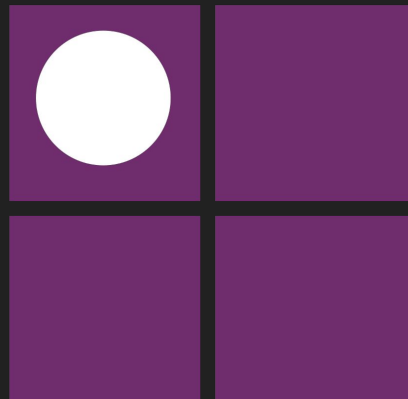


# Amazon SimpleDB

Hannes Schwanzer & Isa Kilic



# Gliederung

- Einführung
    - Was ist Amazon SimpleDB?
  - Vorteile
    - Welche Vorteile hat man?
  - Einschränkungen
    - Wodurch wird man eingeschränkt?
  - Use Cases
    - Wofür nutzt man Amazon SimpleDB?
- 

- Datenspeicherung
  - Wie werden die Daten gespeichert?
- Kosten
  - Wie viel kostet das alles?
- Select-Queries
  - Wie führt man Selects aus?
- API-Befehle
  - Welche Optionen hat man?
- Rest Request
  - Kurzes Beispiel zur Ausführung von Rest Requests

→ Unterstützte Sprachen und Python Beispiele

# Amazon SimpleDB

- NoSQL-Datenbank
  - Auslagerung des Arbeitsaufwands bei der Datenbankadministration
  - Speichern und Abfragen von Datenelementen per Web-Serviceanforderungen
- Datenkopien werden auf Geographisch unterschiedlichen Orten gespeichert
- Kosten beruhen auf tatsächlich genutzten Ressourcen
- Anmerkung:
  - Zur Nutzung mit anderen AWS-Services konzipiert

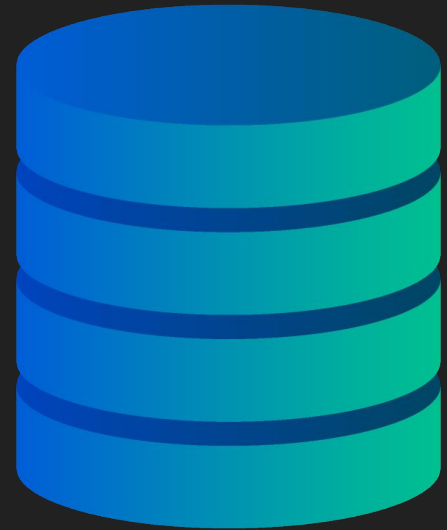
# Amazon SimpleDB – Anmerkung

- Nutzung mit anderen AWS-Services
  - Nutzer können Anwendung per Amazon Cloud laufen lassen
  - Datenobjekte der Anwendung auf Amazon S3 speichern
  - Per SimpleDB Anfragen ausführen

# Mentimeter Umfrage

Welche Punkte werden von NoSQL Datenbanken  
meist berücksichtigt?

menti.com → 2239 2443



# Vorteile

- Low Touch
  - Weniger Verwaltungsaufwand
- Hohe Verfügbarkeit
  - Mehrere Kopien auf geographisch unterschiedlichen Orten
  - Hohe Erreichbarkeit und Datenhaltbarkeit
- Sicher
  - HTTPS Endpunkt
  - Ähnliches Rechtesystem wie Linux
  - Kommunikation zwischen Anwendungen bzw. Client und Domain verschlüsselt

# Vorteile

- Einfache Nutzung
  - Wenig Datenbank Erfahrung notwendig
  - Höhere Produktivität
- Preis
  - Nur tatsächlicher Aufwand des Speicherns bzw. Lesens muss bezahlt werden
    - Beispiel folgt

# Vorteile

- Flexibilität
  - Änderungen in Applikation sind nicht vom Datenbankschema abhängig
  - Konsistent Halten von Änderungen und der Flexibilität der Read-Performance
  - Entscheidung zwischen Eventual Consistency und Consistency möglich:
    - → Eventual Consistency für hohe Geschwindigkeit aber geringere Einheitlichkeit
    - → Consistency für geringere Geschwindigkeit aber höhere Einheitlichkeit






# Einschränkungen

Parameter	Einschränkungen
Domain Größe	10 GB pro Domain / 1 Milliarde Attribute pro Domain
Domains pro Konto	250
Maximale Query Ausführungszeit	5 Sekunden
Maximale Anzahl an Artikeln in einer Select-Antwort	2500
Maximale Größe für die Antwort eines Selects	1 MB
Maximale Anzahl an Vergleichen und einzigartigen Attributen per Select	20

# Einschränkungen – Anmerkung

- Transaktionen
  - Keine relationale Datenbank
  - Bietet keine komplexen Transaktionen und Beziehungen
  - Allerdings sind Bedingte Puts/Deletes ermöglicht
  - Nützlich um verlorene Aktualisierungen zu vermeiden

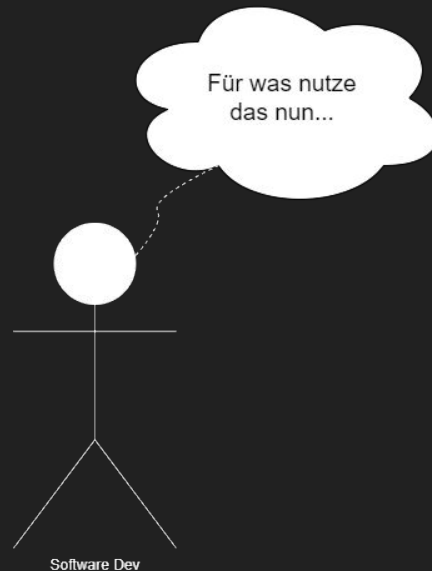
# Use Cases

- Logging
  - 
- Tracking
  - 
- Statusänderungen
  - 

# Mentimeter Umfrage

Für was könnte man SimpleDB verwenden?

menti.com → 2239 2443



# Use Cases

- Logging
  - Server Logs
- Tracking
  - Temperatur / Luftfeuchtigkeit etc.
- Statusänderungen
  - Events
  - Videospiele\*

# Datenspeicherung

- Aufteilung der Daten in verschiedene Indexe
- Speicherung in kleinere Datenelemente
- Verwendung von Laufwerken mit geringer Dichte
- Ähnlich wie Tabellen aus SQL
- Aber
  - Keine festgelegten Typen (z.B. nur Integer)
  - Mehrere Werte pro Zelle möglich
  - Keine Beziehungen zu anderen Domains

# Datenspeicherung – Beispieltabelle

ID	Category	Subcat.	Name	Color	Size	Make	Model
Item_01	Clothes	Sweater	Cathair Sweater	Siamese	Small, Medium, Large		
Item_02	Clothes	Pants	Designer Jeans	Paisley Acid Wash	30x32, 32x32, 32x34		
Item_03	Clothes	Pants	Sweatpants	Blue, Yellow, Pink	Large		
Item_04	Car Parts	Engine	Turbos			Audi	S4
Item_05	Car Parts	Emissions	O2 Sensor			Audi	S4
Item_06	Motorcycle Parts	Bodywork	Fender Eliminator	Blue		Yamaha	R1
Item_07	Motorcycle Parts, Clothing	Clothing	Leather Pants	Black	Small, Medium, Large		

# Kosten

## Monatliche Kosten für Amazon SimpleDB

- Rohe Byte-Größe + 45 Byte Overhead für jedes Element, Attribut, Attribut-Werte-Paar
- Gesamtgröße in GB  $\times$  0.275 USD
- CPU Auslastung in Stunden  $\times$  0.154 USD



# Select – Queries

- Select nur auf eine Domain möglich
- Keine Subqueries möglich
- Ähnlich zu SQL Selects

## Beispiele

- `select * from mydomain where Title = 'The Right Stuff'`
- `select * from mydomain where Pages < '00320'`

# API – Befehle

- CreateDomain
- DeleteDomain
- ListDomain
- DomainMetadata
- (Batch)PutAttributes
- (Batch>DeleteAttributes
- GetAttributes
- Select

# Rest Request – Beispiel

## Request

```
https://sdb.amazonaws.com/?Action=PutAttributes
&DomainName=MyDomain
&ItemName=Item123
&Attribute.1.Name=Color&Attribute.1.Value=Blue
&Attribute.2.Name=Size&Attribute.2.Value=Med
&Attribute.3.Name=Price&Attribute.3.Value=0014.99
&AWSAccessKeyId=your_access_key
&Version=2009-04-15
&Signature=valid_signature
&SignatureVersion=2
&SignatureMethod=HmacSHA256
&Timestamp=2010-01-25T15%3A01%3A28-07%3A00
```

## Response

```
<PutAttributesResponse>
  <ResponseMetadata>
    <StatusCode>Success</StatusCode>
    <RequestId>f6820318-9658-4a9d-89f8-b067c90904fc</RequestId>
    <BoxUsage>0.0000219907</BoxUsage>
  </ResponseMetadata>
</PutAttributesResponse>
```

# Unterstützte Sprachen

- .Net
- C++
- Go
- JavaScript
- Java
- Python
- PHP
- ...

# Python – Beispiel

## Verbinden zur SimpleDB

```
import boto3

client = boto3.client('sdb')
```

## Domain erstellen

```
response = client.create_domain(
    DomainName='string'
)
```

## Werte hinzufügen

```
response = client.batch_put_attributes(
    DomainName='string',
    Items=[
        {
            'Name': 'string',
            'Attributes': [
                {
                    'Name': 'string',
                    'Value': 'string',
                    'Replace': True|False
                },
            ],
        },
    ],
)
```

# Python – Beispiel

## Select-Query

```
response = client.select(  
    SelectExpression='string',  
    NextToken='string',  
    ConsistentRead=True|False  
)
```

## Response

```
{  
    'Items': [  
        {  
            'Name': 'string',  
            'AlternateNameEncoding': 'string',  
            'Attributes': [  
                {  
                    'Name': 'string',  
                    'AlternateNameEncoding': 'string',  
                    'Value': 'string',  
                    'AlternateValueEncoding': 'string'  
                },  
            ],  
        },  
    ],  
    'NextToken': 'string'  
}
```

# Quellen

- 
- [https://docs.aws.amazon.com/de\\_de/AmazonSimpleDB/latest/DeveloperGuide/Welcome.html](https://docs.aws.amazon.com/de_de/AmazonSimpleDB/latest/DeveloperGuide/Welcome.html)
  - <https://boto3.amazonaws.com/v1/documentation/api/latest/reference/services/sdb.html#client>
  - <https://aws.amazon.com/simpliedb/>
  - <https://enlyft.com/tech/products/amazon-simpliedb>
-