

Løsningsskisse til eksamen i TDT4150 13. desember 2014

Oppgave 1 – Spørreoptimalisering – 25 % (5 % på a/c/d, 10 % på b)

- a) Anta at en spørring inneholder bare en seleksjon og en projeksjon. Når kan det være lurt å utføre seleksjon før projeksjon og omvendt?

Svar: Som regel mest interessant å fjerne rader => select.

Men hvis project fjerner mange kolonner og select fjerner få rader => project.

Project kan ikke utføres først hvis den fjerner en kolonne som brukes av en betingelse i select.

Gitt spørringen:

`SELECT * FROM r1, r2 WHERE r1.c1 = r2.c2 AND r2.c3 > 5;`

- b) Hvilken informasjon bør databasesystemet ha for å kunne estimere størrelsen på resultatet av denne spørringen? Lag et symbolsk uttrykk for estimatet? Beskriv de antagelsene du gjør.

Svar: Antall tupler i r1 og r2.

Antall distinkte verdier for r1.c1 og r2.c2

Største og minste verdi for r2.c3

Reduksjonsfaktor for r2: $(\text{maxval} - 5) / (\text{maxval} - \text{minval})$

Reduksjonsfaktor for join: $1 / (\text{max}(\text{unique } r1.c1, \text{unique } r2.c2))$

- c) Hva er fordeler og ulemper med *equiwidth* histogram i forhold til *equidepth*?

Svar: Equiwidth tar mindre plass og er kjappere å bruke (enklere å slå opp)

Equidepth gir mer nøyaktige estimater ved store variasjoner

- d) Anta at tre relasjoner skal joines ved hjelp av *nested loop*. Hvordan bør utføringsplanen være for at man skal oppnå best mulig samlebåndsparallelitet? Begrunn svaret.

Svar: Resultatet av en join bør være ytre relasjon for neste join ettersom hele indre relasjon må være tilgjengelig.

Oppgave 2 – Parallele og distribuerte databaser – 15 % (10% på a og 5 % på b)

- a) Forklar de fire meldingene som blir brukt i tofase-commit (2PC).

Svar: (Variasjoner er OK, avhengig av hvilke pensumlitteratur/foiler de baserer svaret på).

Foiler: Prepare, Vote Commit/Abort, Global Commit/abort, Ack (Forklaring for hver forventes)

Basert på kapittel i Coulouris:

canCommit?(trans) → Yes / No

Call from coordinator to participant to ask whether it can commit a transaction.

Participant replies with its vote.

doCommit(trans)

Call from coordinator to participant to tell participant to commit its part of a transaction.

doAbort(trans)

Call from coordinator to participant to tell participant to abort its part of a transaction.

haveCommitted(trans, participant)

Call from participant to coordinator to confirm that it has committed the transaction.

- b) Ved bruk av masterkopi (primary copy) replikering, har alle objekter (f.eks. en relasjon) en masternode som mottar alle oppdateringer. Bør man bruke samme masternode for alle objekter?

Svar: Ved å ha forskjellige masternoder sprer man last.

Ved å ha samme masternode kan det bli enklere å oppdage konflikter - f.eks. ved låsing.

Oppgave 3 – Skyline – 15 %

- a) Anta en relasjon med todimensjonale data. Forklar hva som er vilkåret for at et tuppel i denne relasjonen skal være med i resultatet av *skyline*-operatoren anvendt på denne relasjonen. Bruk deretter *skyline* på relasjonen R til høyre, og vis hva som er resultatet. Begrunn svaret.

Svar:

Vilkår: *Tuppel ikkje dominert av annan tuppel. Eit punkt dominerer eit anna punkt om det er like godt eller betre i alle dimensjonar og betre i minst ein dimensjon (forventar full forklaring).*

Resultat: *(1,1) (om noen bruker max istedet for min og får rett resultat vil det også bli godkjent)*

X	Y
3	10
5	3
1	6
1	1
1	10
6	1
10	1
1	4
4	3
3	8

- b) Lag et SQL-uttrykk som gjør en skyline-spørring på en relasjon med to attributter som den til høyre (med SQL uten støtte for skyline-operatoren).

Svar: *(Flere varianter kan være korrekt her)*

```
SELECT *
FROM R r
WHERE NOT EXISTS(
  SELECT *
  FROM R rl
  WHERE rl.X <= r.X AND
        rl.Y <= r.Y AND
        (rl.X < r.X OR
         rl.Y < r.Y));
```

Oppgave 4– Kolonnebaserte databasesystemer – 15 % (5 % på a og 10 % på b)

- a) Hva er en projeksjon i C-Store?

Svar: *Fysisk modell i C-Store er basert på projeksjoner. En eller flere kolonner fra én logisk tabell og null eller flere kolonner fra andre tabeller (kun hvis koblet via fremmednøkler til hovedtabellen). Gir max 7p om de ikke har med aspektet med andre tabeller.*

- b) Forklar de tre forskjellige metodene for komprimering av kolonner i C-Store og når disse bør brukes.

Svar:

Sortert, få verdier: (verdi, første radnr, siste radnr)

Usortert, få verdier: (verdi, bitmap av rader med verdi)

Sortert, mange verdier: Lagring av deltaverdier.

Oppgave 5 – Diverse – 30 % (12 % på a/b, 6 % på c)

- a) Forklar de 4 hovedmodulene i en *lagringshandler* (storage manager) i et databasesystem.

Svar: *Forventes detaljer rundt Låshandterer, Logg-handterer, Buffer-pool, og Aksessmetoder. Jfr. Anatomy/Architecture-artikkelen.*

- b) Forklar arkitekturen til SkimpyStash (datastrukturer og hvordan de blir brukt).

Svar: *Jfr. artikkel. "SkimpyStash: RAM space skimpy key-value store on flash-based storage"*

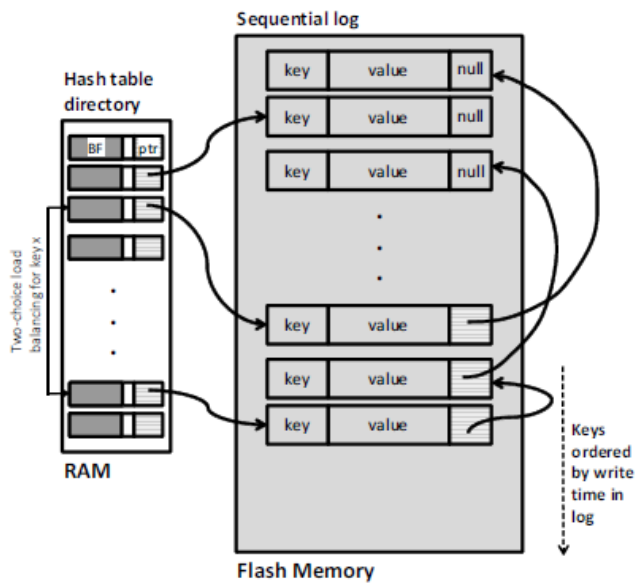


Figure 5: SkippyStash architecture showing the sequential log organization of key-value pair records on flash and enhanced design for the hash table directory in RAM. (RAM write buffer is not shown.)

c) Forklar teknikken ”swizzling”.

Svar: Konvertering av objekter fra disk-format til minneformat, pekere som peker på disk/blokk-adresser peker til objektene i minne i stedet, jfr. artikkelen “In-Memory Performance for Big Data”.