

Titulación: GRADO INGENIERIA INFORMATICA

Año Académico: 2015/2016

Curso: 2º

Asignatura: Ficheros y Bases de Datos

Título: Memoria Práctica 3 – Diseño Físico en Oracle



UNIVERSIDAD CARLOS III DE MADRID

<b>Profesor:</b>	<b>María Dolores Cuadra Fernández</b>	<b>Grupo</b>	<b>50</b>
<b>Alumno/a:</b>	<b>Javier Bautista Rosell</b>	<b>NIA:</b>	<b>100315139</b>
<b>Alumno/a:</b>	<b>Lizaveta Mishkinitse</b>	<b>NIA:</b>	<b>100315144</b>

## 1 Introducción

En esta práctica se realiza un análisis del diseño físico de una base de datos, su rendimiento para los procesos especificados, una propuesta de mejora del diseño físico inicial, su implementación y finalmente la evaluación de los resultados conseguidos con el nuevo diseño.

Con la realización de este ejercicio se busca familiarizarnos con la arquitectura ORACLE y las distintas estructuras auxiliares que aumenten su rendimiento.

Junto con este documento se ha incluido el código SQL con la implementación de las mejoras propuestas en el tercer apartado del ejercicio.

## 2 Análisis

En Oracle, la organización por defecto es serial no consecutiva, con tamaño de espacio de cubo de 8K. El espacio libre distribuido para las modificaciones (PCT\_FREE) es del 10% y el PCT\_USED de 60% por defecto.

Titulación: GRADO INGENIERIA INFORMATICA

Año Académico: 2015/2016

Curso: 2º

Asignatura: Ficheros y Bases de Datos

Título: Memoria Práctica 3 – Diseño Físico en Oracle



UNIVERSIDAD CARLOS III DE MADRID

En primer lugar, queríamos ver el coste de ejecutar la totalidad de las consultas. Así, con el diseño físico inicial el número medio de *consistent gets* es de 28 412.20 bloques:

**TIME CONSUMPTION: 19125 milliseconds.**  
**CONSISTENT GETS: 28412,2 blocks**

En cuanto a cada una de las instrucciones por separado, éstas son las estadísticas que se han recogido tras ejecutar cada una de ellas:

	Consulta 1	Consulta 2	Consulta 3	Consulta 4	Consulta 5	Insertión y Modificación
Consistent Gets Inicial	76	7585	7582	7638	71	2997,8

Como se puede observar, los procesos más costosos son las consultas 2, 3 y 4. Estas consultas son consultas selectivas por clave alternativa. El coste tan elevado de estas consultas se debe a que no solo acceden a varias tablas por claves no identificativas, sino que además acceden a la tabla REP\_LINES, la cual contiene 499 999 entradas.

Analizando el plan de ejecución de cada una de las consultas, está claro que son poco eficientes para diseño físico proporcionado. En todos los casos (salvo la consulta 5, que accede a la tabla CONTRACTS por clave primaria), al ser el valor buscado una clave no identificativa, se accede a la tabla entera al realizar la búsqueda.

Titulación: GRADO INGENIERIA INFORMATICA

Año Académico: 2015/2016

Curso: 2º

Asignatura: Ficheros y Bases de Datos

Título: Memoria Práctica 3 – Diseño Físico en Oracle



UNIVERSIDAD CARLOS III DE MADRID

Consulta 1				Consulta 2			
Operation   Name   Rows				Operation   Name   Rows			
SELECT STATEMENT				SELECT STATEMENT			
FILTER				MERGE JOIN			
HASH GROUP BY				SORT JOIN			
HASH JOIN				VIEW			
TABLE ACCESS FULL				HASH GROUP BY			
TABLE ACCESS FULL				HASH JOIN			
CONTRACTS				TABLE ACCESS FULL			
CONTRACTS				COSTS			
2068				REP_LINES			
				TABLE ACCESS FULL			
				FILTER			
				SORT JOIN			
				TABLE ACCESS FULL			
				PROJECTS			
				63			
				63			
Consulta 3				Consulta 4			
Operation   Name   Rows				Operation   Name   Rows			
SELECT STATEMENT				SELECT STATEMENT			
MINUS				SORT ORDER BY			
SORT UNIQUE				HASH JOIN			
TABLE ACCESS FULL				TABLE ACCESS FULL			
CONTRACTS				PROJECTS			
SORT UNIQUE				VIEW			
TABLE ACCESS FULL				HASH GROUP BY			
REP_LINES				HASH JOIN			
				TABLE ACCESS FULL			
				CONTRACTS			
				HASH JOIN			
				TABLE ACCESS FULL			
				COSTS			
				HASH JOIN			
				TABLE ACCESS FULL			
				COSTS			
				TABLE ACCESS FULL			
				REP_LINES			
				441K			
Consulta 5							
Operation   Name   Rows							
SELECT STATEMENT							
FILTER							
HASH GROUP BY							
HASH JOIN							
TABLE ACCESS FULL							
CONTRACTS							
INDEX FAST FULL SCAN							
PK_CONTRACTS							
2068							

Titulación: GRADO INGENIERIA INFORMATICA

Año Académico: 2015/2016

Curso: 2º

Asignatura: Ficheros y Bases de Datos

Título: Memoria Práctica 3 – Diseño Físico en Oracle



UNIVERSIDAD CARLOS III DE MADRID

Respecto a los otros dos procesos analizados – las inserciones y las modificaciones, son bastante eficientes con el diseño físico dado. Ello es debido a que la organización sería la más eficiente en cuanto a las inserciones (únicamente se inserta al final del fichero). Aunque la organización podría no ser la óptima para las modificaciones, al ser en este caso no consecutiva y al disponer de un 10% de ELD, las modificaciones tienen poco coste.

### 3 Diseño Físico

Tal y como se ha comentado en el análisis, el mayor problema del diseño físico actual es que al realizar los procesos de consulta se accede a toda la tabla (*full scan*). Para evitarlo, una posible solución sería crear índices sobre las claves alternativas para los procesos selectivos.

Antes de proceder a la creación de los índices, se han tenido en cuenta las siguientes consideraciones:

- Es recomendable crear un índice si se quiere obtener menos del 15% de las entradas de una tabla grande. Por tanto, la tabla REP\_LINES sería buena candidata para tener un índice.
- La indización de las columnas sobre las que se realiza un join mejora el rendimiento del join de varias tablas. Todas las consultas que realizamos tienen un join, por lo que habrá que tenerlo en cuenta.
- Las tablas pequeñas no necesitan índices. Por consiguiente, no se hará ningún índice sobre la tabla COSTS, que solo contiene 250 entradas.
- Son buenos candidatos para índices regulares las columnas con un amplio rango de valores, y para índices bitmap las columnas con un rango de valores reducido.

Titulación: GRADO INGENIERIA INFORMATICA

Año Académico: 2015/2016

Curso: 2º

Asignatura: Ficheros y Bases de Datos

Título: Memoria Práctica 3 – Diseño Físico en Oracle



UNIVERSIDAD CARLOS III DE MADRID

- Las columnas con muchos valores 'null' no son buenas para la indización. En nuestro caso ocurre en la tabla CONTRACTS con el campo 'END\_DATE'.
- El tamaño de una entrada de índice no puede exceder aproximadamente la mitad del espacio disponible del bloque.
- Se debería limitar el número de índices por tabla si se realizan frecuentes modificaciones sobre la misma, debido a que las modificaciones sobre la tabla supone una modificación de entradas de índice, lo que puede llegar a ser muy costoso. Para nuestro diseño, debemos contrastar la reducción de costes en las consultas con el aumento de costes al tener que actualizar las entradas del índice con inserciones y modificaciones.

De los procesos analizados, los más costosos son las consulta 2, 3 y 4. Por consiguiente, se analizará cada una de ellas por separado.

**a) Consulta 2: Proyectos no finalizados cuyo gasto ya se ha excedido de su presupuesto**

Frecuencia: 0.06%

Consistent gets: 7585

Rows processed: 1

<b>Tablas asociadas</b>	PROJECTS, COSTS, REP_LINES
<b>Clave filtrado</b>	PROJECTS: <i>BUDGET, END_DATE</i>
<b>Clave Acceso</b>	<ul style="list-style-type: none"><li>- PROJECTS: <i>IDPROJECTS</i></li><li>- COSTS: <i>CAT_NUM, YEAR</i></li><li>- REP_LINES: <i>REP_DATE, IDPROJECTS, JOB</i></li></ul>
<b>Tipo de proceso</b>	Selectivo por clave alternativa

Titulación: GRADO INGENIERIA INFORMATICA

Año Académico: 2015/2016

Curso: 2º

Asignatura: Ficheros y Bases de Datos

Título: Memoria Práctica 3 – Diseño Físico en Oracle



UNIVERSIDAD CARLOS III DE MADRID

En esta consulta, igual que en las siguientes, la tabla con más entradas y por tanto la que más accesos necesita, es REP\_LINES, y se accede por una clave alternativa.

Para mejorar el rendimiento, se puede crear un índice para REP\_LINES sobre la clave por la que accede: REP\_DATE, IDPROJECTS, JOB. De esta manera, en vez de buscar en toda la tabla, buscará en el índice la entrada que corresponde y accederá solo a los bloques correspondientes.

Al ser el índice de tipo árbol B, el coste de selección se reduciría de buscar en toda la tabla a ser solo la recuperación de la entrada (número de niveles -1, ya que la raíz está en memoria) más tantos cubos de datos como punteros tenga la entrada.

Como ya se ha mencionado, no compensa crear un índice sobre COSTS por el reducido tamaño de la tabla.

Finalmente, para mejorar el rendimiento del índice, es recomendable que esté en un tablespace distinto al de la tabla. Por ello, este índice lo crearemos en el tablespace TABL\_16K

**b) Consulta 3: Empleados activos en el mes de Abril de 2014 que no cogieron vacaciones ese mes.**

Frecuencia: 0.06%

Consistent gets: 7582

Rows processed: 119

<b>Tablas asociadas</b>	CONTRACTS, REP_LINES
<b>Clave filtrado</b>	CONTRACTS: <i>END_DATE, START_DATE</i> REP_LINES: <i>TASK, REP_DATE</i>
<b>Clave Acceso</b>	-
<b>Tipo de proceso</b>	Selectivo por clave alternativa

Titulación: GRADO INGENIERIA INFORMATICA

Año Académico: 2015/2016

Curso: 2º

Asignatura: Ficheros y Bases de Datos

Título: Memoria Práctica 3 – Diseño Físico en Oracle



UNIVERSIDAD CARLOS III DE MADRID

En esta consulta, la tabla con más entradas es REP\_LINES, y se accede por una clave alternativa.

Siguiendo la misma lógica que en la consulta anterior, se creará un índice sobre los valores por cuales se accede a REP\_LINES: REP\_DATE, IDPROJECTS, JOB, TASK

**c) Consulta 4: Cuenta total de resultados por proyecto, ordenada por fecha de finalización**

Frecuencia: 0.06%

Consistent gets: 7638

Rows processed: 698

<b>Tablas asociadas</b>	PROJECTS, CONTRACTS, COSTS, REP_LINES
<b>Clave filtrado</b>	CONTRACTS: <i>START_DATE, END_DATE</i> REP_LINES: <i>REP_DATE</i>
<b>Clave Acceso</b>	PROJECTS: <i>IDPROJECTS</i> CONTRACTS: <i>SSN, CAT_NUM</i> COSTS: <i>CAT_NUM, YEAR</i> REP_LINES: <i>REP_DATE, SSN, IDPROJECTS, JOB</i>
<b>Tipo de proceso</b>	Selectivo por clave alternativa

Igual que en las dos consultas anteriores, se creará un índice para REP\_LINES sobre los campos por los que accede: REP\_DATE, SSN, IDPROJECTS, JOB.

Titulación: GRADO INGENIERIA INFORMATICA

Año Académico: 2015/2016

Curso: 2º

Asignatura: Ficheros y Bases de Datos

Título: Memoria Práctica 3 – Diseño Físico en Oracle



UNIVERSIDAD CARLOS III DE MADRID

Hay que tener en cuenta que crear índices sobre REP\_LINES supone un coste adicional para las inserciones y modificaciones, debido a que hay que actualizar el índice. Una vez creados los índices habría que calcular el aumento del coste para las inserciones.

También podemos modificar los parámetros PCT\_FREE y PCT\_USED de los bloques de datos, controlando así el uso del espacio libre dentro del bloque, y consiguiendo una mejora en la escritura y recuperación de datos.

En las tablas que no tienen modificaciones, podemos disminuir el porcentaje de PCT\_FREE (el porcentaje reservado a posibles modificaciones de filas que ya están en el bloque) al mínimo (las tablas EMPLOYEES, CONTRACTS, PROJECTS y REPORTS).

Por otro lado, la tabla REP\_LINES no solo tiene modificaciones, sino que además aumenta el tamaño del bloque al aumentar el tamaño del campo 'task'. Por ello, en este caso nos interesa aumentar el PCT\_FREE para conseguir una mejora del rendimiento.



Titulación: GRADO INGENIERIA INFORMATICA  
 Año Académico: 2015/2016  
 Curso: 2º  
 Asignatura: Ficheros y Bases de Datos  
 Título: Memoria Práctica 3 – Diseño Físico en Oracle



UNIVERSIDAD CARLOS III DE MADRID

## 4 Evaluación

Tras implementar los 3 índices y modificar el PCT\_FREE de las tablas, obtenemos los siguientes *consistent gets*:

<b>Consistent Gets Iniciales</b>	28412,2	RESULTS AT 06/05/16 TIME CONSUMPTION: 19125 milliseconds. CONSISTENT GETS: 28412,2 blocks
<b>Consistent Gets Diseño Nuevo</b>	6017,2	RESULTS AT 14/05/16 TIME CONSUMPTION: 44891 milliseconds. CONSISTENT GETS: 6017,2 blocks
<b>Variación (%)</b>	<b>-78,82%</b>	

En total, hemos conseguido mejorar el número de accesos por un 78,82%.

Si analizamos las mejoras de cada consulta por separado, estos son los resultados obtenidos:

### a) Consulta 2

IND_query2 ON rep_lines (REP_DATE, job, IDPROJECT) TABLESPACE TABL_16K		
<b>Consistent Gets Iniciales</b>	7585	Estadísticas ----- 0 recursive calls 0 db block gets 7585 consistent gets
<b>Consistent Gets Con Índice</b>	1360	Estadísticas ----- 0 recursive calls 0 db block gets 1360 consistent gets
<b>Variación (%)</b>	<b>-82,07%</b>	

Titulación: GRADO INGENIERIA INFORMATICA  
 Año Académico: 2015/2016  
 Curso: 2º  
 Asignatura: Ficheros y Bases de Datos  
 Título: Memoria Práctica 3 – Diseño Físico en Oracle



UNIVERSIDAD CARLOS III DE MADRID

### b) Consulta 3

IND_query3 ON rep_lines (REP_DATE, job, task, IDPROJECT) TABLESPACE TABL_16K		
<b>Consistent Gets Iniciales</b>	7582	Estadísticas ----- 18 recursive calls 0 db block gets 7582 consistent gets
<b>Consistent Gets Con Índice</b>	48	Estadísticas ----- 0 recursive calls 0 db block gets 48 consistent gets
<b>Variación (%)</b>	<b>-99,37%</b>	

### c) Consulta 4:

IND_query4 ON rep_lines (REP_DATE, job, ssn, IDPROJECT) TABLESPACE TABL_16K		
<b>Consistent Gets Iniciales</b>	7638	Estadísticas ----- 0 recursive calls 0 db block gets 7638 consistent gets
<b>Consistent Gets Con Índice</b>	1929	Estadísticas ----- 0 recursive calls 0 db block gets 1929 consistent gets
<b>Variación (%)</b>	<b>-74,74%</b>	

Como se puede observar, en los tres casos los accesos con índices han mejorado de manera significativa. No obstante, ahora son más caros los procesos de inserción y modificación:

Titulación: GRADO INGENIERIA INFORMATICA  
 Año Académico: 2015/2016  
 Curso: 2º  
 Asignatura: Ficheros y Bases de Datos  
 Título: Memoria Práctica 3 – Diseño Físico en Oracle



UNIVERSIDAD CARLOS III DE MADRID

IND\_query2 ON rep\_lines (REP\_DATE, job, IDPROJECT) TABLESPACE  
 TABL\_16K  
 IND\_query3 ON rep\_lines (REP\_DATE, job, task, IDPROJECT) TABLESPACE  
 TABL\_16K  
 IND\_query4 ON rep\_lines (REP\_DATE, job, ssn, IDPROJECT) TABLESPACE  
 TABL\_16K

<b>Consistent Gets Iniciales</b>	2997,8	RESULTS AT 14/05/16 TIME CONSUMPTION: 1953 milliseconds. CONSISTENT GETS: 2997,8 blocks
<b>Consistent Gets Con Índice</b>	3103,7	RESULTS AT 14/05/16 TIME CONSUMPTION: 3047 milliseconds CONSISTENT GETS: 3103,7 blocks
<b>Variación (%)</b>	+3,53%	

Para solucionarlo, hemos modificado el valor de PCT\_FREE. En las tablas donde no hay modificaciones (las tablas EMPLOYEES, CONTRACTS, PROJECTS y REPORTS), lo hemos disminuido de 10% a 2%, ya que no se va a utilizar.

Por otro lado, se realizan 800 modificaciones en la tabla REP\_LINES, aumentando el tamaño del campo 'task'. En este caso, hemos subido el PCT\_FREE hasta el 60%.

Como se puede observar, ahora disminuyen los accesos para las inserciones y modificaciones (con tres índices creados):

Titulación: GRADO INGENIERIA INFORMATICA

Año Académico: 2015/2016

Curso: 2º

Asignatura: Ficheros y Bases de Datos

Título: Memoria Práctica 3 – Diseño Físico en Oracle



UNIVERSIDAD CARLOS III DE MADRID

IND\_query2 ON rep\_lines (REP\_DATE, job, IDPROJECT) TABLESPACE  
TABL\_16K

IND\_query3 ON rep\_lines (REP\_DATE, job, task, IDPROJECT) TABLESPACE  
TABL\_16K

IND\_query4 ON rep\_lines (REP\_DATE, job, ssn, IDPROJECT) TABLESPACE  
TABL\_16K

Alter table projects pctfree 2;

Alter table contracts pctfree 2;

Alter table costs pctfree 2;

Alter table Employees pctfree 2;

Alter table reports pctfree 2;

Alter table Rep\_lines pctfree 60;

Consistent Gets Inicial	2997,8	RESULTS AT 14/05/16 TIME CONSUMPTION: 1953 milliseconds. CONSISTENT GETS: 2997,8 blocks
Consistent Gets Con Índice	2528,6	RESULTS AT 14/05/16 TIME CONSUMPTION: 1922 milliseconds. CONSISTENT GETS: 2528,6 blocks
<b>Variación (%)</b>		<b>-15,65%</b>

Titulación: GRADO INGENIERIA INFORMATICA

Año Académico: 2015/2016

Curso: 2º

Asignatura: Ficheros y Bases de Datos

Título: Memoria Práctica 3 – Diseño Físico en Oracle



UNIVERSIDAD CARLOS III DE MADRID

## 5 Conclusiones Finales

La realización de esta última práctica nos resultó algo más difícil que las dos anteriores, ya que no era tanto programar SQL como hacer un análisis del rendimiento de las consultas. Al ser más abstracto y al no poder realizar todas las pruebas con SQLDeveloper (la herramienta con la que hemos trabajado hasta ahora), hemos tenido que dedicarle más tiempo a la práctica.

Respecto a los resultados obtenidos, hemos conseguido mejorar considerablemente el diseño físico inicial. No obstante, no conseguimos implementar ningún cluster que mejorase el diseño, ni tampoco utilizar índices bitmap.

Gracias a esta práctica ahora tenemos un entendimiento mejor de la estructura de una base de datos y de la importancia de no solo realizar buen diseño lógico, sino que también el diseño físico, ya que puede reducir drásticamente los costes de recuperación de la información.

Finalmente, respecto a la asignatura en general, nos ha parecido que está bien estructurada y que hemos obtenido unos conocimientos básicos de ficheros y bases de datos. Lo que nos resultó más difícil es la falta de material para la realización de las prácticas (SQLPlus que está disponible es los ordenadores de la universidad no tiene todos los permisos de usuario necesarios, no es una herramienta muy cómoda y además si no se dispone de otra herramienta, como es el caso de esta práctica, se depende del horario de la universidad), y los últimos temas de ficheros, que son más densos que el temario anterior pero se le dedica menos tiempo.