



Sallai Tamás

Amazon DynamoDB

Mi ez?

- ▼ NoSQL DB
- ▼ Nem JPA
- ▼ Táblákból áll
- ▼ Kulcs-érték párok
- ▼ Schemaless

Kulcs-érték párok

- ▼ Értékek

- ▼ S,N,B,SS,NS,BS

- ▼ Kulcsok

- ▼ Hash

- ▼ Range

Műveletek

- ▼ Get, put, delete + batch
- ▼ Query
- ▼ Scan

Query / scan

- ▼ Paginálás
- ▼ Condition-ök
- ▼ Limit
- ▼ Exclusive Startkey
- ▼ Forward/back

Hierarchia kialakítása

Blog

- ▼ Hash: id

BlogPost

- ▼ Hash: Blog.id
- ▼ Range: timestamp

Többszintű hierarchia

BlogPostComment

- ▼ Hash: Blog.id + BlogPost.timestamp
- ▼ Range: timestamp

Tranzakcionalitás

Full ACID*

* csak entitáson belül

Tranzakcionalitás

Expected Value Put



Pesszimista és optimista konkurenciakezelés

Local Secondary Index - LSI

- ▼ Új feature
- ▼ Csak az új API-ban
- ▼ Táblán belüli index

Denormalizálás

- ▼ Utolsó komment szerint listázni a blogokat
- ▼ Új tábla:
 - ▼ Hash: konstans
 - ▼ Range: BlogPostComment.timestamp + Blog.id + padding

Multitenancy

- ▼ Adatközpontok között
- ▼ Prefixelt táblák

Költségmodell

- ▼ Provisioned throughput
- ▼ Eventual / strong consistency
- ▼ Query / scan különbség

Előnyök

- ▼ Gyors
- ▼ Megbízható
- ▼ Skálázható
- ▼ Leállásmentes
- ▼ MapReduce

Hátrányok

- ▼ Kifejezőereje kicsi
- ▼ Nincs tranzakcionalitás
- ▼ Költségmodell fura
- ▼ Nincs napon belüli skálázódás

Java

- ▼ Lokális mock: AlternatorDB
- ▼ SDK + Mapper
- ▼ AWS access + secret key
- ▼ Exponential backoff

Mapper

- ▼ Annotációk: Getterre
- ▼ Custom marshallerek
- ▼ Optimista konkurrenciakezelés
- ▼ Query / scan automatikus paginálás

Köszönöm a figyelmet!

Elérhetőségeim:



: gsashee@gmail.com



: <http://www.linkedin.com/in/sallait>



: <https://profiles.google.com/gsashee>



: <https://www.facebook.com/gsashee>