

**IS-203**

**Semester project: Part 4**

**Due date: 15.04.2011**

## **Task #1 – Tablespaces, Tables, Indexes, and Users**

We have learned how one creates tablespaces and, along the way, have learned about the various ways of managing extents and segments/data blocks. We have also learned how one can create tables and specify how they use storage within the database. But if we allow Oracle to determine such things as how large the extents shall be and how empty data blocks shall be before new rows are inserted into them, some of these table parameters will have no effect. The task is to find out what effect various means of managing extents and segments/blocks have on the use of table parameters when tables are created.

Create two tablespaces described in Table 1 (XX = (possibly abbreviated) group name). Do not use file sizes larger than 5M. We need to be a bit careful about use of storage on the server. Create the files in the folder 'c:\app\oracle\oradata\dbadb\oppgave4'.

**Table 1 Tablespaces**

<i>Tablespace</i>	<i>Extent management</i>	<i>Segment management</i>
XX_TBS1	Automatic	Automatic
XX_TBS2	Uniform 1M	Manual

Look up in the data dictionary the values for the parameters listed in Table 2. In the *explanation* column, explain why the values are what they are.

**Table 2 Tablespace parameters**

xx_TBS1		
<i>parameter</i>	<i>value</i>	<i>explanation</i>
INITIAL_EXTENT		
NEXT_EXTENT		
MIN_EXTENTS		
MAX_EXTENTS		
xx_TBS2		
<i>parameter</i>	<i>value</i>	<i>explanation</i>

INITIAL_EXTENT		
NEXT_EXTENT		
MIN_EXTENTS		
MAX_EXTENTS		

Create new tables (STUDENT1 og STUDENT2), both with the columns:

Stud_id	CHAR(11)
Stud_fname	VARCHAR(50)
Stud_lname	VARCHAR(50)
Stud_State	CHAR(2)
Stud_Country	VARCHAR(50)

and parameters

PCTFREE	15
Initial extent size	200K
Next extent size	100K
Minimum extents	2
Maximum extents	200

Create STUDENT1 in XX\_TBS1 and STUDENT2, in XX\_TBS2.

Fill out the empty cells in the table with the actual values you find in DBA\_TABLES. If the actual values are different from the values specified above, try to give an explanation why.

**Table 3 Table parameters**

STUDENT1 in xx_TBS1			
	<i>Specified</i>	<i>actual</i>	<i>Explanation</i>
PCT_FREE	15		
INITIAL_EXTENT	200K		
NEXT_EXTENT	100K		
MIN_EXTENTS	2		
MAX_EXTENTS	200		
STUDENT2 in xx_TBS2			
	<i>Specified</i>	<i>actual</i>	<i>Explanation</i>
PCT_FREE	15		

INITIAL_EXTENT	200K		
NEXT_EXTENT	100K		
MIN_EXTENTS	2		
MAX_EXTENTS	200		

### Summary question :

*How much control of the table parameters does Oracle assume when one uses tablespaces with automatic segment space management and extent management? With manual segment space management and extent management? Is one technique better than the other? Explain.*

## Task #2 -- Skiklubben

You will now use some of what you have learned to take the skiklubben database a step further.

Use the following assumptions

- There are 19 fylker.
- There are now 20,000 active members, and we are going to load all 20,000 into the table at the outset.
- Each year, 2000 new members join skiklubben.
- The number of active members is quite stable; approximately 2000 new members become active and 2000 leave for one reason or another each year. For example, after a member reaches the age of 14 s/he is no longer active. But the data on inactive members is not deleted (the table grows).
- On average, each active member registers 15 activities per year (season).
- Among the activities, 40 % are cross-country skiing, 40% are alpine skiing, and 20% are ski jumping.
- Each member is a member of an average of two groups.
- Each group has an average of 10 members.
- There are an average of 150 organizations in each fylke. The number of organizations is very stable (almost no growth).

1. Create tablespaces that will be used for the application.
  - a. Calculate the maximum size of a row in each table. To calculate the maximum row length you must examine the data types for each column. You may assume that NUMBER/INTEGER data types occupies 4 bytes. DATE data type occupies 7 bytes, and VARCHAR(x) and CHAR(x) occupy x bytes.

Use the following formula:

$$\text{max row length} = 3 + (\text{sum of the column sizes}) + (1 * \text{number of columns} < 250 \text{ bytes}) + (3 * \text{number of columns} > 250 \text{ bytes})$$

- b. Calculate the size of each table, if the table is to store a year's worth of data (see the assumptions above). In order to calculate the size, you can use the formula found in the lecture PPT ("Database objekter/Database Administration Tables & Indexes", page 65 / slide 120). Decide what value of PCTFREE is appropriate in each table.
  - c. Determine an appropriate INITIAL and NEXT extent size for each table. Use the following heuristics:
    - i. It is good to use a single size for all extents in a tablespace.
    - ii. Large tables/segments should use larger extent sizes than small tables.
    - iii. The NEXT extent size should be large enough to store about a half-year's worth of data.
    - iv. The INITIAL extent size should be equal to the NEXT (if the table currently has no data or less data than the size of the NEXT extent) or a multiple of NEXT size (if the table initially shall be loaded with more data than the size of the NEXT extent).
    - v. There is little point in using extent sizes less than 64K.
    - vi. Use the following extent sizes: 64K, 128K, 256K, 512K, 1M, 2M, 4M, etc.
  - d. Put the tables into groups such that tables with similar characteristics (with NEXT extent sizes that are equal or fit together well) will be created in the same tablespace.
  - e. *Question: Does it make sense to adjust the NEXT extent size of individual tables so that they fit together better in a tablespace? Or is it better to create multiple tablespaces, one for each extent size? For example, if you have 4 tables with extent sizes of 64K, 128K, 128K, 256K, do you think it is better to (a) create one tablespace for all four, (b) use one NEXT extent size for all of the tables, (c) create three tablespaces for the 64K, 128K, and 256K extent sizes?, (d) something else? What do you think? Why?*
  - f. Decide which extent management and segment space management is the best for these tables.
  - g. Calculate how large each datafile should be to store a year's worth of data.
  - h. Create the necessary tablespaces. **NOTE! Do not use file sizes larger than 5M** on these (but you can use the AUTOEXTEND on the files) and I recommend that you use small extent sizes (64K max) since you will have a number of segments that need extents.
- 2. Move all of the tables created in assignments 1-3 to these tablespaces.
  - 3. Create (or move) the necessary indexes into the INDX tablespace.

4. Create four roles such that
  - a. The first has the least possible privileges, to only read skiklubben tables. It should not have the privileges to create anything in the database.
  - b. The second shall have the privilege to execute all of the procedures and functions in the skiklubben\_pkg package.
  - c. The third should have the right to insert a new member, insert a new group, register activities, and become a member in a group.
  - d. The fourth shall have the privilege to give out medals, give medals to all, and find the cup winners.
  - e. What difficulties did you encounter in (c) and (d) in comparison to (a) and (b)?
  - f. How can this problem be solved?
5. Create four users
  - a. Give the users name that start with XX\_1, XX\_2, etc. Where XX is the (possibly abbreviated) group name.
  - b. Create a PROFILE with reasonable characteristics and use it when creating users.
  - c. Give all users the default tablespace USERS with 5M quota.
  - d. Grant the first user role (4.a); the second users, role (4.b); the third user, role (4.c); the fourth user, roles (4.a) and (4.d).

## **Deliverables**

### **Task #1**

1. Tables with the cells filled in
2. Answer to the summary question
3. SQL DDL queries you used to create the tablespaces and tables
4. SQL used to look up the actual tablespace and table parameter values in the data dictionary.

### **Task #2**

5. Answers to 1.a-e, with explanation
6. SQL DDL queries you used to create tablespaces and move tables to these tablespaces, and move indexes to the INDX tablespace.
7. SQL DDL queries used to create users and roles, and give privileges to users.
8. Answers to 4.e,f

## Reflection

9. Each member of the group shall also submit a reflection. Answer two questions among those found at <http://escalate.ac.uk/resources/reflection/09.html>