

B-Baum

- Für den Hintergrundspeicher optimierte Bäume, deren Knotengrößen auf die Seitenkapazität abgestimmt werden (1 Knoten entspricht einer Seite)
- Seitenwechsel ist nur notwendig, wenn eine Kante verfolgt wird -> Max. Anzahl Seitenzugriffe entspricht also der Höhe des Baums
- Balancierter Baum -> Jede Weg von Wurzel zu Blatt ist gleich lang
- Jeder Eintrag besteht aus Schlüssel S_i und Datensatz D_i (der beim Sekundärindex einer TID entspricht).
- Zusätzlich gibt es jeweils einen Verweis V_{i-1} (Knoten mit kleineren Schlüsseln) und V_i (Knoten mit größeren Schlüsseln)

B-Baum – Eigenschaften mit Grad k

1. Jeder Weg von der Wurzel zu einem Blatt hat die gleiche Länge.
2. Jeder Knoten außer der Wurzel hat mindestens k und höchstens $2k$ Einträge. Die Wurzel hat zwischen einem und $2k$ Einträgen. Die Einträge werden in allen Knoten sortiert gehalten.
3. Alle Knoten mit n Einträgen, außer den Blättern, haben $n+1$ Kinder.
4. Seien S_1, \dots, S_n die Schlüssel eines Knotens mit $n+1$ Kindern. V_0, V_1, \dots, V_n seien die Verweise auf diese Kinder. Dann gilt:
 - a) V_0 weist auf den Teilbaum mit Schlüsseln kleiner als S_1
 - b) $V_i (i=1, \dots, n-1)$ weist auf den Teilbaum, dessen Schlüssel zwischen S_i und S_{i+1} liegen.
 - c) V_n weist auf den Teilbaum mit Schlüsseln größer als S_n .
 - d) In den Blattknoten sind die Zeiger nicht definiert.

B-Baum – Einfügen

1. Führe eine Suche nach dem Schlüssel durch; diese endet an der Einfügestelle.
2. Füge den Schlüssel dort ein.
3. Ist der Knoten überfüllt, teile ihn:
 - Erzeuge einen neuen Knoten und belege ihn mit den Einträgen des überfüllten Knotens, deren Schlüssel größer ist als der des mittleren Eintrags.
 - Füge den mittleren Eintrag im Vaterknoten des überfüllten Knotens ein.
 - Verbinde den Verweis rechts des neuen Eintrags im Vaterknoten mit dem neuen Knoten.
4. Ist der Vaterknoten jetzt überfüllt?
 - Handelt es sich um die Wurzel, so lege eine neue Wurzel an.
 - Wiederhole Schritt 3 mit dem Vaterknoten.

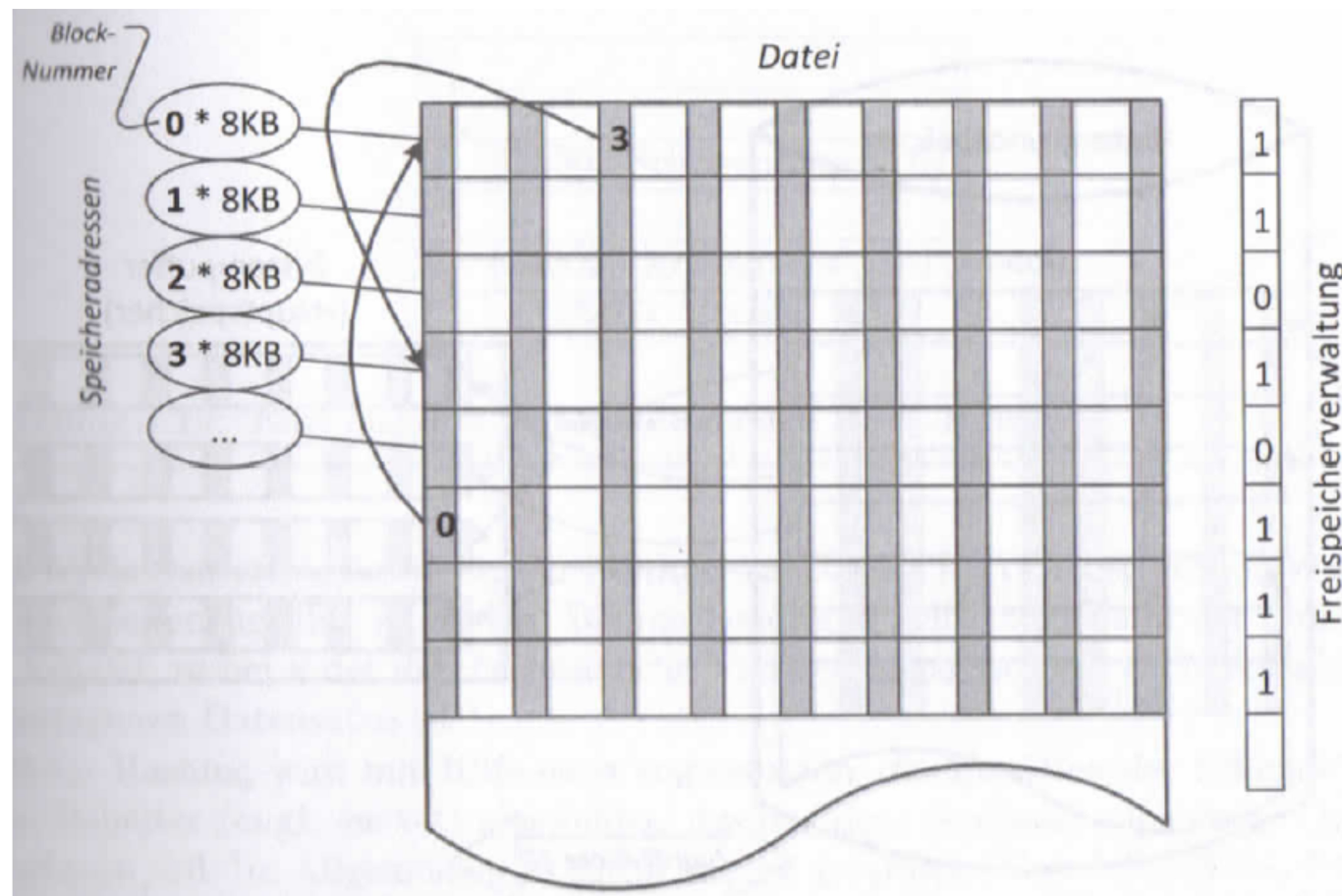
B-Baum - Löschen

- Befindet sich der Eintrag in einem Blattknoten, kann er einfach entfernt werden.
- Bei einem inneren Knoten wird der nächstgrößere (oder nächstkleinere) Schlüssel gesucht und an die Stelle des alten Schlüssels plaziert.
- Bleibt ein Blattknoten unterbelegt, wird der Knoten mit einem seiner Nachbarn entweder ausgeglichen oder verschmolzen (letzteres ist nur möglich, wenn beide Knoten minimal belegt sind).
- Beim Verschmelzen tritt an deren Stelle ein Knoten, der zusätzlich zu deren Inhalt noch den zugehörigen Schlüssel aus dem Vaterknoten enthält (kann somit eine Unterbelegung des Vaterknotens auslösen....).

Hintergrundspeicher-Struktur B-Baum

- Knoten eines Baumes entsprechen Seiten des Hintergrundspeichers.
- Seiten (zumindest die der inneren Knoten) sollten möglichst im Hauptspeicher (Pufferspeicher) gehalten werden. (Zugriffslücke von 10^5)
- Reihenfolge egal (Wurzel muss nicht an erster Stelle stehen, da sich der Baum ja dynamisch verändern kann).
- Einsatz einer Freispeicherverwaltung als Bitvektor, um Reorganisationsaufwand gering zu halten.

Hintergrundspeicher-Struktur B-Baum



Quelle: Kemper, Alfons & Eickler, André: Datenbanksysteme (9. Auflage), Oldenbourg Verlag, 2013, Seite 231



Berechnen Sie den möglichen Verzweigungsgrad unter folgender Annahme:

- Seitengröße: 8KB, Zeigergröße: 4 Byte, TID: 8 Byte, Schlüssel: 4 Byte (long)