

# Oracle Performance Tuning

## Übungsprüfung WS 2001/2002

Alex Schröder

2002-01-19

### **Allgemeines**

- Sie haben 45 min. Zeit um die Prüfungsfragen zu beantworten.
- Es gibt 5 Fragen à 20 Punkte.
- Die Fragen sind nicht alle gleich schwer.
- Beachten sie, dass die Fragen in Teilfragen unterteilt wurden.
- Pro Teilfrage finden sie die maximal erreichbaren Punkte aufgeführt.
- Zu gewissen Teilfragen sind mehrere, gleichwertige Antworten möglich.
- Halten sie sich kurz. Die Musterlösung ist nie länger als zehn Sätze pro Teilfrage.

# 1 Speicherstrukturen

1. Zeichnen sie schematisch die Struktur der Shared Global Area (SGA). Erklären sie in ein oder zwei Sätzen die Funktion der einzelnen Komponenten. (10 Punkte)
2. Erklären sie, unter welchen Umständen der Buffer Cache besonders ineffizient sein kann. In diesen Fällen bringt es meistens nichts, den Buffer Cache zu vergrößern. Welche Lösung bietet sich an? Erklären sie, warum diese Lösung funktioniert. (10 Punkte)

# 2 Normalisieren, Denormalisieren

1. Zeichnen sie ein ER Diagramm für die Tabellen, welche durch folgende SQL Statements beschrieben werden. (5 Punkte)

```
create table apparat (  
  apparatenummer number primary key,  
  werk_id number,  
  fertigungsauftrag number);
```

```
create table parameter (  
  parameter_id number primary key,  
  soll_wert number,  
  einheit varchar2(30),  
  bezeichnung varchar2(60));
```

```
create table apparateparameter (  
  apparatenummer number references apparat,  
  parameter_id references parameter,  
  wert number);
```

```
alter table apparateparameter primary key (  
  apparatenummer,  
  parameter_id);
```

2. Schreiben sie ein SQL Statement, welches für den Fertigungsauftrag 75 alle Apparatenummern auflistet, und zu jeder Apparatenummer den Wert der Parameter 5, 7, 165 und 351. Gehen sie davon aus, dass die aufgeführten Parameter für alle Apparate existieren; ein outerjoin ist also nicht nötig. (5 Punkte)

Hier ein Beispiel für den gewünschten Output:

```
SQL> select ...
```

APPARATENUMMER	P5	P7	P165	P351
6589421865	220	5	13	102
6589421866	220	5	14	100
6589421867	220	5	12	112

3. Nehmen sie an, es gibt zu jedem Apparat im Durchschnitt etwa 1500 Werte. Die APPARATEPARAMETER Tabelle wird also sehr gross sein. Um die Performance des vorherigen Statements zu verbessern, könnten sie Denormalisieren. Was schlagen sie vor, und warum erwarten sie einen Performance Gewinn? Geben sie an, welche Nachteile ihre Lösung mit sich bringt. (10 Punkte)

### 3 Indexe

1. Erklären sie kurz, was der Vorteil eines B-Baumes gegenüber einem Suchbaum ist. Erklären sie, welchen Vorteil dies für die Performance bietet. (10 Punkte)
2. Zeichnen sie einen B-Baum der Ordnung 5 und füllen sie die Zahlen 1 bis 8 in den B-Baum. Die Reihenfolge ist relevant. (10 Punkte)

### 4 Optimizer

1. Erklären sie den Unterschied zwischen dem rule-based optimizer (RBO) und dem cost-based optimizer (CBO). Führen sie Vor- und Nachteile des CBO gegenüber dem RBO auf.
2. Erklären sie den Unterschied zwischen den Optimizer Goals FIRST\_ROWS und ALL\_ROWS. (Dies sind die zwei unterschiedlichen Optimierungsvarianten des CBO.)

### 5 SQL Tuning

Gegeben sind die beiden Tabellen EMP und SALGRADE. In der Tabelle EMP sind die Mitarbeiter abgespeichert. In der Tabelle SALGRADE sind die einzelnen Lohnkategorien abgespeichert.

SQL> desc emp

Name	Null?	Type
EMPNO	NOT NULL	NUMBER(4)
ENAME		VARCHAR2(10)
JOB		VARCHAR2(9)
MGR		NUMBER(4)
HIREDATE		DATE
SAL		NUMBER(7,2)
COMM		NUMBER(7,2)
DEPTNO		NUMBER(2)

SQL> desc salgrade

Name	Null?	Type
GRADE		NUMBER
LOSAL		NUMBER

HISAL

NUMBER

```
SQL> select * from salgrade;
```

GRADE	LOSAL	HISAL
1	700	1200
2	1201	1400
3	1401	2000
4	2001	3000
5	3001	9999

Gegeben ist nun folgendes Statement samt Execution Plan:

```
SQL> select ename
       from emp, salgrade
       where grade = 3
       and sal between losal and hisal;
```

ENAME

-----  
ALLEN  
TURNER

Execution Plan

```
-----  
0      SELECT STATEMENT Optimizer=CHOOSE  
1      0      NESTED LOOPS  
2      1      TABLE ACCESS (FULL) OF 'SALGRADE'  
3      1      TABLE ACCESS (FULL) OF 'EMP'
```

1. Die Performance des Statements ist nicht optimal. Woran erkennt man das im Execution Plan? (5 Punkte)
2. Um die Performance zu steigern, kann man Indexe anlegen. Welche Indexe würden sie anlegen? Geben sie Tabellenname und die Spaltennamen in der richtigen Reihenfolge an. Die Reihenfolge ist relevant, falls sie einen zusammengesetzten Index anlegen wollen. Falls sich ein Index nur unter gewissen Umständen lohnen würde, begründen sie ihre Entscheidung. (5 Punkte)

Im obigen Execution Plan wird ein *Nested Loop* ausgeführt, um den Join zu machen. Die Alternative hierfür wäre ein *Sort Merge*. Ein Sort Merge Join kann allerdings nur bei einem equijoin ausgeführt werden.

1. Erklären sie, warum es sich hier nicht um einen equijoin handelt (3 Punkte).
2. Erklären sie, wie ein Sort Merge Join funktioniert (4 Punkte).
3. Erklären sie, basierend auf ihrer vorherigen Erklärung des Sort Merge Joins, warum dieser nur für equijoins verwendet werden kann (3 Punkte).