispiel für Graphdatensatz ist soziales Netzwerk
 - Nutzer haben Freunde
 - Posts haben Kommentare
 - Kommentare werden von Nutzern geliked
 - Nutzer teilen und liken Posts
 - alles durch ein Netz von verbundenen Datenpunkten abgebildet
- Hochverfügbar über 3 AZs mit bis zu 15 Replikat-DBs
- Gut für Wissensgraphen → zb. Wikipedia

Timestream

- Vollständig verwaltet, schnell, Skalierbar, serverlos, Zeitreihendaten
- Skaliert automatisch hoch und runter
- kann pro Tag mehrere Billionen (→12 Nullen) Anfragen verarbeiten
- 1000 mal schneller als relationale Datenbank und 10 mal günstiger
- eingebaute Analyse

QLDB

- Quantum Ledger Database
- zentralisierte Blockchain
- für aufnahmen von Finanztransaktionen
- komplette Geschichte aller Veränderungen der Daten sind einsehbar
- alter Stand ist kryptografisch unveränderbar

Managed Blockchain

- dezentralisierte Blockchain
- ermöglicht Transaktionen ohne eine zentralisierte Autorität
- kompatibel mit Hyperledger Fabric & Ethereum

Glue

- Service für: Extraktion, Transformation und Laden (ETL) von Daten
- serverless
- Nutzung um Daten vorzubereiten für Analysezwecke

DMS - Database Migration Service

Grafik: Quelldatenbank → EC2 auf der DMS läuft → Zieldatenbank

- schnell, sicher
- Quelldatenbank bleibt während der Migration verfügbar

Unterstützt:

- Homogen: gleicher Datenbanktyp auf gleicher Datenbanktyp: z.B. Oracle → Oracle
- Heterogen: Datenbanktyp auf anderen Datenbanktyp: z.B. Microsoft SQL → Aurora

Amazon Athena

- serverless Anfrageservice für Analyse von S3 Objekten
- Sprache: nutzt standard SQL
- Datentypen die unterstützt werden: CSV, JSON, ORC, Avro und Parquet (auf Presto)
- Kosten: \$ 5 / TB das gescannt wird
- Nutzung
 - Business intelligence
 - analytics
 - reporting
 - Logs
 - CloudTrail trails
 - etc.
- Prüfung: analysiere Daten in S3 mit serverless SQL → Athena

Amazon EMR - Elastic Map Reduce

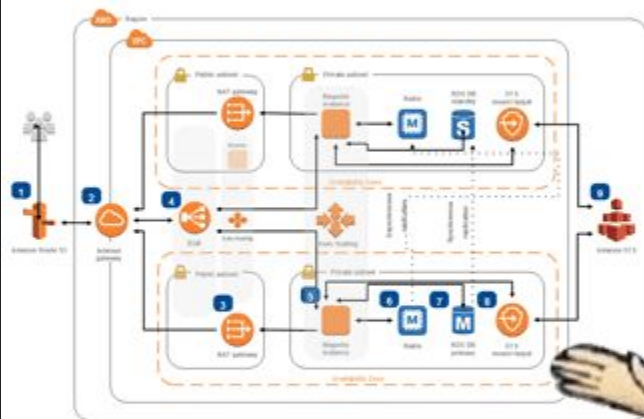
- Hadoop clusters (Big Data)
 - Analyse
 - Kann cluster von hunderter von EC2-Instanzen erfassen
 - AWS übernimmt die Konfiguration
 - Auto-scaling
- Nutzung: Datenverarbeitung, machine learning, big data, ...

Quiz zum gibt es heute Mittag :)

und die Folien auch :)

meme auf der nächsten Seite

**ME: I JUST NEED TO HOST
'HELLO WORLD' ON THE CLOUD.**



**AWS: NO PROBLEM. HAVE YOU
CHECKED ALL OF OUR COOL NAMED
PRODUCTS YOU'LL NEVER UNDERSTAND?**