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

November 10, 2017

Databases IIB: DBMS-Implementation
— Exercise Sheet 5 —

Part a) to e) will be discussed in class, you only have to submit Part f). But please think about the questions in a) before the meeting. Send your solution for the homework to the instructor via EMail: `brass@informatik.uni-halle.de` (with "`dbi17`" in the subject line). The official deadline is November 15, 12:00.

Repetition Questions

a) What would you answer to the following questions in an oral exam?

- What is a tablespace in Oracle?
- What is the relationship between tables and data files in Oracle, i.e.:
  - Is a table stored in one file only, or in many files?
  - Does a file only data of one table, or of many tables?

  How do tablespaces fit in the picture?
- What is the purpose of the `SYSTEM` tablespace?
- What is the purpose of a temporary tablespace?
- Why is it good to define a quota for every user on every tablespace accessible to that user?
- Which alternatives are there for identifying users?
- Oracle stores user passwords encrypted in the database (in the system catalog). What problem occurs when an administrator needs to start the database? What solutions are there?
- What does it mean that a data file is "auto extensible"? Why should one define a relatively large `INCREMENT_BY` size (i.e. certainly more than a single block)?
- What is the purpose of control files in Oracle?
- What is the purpose of redo log files in Oracle?
- Why is it a good idea to write modified blocks of data files that are buffered in main memory only from time to time, while a log of changes is written to the redo log files at least at every commit? I.e. why would it take much longer to write the data blocks modified by a transaction than to write the log entries?

In-Class Exercises

b) Which Oracle processes are running on the server machine? Try

```
ps -ef | fgrep ora
```

c) Write SQL queries for the following questions:

- How many tables do you own?
- How many table-like objects are accessible to you? Print a separate number for each `TABLE_TYPE`.
- How many columns have the tables (and views) listed in `ALL_TAB_COLUMNS` in average? What is the minimal and maximal number of columns?
- What percentage of the data dictionary table names (except the `V$`-tables) ends in "`S`" (i.e. probably use a plural form)?

d) What is the database block size? If you are logged in as `SYSTEM`, you can get the value of the parameter "`db_block_size`" from the table "`V$PARAMETER`". Otherwise, you can simply divide `BYTES` by `BLOCKS` for any table that has both columns (e.g. `USER_SEGMENTS`).

e) How many blocks are attributed to you in `USER_TS_QUOTAS`? You can check the entries in `USER_SEGMENTS` to see the number of blocks for each database object. A segment is a logical file within a tablespace, and normally, their is one file for each table or index.

Homework Exercise

f) Develop an SQL*Plus script `files.sql` that lists all files of the database, i.e. at least

- the data files together with the tablespace to which they belong and their size in Megabytes,
- the temporary data files (also with tablespace and size),
- the control files,
- the log files with their group number and their size (ordered by group number).

You might need the tables `V$DATAFILE`, `V$TABLESPACE`, `V$TEMPFILE`, `V$CONTROLFILE`, `V$LOGFILE`, `V$LOG`. You must have administrator rights to access these tables (i.e. log in as `SYSTEM` or give your account the "`SELECT ANY TABLE`" system privilege). The SQL*Plus command `describe` shows you the column names, e.g. try

```
describe v$datafile
```

You can get more information on the columns of these tables from the "Oracle Database Reference":

[https://docs.oracle.com/cd/E11882_01/server.112/e40402/]

These tables are documented in Part III: "Dynamic Performance Views".

To compute Megabytes from Bytes, divide by `1024*1024` and use `ROUND(`x`)` to round the result to the nearest integer.

An SQL*Plus script can contain any number of SQL queries and SQL*Plus commands. You should write an SQL query for each file type.

Please make the output look nice. Use the SQL*Plus command `column` to make the output columns suffiently small, so that each result row fits into one line, e.g.

```
column FILENAME format A40
```

This assumes that you have renamed the column for the file name to "`FILENAME`". Try to use understandable column names. Note that the `COLUMN` command must appear in the SQL*Plus script before the query (when the query is executed, SQL*Plus looks for matching column format definitions).

Include headlines in the script with the SQL*Plus command `PROMPT`, which prints the rest of the line, e.g.

```
prompt Hello, world!
```


If you want to print the name of the database at the top of the report, you find that in the table `V$DATABASE`. You may include the name of the server parameter file in the output, this is the value of the parameter `spfile` listed in `V$PARAMETER`.

Unfortunately, the location of the ALERT file cannot be directly read from the data dictionary. The location on our system is

```
/app/oracle/diag/rdbms/orcl/orcl/trace/alert_orcl.log
```

The following webpage contains an SQL query that computes the directory:

[http://csl-oracle.blogspot.de/2013/01/finding-location-of-oracle-alert-log.html]

The query concatenates the following strings (I added the last part for the file name):

- The value of the parameter "`diagnostic_test`",
- the fixed string "`/diag/rdbms/`",
- the database name (parameter "`db_name`"),
- a "`/`",
- the instance name (parameter "`instance_name`", this is the same as the "`SID`"),
- the fixed string "`/trace/alert_`",
- the instance name/SID again,
- the fixed string "`.log`".

An Oracle instance consists of server processes and a shared memory region for these processes, the database consists of the files on disk. For high performance systems, it is possible to have several instances that open the same database.

There is also an XML version of the alert log in

```
/app/oracle/diag/rdbms/orcl/orcl/alert/log.xml
```

You may also want to print the directory "`background_dump_dest`" (another parameter from `V$PARAMETER`). There is also a "`core_dump_dest`" and a "`user_dump_dest`".

For Oracle 11g, the alert log file is actually contained in the background dump destination directory, but the above webpage states that this has changed in Oracle 12g.

If you want, you can also write comments in the SQL*Plus script with "`REM ...`".

If necessary, you can switch the automatic printing of table headlines off with

```
set heading off
```