1 Generelles

Erläutern Sie die Motivation hinter der Datenbanktheorie. Gehen Sie insbesondere auf die Probleme ein, und beschreiben Sie was es heißt wenn ein Datum integer ist.

fhwedel
UNIVERSITY OF APPLIED SCIENCES

Motivation

- Daten sollen dauerhaft gespeichert sein und jederzeit schnell verfügbar sein
- Integrität der Daten muss zu jedem Zeitpunkt gewährleistet sein (Richtigkeit d. Daten)
- Mehrbenutzerbetrieb mit stark unterschiedlichen Nutzungsszenarien (Endanwender, Programmierer, Administrator)
- Probleme hierbei:
  - Konflikte um Ressourcen -> Anfrageoptimierung notwendig
  - Inkonsistente Zustände -> Deadlocks
  - Transaktionsverwaltung -> 2 Phasen-Sperrprotokoll
  - Datenschutz/Autorisierung -> Grant/Revoke

Um diese Aufgaben zu erleichtern und einzelne Bereiche eines DBMS unabhängig vom anderen zu machen, entstanden mehrstufige Architekturmodelle

Datenbanktheorie und Implementierung - FH-Wedel SS 2019 - Michael Predeschly 3

Erläutern Sie das ANSI-SPARC Schema. Welches Ziel wird hierbei verfolgt.

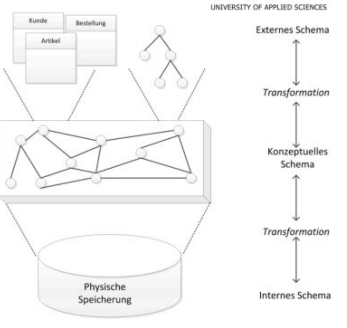
fhwedel
UNIVERSITY OF APPLIED SCIENCES

Architektur von Datenbanken

ANSI/SPARC 3 Schema-Architektur (1975)

Unterscheidung von drei Ebenen

- **Externe Schema** (Benutzersicht)
- **Konzeptuelles Schema** (Referenzschema)
- **Internes Schema** (physische Realisierung)



Ziel:
Trennung der einzelnen Schichten, um Modularität und Austauschbarkeit zu gewährleisten (einzelne Schichten müssen nur Schnittstellen kennen).

Datenbanktheorie und Implementierung - FH-Wedel SS 2019 - Michael Predeschly 4

Wie werden Daten auf einer Magnetplatte adressiert? Nennen Sie sowohl den Überbegriff als auch die einzelnen Komponenten aus denen sich dieser zusammensetzt.

Wägen Sie ab warum es sinnvoll sein kann bei der Speicherung von Daten auf einer Magnetplatte auf größere Datenblöcke zu setzen.

Bei zu kleinen Datenblöcken müssen die zu speichernden Daten öfter aufgeteilt werden. Da diese im Zweifelsfall über die gesamte Platte verteilt sein können wird viel Zeit für diverse Schreib- und Lesevorgänge benötigt.

Bei zu großen Datenblöcken kann es vorkommen, dass eine kleines Datum (z.B. eine Zahl) nur einen geringen Teil der verfügbaren Blockgröße verwendet.

Erläutern Sie grob wie ein Raid-System funktioniert. Was versteht man unter dem sog. Striping sowie dem Mirroring ? ¡missing IDENTIFIER¿

fhwedel
UNIVERSITY OF APPLIED SCIENCES

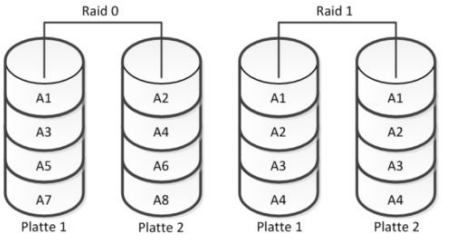
Physische Speicherung - Raid

Festplatten können zu sogenannten RAIDs (**R**edundant **A**rray of **I**ndependent **D**isks) zusammengeschalten werden, um die Ausfallsicherheit oder die Zugriffsgeschwindigkeit zu erhöhen.

Wie funktioniert ein Raid?

RAID 0 - Striping (max. Performance)
↳ Daten werden auf zwei Platten aufgeteilt

RAID 1 - Mirroring
↳ Daten werden kopiert & auf zwei Platten gespeichert

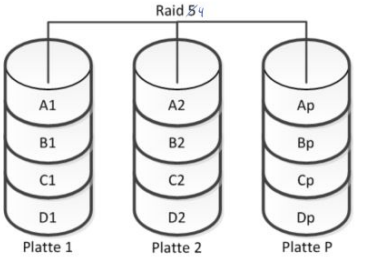


Erläutern Sie was man unter einem Raid-4 System versteht. Wie wird hierbei die Parität ermittelt?

fhwedel
UNIVERSITY OF APPLIED SCIENCES

RAID 5 - Parität auf mehreren Platten

- Vorteile von RAID 0 und RAID 1 kombiniert
- Parity Information auf Platte P erlaubt bei Ausfall einer Platte die Wiederherstellung der Daten der ausgefallenen Platte.
- RAID 5 ersetzt kein Backup!!
- Werden Daten gelöscht, können diese auch mit der Parity-Information nicht wiederhergestellt werden



fhwedel
UNIVERSITY OF APPLIED SCIENCES

Beispiel Paritätsberechnung mittels XOR

Platte 1: `0101 1110...`

Platte 2: `1110 0011...`

Paritätsplatte: `1011 1101...`

Wie vorgehen bei mehr als 2+1 Platten?

x	y	x XOR y
0	0	0
0	1	1
1	0	1
1	1	0

¡img src=paste-18150531792897.jpg

a) Berechnen Sie die Parity Information der Platte P.
b) Nach erfolgreicher Parity-Berechnung fällt Platte 2 aus. Rekonstruieren Sie diese auf einer Platte 2'. Wichtig ist hierbei der Rechenweg.

¡/div¿¡missing IDENTIFIER¿

fhwedel
UNIVERSITY OF APPLIED SCIENCES

Übungsaufgabe

Gegeben sind folgende 4 Platten:

Platte 1: 1111 0111
Platte 2: 1011 1100
Platte 3: 0111 0011

Pl.1	1 1 1 1	0 1 1 1
Pl.3	0 1 1 1	0 0 1 1
Par.	0 0 1 1	1 0 0 0
Pl.2	1 0 1 1	1 1 0 0

Platte P: 0011 1000

a) Berechnen Sie die Parity Information der Platte P.
b) Nach erfolgreicher Parity-Berechnung fällt Platte 2 aus. Rekonstruieren Sie diese auf einer Platte 2'. Wichtig ist hierbei der Rechenweg.

Datenbanktheorie und Implementierung - FH-Wedel SS 2019 - Michael Predeschly 9

Warum erhöht ein Raid 4 die Ausfallsicherheit, ersetzt aber kein Backup?

Manuelles Löschen führt weiterhin zum Datenverlust.

Weswegen wird in der Datenbanktheorie eine möglichst effiziente Pufferverwaltung angestrebt? Beschreiben Sie grob wie so eine aussehen könnte.

fhwedel
UNIVERSITY OF APPLIED SCIENCES

Motivation

Wenn Daten von Festplatte gelesen werden ist dies sehr viel langsamer als wenn diese im Arbeitsspeicher liegen. Die Zugriffszeit gibt dabei an, wie schnell Daten bereitgestellt werden können. Typische Zugriffszeiten heutiger Speichermedien:

- Festplatten: 9 ms
- SSD: 250 µs
- Arbeitsspeicher 60-70 ns
  (Quelle: https://de.wikipedia.org/wiki/Zugriffszeit)

-> Daten sollten komplett im Arbeitsspeicher gehalten werden. Dies ist aufgrund hoher Kosten meist nicht komplett möglich.

-> Intelligente Auslagerung der am häufigsten/dringendsten benötigten Daten in den Arbeitsspeicher

Datenbanktheorie und Implementierung - FH-Wedel SS 2019 - Michael Predeschly 12

Was versteht man unter dem Mapping von Speicheradressen?

Zuordnung von Speicheradressen auf Scheiben, Spuren und Sektoren.

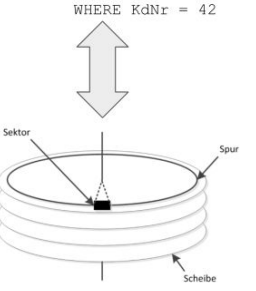
fhwedel
UNIVERSITY OF APPLIED SCIENCES

Speicherverwaltung

- Verbindung der Daten mit dem Festplatten-Speicher (Mapping von Speicheradressen auf Scheiben, Spuren und Sektoren)
- Abstraktion von konkreter Hardware auf der Speicherverwaltung realisiert
  -> Technologieunabhängigkeit
- Effizientes Auslesen der angeforderten Daten in den Arbeitsspeicher
- Schnelles Auffinden freier Datenblöcke

```
SELECT *
FROM Kunde

UPDATE Kunde
SET Name = Bilbo
WHERE KdNr = 42
```



Datenbanktheorie und Implementierung - FH-Wedel SS 2019 - Michael Predeschly 13

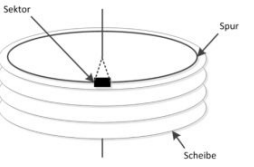
Beschreiben Sie generell den Begriff der Speicherverwaltung.

Speicherverwaltung

- Verbindung der Daten mit dem Festplatten-Speicher (Mapping von Speicheradressen auf Scheiben, Spuren und Sektoren)
- Abstraktion von konkreter Hardware auf der Speicherverwaltung realisiert
  -> Technologieunabhängigkeit
- Effizientes Auslesen der angeforderten Daten in den Arbeitsspeicher
- Schnelles Auffinden freier Datenblöcke

```
SELECT *
FROM Kunde
```

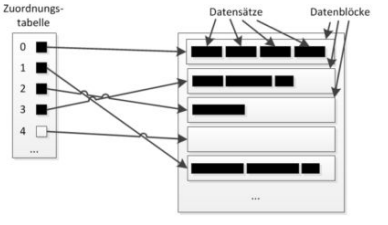
```
UPDATE Kunde
SET Name = Bilbo
WHERE KdNr = 42
```



Beschreiben Sie grob den Begriff der Speicherzuordnung. Mit welchem Hilfsmittel ist es hierbei möglich die Datenblöcke zu lokalisieren? Müssen die Datenblöcke stets die gleich Länge aufweisen?

Speicherzuordnung

- **Zuordnungstabelle** als Inhaltsverzeichnis für Datenblöcke
- Datenblöcke enthalten mehrere **Datensätze** unterschiedlicher Länge
- Eine Datei kann mehrere Einträge in der Zuordnungstabelle beanspruchen
- Freie Datenblöcke können leicht über Zuordnungstabelle gefunden werden



Was sind die Ziele der Pufferverwaltung ? ¡missing IDENTIFIER¿

Pufferverwaltung

Ziel Pufferverwaltung:

- Reduktion von physischer Ein-/Ausgabe
- Häufig genutzte Daten werden in **Seiten** im Hauptspeicher vorgehalten
- **Verdrängung** alter Daten im Hauptspeicher durch intelligente Verfahren

-> Weniger Zugriffe auf die Festplatte und damit schnellere Verarbeitung der Anfragen möglich

Pufferverwaltung in Datenbanken hat viele Ähnlichkeiten zum Vorgehen von Cache-mechanismen bei Betriebssystemen.

Datenbanktheorie und Implementierung - FH-Wedel SS 2019 - Michael Predeschly 15

Wo werden die Pufferdaten zwischengelagert?

In sog. Seiten im Hauptspeicher (RAM).

Pufferverwaltung

Ziel Pufferverwaltung:

- Reduktion von physischer Ein-/Ausgabe
- Häufig genutzte Daten werden in **Seiten** im Hauptspeicher vorgehalten
- **Verdrängung** alter Daten im Hauptspeicher durch intelligente Verfahren

-> Weniger Zugriffe auf die Festplatte und damit schnellere Verarbeitung der Anfragen möglich

Pufferverwaltung in Datenbanken hat viele Ähnlichkeiten zum Vorgehen von Cache-mechanismen bei Betriebssystemen.

Datenbanktheorie und Implementierung - FH-Wedel SS 2019 - Michael Predeschly 15

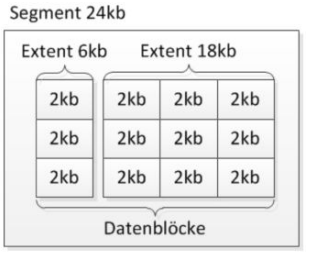
Beschreiben Sie anhand der Datenbanken vom Hersteller Oracle was man unter einem Segment sowie einem Extent versteht. Wie kann es sein, dass die Größe des Extents von den kleinsten logischen Einheiten einer Festplatte / SSD abweichen kann?

Beispiel Oracle

- **Segmente**: logische Einheit
- **Extent**: kleinste logische Speichereinheit in Oracle

Segmente besteht aus einem oder mehreren Extents, die die Informationen zu einem Objekt gespeichert haben (z.B. Relation).

Reicht Speicherplatz für Segment nicht aus, werden weitere Speicherbereiche im Rahmen des Tablespaces allokiert.



Datenbanktheorie und Implementierung - FH-Wedel SS 2019 - Michael Predeschly 17

Da eine Datenbank i.d.R. ein eigenes Filesystem anlegt ist es komplett losgelöst von sämtlichen Restriktionen von Festspeicher / Betriebssystem.

Was ermöglicht das Konzept eines Segments in der Datenbanktheorie?

Segmente

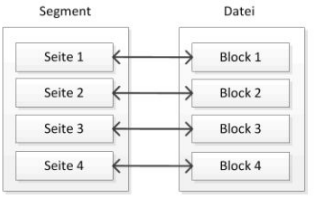
- Abstraktionsebene, die Blöcke und Dateien verbirgt.
- Konzept des Segments ermöglicht:
  - verschiedene Einbringungsstrategien (direktes Einbringen/update-in-place, indirektes Einbringen)
  - Segmente als kleinste Einheit
    - des Sperrens für Transaktionen
    - der Wiederherstellung bei Gerätefehlern
    - der Zugriffskontrolle
- Behälter für Relationen, Zugriffspfade, ...
- Man unterscheidet:
  - Öffentliche Segmente (gemeinsam genutzte Datensätze, erlaubter konkurrierender Zugriff)
  - Private Segmente (z.B. Speicherung von Log-Informationen)
  - Temporäre Segmente (z.B. Sortieren, Speichern von Zwischenergebnissen)

Datenbanktheorie und Implementierung - FH-Wedel SS 2019 - Michael Predeschly 18

Beschreiben Sie das Prinzip der direkten Seitenadressierung an einem Bild.

Seitenzuordnung/ direkte Seitenadressierung

- Managed **Segmente** im Arbeitsspeicher, die aus Seiten (pages) bestehen.
- Größe der Seiten wird so gewählt, dass sie jeweils einem Block der Festplatte entspricht (egal ob voll oder leer).
  -> je 1 Zugriff je Seite/Block
- Bei **direkter Seitenadressierung** werden Segmente analog zu dem Bild rechts zugeordnet.



direkte Seitenadressierung

Datenbanktheorie und Implementierung - FH-Wedel SS 2019 - Michael Predeschly 19

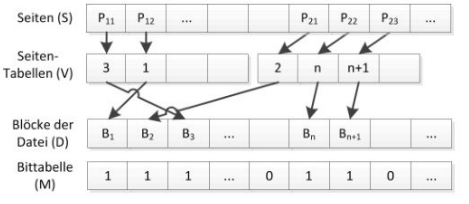
Beschreiben Sie das Prinzip der indirekten Speicheradressierung. Gehen Sie insbesondere auf die beiden verwendeten Hilfsstrukturen ein.

Indirekte Seitenadressierung

- erhöhte Zugriffskosten mit besserer Speicherauslastung
- maximale Flexibilität bei Blockzuordnung
- arbeitet mit 2 Hilfsstrukturen
- dynamisch änderbar

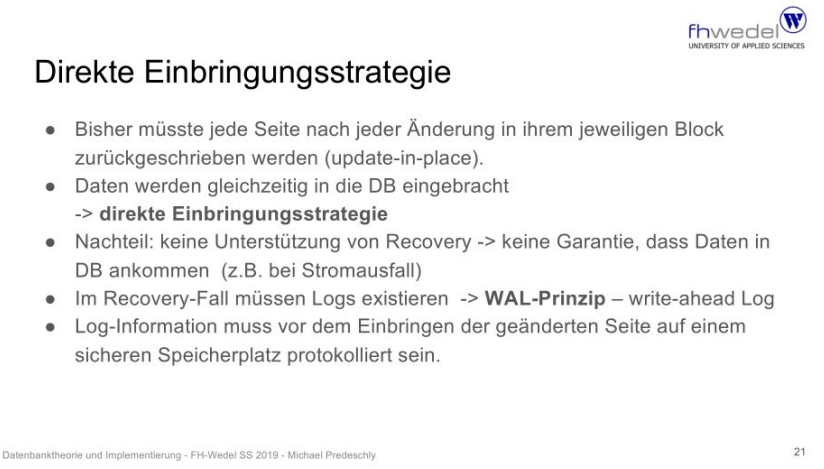
Hilfsstrukturen:

- eine **Seitentabelle** (V) je Segment
- **Bittabelle** (**Map** - M) zur Freispeicherverwaltung



Datenbanktheorie und Implementierung - FH-Wedel SS 2019 - Michael Predeschly 20

Beschreiben Sie das Prinzip der direkten Einbringunsstrategie. Welche Risiken können hierbei auftreten und wie kann man diesen entgegenwirken?



WAL-Prinzip: Man schreibt einen Log über die Dinge die als nächstes getan werden sollen (wie eine Todo-Liste). Falls hierbei ein Stromausfall dazwischen kommt weiß man genau welche Daten noch nicht kopiert wurden / noch gelöscht werden müssen um die Daten konsistent zu halten.

Beschreiben Sie den Begriff der verzögerten Einbringungsstrategie am Beispiel des Schattenspeicherkonzepts. ¡missing IDENTIFIER¿

Verzögerte Einbringungsstrategie

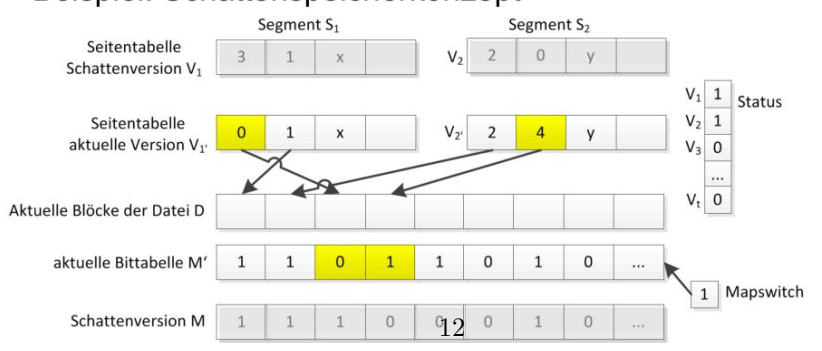
- **Verzögerte Einbringungsstrategien** unterscheiden zwischen Zurückschreiben von Seiten und ihrem Einbringen.
- Hierbei kann Logging reduziert werden oder sogar ganz entfallen.
- Ein Beispiel für ein solches Verfahren ist das **Schattenspeicherkonzept** aus IBMs System R (1977).

Idee:
Alle belegten Segmente, Seitentabellen und Bittabellen werden in **Sicherungspunkten** gespeichert. Sicherungspunkte sind dabei „eingefrorene" Zustände auf einem nichtflüchtigen externen Speicher. Seitenänderungen führen dann zu neuen Seiten (in freien Bereichen), während alte Inhalte unverändert in **Schattenseiten** abgelegt bleiben.

Beispiel: Schattenspeicherkonzept

- Vorteile:
  - Unterstützung von Recovery
  - WAL wird vermieden -> flexibleres Schreiben der Logfiles
- Nachteile:
  - Erhöhte Zugriffszeiten durch verlorene Clustereigenschaft von Blöcken (Daten liegen verteilter)
  - Seitentabellen und Bittabellen sehr aufwendig, d.h. passen nicht mehr in den Hauptspeicher (nur bei großen Datenbanken ein Problem)
- Zu Beginn einer Änderung werden für alle Segmente S_x, die geändert werden sollen, Schattenversionen der Seitentabelle V_x angelegt.
- Zusätzlich wird eine Schattenkopie der Bittabelle M angelegt (Mapswitch zeigt aktuelle Version an).
- Über den Status ist ersichtlich, welches Segment gerade geändert wird.

Beispiel: Schattenspeicherkonzept

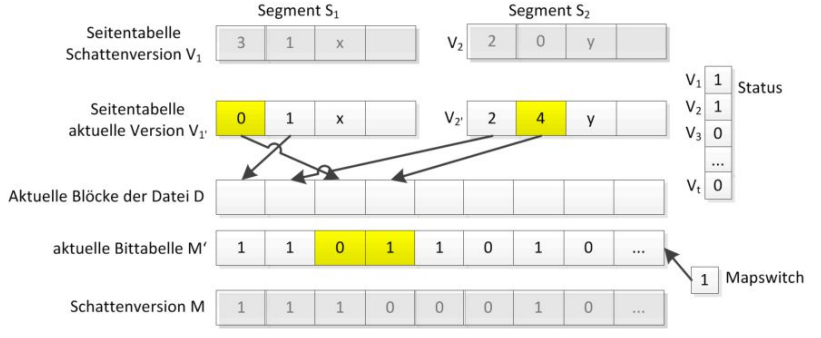


Erläutern Sie das Konzept des Schattenspeicherkonzepts. ¡missing IDENTIFIER¿

Beispiel: Schattenspeicherkonzept

- Vorteile:
  - Unterstützung von Recovery
  - WAL wird vermieden -> flexibleres Schreiben der Logfiles
- Nachteile:
  - Erhöhte Zugriffszeiten durch verlorene Clustereigenschaft von Blöcken (Daten liegen verteilter)
  - Seitentabellen und Bittabellen sehr aufwendig, d.h. passen nicht mehr in den Hauptspeicher (nur bei großen Datenbanken ein Problem)
- Zu Beginn einer Änderung werden für alle Segmente S_x, die geändert werden sollen, Schattenversionen der Seitentabelle V_x angelegt.
- Zusätzlich wird eine Schattenkopie der Bittabelle M angelegt (Mapswitch zeigt aktuelle Version an).
- Über den Status ist ersichtlich, welches Segment gerade geändert wird.

Beispiel: Schattenspeicherkonzept



Erläutern Sie wie die Verwaltung des Puffers funktioniert? Gehen Sie insbesondere auf die Funktion des Pufferrahmens sowie des Pufferkontrollblocks ein.

Verwaltung des Puffers

- Alle Lese- und Schreiboperationen aller (parallelen Transaktionen) der DB werden über den Puffer abgewickelt.
  -> GB-Bereich
  -> Puffer kann nur Bruchteil einer DB aufnehmen
- Puffer besteht aus N **Pufferrahmen** für N Datenbankseiten
- **Pufferkontrollblock** (PKB) enthält Verwaltungsdaten wie z.B. Änderungsvermerke
- Es werden Kontroll- und Zuteilungsaufgaben notwendig, um den Puffer zu verwalten.

Datenbanktheorie und Implementierung - FH-Wedel SS 2019 - Michael Predeschly 25

Welche Kontroll- und Zuteilungsaufgaben besitzt die Speicherverwaltung einer Datenbank?

Kontroll- und Zuteilungsaufgaben

- Speicherverwaltung zur Zuteilung von Pufferrahmen, sowie zur Suche und Ersetzung von Seiten
- Lastkontrolle zur Anpassung der Anzahl der aktiven Transaktionen an das momentane Optimum, das vom jeweiligen Betriebszustand abhängt
- Ziel ist die Reduktion der Input/Output Zugriffe.

- Vorgehen:
  - Seitennummer ermitteln
  - Zugehörige Seite anfordern (**logische Seitenreferenz**)
  - Festlegen, ob Seite: gelesen oder geändert werden soll
- **FIX-Operation** legt fest, dass Seite im Pufferrahmen bleibt (**UNFIX = Freigabe**)

Datenbanktheorie und Implementierung - FH-Wedel SS 2019 - Michael Predeschly 26

Gehen Sie auf die verschieden Szenarien ein welche beim Arbeiten mit einem Puffer auftreten können.

Logische Seitenreferenz

a) **Seite im Puffer**

-> geringer Aufwand (ca. 100 CPU Instruktionen)

- Seite finden
- FIX-Operation ausführen
- Wartungsarbeiten im PKB ausführen
- Pufferadresse an rufende Komponente übergeben

b) **Seite nicht im Puffer**

-> Logische Seitenreferenz führt zu einer physischen Seitenreferenz

- erfolglose Suche im Puffer
- 2 Input/Output – Vorgänge (je I/O etwa 2500 CPU Instruktionen pro Vorgang + 9 ms für HD)
  - Auswahl einer Seite zur Verdrängung
  - Herausschreiben der Seite bei Änderungsvermerk
  - Neue Seite lesen

Datenbanktheorie und Implementierung - FH-Wedel SS 2019 - Michael Predeschly 27

Geben Sie eine grobe Übersicht über die verschiedenen Suchstrategien der Pufferverwaltung.

Auffinden einer Seite

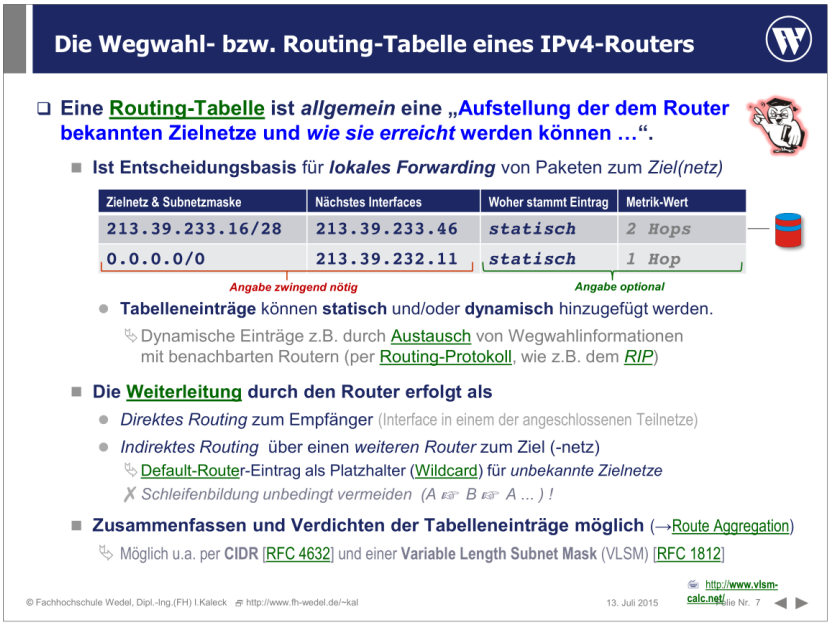
- **Forderung:** Hocheffiziente Suchstrategie, da der Vorgang extrem häufig vorkommt.

Datenbanktheorie und Implementierung - FH-Wedel SS 2019 - Michael Predeschly 29

Erläutern Sie nun noch kurz, wozu bei einem Router (oder einer Arbeitsstation) eine Wegwahl- bzw. Routingtabelle eigentlich dient. Was ist das bzw. was wird hier prinzipielle gespeichert?

In paketvermittelten Netzen werden Ende-zu-Ende-Verbindungen zwischen zwei Endgeräten dadurch hergestellt, dass Datenpakete von einer sendenden Station zu einer Empfangsstation übertragen werden. Die Stationen müssen über ihre Netzwerkadressen eindeutig identifizierbar sein. Zwischen den beiden Stationen liegen Router, die die Datenpakete gemäß ihrer Routingtabellen von der Sendestation über den ersten Router zum zweiten, zum dritten usw. weiterleiten, bis der letzte Router auf dem Weg durch das Datenpaketnetz die Datenpakete an die Empfängerstation ausliefert.

Routingtabellen werden von den Routing-Protokollen für die Pfadermittlung erstellt und können je nach Routing-Protokoll unterschiedlich sein. Beim statischen Routing ist der Routingpfad fest vorgegeben. Beim dynamischen Routing werden die Routingtabellen durch aktuelle Routinginformationen aktualisiert. Die Aktualisierung übernehmen Routing-Protokolle, die die Informationen eintragen. Das können Information über die Verknüpfungen der Netzwerke, bzw. der Endgeräte und der Hops mit den dazu gehörenden IP-Adressen sein, oder die Routing-Metriken mit der die Route festgelegt wurde.



Was versteht man unter einem logischen Seitenreferenzstring?

Seitenreferenzstrings

- **Jede Datenanforderung ist eine *logische Seitenreferenz***
- **Aufgabe der DB-Pufferverwaltung:**
  Minimierung der *physischen Seitenreferenzen*
- **Referenzstring** $R = \langle r_1, r_2, \ldots r_i, \ldots r_n \rangle$
  mit $r_i = (T_i, D_i, S_i)$

  T_i zugreifende Transaktion
  D_i referenzierte DB-Partition
  S_i referenzierte DB-Seite

fhwedel
UNIVERSITY OF APPLIED SCIENCES

Logischer Seitenreferenzstring

- zeitliche Folge der logischen Seitenanforderungen aller parallelen Transaktionen innerhalb eines Zeitabschnittes
- beschreibt aktuelles Zugriffsverhalten des DBS
- **Problem:** Erzeugung eines optimalen physischen Seitenreferenzstrings (= minimale Zugriffe) Wesentlich beeinflusst durch:
  - Größe des Systempuffers und Wahl der Seitenersetzungsstrategie
  - Ein logischer Seitenreferenzstring kann unterschiedliche physische Referenzstrings zur Folge haben.
- Die Optimierung der Seitenersetzungsstrategie im Systempuffer ist von entscheidender Bedeutung.

Geben Sie eine kurze Aufschlüsselung über die möglichen Suchstrategien bei der Pufferverwaltung. ¡missing IDENTIFIER¿

Erläutern Sie die Vorgehensweise bei der direkten Suche im Pufferrahmen, wie viele Rahmen werden durchschnittlich durchsucht und was ist der wesentliche Nachteil dieser Technik?

fhwedel
UNIVERSITY OF APPLIED SCIENCES

Direkte Suche im Pufferrahmen

- Im **Seitenkopf** (page header) wird geprüft, ob die im Rahmen vorhandene Seitennummer mit der angeforderten übereinstimmt.
- Im Erfolgsfall (bei N Rahmen) sind durchschnittlich N/2, bei Misserfolg N Rahmen aufzusuchen.
- Hoher Aufwand besonders bei virtueller Systemumgebung (verborgener Zusatzaufwand durch Paging).