

Tema 2 DATC

Document no-sql database. Azure Cosmos DB.

NIȚU Diana-Adina
An 4 IS, grupa 4.1

Document-oriented databases reprezintă una din principalele categorii de baze de date no-sql. Popularitatea termenului „document-oriented databases” a crescut odată cu utilizarea termenului no-sql în sine.

Ce este mai exact? E un program conceput pentru stocarea, regăsirea și gestionarea informațiilor orientate către documente, cunoscute ca și date semi-structurate.

Printre subclasele pe care document-oriented databases le are se regăsesc bazele de date XML care sunt optimizate pentru a lucra cu documente XML (Extensible Markup Language). De asemenea mai sunt și bazele de date grafice ce sunt similare cu cele XML, diferența constând în stratul „relație” care permite conectarea documentelor pentru traversări rapide.

Document-oriented databases reprezintă în mod inerent o subclasă a store-ului key-value, un alt concept no-sql. Diferența între document-oriented databases și key-value store constă în modul în care sunt procesate datele. Astfel că în key-value store datele sunt considerate a fi opace în mod inerent bazei de date, în timp ce un sistem document-oriented se bazează pe structura internă a documentului pentru a putea extrage metadate pe care motorul bazei de date le folosește pentru optimizarea ulterioară.

Document databases sunt extrem de diferite de bazele de date relaționale. Acestea din urmă stochează în general date în tabele separate definite de programator și un singur obiect poate fi împărțit pe mai multe tabele. Document-oriented databases stochează toate informațiile pentru un anumit obiect într-o singură instanță din baza de date și fiecare obiect stocat poate fi diferit de celălalt. Acest lucru elimină nevoia de mapare obiect-relațională în timp ce se încarcă date în baza de date.

Conceptul central al document-oriented databases este noțiunea de **document**. În timp ce fiecare implementare a bazelor de date orientate spre documente diferă în funcție de detaliile acestei definiții, în general, toate își asumă documentele încapsulând și codificând datele (sau informațiile) într-un format standard sau într-o codificare. Encodurile utilizate includ XML, YAML, JSON și BSON, precum și formulare binare sau documente PDF și Microsoft Office (MS Word, Excel și așa mai departe).

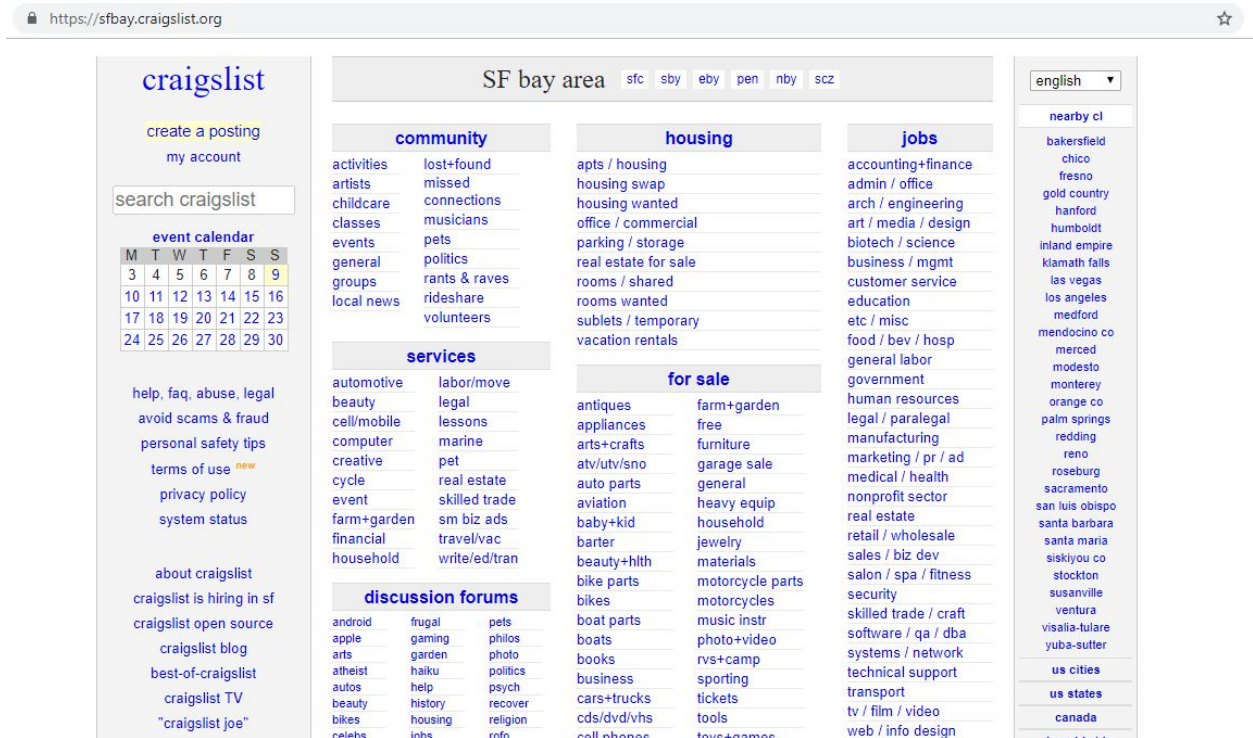
Documentele dintr-un store de documente sunt aproape echivalente cu conceptul de programare al unui obiect. Ele nu sunt obligate să adere la o schemă standard și nici nu vor avea aceleași secțiuni, sloturi, părți sau chei. În general, programele care utilizează obiecte au multe tipuri diferite de obiecte, iar aceste obiecte au deseori multe câmpuri opționale. Fiecare obiect, chiar și cele din aceeași clasă pot arăta foarte diferit. Depozitele de documente sunt similare prin faptul că permit diferite tipuri de documente într-un singur store, permit ca câmpurile din ele să fie opționale și adesea să le permită să fie codificate utilizând sisteme de codificare diferite.

Operațiile de bază pe care le suportă o bază de date orientată pe documente sunt similare cu cele ale altor baze de date și, deși terminologia nu este perfect standardizată, cei mai mulți utilizatori le vor recunoaște ca CRUD (Creation, Retrieval, Update, Deletion):

- Creare (sau inserare)
- Recuperare (sau interogare, căutare, citire, găsim)
- Actualizare (sau editare)
- Ștergere (sau eliminare)

Studiu de caz - [Craiglist](#):

Craiglist este o comunitate populară de anunțuri și locuri de muncă, care deservește 570 de orașe din 50 de țări. Cu 1,5 milioane de anunțuri noi publicate în fiecare zi, Craiglist trebuie să arhiveze miliarde de înregistrări în multe formate diferite și trebuie să poată interoga și raporta aceste arhive în timpul rulării. În trecut, Craiglist și-a păstrat informațiile într-un cluster MySQL, dar lipsa de flexibilitate și costurile de management au devenit niște bariere pentru utilizare. În 2011, Craiglist a migrat peste două miliarde de documente către MongoDB (o bază de date orientată pe documente cu scalabilitatea, flexibilitatea și cu interogarea și indexarea de care numeroși utilizatori au nevoie în ziua de azi) pentru scalabilitatea și schema flexibilă pe care le oferea.



Pentru o bună perioadă de timp, pentru Craigslist, MySQL a fost singura opțiune pentru stocarea datelor, inclusiv arhiva. Cererea inițială de arhivare Craigslist a preluat datele existente din baza de date live și a copiat-o în sistemul de arhivare. Dar folosirea unui sistem de baze de date relaționale a limitat flexibilitatea și a cauzat întârzieri îndelungate, deoarece modificările aduse schemei de bază de date live trebuiau să fie propagate în sistemul de arhivare. Când se efectuau modificări la miliarde de rânduri în clusterul MySQL, aceste modificări nu puteau fi mutate în arhivă. Datele din arhivă au fost strânse în baza de date a producției; performanțele de pe baza de date live s-au deteriorat. Pentru a preveni alte impedimente în calea creșterii și capacității companiei de a-și servi clienții, echipa a început să caute soluții.

După evaluarea mai multor opțiuni NoSQL, Craigslist s-a stabilit la MongoDB. Un motiv convingător pentru care MongoDB a fost ales este că acesta se mândrește cu scalabilitatea încorporată. Fiecare post și metadatele sale pot fi stocate ca un singur document. Pe măsură ce se schimba schema pe baza de date live, MongoDB putea găzdui aceste modificări fără migrații costisitoare ale schemelor. MongoDB a permis Craigslist-ului să ajusteze pe orizontală hardware-ul mărfurilor, fără a fi nevoit să scrie și să mențină un cod complex, personalizat, de împărțire.

Mai mult, folosind auto-sharding, implementarea inițială a MongoDB de către Craigslist a fost proiectată pentru a conține peste 5 miliarde de documente și 10 TB de date.

Conceptele și caracteristicile MongoDB sunt similare, în multe privințe, cu bazele de date relaționale, astfel încât dezvoltatorii Craigslist au putut realiza tranziția fără probleme.

Azure Cosmos DB - O bază de date pentru construirea rapidă de aplicații la scară mondială, cu suport nativ pentru NoSQL.

Azure Cosmos DB a fost construit de la început cu o distribuție globală și o scală orizontală în centrul său. Oferă distribuție globală la cheie în orice număr de regiuni Azure prin scalarea și replicarea transparentă a datelor oriunde s-ar afla utilizatorii.

Azure Cosmos DB oferă suport nativ pentru API-urile NoSQL și OSS, inclusiv MongoDB, Cassandra, Gremlin și SQL. De asemenea, oferă mai multe modele de consistență bine definite, garantează o latență de citire și scriere de 99% și garantează o disponibilitate ridicată de 99,999 cu multi-homing oriunde în lume - toate susținute de către industria de lider, servicii de nivel cuprinzător de servicii (SLAs - service-level agreement).

Distribuție globală a cheii - Proiectat ca un sistem de baze de date distribuit global, Azure Cosmos DB permite scrierea și citirea din replicile locale ale bazei de date Cosmos DB, care este reprodusă în orice număr de regiuni Azure.

Scalabilitate nelimitată și elastică a scrierilor și citirilor - Azure Cosmos DB scalează elastic citirile și scrierile globale și utilizatorii plătesc numai pentru capacitatea de stocare de care au nevoie.

Latență redusă garantată 99% - Cu noul protocol de replicare multi-master și cu motorul bazat pe baze de date optimizate pentru scriere, Azure Cosmos DB garantează latențe mai mici de 10 ms pentru scrierea și citirea (indexată) 99%, în întreaga lume.

Alegeri de consistență bine definite - Protocolul de replicare multi-master Azure Cosmos DB este proiectat pentru a oferi cinci opțiuni de consistență bine definite.

Multi-model cu suport nativ pentru API-urile NoSQL - Azure Cosmos DB permite modelarea datelor din lumea reală utilizând modele cheie-valoare, grafice, coloane-familie și documente de date.

Performanță și securitate la nivel de întreprindere - Azure Cosmos DB oferă securitate la nivel de întreprindere și este primul și singurul serviciu care oferă SLA-uri de top din domeniu.

Concluzie:

Document oriented databases, mai exact bazele de date no-sql reprezintă acum cel mai avansat nivel de baze de date, depășind fără dar și poate bazele de date sql. De ce? Datorită flexibilității și scalabilității pe care le au.

După cum s-a putut observa, este destul de ușor de lucrat cu bazele de date orientate pe documente deoarece acestea se concentrează în jurul documentelor, iar store-urile de documente folosesc conceptul de obiect în ceea ce privește manipularea datelor, două idei cu care utilizatorii sunt extrem de familiarizați având în vedere contextul actual în materia tehnologiei informației.

Mai mult, numeroase platforme, afaceri de succes ce necesită utilizarea bazelor de date fac tranziția ușor-ușor la cele no-sql, manipularea informațiilor făcându-se la fel de simplu.

Bibliografie:

- https://en.wikipedia.org/wiki/Document-oriented_database
- <https://www.mongodb.com/post/15781260117/mongodb-case-study-craigslist>
- <https://www.mongodb.com/what-is-mongodb>
- <https://docs.microsoft.com/en-us/azure/cosmos-db/introduction>