

## TEMA 2 DATC

### PARTITIONAREA DATELOR PENTRU O APLICATIE DE TIP CLOUD

**Cloud computing** este o tehnologie care ne oferă diferite servicii gratuite. Unul dintre serviciile principale este acela că oferă facilitatea de a stoca date într-un cloud(nor), care poate fi accesat oriunde, oricând cu World Wide Web (WWW). Nu numai că reduce costul, dar și reduce riscul de date pierderi.

Mai devreme am văzut cât de ușor este să scalăm nivelul web al unei aplicații cloud, adăugând și eliminând serverele web. Dar, dacă toți accesează aceleași date nivelul de date este cel mai greu scalabil. În acest capitol, ne uităm la modul în care putem face scalabilitatea datelor prin împărțirea datelor în mai multe baze de date relaționate sau prin combinarea stocării bazelor de date relaționate cu alte opțiuni de stocare a datelor.

Pentru a determina dacă aveți nevoie de o strategie de partiționare și care ar trebui să fie, luați în considerare trei întrebări despre datele dvs.:

- Volum - Cât de mult veți stoca în final? Câțiva gigabytes? O sută de gigabytes? Terabytes?
- Viteza - Care este rata la care vor crește datele dvs.? Este o aplicație internă care nu generează o mulțime de date? O aplicație externă pe care clienții vor încărca imagini și videoclipuri?
- Varietate - Ce tip de date veți păstra? Imagini relaționale, imagini, perechi cheie-valoare, grafice sociale?

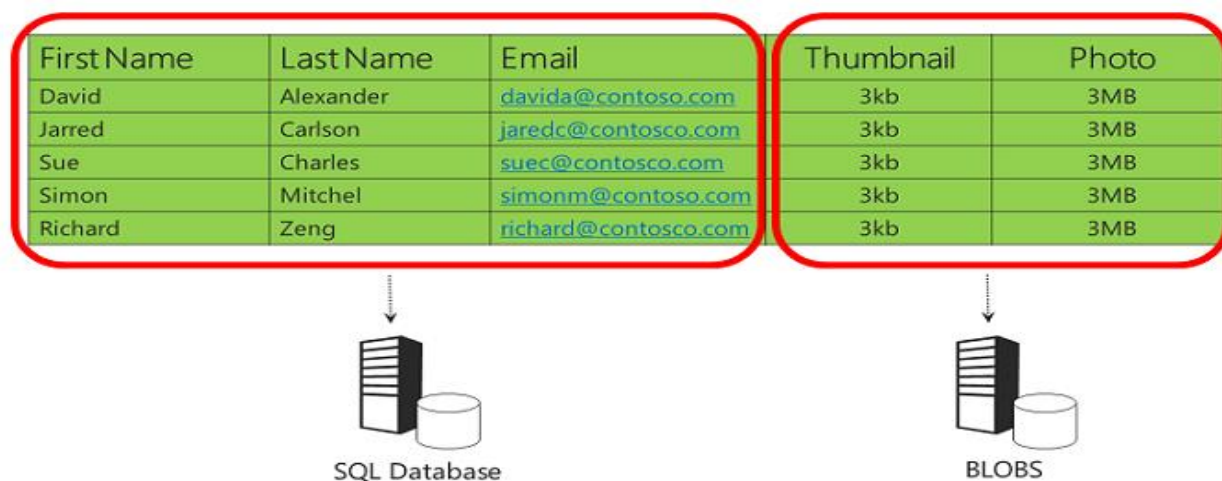
Dacă credeți că veți avea o mulțime de date, viteză sau varietate, trebuie să vă gândiți cu grijă ce tip de sistem de partiționare veți folosi pentru a permite aplicației să se adapteze eficient și pentru a vă asigura că nu vor fi pierderi de date. Există în principiu trei abordări de partiționare:

- **Partiționare verticală**
- **Partiționare orizontală**
- **Partiționare hibridă**

1. **Partiționare verticală** - este ca împărțirea unui tabel pe coloane: un set de coloane trece într-un singur depozit de date și un alt set de coloane trece într-un alt depozit de date

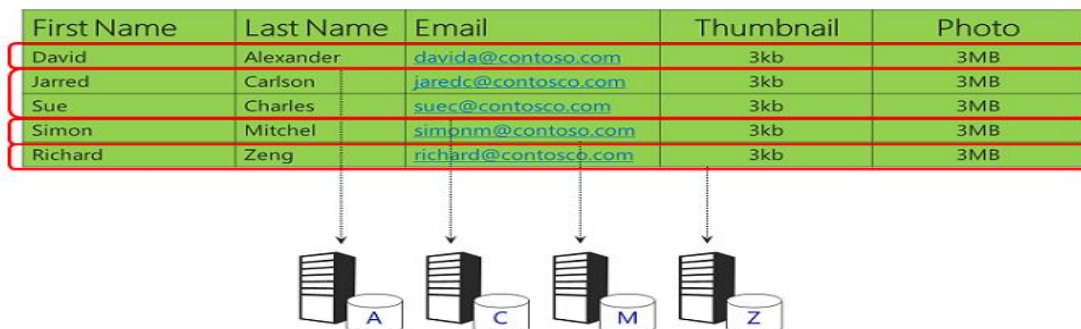
| First Name | Last Name | Email  | Thumbnail | Photo |
|------------|-----------|--|-----------|-------|
| David      | Alexander | <a href="mailto:davida@contoso.com">davida@contoso.com</a>     | 3kb       | 3MB   |
| Jarred     | Carlson   | <a href="mailto:jaredc@contosco.com">jaredc@contosco.com</a>   | 3kb       | 3MB   |
| Sue        | Charles   | <a href="mailto:suec@contosco.com">suec@contosco.com</a>       | 3kb       | 3MB   |
| Simon      | Mitchel   | <a href="mailto:simonm@contoso.com">simonm@contoso.com</a>     | 3kb       | 3MB   |
| Richard    | Zeng      | <a href="mailto:richard@contosco.com">richard@contosco.com</a> | 3kb       | 3MB   |

Atunci când reprezentați aceste date ca tabele și vedeți diferitele tipuri de date, puteți observa că cele trei coloane din stânga au date de tip șir care pot fi stocate eficient într-o bază de date relațională, în timp ce cele două coloane din dreapta sunt în esență array-uri de octeți care provin din fișiere imagine. Este posibil să stocați date de tip imagine într-o bază de date relațională și o mulțime de oameni fac asta deoarece nu doresc să salveze datele în sistemul de fișiere. Este posibil ca aceștia să nu aibă un sistem de fișiere capabil să stocheze volumele de date necesare sau să nu vrea să gestioneze un sistem separat de backup și restaurare. Această abordare funcționează bine pentru bazele de date locale și pentru cantități mici de date în baze de date cloud. În mediul local, ar putea fi mai ușor să lăsați DBA să aibă grijă de tot. Dar într-o bază de date cloud, stocarea este relativ scumpă și un volum mare de imagini ar putea face ca dimensiunea bazei de date să crească dincolo de limitele la care poate funcționa eficient. Puteți rezolva aceste probleme prin împărțirea verticală a datelor, ceea ce înseamnă că **alegeți cel mai adecvat mod de stocare pentru fiecare coloană din tabelul dvs. de date**. Ceea ce ar putea funcționa cel mai bine pentru acest exemplu este de a pune datele de șir într-o bază de date relațională și imaginile din spațiul de stocare Blob.



Stocarea imaginilor în spațiul de stocare Blob în locul unei baze de date este mai practică în cloud decât într-un mediu local, deoarece nu trebuie să vă faceți griji în ceea ce privește configurarea serverelor de fișiere sau gestionarea datelor de rezervă și restaurarea datelor stocate în afara bazei de date relaționale : tot ceea ce vă gestionează automat serviciul de stocare Blob.

- 2. Partiționare orizontală** - partiționarea orizontală este ca și cum ați diviza un tabel pe rânduri: un set de rânduri intră într-un singur magazin de date, iar un alt set de rânduri intră într-un alt magazin de date.

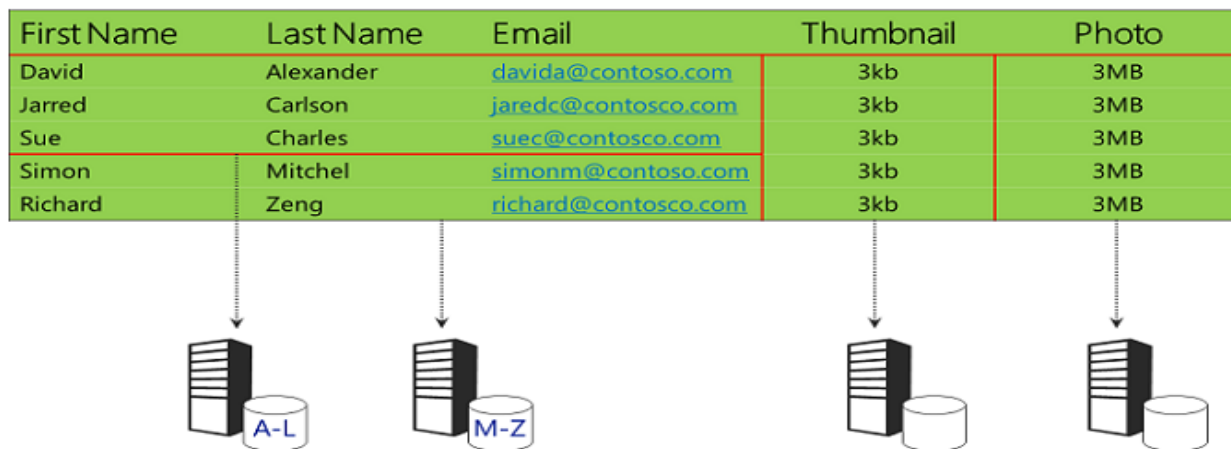


În exemplul de mai sus, presupunându-se că vom avea o bază de date vastă de clienți, s-a împărțit în mai multe tabele(una pentru numele care încep cu litera A, alta cu C, alta cu M si alta cu Z).

Un dezavantaj al partiționării orizontale este că ar fi greu să faci interogări în toate datele. În acest exemplu, o interogare ar trebui să tragă din până la 26 de baze de date diferite pentru a obține toate datele stocate de aplicație.

În această tehnică, datele au fost împărțite orizontal și apoi aceste partiții stocate pe diferite mașini. Această tehnică urmează principiul partiționării statice deci partițiile odată formate, nu se pot schimba. Ele rămân aceleași pentru totdeauna.

- 3. Partiționarea hibridă**– se pot combina partiționarea verticală și orizontală. Ca în exemplul de mai jos, se pot stoca imaginile în memoria Blob și se pot împărți orizontal datele de tip șir



În concluzie, o schemă eficientă de partiționare poate permite aplicației dvs. din cloud să scadă până la câțiva petabytes de date din cloud fără blocaje. Și nu trebuie să plătiți în fața mașinilor masive sau a unei infrastructuri extinse, cum ar fi dacă utilizați aplicația într-un centru de date local. În nor puteți să adăugați în mod incremental capacitatea de care aveți nevoie și veți plăti doar pentru cât utilizați atunci când o utilizați.