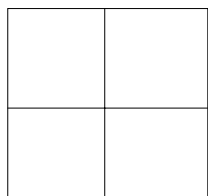
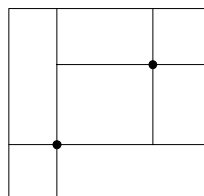


Kap. 11 Grid-File

Nievergelt, Hinterberger, Sevcik: The Grid-File.
ACM – TODS, 9, 1, März 1984



Partitionierung des
Suchraumes bei
Region-QT

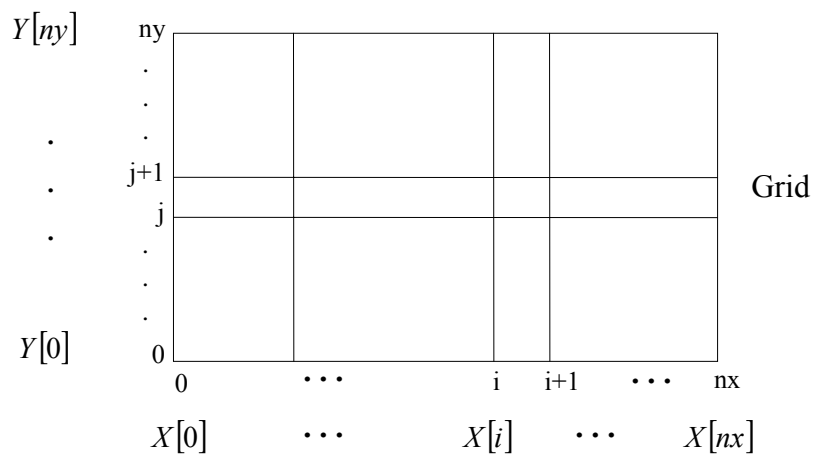


Partitionierung der
Schlüsselmenge bei
Point-QT, auch bei
B-Baum

Bei Grid-F: schlüsselabhängige Raum – Partitionierung, hybride DS.

1

1. Trennschlüsselvektoren X, Y



2

2. Grid:

Matrix von Zeigern auf Buckets

$G[0 : nx - 1, 0 : ny - 1]$

$G[i, j]$ zeigt auf Bucket mit Schlüssel (x, y) , bzw. $((x, y), \alpha)$ und Bedingung:

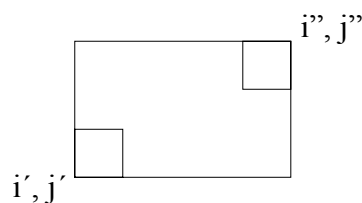
$$X[i] \leq x < X[i + 1]$$

$$Y[j] \leq y < Y[j + 1]$$

i.a. zeigen mehrere Grid-Einträge auf denselben Bucket
(wie bei dynamic hashing)

3

Konvexitäts Bedingung für Gridausschnitt:



$$G[i', j'] = G[i'', j'']$$

$$\Rightarrow \forall i, j \quad \text{mit } i' \leq i \leq i'' \quad \text{und} \\ j' \leq j \leq j'' \quad \text{gilt:}$$

$$G[i', j'] = G[i, j] = G[i'', j'']$$

Hinw.: Konvexitätsbedingung auf G entsteht automatisch durch Spaltvorgänge.

Grundidee von Grid-File:

Vektoren X, Y von Trennschlüsseln stehen im AS. Grid G auf Platte irgendwie paginiert.

4

Suchalgorithmus: (x, y) :

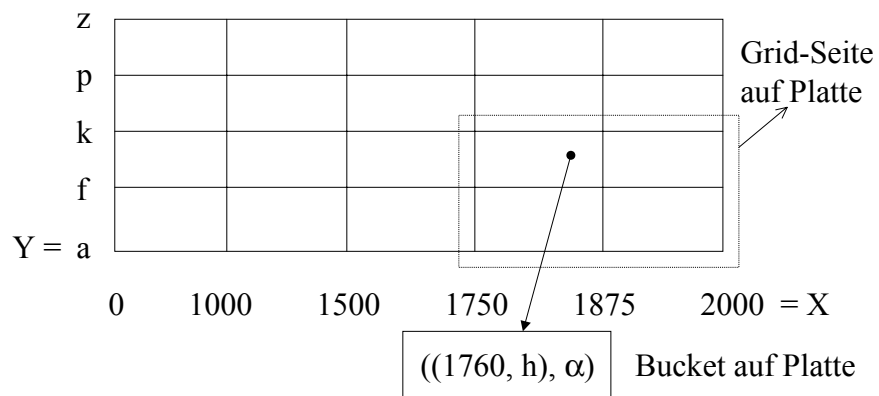
1. Bestimme i mit $X[i] \leq x < X[i+1]$
j mit $Y[j] \leq y < Y[j+1]$
2. Berechne Adresse der Seite im paginierten G, die $G[i, j]$ enthält: $P(i, j)$
3. Hole $P(i, j)$ von Platte
4. Extrahiere $G[i, j]$ aus $P(i, j)$
5. Hole bucket, auf den $G[i, j]$ zeigt, enthält gesuchten Satz
6. Extrahiere und bearbeite Record $((x, y), \alpha)$
 \Rightarrow 2 Plattenzugriffe für Suche!

5

Beispiel:

$X = (0, 1000, 1500, 1750, 1875, 2000)$

$Y = (a, f, k, p, z)$



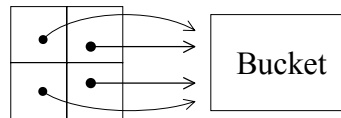
nur X, Y im AS plus gecachte Grid-Seiten u. Buckets

6

Split-Vorgänge bei insert:

Bei B-Bäumen einheitlicher Split-Algorithmus, hier 2 wesentlich verschiedene: Grid-Split und Bucket-Split.

1. Bucket-Split: wenn mehrere "buddy-Zeiger" auf dieselbe Bucket-Seite zeigen, z.B.



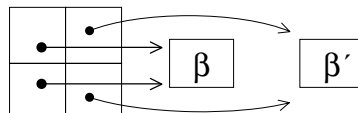
$$\beta = G[i, j] = G[i + 1, j] = G[i, j + 1] = G[i + 1, j + 1]$$

siehe Konvexitätsbedingung

7

Spalte Bucket β in β und β' längs X-Achse oder Y-Achse,
verteile $((x, y), \alpha)$ entsprechend neu auf β und β' ,
schreibe Buckets β und β' neu,
schreibe Grid-Seite neu

bei Spaltung längs X-Achse entsprechend vorhandenem
Trennschlüssel:



8

2. Grid-Split: z.B. In X-Achse bei 1630

2.1 neuer Trennschlüssel 1630 durch insert in X-Struktur

2.2 $G[i, j + 1] := G[i, j]$

$\forall j : k < j \quad \forall i$ wenn Trennschlüssel an $X[k]$ eingefügt wird.

i.e. Verschiebung des Grids u. neue Paginierung auf Platte

d.h. manchmal ganz G neu schreiben

Hinweis: Durch viele Schreibvorgänge $O(n)$ wird "bezahlt" für 2 Lesevorgänge beim Suchen.

9

Datenstruktur für G? Erfordert Operationen

- einschieben einer Zeile
- einschieben einer Spalte } in G

i.a. n-Dimension-Grid, einschieben einer n-1 dimensionalen Untermatrix = Hyperebene in einem Hyperrechteck

10