Prof. Dr. Stefan Brass
Institut für Informatik
MLU Halle-Wittenberg

03. Februar 2022

Datenbanken II B: DBMS-Implementierung
— 13. Übungsblatt: Anfrageauswertung —

Hausaufgaben

Geben Sie die Aufgaben dieses Abschnitts bis Mittwoch, 09.02.2022, 18^{00}, über die Übungsplattform in StudIP ab. Schreiben Sie die Lösungen in eine `.txt`-Datei bzw. `.sql`-Datei.

Sie haben in den Übungen Zugriff auf eine Oracle-Datenbank (Version 18c) über die Adminer-Webschnittstelle bekommen:

https://dbs.informatik.uni-halle.de/db2b/adminer?
oracle=oracle-18.4-xe-db2b%2FXEPDB1&username=&db=USERS

Es sei die bekannte Punkte-Datenbank gegeben:

- STUDENTEN(SID, VORNAME, NACHNAME, EMAIL^o)
- AUFGABEN(ATYP, ANR, THEMA, MAXPT)
- BEWERTUNGEN(SID→STUDENTEN, (ATYP, ANR)→AUFGABEN, PUNKTE)

Damit die Aufgabe etwas interessanter ist, wollen wir annehmen, dass es sich um einen „Massive Open Online Course (MOOC)“ handelt mit sehr vielen Studierenden.

Die folgende Anfrage ist gegeben:

```
SELECT S.SID, S.VORNAME, S.NACHNAME
FROM   STUDENTEN S, BEWERTUNGEN B
WHERE  B.ATYP = 'H' AND B.ANR = 12
AND    B.PUNKTE > 10
AND    S.SID = B.SID
```

Beschreiben Sie, wie man die Anfrage auswerten kann, wenn folgende Indexe gegeben sind. Es reicht, wenn Sie die Vorgehensweise mit Worten erläutern. Sie brauchen keinen Oracle QEP zu zeichnen.

a) **(3 Punkte)** Kein Index. (Dann kann man natürlich auch keinen Schlüssel auf den Tabellen deklarieren, und damit auch keinen Fremdschlüssel. Alle mir bekannten Datenbanksysteme würden zur Überwachung eines Schlüssels automatisch einen Index anlegen.)

b) **(3 Punkte)** Indexe, die den Schlüsseln entsprechen: `I1` über `STUDENTEN(SID)` und `I2` über `BEWERTUNGEN(SID, ATYP, ANR)`. Beide Indexe sind `UNIQUE`.

c) **(3 Punkte)** Würde es einen Unterschied machen, wenn man den Index `I2` ersetzt durch einen Index `I3` über `BEWERTUNGEN(ATYP, ANR, SID)`?

d) **(3 Punkte)** Wenn Sie sich einen oder mehrere Indexe aussuchen könnten, um Anfragen dieses Typs sehr effizient auszuwerten, welchen Index bzw. welche Indexe würden Sie wählen?

e) **(3 Punkte)** Wie wertet Oracle die Anfrage aus, wenn die Tabellen sehr klein sind, wie bei der Datenbank aus der Einführungs-Vorlesung?

[http://www.informatik.uni-halle.de/~brass/dbi21/homework/bsp_db.sql]

Führen Sie den „`ANALYZE TABLE`“ Befehl für die benutzen Tabellen aus. Sie können es auch einmal vor und einmal nach dem „`ANAYZE TABLE`“ probieren.

f) **(3 Punkte)** Legen Sie Ihren bestmöglichen Index (bzw. Ihre bestmöglichen Indexe) an und probieren Sie aus, ob Oracle die Anfrage wie beabsichtigt ausführt.

g) **(3 Punkte)** Installieren Sie nun eine größere Version der Tabellen (die existierenden Tabellen werden dabei gelöscht). Eventuell müssen Sie vorher andere große Tabellen löschen, die Sie noch haben. Falls das Laden der Bewertungen abbricht, weil Ihre Quota nicht ausreicht, nehmen Sie die Zeilen, die noch geladen wurden.

[http://www.informatik.uni-halle.de/~brass/dbi21/homework/bsp_db2.sql]
[http://www.informatik.uni-halle.de/~brass/dbi21/homework/stud.sql]
[http://www.informatik.uni-halle.de/~brass/dbi21/homework/bew.sql]

Die Studentennamen stammen aus Beispieldaten der folgenden Webseite:

[https://www.briandunning.com/sample-data/]

Verwendet Oracle jetzt einen anderen Auswertungsplan?

Wiederholungsaufgaben

Die „Wiederholungsaufgaben“ brauchen Sie nicht abzugeben. Beschäftigen Sie sich aber bitte auch mit diesen Aufgaben. Notieren Sie sich Fragen, die Sie gerne in der Übung geklärt haben wollen.

h) Wie würden Sie in einer mündlichen Prüfung auf folgende Fragen zu Indexen antworten?

- Vergleichen Sie einen „Full Table Scan“ mit einem Zugriff auf die Tabelle über einen Index (inklusive „Dereferenzierung“ der ROWIDs aus dem Index). Was muss man wissen, um zu entscheiden, welcher Auswertungsplan besser ist? Geben Sie ein Beispiel, bei dem die Index-Nutzung deutlich schlechter ist als ein vollständiges Lesen der Tabelle.
- In welchen Fällen ist ein Index nützlich zur Auswertung einer `ORDER BY`-Klausel? Beschreiben Sie auch mögliche Nachteile.
- Was sind die Nachteile von Indexen? In welchen Fällen sollte man keinen Index über einer Spalte A einer Relation R definieren?

i) Wie würden Sie in einer mündlichen Prüfung auf folgende Fragen zu weiteren Datenstrukturen zur Speicherung von Relationen antworten?

- Die Standard-Datenstruktur für eine Tabelle ist die Heap-Datei. Nennen Sie mindestens eine weitere Datenstruktur, die in Oracle zur Speicherung von Tabellenzeilen genutzt werden kann.
- Was ist eine „Index-Organized Table“? Nennen Sie Vorteile und Nachteile im Vergleich mit einer Heap-Datei und einem normalen Index. Was ist das Problem, wenn man zusätzliche Indexe auf einer „Index-Organized Table“ anlegen will?
- Was ist ein „Cluster“ in Oracle? Zu welchem Zweck wird er eingesetzt bzw. was ist der wesentliche Vorteil? Welchen Preis muss man dafür zahlen?
- Was ist ein „Hash Cluster“ in Oracle? Wie kann man eine Zeile mit einem gegebenen Schlüsselwert in einem Blockzugriff finden?
- Was ist ein „Bitmap Index“? In welchen Situationen werden solche Indexe eingesetzt?
- Was ist der Vorteil einer partitionierten Tabelle?

j) Wie würden Sie in einer mündlichen Prüfung auf folgende Fragen über die Anfrageauswertung antworten?

- Ein Anfrageauswertungsplan oder Zugriffsplan ist recht ähnlich zu einem Ausdruck in der relationalen Algebra. Was sind die wesentlichen Unterschiede?
- Nennen Sie einige Operationen, die in Oracle QEPs (query execution/evaluation plans) auftreten können.

- Wie können Sie den „Query Evaluation Plan“, den der Oracle Optimierer für eine Anfrage gewählt hat, anschauen?
- Wie können Sie prüfen, ob Oracle einen Index, den man zur Beschleunigung einer Anfrage angelegt hat, auch benutzt? Was können Sie tun, wenn sich der Optimierer für einen anderen Index entschlossen hat, der aus Ihrer Sicht schlechter ist?
- Warum sollte man die Materialisierung von Zwischenergebnissen bei der Auswertung eines QEP möglichst vermeiden (im Normalfall)? Warum würde man zur Auswertung eines Ausdrucks in relationaler Algebra nicht an jedem Knoten eine temporäre Zwischenrelation erzeugen? Was ist die Alternative?
- Bei welcher Operation ist die Materialisierung von Zwischenergebnissen unvermeidbar? Welche Operation benötigt also größere Mengen an temporärem Speicherplatz?
- Gibt es auch Situationen, in denen die Materialisierung von Zwischenergebnissen nützlich sein kann? Diskutieren Sie die Vor- und Nachteile der „Pipelined Evaluation“ (oder „Lazy Evaluation“).
- Beschreiben Sie ein mögliches Interface für Knoten im QEP, wenn man „Pipelined Evaluation“ nutzen will.
- Erklären Sie die Parameter `sort_area_size` und `sort_area_retained_size` des Oracle Servers. Warum macht es Sinn, die `sort_area_retained_size` kleiner als die `sort_area_size` zu wählen, wenn Hauptspeicher knapp ist?
- Wo speichert Oracle temporäre Daten für die Sortierung wenn der Hauptspeicher nicht ausreicht?
- Beschreiben Sie, wie der „Mergesort“ Algorithmus funktioniert. Wie werden die „Läufe“ gemischt? Wie kann man den Algorithmus mit vier Dateien implementieren?
- Was ist die Komplexität des „Mergesort“ Verfahrens? Was ist die bestmögliche Komplexität von Sortierverfahren?
- Wie kann man den „Mergesort“ Algorithmus verbessern, wenn die Eingabe schon partiell sortiert ist?
- Wie kann man vorhandenen Hauptspeicher beim „Mergesort“ Algorithmus verwenden? (Es sei hier angenommen, dass der Hauptspeicher nicht ausreicht, um die ganzen Daten im Hauptspeicher zu sortieren. Sonst würde man vermutlich einen anderen Sortieralgorithmus verwenden.)
- Nennen Sie vier verschiedene Join-Verfahren.
- Erklären Sie den „Nested Loop Join“. Was ist die Komplexität dieses Verfahrens, wenn beide Tabellen n Tupel enthalten?
- Wie kann man den „Nested Loop Join“ verbessern, um vorhandenen Hauptspeicher effektiv zu nutzen?

- Erklären Sie den „Merge Join“. Was ist die Komplexität dieses Verfahrens, wenn beide Eingaberelationen n Tupel enthalten, und das Join-Attribut auf einer Seite Schlüssel ist?
- Vergleichen Sie den „Nested Loop Join“ und den „Merge Join“. Was kann der „Nested Loop Join“, was mit dem „Merge Join“ nicht möglich ist?
- Erläutern Sie den „Index Join“. Was ist seine Komplexität? Wann ist dieses Verfahren dem „Nested Loop Join“ und dem „Merge Join“ klar überlegen? Was ist umgekehrt ein Vorteil des „Merge Join“ Verfahrens?
- Erklären Sie den „Hash Join“. Was ist die grundlegende Idee?

k) Wie würden Sie in einer mündlichen Prüfung auf folgende Fragen über die Anfrageoptimierung antworten? (Diese Fragen beziehen sich auf Stoff, der erst in der nächsten Vorlesung behandelt wird.)

- Nennen Sie einige Äquivalenzen der relationalen Algebra, die für die algebraische Anfrageoptimierung genutzt werden können.
- Warum ist es normalerweise besser, eine Selektion vor einem Join auszuführen statt danach? Geben Sie ein Beispiel, bei dem diese Heuristik zu einem klar schlechteren Ergebnis führt.
- Angenommen, eine SQL Anfrage deklariert Tupelvariablen über vier Relationen: `R1`, `R2`, `R3`, `R4`. Welche Struktur von Joins wird ein klassisches DBMS in die Auswertungsmöglichkeiten einbeziehen? Was ist ein „bushy join“?
- Was ist die Grundidee des regelbasierten Optimierers, den Oracle früher verwendet hat?
- Beschreiben Sie die grundsätzliche Funktionsweise eines kostenbasierten Optimierers.
- Was ist das klassische Kostenmaß für Auswertungspläne bei der kostenbasierten Anfrageoptimierung? Warum reicht das heute möglicherweise nicht mehr ausreichend?
- Wie kann man die Größe eines Selektionsergebnisses $\sigma_{A=c}(R)$ schätzen? Welche Daten benötigt man dafür?
- Welche Methoden sollte die Klasse für die Knoten im Auswertungsplan haben, um die Kostenberechnung zu unterstützen?
- Warum macht es einen Unterschied, ob man die Zeit bis zur ersten Zeile des Anfrageergebnisses oder die Zeit bis zur letzten Zeile des Anfrageergebnisses minimieren will?