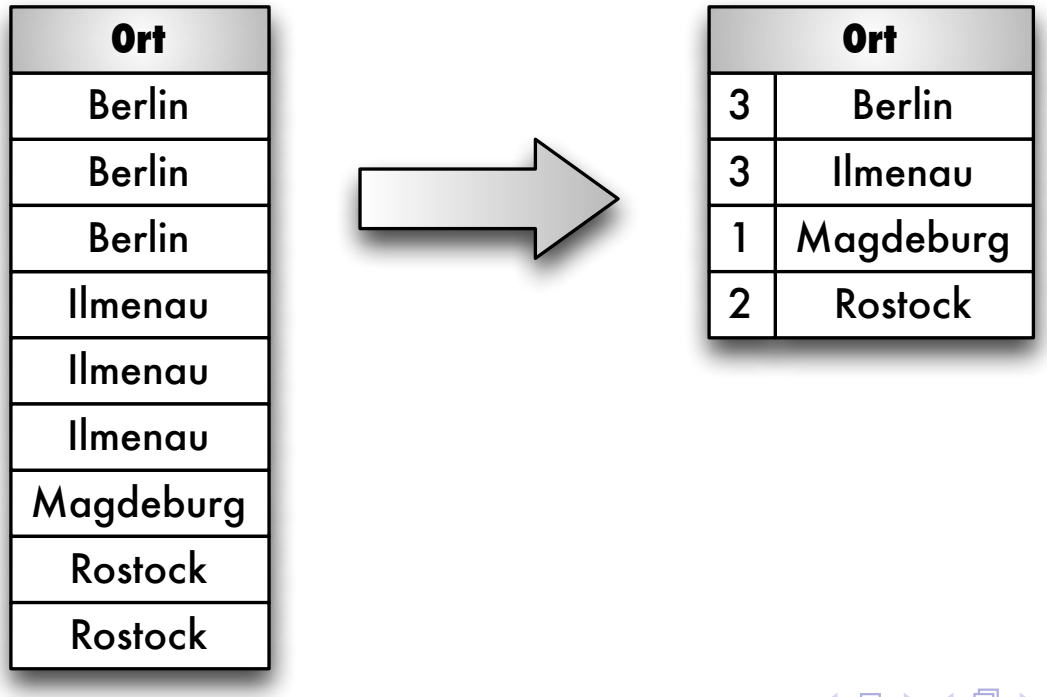
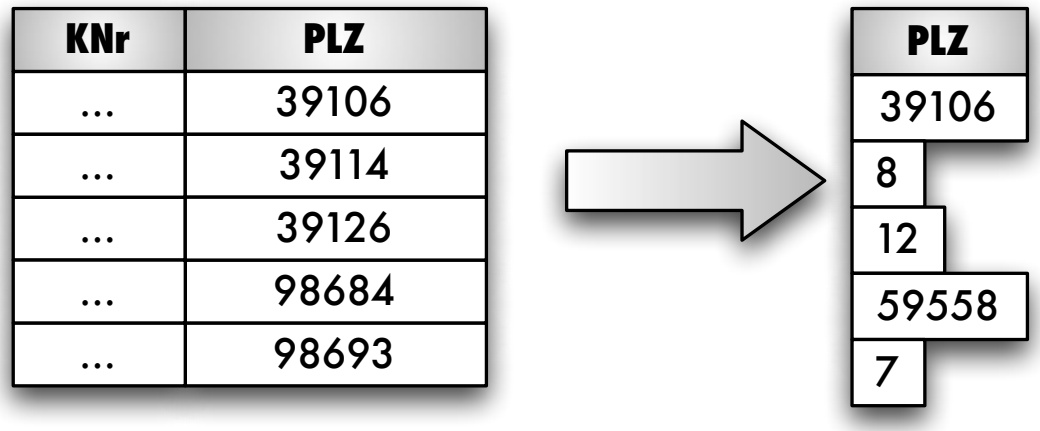
Run Length Encoding

- auch Lauflängenkodierung
- lange Folgen gleicher Werte werden durch das einmalige Speichern des Wertes zusammen mit der Häufigkeit der Wiederholung ersetzt
- insbesondere bei spaltenorganisierter Datenorganisation; durch Sortierung weiter unterstützt



Delta Coding

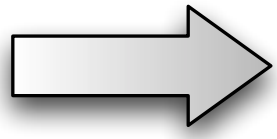
- Speicherung der Wertdifferenz zum Vorgänger anstelle des Wertes
- insbesondere bei aufeinanderfolgenden Werten mit geringer Differenz
- Unterstützung durch Sortierung



Bit-Vector Encoding

- bei kleiner Anzahl verschiedener Werte: pro Spaltenwert ein Bitstring (1, wenn Tupel an der Position den Wert hat, sonst 0)
- Länge des Bitstrings entspricht Anzahl der Tupel
- Verwendung u.a. bei Bitmap-Indexen

KNr	Kundenstatus
...	Premium
...	Silber
...	Standard
...	Standard
...	Standard
...	Premium
...	Silber
...	Standard

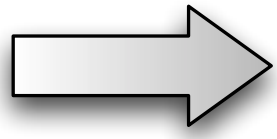


Premium:	1000.0100
Silber:	0100.0010
Standard:	0011.1001

Dictionary Encoding

- Verwendung eines Wörterbuchs für alle (String-)werte und Eintrag eines Codes für den eigentlichen Spaltenwert
- insbesondere bei häufigen und langen Werten

KNr	Bundesland
...	Thüringen
...	Thüringen
...	Sachsen
...	Sachsen-Anh.
...	Hessen
...	Bayern
...	Hessen
...	Sachsen-Anh.



Bundesland
0100
0100
0010
0011
0001
0000
0001
0011

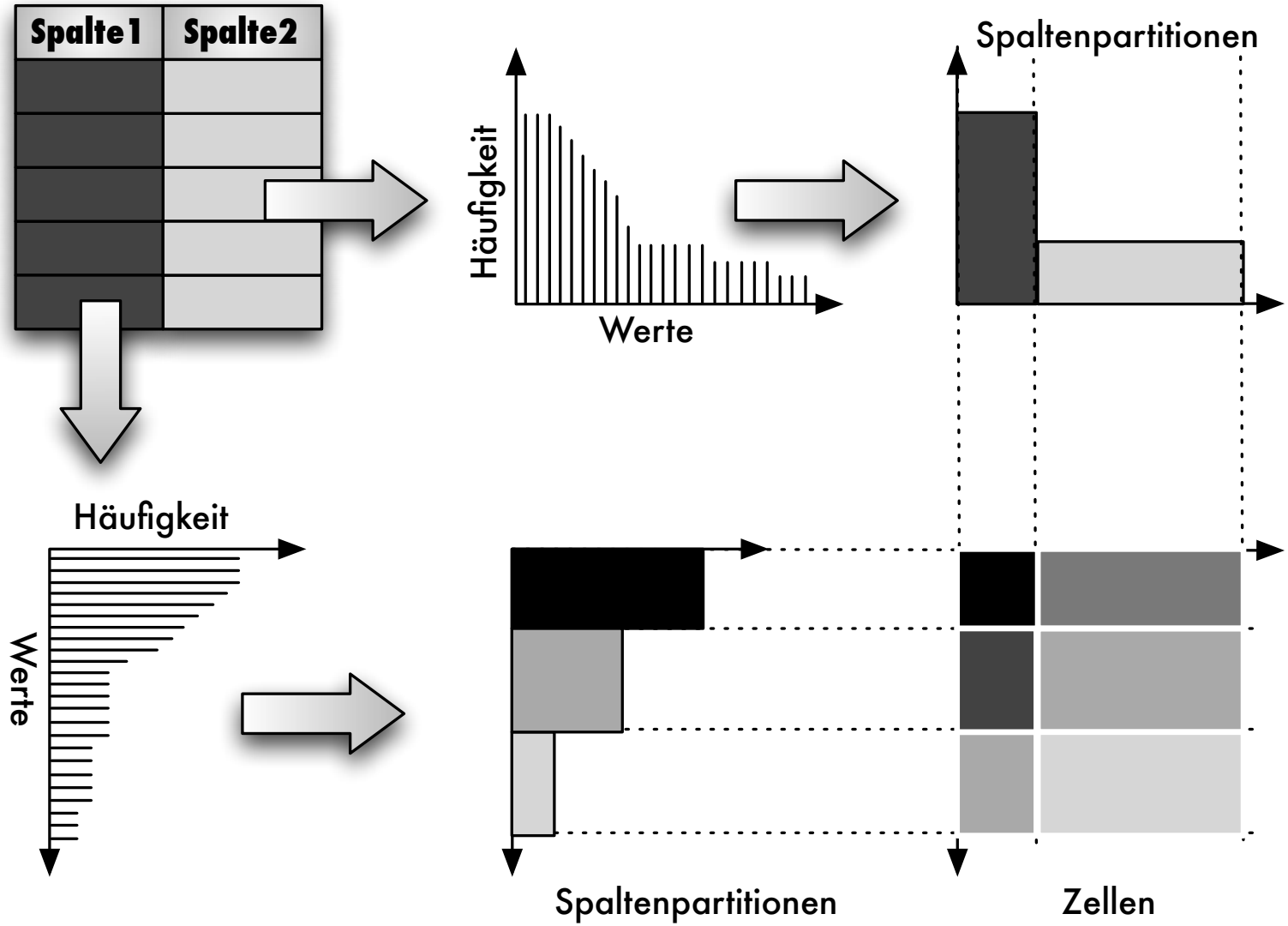
Dictionary

Bayern	0000
Hessen	0001
Sachsen	0010
Sachsen-Anh.	0011
Thüringen	0100

Frequency Partitioning

- entwickelt im Rahmen des BLINK-Projektes von IBM
- Kompression kompletter Tupel, Kodierung der Spaltenwerte durch Dictionary Encoding mit Entropiekodierung
- Vermeidung des Overheads beim Lesen des n-ten Spaltenwertes durch Codes unterschiedlicher Länge (2 ns Overhead pro Spaltenwert)
- Idee: Gruppierung von Tupeln auf Basis der Spaltenwerte derart, dass Gruppen von Tupeln (Partitionen) Spaltencodes gleicher Länge haben
  - Partitionen nach Häufigkeit des Vorkommens der Spaltenwerte bilden
  - Partitionen mit Entropiekodierung komprimieren
  - pro Partition werden feste Codelängen verwendet

Frequency Partitioning: Prinzip



Frequency Partitioning: Partitionierung

- Sortierung der Spaltenwerte nach Häufigkeiten + Zerlegung in Intervalle
- Intervallgröße: Zweierpotenz (da Bitkodierung) mit Ausnahme des letzten Intervalls
- optimale Partitionierung durch dynamische Programmierung: Zielfunktion = durchschnittliche Größe der kodierten Spalte
- vollständige Suche über alle Kombinationen von Spaltenpartitionen nicht möglich, daher Greedy: welche Spalte zieht den größten Gewinn aus einer zusätzlichen Partition?