

Database In-Memory By Example
Andy Rivenes, Product Manager, Oracle
Copyright (c) 2018, Oracle and/or its affiliates. All rights reserved.

DOAG
May 14-15, 2018

Safe Harbor Statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

Part 1:

```
[oracle@vbgeneric Part1]$ sqlplus /nolog
```

```
SQL*Plus: Release 12.2.0.1.0 Production on Tue Jan 16 18:59:42 2018
```

```
Copyright (c) 1982, 2016, Oracle. All rights reserved.
```

```
SQL> @01_show_sga.sql
```

```
Connected.
```

```
SQL> set numwidth 20
```

```
SQL>
```

```
SQL> -- This command shows the SGA allocations
```

```
SQL>
```

```
SQL> show sga
```

```
Total System Global Area      11039408128 bytes
Fixed Size                      8804520 bytes
Variable Size                   5972690776 bytes
Database Buffers                 4697620480 bytes
Redo Buffers                     24748032 bytes
In-Memory Area                   335544320 bytes
```

```
SQL>
```

```
SQL> set echo off
```

```
SQL> @02_show_inmemory.sql
```

```
Connected.
```

```
SQL>
```

```
SQL> -- This command shows the settings of all in-memory init.ora parameters
```

```
SQL>
```

```
SQL> show parameter inmemory
```

NAME	TYPE	VALUE
inmemory_adg_enabled	boolean	TRUE
inmemory_clause_default	string	
inmemory_expressions_usage	string	ENABLE
inmemory_force	string	DEFAULT
inmemory_max_populate_servers	integer	4
inmemory_query	string	ENABLE
inmemory_size	big integer	320M
inmemory_trickle_repopulate_servers_	integer	1
percent		
inmemory_virtual_columns	string	ENABLE
optimizer_inmemory_aware	boolean	TRUE

```
SQL>
```

```
SQL> set echo off
```

```

SQL> @03_im_usage.sql
Connected.
SQL>
SQL> -- This query displays how much space is allocated/used in the In-Memory Column Store
SQL>
SQL> SELECT *
  2 FROM v$inmemory_area;

```

POOL	ALLOC_BYTES	USED_BYTES	POPULATE_STATUS	CON_ID
1MB POOL	4592762880	0	DONE	3
64KB POOL	1157627904	0	DONE	3

```

SQL>
SQL> set echo off
SQL> @04_desc.sql
Connected.
SQL>
SQL> -- This command displays the columns in the view v$IM_SEGMENTS
SQL>
SQL> desc v$IM_SEGMENTS

```

Name	Null?	Type
OWNER		VARCHAR2(128)
SEGMENT_NAME		VARCHAR2(128)
PARTITION_NAME		VARCHAR2(128)
SEGMENT_TYPE		VARCHAR2(18)
TABLESPACE_NAME		VARCHAR2(128)
INMEMORY_SIZE		NUMBER
BYTES		NUMBER
BYTES_NOT_POPULATED		NUMBER
POPULATE_STATUS		VARCHAR2(13)
INMEMORY_PRIORITY		VARCHAR2(8)
INMEMORY_DISTRIBUTE		VARCHAR2(15)
INMEMORY_DUPLICATE		VARCHAR2(13)
INMEMORY_COMPRESSION		VARCHAR2(17)
INMEMORY_SERVICE		VARCHAR2(12)
INMEMORY_SERVICE_NAME		VARCHAR2(129)
CON_ID		NUMBER

```

SQL>
SQL> set echo off
SQL> @05_im_segments.sql
Connected.
SQL>
SQL> -- This query displays what objects are populated in the In-Memory Column Store
SQL>
SQL> Select v.owner, v.segment_name name, v.populate_status status From v$im_segments v;

```

no rows selected

```

SQL>
SQL> set echo off
SQL> @06_im_alter_table.sql
Connected.
SQL>
SQL> -- This query enables objects to be populated in the In-Memory Column Store
SQL>
SQL> alter table LINEORDER inmemory;

```

Table altered.

```

SQL> alter table PART inmemory;

```

Table altered.

```
SQL> alter table CUSTOMER inmemory;
```

Table altered.

```
SQL> alter table SUPPLIER inmemory;
```

Table altered.

```
SQL> alter table DATE_DIM inmemory;
```

Table altered.

```
SQL>
```

```
SQL> set echo off
```

```
SQL> @07_im_attributes.sql
```

```
Connected.
```

```
SQL>
```

```
SQL> -- This query allows you to review the current attributes of the tables in the SSB schema
```

```
SQL>
```

```
SQL> select table_name, cache, buffer_pool, compression, compress_for, inmemory,  
2         inmemory_priority, inmemory_distribute, inmemory_compression  
3 from user_tables;
```

TABLE_NAME	CACHE	BUFFER_POOL	DISK COMPRESSION	COMPRESS_FOR	INMEMORY ENABLED	INMEMORY PRIORITY	INMEMORY DISTRIBUTE	INMEMORY COMPRESSION
CUSTOMER	Y	KEEP	ENABLED	BASIC	ENABLED	NONE	AUTO	FOR QUERY LOW
SUPPLIER	Y	KEEP	ENABLED	BASIC	ENABLED	NONE	AUTO	FOR QUERY LOW
LINEORDER	Y	KEEP	ENABLED	BASIC	ENABLED	NONE	AUTO	FOR QUERY LOW
PART	Y	KEEP	ENABLED	BASIC	ENABLED	NONE	AUTO	FOR QUERY LOW
DATE_DIM	Y	KEEP	ENABLED	BASIC	ENABLED	NONE	AUTO	FOR QUERY LOW

```
SQL>
```

```
SQL> set echo off
```

```
SQL> @08_im_start_pop.sql
```

```
Connected.
```

```
SQL>
```

```
SQL> select /*+ full(LINEORDER) noparallel(LINEORDER) */ count(*) from LINEORDER;
```

```
          COUNT(*)  
-----  
59986052
```

```
SQL> select /*+ full(PART) noparallel(PART) */ count(*) from PART;
```

```
          COUNT(*)  
-----  
800000
```

```
SQL> select /*+ full(CUSTOMER) noparallel(CUSTOMER) */ count(*) from CUSTOMER;
```

```
          COUNT(*)  
-----  
300000
```

```
SQL> select /*+ full(SUPPLIER) noparallel(SUPPLIER) */ count(*) from SUPPLIER;
```

```
          COUNT(*)  
-----  
20000
```

```
SQL> select /*+ full(DATE_DIM) noparallel(DATE_DIM) */ count(*) from DATE_DIM;
```

COUNT(*)

2556

SQL>
SQL> set echo off

top - 19:06:23 up 27 min, 2 users, load average: 2.66, 0.97, 0.86
Tasks: 261 total, 6 running, 255 sleeping, 0 stopped, 0 zombie
%Cpu(s): 82.6 us, 12.5 sy, 0.0 ni, 0.0 id, 0.0 wa, 0.0 hi, 4.9 si, 0.0 st
KiB Mem : 12027112 total, 123456 free, 1138924 used, 10764732 buff/cache
KiB Swap: 3145724 total, 3133628 free, 12096 used, 658704 avail Mem

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
5708	oracle	20	0	10.924g	211880	104336	R	61.1	1.8	0:28.34	ora_w001_orcl12
6078	oracle	20	0	10.793g	146448	75152	R	36.9	1.2	0:10.66	ora_w002_orcl12
5704	oracle	20	0	10.930g	232864	98052	R	36.2	1.9	0:11.09	ora_w000_orcl12
6538	oracle	20	0	10.764g	199384	158556	S	36.2	1.7	0:07.60	ora_w003_orcl12
6575	oracle	20	0	10.712g	154872	137616	R	18.9	1.3	0:00.57	ora_m000_orcl12
35	root	20	0	0	0	0	S	3.0	0.0	0:18.22	kswapd0
5660	oracle	-2	0	10.698g	7144	4012	S	2.0	0.1	0:11.60	ora_vkrm_orcl12
1205	root	20	0	264016	25672	2508	S	1.0	0.2	0:19.31	Xorg
3329	oracle	20	0	1099792	10432	2280	S	0.7	0.1	0:01.55	gnome-settings-
3373	oracle	20	0	1610852	120324	18380	S	0.7	1.0	0:43.41	gnome-shell
5953	oracle	20	0	10.709g	161552	149592	S	0.7	1.3	0:03.04	ora_cjq0_orcl12
3	root	20	0	0	0	0	S	0.3	0.0	0:00.33	ksoftirqd/0
18	root	20	0	0	0	0	S	0.3	0.0	0:00.32	rcuos/1
4169	oracle	20	0	561864	13624	5840	S	0.3	0.1	0:08.36	gnome-terminal-
5444	root	20	0	0	0	0	S	0.3	0.0	0:00.19	kworker/1:0
5682	oracle	20	0	10.698g	10364	7236	S	0.3	0.1	0:01.79	ora_vkrm_orcl12
1	root	20	0	190868	3568	2104	S	0.0	0.0	0:02.00	systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:00.01	kthreadd
5	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	kworker/0:0H
7	root	20	0	0	0	0	S	0.0	0.0	0:00.83	rcu_sched

SQL> @09_im_populated.sql
Connected.
SQL>
SQL> -- Query the view v\$IM_SEGMENTS to shows what objects are in the column store
SQL> -- and how much of the objects were populated. When the BYTES_NOT_POPULATED is 0
SQL> -- it indicates the entire table was populated.
SQL>
SQL> SELECT v.owner, v.segment_name name, v.populate_status status, v.bytes_not_populated
2 FROM v\$im_segments v;

OWNER	NAME	STATUS	BYTES_NOT_POPULATED
SSB	PART	COMPLETED	0
SSB	CUSTOMER	COMPLETED	0
SSB	LINEORDER	COMPLETED	3383361536
SSB	DATE_DIM	COMPLETED	0
SSB	SUPPLIER	COMPLETED	0

SQL>
SQL> set echo off
SQL> @10_im_usage.sql
Connected.
SQL>
SQL> -- This query displays how much space is allocated/used in the In-Memory Column Store
SQL>

```
SQL> SELECT *
      2 FROM v$inmemory_area;
```

POOL	ALLOC_BYTES	USED_BYTES	POPULATE_STATUS	CON_ID
1MB POOL	4592762880	308281344	DONE	3
64KB POOL	1157627904	3276800	DONE	3

```
SQL>
SQL> set echo off
SQL> @11_cat_alert_log.sql
Connected.
SQL>
SQL> -- cat the alert.log to show out of memory message
SQL>
SQL> host tail -10 /u01/app/oracle/diag/rdbms/orcl12c/orcl12c/trace/alert_orcl12c.log
user-specified limit on the amount of space that will be used by this
database for recovery-related files, and does not reflect the amount of
space available in the underlying filesystem or ASM diskgroup.
2018-01-16T18:57:43.085055-05:00
ORCL(3):Setting Resource Manager plan SCHEDULER[0x4ABE]:DEFAULT_MAINTENANCE_PLAN via scheduler window
ORCL(3):Setting Resource Manager plan DEFAULT_MAINTENANCE_PLAN via parameter
2018-01-16T19:06:32.556256-05:00
Shared IO Pool defaulting to 32MB. Trying to get it from Buffer Cache for process 6575.
2018-01-16T19:06:47.224045-05:00
ORCL(3):Insufficient memory to populate table to inmemory area
```

```
SQL>
SQL> set echo off
SQL> @12_alter_inmemory_size.sql
Connected.
SQL>
SQL> -- This command increases the size of the IM column store
SQL>
SQL> alter system set inmemory_size=5504M;
```

System altered.

```
SQL>
SQL> set echo off
```

```
top - 19:19:54 up 41 min, 2 users, load average: 2.49, 1.06, 0.82
Tasks: 262 total, 5 running, 257 sleeping, 0 stopped, 0 zombie
%Cpu(s): 90.1 us, 7.5 sy, 0.0 ni, 0.0 id, 0.0 wa, 0.0 hi, 2.4 si, 0.0 st
KiB Mem : 12027112 total, 138960 free, 1261928 used, 10626224 buff/cache
KiB Swap: 3145724 total, 3137488 free, 8236 used, 1423988 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
6743	oracle	20	0	10.923g	615120	495844	R	78.6	5.1	0:57.72	ora_w000_orcl12
6777	oracle	20	0	10.750g	355500	322132	R	39.8	3.0	0:32.67	ora_w005_orcl12
6644	oracle	20	0	10.939g	446072	294576	S	35.2	3.7	0:32.84	ora_w002_orcl12
6808	oracle	20	0	10.954g	470832	311868	R	32.6	3.9	0:32.68	ora_w006_orcl12
3373	oracle	20	0	1610916	123888	15940	S	5.3	1.0	1:08.04	gnome-shell
1205	root	20	0	264016	26288	3124	S	2.0	0.2	0:32.32	Xorg
5660	oracle	-2	0	10.698g	7144	4012	S	1.6	0.1	0:26.77	ora_vktn_orcl12
4169	oracle	20	0	562352	17500	9120	S	1.3	0.1	0:13.40	gnome-terminal-
5716	oracle	20	0	10.721g	174956	148904	S	1.0	1.5	0:02.58	ora_mmon_orcl12
5688	oracle	20	0	10.701g	25168	19200	S	0.7	0.2	0:00.88	ora_dia0_orcl12
3	root	20	0	0	0	0	S	0.3	0.0	0:00.43	ksoftirqd/0
35	root	20	0	0	0	0	S	0.3	0.0	0:19.51	kswapd0
3329	oracle	20	0	1099792	11572	3412	S	0.3	0.1	0:02.18	gnome-settings-
5682	oracle	20	0	10.698g	13216	10088	S	0.3	0.1	0:04.54	ora_vkrm_orcl12
5953	oracle	20	0	10.709g	180060	168100	S	0.3	1.5	0:04.97	ora_cjq0_orcl12
7295	oracle	20	0	157832	4400	3600	R	0.3	0.0	0:00.06	top

```

1 root      20   0 190868  3588  2124 S  0.0  0.0  0:02.09 systemd
2 root      20   0     0     0     0 S  0.0  0.0  0:00.01 kthreadd
5 root       0 -20     0     0     0 S  0.0  0.0  0:00.00 kworker/0:0H
7 root      20   0     0     0     0 S  0.0  0.0  0:01.03 rcu_sched

```

SQL> @14_im_populated.sql

Connected.

SQL>

SQL> -- Query the view v\$IM_SEGMENTS to shows what objects are in the column store
SQL> -- and how much of the objects were populated. When the BYTES_NOT_POPULATED is 0
SQL> -- it indicates the entire table was populated.

SQL>

```

SQL> SELECT v.owner, v.segment_name name, v.populate_status status, v.bytes_not_populated
2 FROM v$im_segments v;

```

OWNER	NAME	STATUS	BYTES_NOT_POPULATED
SSB	PART	COMPLETED	0
SSB	CUSTOMER	COMPLETED	0
SSB	LINEORDER	STARTED	1676083200
SSB	DATE_DIM	COMPLETED	0
SSB	SUPPLIER	COMPLETED	0

SQL>

SQL> set echo off

SQL> /

OWNER	NAME	STATUS	BYTES_NOT_POPULATED
SSB	PART	COMPLETED	0
SSB	CUSTOMER	COMPLETED	0
SSB	LINEORDER	COMPLETED	0
SSB	DATE_DIM	COMPLETED	0
SSB	SUPPLIER	COMPLETED	0

SQL>

SQL> @15_im_usage.sql

Connected.

SQL>

SQL> -- This query displays how much space is allocated/used in the In-Memory Column Store

SQL>

```

SQL> SELECT *
2 FROM v$inmemory_area;

```

POOL	ALLOC_BYTES	USED_BYTES	POPULATE_STATUS	CON_ID
1MB POOL	4592762880	3202351104	DONE	3
64KB POOL	1157627904	23134208	DONE	3

SQL>

SQL> set echo off

SQL> @16_comp_ratio.sql

Connected.

SQL>

SQL> -- This query compares the actual size of the tables on disk to the size in the

SQL> -- In-Memory Column Store, in order to calculate the compression ratio

SQL> -- (Does not factor in on-disk compression)

SQL>

```

SQL> SELECT v.owner,
2         v.segment_name      name,
3         v.bytes             on_disk_size,
4         v.inmemory_size    in_mem_size,
5         ROUND(v.bytes / v.inmemory_size, 2) comp_ratio

```

```
6 FROM v$im_segments v
7 ORDER BY 4;
```

OWNER	NAME	ON_DISK_SIZE	IN_MEM_SIZE	COMP_RATIO
SSB	DATE_DIM	122880	1310720	.09
SSB	SUPPLIER	1761280	2359296	.75
SSB	PART	56721408	18284544	3.1
SSB	CUSTOMER	24862720	23330816	1.07
SSB	LINEORDER	3693232128	3180199936	1.16

```
SQL>
SQL> set echo off
```

```
*****
Part 2:
*****
```

```
SQL> @01_im_query_stats.sql
Connected.
SQL>
SQL> -- In-Memory Column Store query
SQL>
SQL> select      max(lo_ordtotalprice) most_expensive_order From LINEORDER;
```

```
MOST_EXPENSIVE_ORDER
-----
57346348
```

```
Elapsed: 00:00:00.12
SQL>
SQL> set echo off
Hit enter ...
```

```
PLAN_TABLE_OUTPUT
```

```
-----
SQL_ID 7htp8zu9x0vq7, child number 0
-----
select max(lo_ordtotalprice) most_expensive_order From LINEORDER

Plan hash value: 2267213921
```

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT				4888 (100)	
1	SORT AGGREGATE		1	6		
2	TABLE ACCESS INMEMORY FULL	LINEORDER	59M	343M	4888 (7)	00:00:01

```
14 rows selected.
Hit enter ...
```

NAME	VALUE
IM scan CUs columns accessed	222
IM scan CUs columns theoretical max	1887
IM scan CUs current	111
IM scan CUs memcompress for query low	111
IM scan CUs no cleanout	111
IM scan CUs pcode aggregation pushdown	111

```

IM scan CUs readlist creation accumulated time      46
IM scan CUs readlist creation number              111
IM scan CUs split pieces                          190
IM scan bytes in-memory                          3112998500
IM scan bytes uncompressed                       13186858332
IM scan rows                                     59986052
IM scan rows pcode aggregated                    59986052
IM scan rows projected                          111
IM scan rows valid                              59986052
IM simd decode unpack calls                     111
parse time cpu                                  28
parse time elapsed                              37
physical reads cache                            72
redo size                                        744
session logical reads                          450908
session logical reads - IM                    450834
session pga memory                             10848440
session pga memory max                        10848440

```

21 rows selected.

```

SQL>
SQL> @02_buffer_query_stats.sql
Connected.
SQL>
SQL> -- Buffer Cache query with the column store disabled via NO_INMEMORY hint
SQL>
SQL> select /*+ NO_INMEMORY */ max(lo_ordtotalprice) most_expensive_order From LINEORDER;

```

```

MOST_EXPENSIVE_ORDER
-----
57346348

```

```

Elapsed: 00:00:09.10
SQL>
SQL> set echo off
Hit enter ...

```

PLAN_TABLE_OUTPUT

```

SQL_ID 9pnpqzggghwb1g, child number 0
-----
select /*+ NO_INMEMORY */ max(lo_ordtotalprice) most_expensive_order
From LINEORDER

```

Plan hash value: 2267213921

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT				123K(100)	
1	SORT AGGREGATE		1	6		
2	TABLE ACCESS FULL	LINEORDER	59M	343M	123K (1)	00:00:05

15 rows selected.

Hit enter ...

```

NAME
-----
VALUE
-----

```



```

IM scan segments disk          1
redo size                      876
session logical reads         450929
session pga memory            11045048
session pga memory max        11045048

```

```

SQL>
SQL> @03_single_key_im.sql
Connected.
SQL>
SQL> -- In-Memory Column Store query
SQL>
SQL> select lo_orderkey, lo_custkey, lo_revenue
  2 from LINEORDER
  3 where lo_orderkey = 5000000;

```

```

      LO_ORDERKEY      LO_CUSTKEY      LO_REVENUE
-----
      50000000         48647          2456268

```

Elapsed: 00:00:00.04

```

SQL>
SQL> set echo off
Hit enter ...

```

PLAN_TABLE_OUTPUT

```

SQL_ID 513g163sj3cv2, child number 0
-----
select lo_orderkey, lo_custkey, lo_revenue from LINEORDER where
lo_orderkey = 5000000

```

Plan hash value: 4017770458

```

-----
| Id | Operation | Name | Rows | Bytes | Cost (%CPU) | Time |
-----
| 0 | SELECT STATEMENT | | | | 4808 (100) | |
|* 1 | TABLE ACCESS INMEMORY FULL | LINEORDER | 4 | 68 | 4808 (5) | 00:00:01 |
-----

```

Predicate Information (identified by operation id):

```

-----
1 - inmemory("LO_ORDERKEY"=5000000)
   filter("LO_ORDERKEY"=5000000)

```

20 rows selected.

```

SQL>
SQL> @04_single_key_buffer.sql
Connected.
SQL>
SQL> -- Buffer Cache query with the column store disabled via INMEMORY_QUERY parameter
SQL>
SQL> select /*+ NO_INMEMORY */
  2 lo_orderkey, lo_custkey, lo_revenue
  3 from LINEORDER
  4 where lo_orderkey = 5000000;

```

```

      LO_ORDERKEY      LO_CUSTKEY      LO_REVENUE
-----

```

5000000 48647 2456268

Elapsed: 00:00:06.48
SQL>
SQL> set echo off
Hit enter ...

PLAN_TABLE_OUTPUT

SQL_ID 4rcm8pc7xktun, child number 0

select /*+ NO_INMEMORY */ lo_orderkey, lo_custkey, lo_revenue
from LINEORDER where lo_orderkey = 5000000

Plan hash value: 4017770458

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT				122K(100)	
* 1	TABLE ACCESS FULL	LINEORDER	4	68	122K (1)	00:00:05

Predicate Information (identified by operation id):

1 - filter("LO_ORDERKEY">=5000000)

19 rows selected.

Hit enter ...

NAME	VALUE
IM scan segments disk	1
parse time cpu	1
parse time elapsed	1
redo size	876
session logical reads	450930
session pga memory	11110584
session pga memory max	11110584

7 rows selected.

SQL>
SQL> @05_index_comparison.sql
Connected.
SQL>
SQL> -- Enable the use of invisible indexes
SQL>
SQL> alter session set optimizer_use_invisible_indexes=true;

Session altered.

SQL>
SQL> -- Execute the query again include a new comment to ensure a hard parse
SQL>
SQL> set timing on
SQL>
SQL> Select /* With index */ lo_orderkey, lo_custkey, lo_revenue
2 From LINEORDER

```
3 Where lo_orderkey = 5000000;
```

LO_ORDERKEY	LO_CUSTKEY	LO_REVENUE
5000000	48647	2456268

Elapsed: 00:00:00.15

SQL>

SQL> set timing off

SQL>

SQL> pause Hit enter ...

Hit enter ...

SQL>

SQL> select * from table(dbms_xplan.display_cursor());

PLAN_TABLE_OUTPUT

SQL_ID 8kgqurq43dgg1, child number 0

Select /* With index */ lo_orderkey, lo_custkey, lo_revenue From
LINEORDER Where lo_orderkey = 5000000

Plan hash value: 747895665

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT				4 (100)	
1	TABLE ACCESS BY INDEX ROWID BATCHED	LINEORDER	4	68	4 (0)	00:00:01
* 2	INDEX RANGE SCAN	STEP3_3	4		3 (0)	00:00:01

Predicate Information (identified by operation id):

2 - access("LO_ORDERKEY"=5000000)

20 rows selected.

SQL>

SQL> -- Compare the Elapsed time of the query In-Memory and in the index access

SQL>

SQL> alter session set optimizer_use_invisible_indexes=false;

Session altered.

SQL>

SQL> set echo off

SQL> @06_storage_index.sql

Connected.

SQL>

SQL> -- Execute the In-Memory Column Store query

SQL>

SQL> select lo_orderkey, lo_custkey, lo_revenue

2 from LINEORDER

3 where lo_orderkey = 5000000;

LO_ORDERKEY	LO_CUSTKEY	LO_REVENUE
5000000	48647	2456268

Elapsed: 00:00:00.01

```
SQL>
SQL> set echo off
Hit enter ...
```

PLAN_TABLE_OUTPUT

SQL_ID 513g163sj3cv2, child number 0

```
select lo_orderkey, lo_custkey, lo_revenue from LINEORDER where
lo_orderkey = 5000000
```

Plan hash value: 4017770458

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT				4808 (100)	
* 1	TABLE ACCESS INMEMORY FULL	LINEORDER	4	68	4808 (5)	00:00:01

Predicate Information (identified by operation id):

```
1 - inmemory("LO_ORDERKEY"=5000000)
   filter("LO_ORDERKEY"=5000000)
```

20 rows selected.

Hit enter ...

NAME	VALUE
IM scan CUs columns accessed	2
IM scan CUs columns theoretical max	1887
IM scan CUs current	111
IM scan CUs memcompress for query low	111
IM scan CUs no cleanout	111
IM scan CUs pcode pred evaled	1
IM scan CUs predicates applied	111
IM scan CUs predicates optimized	110
IM scan CUs predicates received	111
IM scan CUs pruned	110
IM scan CUs readlist creation accumulated time	36
IM scan CUs readlist creation number	111
IM scan CUs split pieces	190
IM scan bytes in-memory	3112998500
IM scan bytes uncompressed	13186858332
IM scan rows	59986052
IM scan rows optimized	59441024
IM scan rows projected	1
IM scan rows valid	545028
IM scan segments minmax eligible	111
IM simd compare calls	1
IM simd decode calls	3
redo size	744
session logical reads	450908
session logical reads - IM	450834
session pga memory	10979512
session pga memory max	10979512

27 rows selected.

```

SQL>
SQL> @07_multi_preds.sql
Connected.
SQL>
SQL> -- In-Memory Column Store query
SQL>
SQL> select lo_orderkey, lo_custkey, lo_revenue
  2 from LINEORDER
  3 where lo_custkey = 5641
  4 and lo_shipmode = 'SHIP'
  5 and lo_orderpriority = '5-LOW';

```

LO_ORDERKEY	LO_CUSTKEY	LO_REVENUE
182852	5641	4117386
13644419	5641	3786727
16925125	5641	5711620
16925125	5641	6400967
18139779	5641	4508300
22534688	5641	705728
22534688	5641	3400905
27479847	5641	7247038
27479847	5641	4341522
28390534	5641	486770
40154336	5641	3052806
51805731	5641	4164048
59428583	5641	6083192
59428583	5641	3904048

14 rows selected.

```

Elapsed: 00:00:00.08
SQL>
SQL> set echo off
Hit enter ...

```

PLAN_TABLE_OUTPUT

```

SQL_ID gcrg8wtsj5wxh, child number 0
select lo_orderkey, lo_custkey, lo_revenue from LINEORDER where
lo_custkey = 5641 and lo_shipmode = 'SHIP' and
lo_orderpriority = '5-LOW'

```

Plan hash value: 4017770458

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT				5709 (100)	
* 1	TABLE ACCESS INMEMORY FULL	LINEORDER	9	396	5709 (20)	00:00:01

Predicate Information (identified by operation id):

```

1 - inmemory(("LO_CUSTKEY"=5641 AND "LO_SHIPMODE"='SHIP' AND
"LO_ORDERPRIORITY"='5-LOW'))
filter(("LO_CUSTKEY"=5641 AND "LO_SHIPMODE"='SHIP' AND
"LO_ORDERPRIORITY"='5-LOW'))

```

23 rows selected.

Hit enter ...

NAME	VALUE
IM scan CUs columns accessed	121
IM scan CUs columns theoretical max	1887
IM scan CUs current	111
IM scan CUs memcompress for query low	111
IM scan CUs no cleanout	111
IM scan CUs pcode pred evaled	305
IM scan CUs pcode selective done	97
IM scan CUs predicates applied	333
IM scan CUs predicates received	333
IM scan CUs readlist creation accumulated time	224
IM scan CUs readlist creation number	111
IM scan CUs split pieces	190
IM scan bytes in-memory	3112998500
IM scan bytes uncompressed	13186858332
IM scan rows	59986052
IM scan rows projected	14
IM scan rows valid	59986052
IM scan segments minmax eligible	111
IM simd compare calls	163
IM simd compare selective calls	97
IM simd decode calls	50
parse time cpu	2
parse time elapsed	2
physical reads cache	2
redo size	1860
session logical reads	451008
session logical reads - IM	450834
session pga memory	11176120
session pga memory max	11176120

29 rows selected.

```
SQL>
SQL> @08_multi_col.sql
Connected.
SQL>
SQL> -- In-Memory Column Store query
SQL>
SQL> Select lo_orderkey, lo_revenue
  2 From LINEORDER
  3 Where lo_revenue = (Select min(lo_revenue)
  4                      From LINEORDER
  5                      Where lo_supplycost = (Select max(lo_supplycost)
  6                                                  From LINEORDER
  7                                                  Where lo_quantity > 10)
  8                      And lo_shipmode LIKE 'TRUCK%'
  9                      And lo_discount between 2 and 5
 10 );
```

LO_ORDERKEY	LO_REVENUE
3842596	199404
5335335	199404
21888516	199404
32749476	199404
36111875	199404
37275106	199404

49976640

199404

7 rows selected.

Elapsed: 00:00:00.49

SQL>

SQL> set echo off

Hit enter ...

PLAN_TABLE_OUTPUT

SQL_ID bnysz8dqbamnd, child number 0

```

Select lo_orderkey, lo_revenue From LINEORDER Where lo_revenue =
(Select min(lo_revenue) From LINEORDER
 Where lo_supplycost = (Select max(lo_supplycost)
 From LINEORDER
 Where lo_quantity > 10) And
lo_shipmode LIKE 'TRUCK%' And lo_discount between
2 and 5 )

```

Plan hash value: 482539359

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT				15879 (100)	
* 1	TABLE ACCESS INMEMORY FULL	LINEORDER	10	120	5162 (12)	00:00:01
2	SORT AGGREGATE		1	25		
* 3	TABLE ACCESS INMEMORY FULL	LINEORDER	284	7100	5555 (18)	00:00:01
4	SORT AGGREGATE		1	8		
* 5	TABLE ACCESS INMEMORY FULL	LINEORDER	48M	373M	5162 (12)	00:00:01

Predicate Information (identified by operation id):

- 1 - inmemory("LO_REVENUE"=)
 - filter("LO_REVENUE"=)
- 3 - inmemory(("LO_SHIPMODE" LIKE 'TRUCK%' AND "LO_DISCOUNT"<=5 AND "LO_DISCOUNT">=2 AND "LO_SUPPLYCOST"=))
 - filter(("LO_SHIPMODE" LIKE 'TRUCK%' AND "LO_DISCOUNT"<=5 AND "LO_DISCOUNT">=2 AND "LO_SUPPLYCOST"=))
- 5 - inmemory("LO_QUANTITY">10)
 - filter("LO_QUANTITY">10)

35 rows selected.

Hit enter ...

NAME	VALUE
IM scan CUs columns accessed	658
IM scan CUs columns theoretical max	5661
IM scan CUs current	333
IM scan CUs memcompress for query low	333
IM scan CUs no cleanout	333
IM scan CUs pcode aggregation pushdown	114
IM scan CUs pcode pred evald	777
IM scan CUs predicates applied	444
IM scan CUs predicates received	444

```

IM scan CUs readlist creation accumulated time      235
IM scan CUs readlist creation number                333
IM scan CUs split pieces                            570
IM scan bytes in-memory                            9338995500
IM scan bytes uncompressed                          39560574996
IM scan rows                                        179958156
IM scan rows pcode aggregated                       47988534
IM scan rows projected                              121
IM scan rows valid                                  179958156
IM scan segments minmax eligible                    222
IM simd compare calls                               303
IM simd decode calls                                14
IM simd decode unpack calls                          234
IM simd set membership calls                        111
parse time cpu                                      1
parse time elapsed                                  1
redo size                                           3800
session logical reads                               1352633
session logical reads - IM                          1352502
session pga memory                                  10913976
session pga memory max                              10913976

```

30 rows selected.

```

SQL>
SQL> @00_pre-ime.sql
Connected.
SQL> set numwidth 20
SQL>
SQL> -- In-Memory Column Store query - No IMEs
SQL>
SQL> Select lo_shipmode, sum(lo_ordtotalprice),
2         sum(lo_ordtotalprice - (lo_ordtotalprice*(lo_discount/100)) + lo_tax) discount_price
3 From   LINEORDER
4 group by
5     lo_shipmode;

```

LO_SHIPMOD	SUM(LO_ORDTOTALPRICE)	DISCOUNT_PRICE
REG AIR	161864992173671	153805526566678.06
AIR	161811429297122	153751580704864.92
FOB	161837922547928	153778178139883.79
TRUCK	161820421065658	153761833420584.79
SHIP	161872168268007	153814810364218.27
RAIL	161896622020617	15383832879643.73
MAIL	161844311638291	153787676162546.37

7 rows selected.

```

Elapsed: 00:00:17.55
SQL>
SQL> set echo off
Hit enter ...

```

PLAN_TABLE_OUTPUT

```

-----
SQL_ID  g5rn8q8zp92tg, child number 0
-----
Select lo_shipmode, sum(lo_ordtotalprice),          sum(lo_ordtotalprice
- (lo_ordtotalprice*(lo_discount/100)) + lo_tax) discount_price From
LINEORDER group by   lo_shipmode

```

Plan hash value: 3675673598

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT				7393 (100)	
1	HASH GROUP BY		7	161	7393 (39)	00:00:01
2	TABLE ACCESS INMEMORY FULL	LINEORDER	59M	1315M	5401 (16)	00:00:01

16 rows selected.

Hit enter ...

NAME	VALUE
IM scan CUs columns accessed	222
IM scan CUs columns theoretical max	1887
IM scan CUs current	111
IM scan CUs memcompress for query low	111
IM scan CUs no cleanout	111
IM scan CUs pcode aggregation pushdown	111
IM scan CUs readlist creation accumulated time	229
IM scan CUs readlist creation number	111
IM scan CUs split pieces	190
IM scan bytes in-memory	3112998500
IM scan bytes uncompressed	13186858332
IM scan rows	59986052
IM scan rows pcode aggregated	59986052
IM scan rows projected	777
IM scan rows valid	59986052
IM simd decode unpack calls	444
parse time elapsed	1
redo size	1756
session logical reads	450930
session logical reads - IM	450834
session pga memory	11176120
session pga memory max	11176120

22 rows selected.

```
SQL>
SQL> @10_im_usage.sql
Connected.
SQL>
SQL> -- This query displays how much space is allocated/used in the In-Memory Column Store
SQL>
SQL> SELECT *
  2 FROM v$inmemory_area;
```

POOL	ALLOC_BYTES	USED_BYTES	POPULATE_STATUS	CON_ID
1MB POOL	4592762880	3202351104	DONE	3
64KB POOL	1157627904	23134208	DONE	3

```
SQL>
SQL> set echo off
SQL> @11_create_ime.sql
Connected.
SQL>
SQL> -- Create In-Memory Column Expression
SQL>
SQL> alter table lineorder no inmemory;
```

Table altered.

```
SQL> alter table lineorder add v1 invisible as (lo_ordtotalprice - (lo_ordtotalprice*(lo_discount/100)) + lo_tax);
```

Table altered.

```
SQL> alter table lineorder inmemory;
```

Table altered.

```
SQL> select count(*) from lineorder;
```

```
-----  
COUNT(*)  
-----  
59986052
```

```
SQL>
```

```
SQL> @12_im_populated.sql
```

```
SQL> connect ssb/oracle12@orcl
```

```
Connected.
```

```
SQL>
```

```
SQL> set pages 9999
```

```
SQL> set lines 150
```

```
SQL> column name format a30
```

```
SQL> column owner format a20
```

```
SQL> column segment_name format a30
```

```
SQL> column populate_status format a20
```

```
SQL> set echo on
```

```
SQL>
```

```
SQL> -- Query the view v$IM_SEGMENTS to shows what objects are in the column store
```

```
SQL> -- and how much of the objects were populated. When the BYTES_NOT_POPULATED is 0
```

```
SQL> -- it indicates the entire table was populated.
```

```
SQL>
```

```
SQL> SELECT v.owner, v.segment_name name, v.populate_status status, v.bytes_not_populated  
2 FROM v$im_segments v;
```

OWNER	NAME	STATUS	BYTES_NOT_POPULATED
SSB	PART	COMPLETED	0
SSB	CUSTOMER	COMPLETED	0
SSB	LINEORDER	STARTED	3550740480
SSB	DATE_DIM	COMPLETED	0
SSB	SUPPLIER	COMPLETED	0

```
SQL>
```

```
SQL> set echo off
```

```
SQL> /
```

OWNER	NAME	STATUS	BYTES_NOT_POPULATED
SSB	PART	COMPLETED	0
SSB	CUSTOMER	COMPLETED	0
SSB	LINEORDER	STARTED	2178875392
SSB	DATE_DIM	COMPLETED	0
SSB	SUPPLIER	COMPLETED	0

```
SQL> /
```

OWNER	NAME	STATUS	BYTES_NOT_POPULATED
SSB	PART	COMPLETED	0
SSB	CUSTOMER	COMPLETED	0
SSB	LINEORDER	COMPLETED	0

```
SSB          DATE_DIM          COMPLETED          0
SSB          SUPPLIER          COMPLETED          0
```

```
SQL>
SQL> @i3_im_usage.sql
Connected.
SQL>
SQL> -- This query displays how much space is allocated/used in the In-Memory Column Store
SQL>
SQL> SELECT *
  2 FROM v$inmemory_area;
```

POOL	ALLOC_BYTES	USED_BYTES	POPULATE_STATUS	CON_ID
1MB POOL	4592762880	4169138176	DONE	3
64KB POOL	1157627904	23199744	DONE	3

```
SQL>
SQL> set echo off
SQL> @i4_post-ime.sql
Connected.
SQL> set numwidth 20
SQL>
SQL> -- In-Memory Column Store query - With IME created
SQL>
SQL> Select lo_shipmode, sum(lo_ordtotalprice),
  2      sum(lo_ordtotalprice - (lo_ordtotalprice*(lo_discount/100)) + lo_tax) discount_price
  3 From LINEORDER
  4 group by
  5      lo_shipmode;
```

LO_SHIPMOD	SUM(LO_ORDTOTALPRICE)	DISCOUNT_PRICE
REG AIR	161864992173671	153805526566678.06
AIR	161811429297122	153751580704864.92
FOB	161837922547928	153778178139883.79
TRUCK	161820421065658	153761833420584.79
SHIP	161872168268007	153814810364218.27
RAIL	161896622020617	153838322879643.73
MAIL	161844311638291	153787676162546.37

7 rows selected.

```
Elapsed: 00:00:09.61
SQL>
SQL> set echo off
Hit enter ...
```

PLAN_TABLE_OUTPUT

```
SQL_ID g5rn8q8zp92tg, child number 0
-----
Select lo_shipmode, sum(lo_ordtotalprice),      sum(lo_ordtotalprice
- (lo_ordtotalprice*(lo_discount/100)) + lo_tax) discount_price From
LINEORDER group by lo_shipmode
```

Plan hash value: 3675673598

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT				7564 (100)	
1	HASH GROUP BY		7	210	7564 (40)	00:00:01

| 2 | TABLE ACCESS INMEMORY FULL| LINEORDER | 59M| 1716M| 5572 (18)| 00:00:01 |

16 rows selected.

Hit enter ...

NAME	VALUE
IM scan CUs columns accessed	222
IM scan CUs columns theoretical max	1887
IM scan CUs current	111
IM scan CUs memcompress for query low	111
IM scan CUs no cleanout	111
IM scan CUs pcode aggregation IME	111
IM scan CUs pcode aggregation pushdown	222
IM scan CUs readlist creation accumulated time	288
IM scan CUs readlist creation number	111
IM scan CUs split pieces	314
IM scan EU bytes in-memory	730203275
IM scan EU bytes uncompressed	412353699
IM scan EU rows	59986052
IM scan EUs columns accessed	111
IM scan EUs columns theoretical max	1998
IM scan EUs memcompress for query low	111
IM scan EUs split pieces	32
IM scan bytes in-memory	3112998458
IM scan bytes uncompressed	13186858332
IM scan rows	59986052
IM scan rows pcode aggregated	59986052
IM scan rows projected	777
IM scan rows valid	59986052
IM simd decode unpack calls	444
parse time cpu	1
parse time elapsed	5
redo size	744
session logical reads	450910
session logical reads - IM	450834
session pga memory	11176120
session pga memory max	11176120

31 rows selected.

SQL>

Part3:

SQL> @01_join_im.sql
Connected.

SQL>

SQL> -- In-Memory Column Store query

SQL>

```
SQL> Select sum(lo_extendedprice * lo_discount) revenue
 2 From LINEORDER l, DATE_DIM d
 3 Where l.lo_orderdate = d.d_datekey
 4 And l.lo_discount between 2 and 3
 5 And l.lo_quantity < 24
 6 And d.d_date='December 24, 1996';
```

REVENUE

9710699495

Elapsed: 00:00:00.14
SQL>
SQL> set echo off
Hit enter ...

PLAN_TABLE_OUTPUT

SQL_ID b2jysvyzbss5p, child number 0

Select sum(lo_extendedprice * lo_discount) revenue From LINEORDER l,
DATE_DIM d Where l.lo_orderdate = d.d_datekey And l.lo_discount
between 2 and 3 And l.lo_quantity < 24 And d.d_date='December 24,
1996'

Plan hash value: 2403472142

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT				6124 (100)	
1	SORT AGGREGATE		1	43		
* 2	HASH JOIN		3298	138K	6124 (26)	00:00:01
3	JOIN FILTER CREATE	:BF0000	1	25	1 (0)	00:00:01
* 4	TABLE ACCESS INMEMORY FULL	DATE_DIM	1	25	1 (0)	00:00:01
5	JOIN FILTER USE	:BF0000	7935K	136M	6100 (26)	00:00:01
* 6	TABLE ACCESS INMEMORY FULL	LINEORDER	7935K	136M	6100 (26)	00:00:01

Predicate Information (identified by operation id):

2 - access("L"."LO_ORDERDATE"="D"."D_DATEKEY")
4 - inmemory("D"."D_DATE"='December 24, 1996')
filter("D"."D_DATE"='December 24, 1996')
6 - inmemory(("L"."LO_DISCOUNT"<=3 AND "L"."LO_QUANTITY"<24 AND
"L"."LO_DISCOUNT">=2 AND SYS_OP_BLOOM_FILTER(:BF0000,"L"."LO_ORDERDATE")))
filter(("L"."LO_DISCOUNT"<=3 AND "L"."LO_QUANTITY"<24 AND
"L"."LO_DISCOUNT">=2 AND SYS_OP_BLOOM_FILTER(:BF0000,"L"."LO_ORDERDATE")))

32 rows selected.

Hit enter ...

NAME	VALUE
IM scan CUs columns accessed	271
IM scan CUs columns theoretical max	1904
IM scan CUs current	112
IM scan CUs memcompress for query low	112
IM scan CUs no cleanout	112
IM scan CUs pcode pred evald	556
IM scan CUs predicates applied	334
IM scan CUs predicates received	334
IM scan CUs readlist creation accumulated time	178
IM scan CUs readlist creation number	112
IM scan CUs split pieces	204
IM scan bytes in-memory	3113087591
IM scan bytes uncompressed	13187368818

```

IM scan rows 59988608
IM scan rows projected 2132
IM scan rows valid 59988608
IM scan segments minmax eligible 1
IM simd compare calls 334
IM simd decode unpack calls 442
parse time cpu 1
parse time elapsed 1
redo size 744
session logical reads 451064
session logical reads - IM 450849
session pga memory 11634872
session pga memory max 11634872

```

26 rows selected.

```

SQL> @02_join_buffer.sql
Connected.
SQL>
SQL> -- Buffer Cache query with the column store disables via the inmemory_query parameter
SQL>

```

```

SQL> SELECT /*+ NO_INMEMORY */
2      SUM(lo_extendedprice * lo_discount) revenue
3 FROM   lineorder l,
4        date_dim d
5 WHERE  l.lo_orderdate = d.d_datekey
6 AND    l.lo_discount BETWEEN 2 AND 3
7 AND    l.lo_quantity < 24
8 AND    d.d_date='December 24, 1996';

```

```

-----
REVENUE
-----
9710699495

```

```

Elapsed: 00:00:09.57
SQL>
SQL> set echo off
Hit enter ...

```

PLAN_TABLE_OUTPUT

```

-----
SQL_ID 5fn4vtw9a8a7u, child number 0
-----
SELECT /*+ NO_INMEMORY */      SUM(lo_extendedprice * lo_discount)
revenue FROM   lineorder l,      date_dim d WHERE  l.lo_orderdate =
d.d_datekey AND    l.lo_discount BETWEEN 2 AND 3 AND    l.lo_quantity <
24 AND    d.d_date='December 24, 1996'

```

Plan hash value: 2963256899

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT				123K(100)	
1	SORT AGGREGATE		1	43		
* 2	HASH JOIN		3298	138K	123K (1)	00:00:05
* 3	TABLE ACCESS FULL	DATE_DIM	1	25	15 (0)	00:00:01
* 4	TABLE ACCESS FULL	LINEORDER	7935K	136M	123K (1)	00:00:05

Predicate Information (identified by operation id):

```

2 - access("L"."LO_ORDERDATE"="D"."D_DATEKEY")
3 - filter("D"."D_DATE"='December 24, 1996')
4 - filter(("L"."LO_DISCOUNT"<=3 AND "L"."LO_QUANTITY"<24 AND
"L"."LO_DISCOUNT">=2))

```

27 rows selected.

Hit enter ...

NAME	VALUE
IM scan segments disk	2
redo size	964
session logical reads	450958
session pga memory	11176120
session pga memory max	11176120

SQL>

SQL> @03_3join_im.sql

Connected.

SQL>

SQL> -- In-Memory Column Store query - Multi-table join

SQL>

```

SQL> Select /*+ parallel(2) */
2      d.d_year, p.p_brand1, sum(lo_revenue) tot_rev, count(p.p_partkey) tot_parts
3  From lineorder l,
4      date_dim d,
5      part p,
6      supplier s
7  Where l.lo_orderdate = d.d_datekey
8  And   l.lo_partkey   = p.p_partkey
9  And   l.lo_suppkey   = s.s_suppkey
10 And   p.p_category   = 'MFGR#12'
11 And   s.s_region     = 'AMERICA'
12 AND   d.d_year       = 1997
13 Group by d.d_year, p.p_brand1;

```

D_YEAR	P_BRAND1	TOT_REV	TOT_PARTS
1997	MFGR#1211	6852509575	1864
1997	MFGR#121	6409702180	1852
1997	MFGR#1225	6804217225	1848
1997	MFGR#1224	7373166413	2027
1997	MFGR#1231	6839363437	1866
1997	MFGR#1233	6914572704	1871
1997	MFGR#125	6416763567	1821
1997	MFGR#1229	6481227038	1775
1997	MFGR#1237	7041724061	1963
1997	MFGR#1214	6630127600	1834
1997	MFGR#1227	6713851455	1814
1997	MFGR#1234	6695984533	1853
1997	MFGR#1220	6613283998	1796
1997	MFGR#1226	7185780867	1969
1997	MFGR#124	6817087386	1877
1997	MFGR#1216	6877592440	1871
1997	MFGR#1215	7079477060	1888
1997	MFGR#1213	6686343443	1820
1997	MFGR#1235	7075150948	1947
1997	MFGR#1228	7038686432	1957
1997	MFGR#123	6280463233	1773
1997	MFGR#1230	6596531127	1799
1997	MFGR#1232	7404918843	2054

```

1997 MFGR#122          7101221796          1959
1997 MFGR#128          6597639363          1834
1997 MFGR#1240         7056019394          1980
1997 MFGR#1210         6795372926          1879
1997 MFGR#1238         6443513085          1730
1997 MFGR#1218         6841323272          1849
1997 MFGR#1236         6827320374          1938
1997 MFGR#1219         6651946261          1794
1997 MFGR#126          6213627043          1781
1997 MFGR#1222         6551372899          1806
1997 MFGR#1239         6227318243          1761
1997 MFGR#127          6765391794          1902
1997 MFGR#129          6164317597          1713
1997 MFGR#1212         6573951551          1792
1997 MFGR#1223         6465041111          1801
1997 MFGR#1221         6892277556          1911
1997 MFGR#1217         7053041453          1963

```

40 rows selected.

Elapsed: 00:00:00.87

SQL>

SQL> set echo off

Hit enter ...

PLAN_TABLE_OUTPUT

SQL_ID 0kfxnn5ccwcp5, child number 0

```

Select /*+ parallel(2) */          d.d_year, p.p_brand1,
sum(lo_revenue) tot_rev, count(p.p_partkey) tot_parts From
lineorder l,          date_dim d,          part p,          supplier s
Where l.lo_orderdate = d.d_datekey And l.lo_partkey =
p.p_partkey And l.lo_suppkey = s.s_suppkey And p.p_category
= 'MFGR#12' And s.s_region = 'AMERICA' AND d.d_year
= 1997 Group by d.d_year, p.p_brand1

```

Plan hash value: 1077726526

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time	TQ	IN-OUT	PQ Distrib
0	SELECT STATEMENT				3525 (100)				
1	PX COORDINATOR								
2	PX SEND QC (RANDOM)	:TQ10002	1000	73000	3525 (27)	00:00:01	Q1,02	P->S	QC (RAND)
3	HASH GROUP BY		1000	73000	3525 (27)	00:00:01	Q1,02	PCWP	
4	PX RECEIVE		1000	73000	3525 (27)	00:00:01	Q1,02	PCWP	
5	PX SEND HASH	:TQ10001	1000	73000	3525 (27)	00:00:01	Q1,01	P->P	HASH
6	HASH GROUP BY		1000	73000	3525 (27)	00:00:01	Q1,01	PCWP	
* 7	HASH JOIN		96375	6870K	3523 (27)	00:00:01	Q1,01	PCWP	
8	JOIN FILTER CREATE	:BF0000	4000	72000	2 (0)	00:00:01	Q1,01	PCWP	
9	PX RECEIVE		4000	72000	2 (0)	00:00:01	Q1,01	PCWP	
10	PX SEND BROADCAST	:TQ10000	4000	72000	2 (0)	00:00:01	Q1,00	P->P	BROADCAST
11	PX BLOCK ITERATOR		4000	72000	2 (0)	00:00:01	Q1,00	PCWC	
* 12	TABLE ACCESS INMEMORY FULL	SUPPLIER	4000	72000	2 (0)	00:00:01	Q1,00	PCWP	
* 13	HASH JOIN		484K	25M	3520 (27)	00:00:01	Q1,01	PCWP	
14	JOIN FILTER CREATE	:BF0001	365	3650	2 (0)	00:00:01	Q1,01	PCWP	
* 15	TABLE ACCESS INMEMORY FULL	DATE_DIM	365	3650	2 (0)	00:00:01	Q1,01	PCWP	
* 16	HASH JOIN		3195K	137M	3513 (27)	00:00:01	Q1,01	PCWP	
17	JOIN FILTER CREATE	:BF0002	32000	718K	47 (18)	00:00:01	Q1,01	PCWP	
* 18	TABLE ACCESS INMEMORY FULL	PART	32000	718K	47 (18)	00:00:01	Q1,01	PCWP	
19	JOIN FILTER USE	:BF0000	59M	1258M	3381 (25)	00:00:01	Q1,01	PCWP	
20	JOIN FILTER USE	:BF0001	59M	1258M	3381 (25)	00:00:01	Q1,01	PCWP	

21	JOIN FILTER USE	:BF0002	59M	1258M	3381 (25)	00:00:01	Q1,01	PCWP	
22	PX BLOCK ITERATOR		59M	1258M	3381 (25)	00:00:01	Q1,01	PCWC	
* 23	TABLE ACCESS INMEMORY FULL	LINEORDER	59M	1258M	3381 (25)	00:00:01	Q1,01	PCWP	

Predicate Information (identified by operation id):

- 7 - access("L"."LO_SUPPKEY"="S"."S_SUPPKEY")
- 12 - inmemory(:Z>=:Z AND :Z<=:Z AND "S"."S_REGION"='AMERICA')
filter("S"."S_REGION"='AMERICA')
- 13 - access("L"."LO_ORDERDATE"="D"."D_DATEKEY")
- 15 - inmemory("D"."D_YEAR"=1997)
filter("D"."D_YEAR"=1997)
- 16 - access("L"."LO_PARTKEY"="P"."P_PARTKEY")
- 18 - inmemory("P"."P_CATEGORY"='MFGR#12')
filter("P"."P_CATEGORY"='MFGR#12')
- 23 - inmemory(:Z>=:Z AND :Z<=:Z AND SYS_OP_BLOOM_FILTER_LIST(SYS_OP_BLOOM_FILTER(:BF0002,"L"."LO_PARTKEY"),SYS_OP_BLOOM_FILTER(:BF0001,"L"."LO_ORDERDATE"),SYS_OP_BLOOM_FILTER(:BF0000,"L"."LO_SUPPKEY")))
filter(SYS_OP_BLOOM_FILTER_LIST(SYS_OP_BLOOM_FILTER(:BF0002,"L"."LO_PARTKEY"),SYS_OP_BLOOM_FILTER(:BF0001,"L"."LO_ORDERDATE"),SYS_OP_BLOOM_FILTER(:BF0000,"L"."LO_SUPPKEY")))

Note

- Degree of Parallelism is 2 because of hint

62 rows selected.

Hit enter ...

NAME	VALUE
IM scan CUs columns accessed	462
IM scan CUs columns theoretical max	1964
IM scan CUs current	118
IM scan CUs memcompress for query low	118
IM scan CUs no cleanout	118
IM scan CUs pcode pred ealed	118
IM scan CUs pcode selective done	111
IM scan CUs predicates applied	7
IM scan CUs predicates received	7
IM scan CUs readlist creation accumulated time	215
IM scan CUs readlist creation number	118
IM scan CUs split pieces	210
IM scan bytes in-memory	3147114647
IM scan bytes uncompressed	13398657945
IM scan rows	61611164
IM scan rows projected	212343
IM scan rows valid	61611164
IM scan segments minmax eligible	7
IM simd compare calls	118
IM simd decode unpack calls	1088
IM simd rle burst calls	2
IM simd set membership calls	222
IM simd set membership selective calls	111
parse time cpu	11
parse time elapsed	12
redo size	9076
session logical reads	465575
session logical reads - IM	464927
session pga memory	11569336
session pga memory max	11569336

30 rows selected.

```
SQL>
SQL> @04_3join_buffer.sql
Connected.
SQL>
SQL> alter session set inmemory_query = disable;
```

Session altered.

Elapsed: 00:00:00.00

```
SQL>
SQL> -- Buffer Cache query with the column store disables via the inmemory_query parameter
SQL>
SQL> SELECT /*+ NO_PX_JOIN_FILTER parallel(2) */
  2   d.d_year, p.p_brand1, sum(lo_revenue) tot_rev, count(p.p_partkey) tot_parts
  3   From   lineorder l,
  4         date_dim d,
  5         part p,
  6         supplier s
  7   Where  l.lo_orderdate = d.d_datekey
  8   And    l.lo_partkey   = p.p_partkey
  9   And    l.lo_suppkey   = s.s_suppkey
 10   And    p.p_category   = 'MFGR#12'
 11   And    s.s_region     = 'AMERICA'
 12   AND    d.d_year       = 1997
 13   Group by d.d_year, p.p_brand1;
```

D_YEAR	P_BRAND1	TOT_REV	TOT_PARTS
1997	MFGR#1211	6852509575	1864
1997	MFGR#1225	6804217225	1848
1997	MFGR#1213	6686343443	1820
1997	MFGR#121	6409702180	1852
1997	MFGR#1215	7079477060	1888
1997	MFGR#1228	7038686432	1957
1997	MFGR#1235	7075150948	1947
1997	MFGR#1230	6596531127	1799
1997	MFGR#1231	6839363437	1866
1997	MFGR#1232	7404918843	2054
1997	MFGR#123	6280463233	1773
1997	MFGR#1224	7373166413	2027
1997	MFGR#128	6597639363	1834
1997	MFGR#122	7101221796	1959
1997	MFGR#1240	7056019394	1980
1997	MFGR#1233	6914572704	1871
1997	MFGR#125	6416763567	1821
1997	MFGR#1229	6481227038	1775
1997	MFGR#126	6213627043	1781
1997	MFGR#1219	6651946261	1794
1997	MFGR#1237	7041724061	1963
1997	MFGR#1218	6841323272	1849
1997	MFGR#1210	6795372926	1879
1997	MFGR#1236	6827320374	1938
1997	MFGR#1238	6443513085	1730
1997	MFGR#1214	6630127600	1834
1997	MFGR#1234	6695984533	1853
1997	MFGR#1227	6713851455	1814
1997	MFGR#1239	6227318243	1761
1997	MFGR#1222	6551372899	1806
1997	MFGR#127	6765391794	1902
1997	MFGR#1220	6613283998	1796
1997	MFGR#1212	6573951551	1792

1997 MFGR#129	6164317597	1713
1997 MFGR#1226	7185780867	1969
1997 MFGR#1217	7053041453	1963
1997 MFGR#1221	6892277556	1911
1997 MFGR#124	6817087386	1877
1997 MFGR#1223	6465041111	1801
1997 MFGR#1216	6877592440	1871

40 rows selected.

Elapsed: 00:00:13.60
SQL>
SQL> set echo off
Hit enter ...

PLAN_TABLE_OUTPUT

SQL_ID d224atmpv7wv9, child number 0

```

SELECT /*+ NO_PX_JOIN_FILTER parallel(2) */          d.d_year,
p.p_brand1, sum(lo_revenue) tot_rev, count(p.p_partkey) tot_parts From
  lineorder l,      date_dim d,      part p,      supplier
s Where  l.lo_orderdate = d.d_datekey And  l.lo_partkey =
p.p_partkey And  l.lo_suppkey = s.s_suppkey And  p.p_category
= 'MFGR#12' And  s.s_region = 'AMERICA' AND  d.d_year
= 1997 Group by d.d_year, p.p_brand1

```

Plan hash value: 2326395366

Id	Operation	Name	Rows	Bytes	TempSpc	Cost (%CPU)	Time
0	SELECT STATEMENT					130K (100)	
1	HASH GROUP BY		1000	73000		130K (1)	00:00:06
* 2	HASH JOIN		96375	6870K	1096K	130K (1)	00:00:06
* 3	TABLE ACCESS FULL	PART	32000	718K		1898 (1)	00:00:01
* 4	HASH JOIN		1809K	86M		123K (1)	00:00:05
* 5	TABLE ACCESS FULL	SUPPLIER	4000	72000		69 (0)	00:00:01
* 6	HASH JOIN		9103K	277M		123K (1)	00:00:05
* 7	TABLE ACCESS FULL	DATE_DIM	365	3650		15 (0)	00:00:01
8	TABLE ACCESS FULL	LINEORDER	59M	1258M		123K (1)	00:00:05

Predicate Information (identified by operation id):

- 2 - access("L"."LO_PARTKEY"="P"."P_PARTKEY")
- 3 - filter("P"."P_CATEGORY"='MFGR#12')
- 4 - access("L"."LO_SUPPKEY"="S"."S_SUPPKEY")
- 5 - filter("S"."S_REGION"='AMERICA')
- 6 - access("L"."LO_ORDERDATE"="D"."D_DATEKEY")
- 7 - filter("D"."D_YEAR"=1997)

36 rows selected.

Hit enter ...

NAME	VALUE
IM scan segments disk	4
parse time cpu	17

```

parse time elapsed          18
redo size                   3248
session logical reads      459417
session pga memory        11897016
session pga memory max    13076664

```

7 rows selected.

```

SQL>
SQL> @05_use_nojg.sql
Connected.
SQL>
SQL> -- In-Memory Column Store query - No Join Groups defined
SQL>
SQL> Select /*+ NO_VECTOR_TRANSFORM monitor */
  2      d.d_year, p.p_brand1, sum(lo_revenue) rev
  3 From    lineorder l,
  4         date_dim d,
  5         part p,
  6         supplier s
  7 Where   l.lo_orderdate = d.d_datekey
  8 And     l.lo_partkey   = p.p_partkey
  9 And     l.lo_suppkey   = s.s_suppkey
 10 And     p.p_category   = 'MFGR#12'
 11 And     s.s_region     = 'AMERICA'
 12 AND     d.d_year       = 1997
 13 Group by d.d_year, p.p_brand1;

```

D_YEAR	P_BRAND1	REV
1997	MFGR#1211	6852509575
1997	MFGR#1225	6804217225
1997	MFGR#1213	6686343443
1997	MFGR#121	6409702180
1997	MFGR#1215	7079477060
1997	MFGR#1228	7038686432
1997	MFGR#1235	7075150948
1997	MFGR#1230	6596531127
1997	MFGR#1231	6839363437
1997	MFGR#1232	7404918843
1997	MFGR#123	6280463233
1997	MFGR#1224	7373166413
1997	MFGR#128	6597639363
1997	MFGR#122	7101221796
1997	MFGR#1240	7056019394
1997	MFGR#1233	6914572704
1997	MFGR#125	6416763567
1997	MFGR#1229	6481227038
1997	MFGR#126	6213627043
1997	MFGR#1219	6651946261
1997	MFGR#1237	7041724061
1997	MFGR#1218	6841323272
1997	MFGR#1210	6795372926
1997	MFGR#1236	6827320374
1997	MFGR#1238	6443513085
1997	MFGR#1214	6630127600
1997	MFGR#1234	6695984533
1997	MFGR#1227	6713851455
1997	MFGR#1239	6227318243
1997	MFGR#1222	6551372899
1997	MFGR#127	6765391794
1997	MFGR#1220	6613283998
1997	MFGR#1212	6573951551
1997	MFGR#129	6164317597

1997 MFGR#1226	7185780867
1997 MFGR#1217	7053041453
1997 MFGR#1221	6892277556
1997 MFGR#124	6817087386
1997 MFGR#1223	6465041111
1997 MFGR#1216	6877592440

40 rows selected.

Elapsed: 00:00:01.21

SQL>

SQL> set echo off

SQL>

SQL> @06_create_jg.sql

Connected.

SQL>

SQL> -- This script will create Join Groups in the In-Memory Column Store

SQL>

SQL> alter table lineorder no inmemory;

Table altered.

SQL> alter table part no inmemory;

Table altered.

SQL> alter table supplier no inmemory;

Table altered.

SQL> alter table date_dim no inmemory;

Table altered.

SQL>

SQL> CREATE INMEMORY JOIN GROUP lineorder_jg1 (lineorder(lo_orderdate), date_dim(d_datekey));

Join group created.

SQL> CREATE INMEMORY JOIN GROUP lineorder_jg2 (lineorder(lo_partkey), part(p_partkey));

Join group created.

SQL> CREATE INMEMORY JOIN GROUP lineorder_jg3 (lineorder(lo_suppkey), supplier(s_suppkey));

Join group created.

SQL>

SQL> alter table LINEORDER inmemory;

Table altered.

SQL> alter table PART inmemory;

Table altered.

SQL> alter table SUPPLIER inmemory;

Table altered.

SQL> alter table DATE_DIM inmemory;

Table altered.

```
SQL>
SQL> select count(*) from LINEORDER;
```

```
-----
COUNT(*)
-----
59986052
```

```
SQL> select count(*) from PART;
```

```
-----
COUNT(*)
-----
800000
```

```
SQL> select count(*) from SUPPLIER;
```

```
-----
COUNT(*)
-----
20000
```

```
SQL> select count(*) from DATE_DIM;
```

```
-----
COUNT(*)
-----
2556
```

```
SQL>
```

```
SQL> set echo off
```

```
SQL>
```

```
SQL> @07_im_populated.sql
```

```
Connected.
```

```
SQL>
```

```
SQL> -- Query the view v$IM_SEGMENTS to shows what objects are in the column store
```

```
SQL> -- and how much of the objects were populated. When the BYTES_NOT_POPULATED is 0
```

```
SQL> -- it indicates the entire table was populated.
```

```
SQL>
```

```
SQL> SELECT v.owner, v.segment_name name, v.populate_status status, v.bytes_not_populated
2 FROM v$im_segments v;
```

OWNER	NAME	STATUS	BYTES_NOT_POPULATED
SSB	PART	COMPLETED	0
SSB	CUSTOMER	COMPLETED	0
SSB	LINEORDER	STARTED	3383296000
SSB	DATE_DIM	COMPLETED	0
SSB	SUPPLIER	COMPLETED	0

```
SQL>
```

```
SQL> set echo off
```

```
SQL> /
```

OWNER	NAME	STATUS	BYTES_NOT_POPULATED
SSB	PART	COMPLETED	0
SSB	CUSTOMER	COMPLETED	0
SSB	LINEORDER	COMPLETED	0
SSB	DATE_DIM	COMPLETED	0
SSB	SUPPLIER	COMPLETED	0

```
SQL> @08_query_jg.sql
```

```
Connected.
```

```
USER_JOININGROUP QUERY
```

JOININGROUP_NAME	TABLE_NAME	COLUMN_NAME	GD_ADDRESS
------------------	------------	-------------	------------

```

LINEORDER_JG1      LINEORDER      LO_ORDERDATE      00000002DDEE7950
LINEORDER_JG1      DATE_DIM        D_DATEKEY          00000002DDEE7950
LINEORDER_JG2      PART            P_PARTKEY          00000002C5E278C0
LINEORDER_JG2      LINEORDER      LO_PARTKEY         00000002C5E278C0
LINEORDER_JG3      SUPPLIER        S_SUPPKEY          00000002DDEE37950
LINEORDER_JG3      LINEORDER      LO_SUPPKEY         00000002DDEE37950

```

6 rows selected.

```

SQL>
SQL> @09_use_jg.sql
Connected.
SQL>
SQL> -- In-Memory Column Store query - With Join Groups
SQL>
SQL> Select /*+ NO_VECTOR_TRANSFORM monitor */
  2   d.d_year, p.p_brand1, sum(lo_revenue) rev
  3 From   lineorder l,
  4        date_dim d,
  5        part p,
  6        supplier s
  7 Where  l.lo_orderdate = d.d_datekey
  8 And    l.lo_partkey   = p.p_partkey
  9 And    l.lo_suppkey   = s.s_suppkey
 10 And    p.p_category   = 'MFGR#12'
 11 And    s.s_region     = 'AMERICA'
 12 AND    d.d_year       = 1997
 13 Group by d.d_year, p.p_brand1;

```

D_YEAR	P_BRAND1	REV
1997	MFGR#1211	6852509575
1997	MFGR#1225	6804217225
1997	MFGR#1213	6686343443
1997	MFGR#121	6409702180
1997	MFGR#1215	7079477060
1997	MFGR#1228	7038686432
1997	MFGR#1235	7075150948
1997	MFGR#1230	6596531127
1997	MFGR#1231	6839363437
1997	MFGR#1232	7404918843
1997	MFGR#123	6280463233
1997	MFGR#1224	7373166413
1997	MFGR#128	6597639363
1997	MFGR#122	7101221796
1997	MFGR#1240	7056019394
1997	MFGR#1233	6914572704
1997	MFGR#125	6416763567
1997	MFGR#1229	6481227038
1997	MFGR#126	6213627043
1997	MFGR#1219	6651946261
1997	MFGR#1237	7041724061
1997	MFGR#1218	6841323272
1997	MFGR#1210	6795372926
1997	MFGR#1236	6827320374
1997	MFGR#1238	6443513085
1997	MFGR#1214	6630127600
1997	MFGR#1234	6695984533
1997	MFGR#1227	6713851455
1997	MFGR#1239	6227318243
1997	MFGR#1222	6551372899
1997	MFGR#127	6765391794
1997	MFGR#1220	6613283998
1997	MFGR#1212	6573951551

1997 MFGR#129	6164317597
1997 MFGR#1226	7185780867
1997 MFGR#1217	7053041453
1997 MFGR#1221	6892277556
1997 MFGR#124	6817087386
1997 MFGR#1223	6465041111
1997 MFGR#1216	6877592440

40 rows selected.

Elapsed: 00:00:01.14

SQL>

SQL> set echo off

Hit enter ...

```
<html>
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
  <script language="javascript" type="text/javascript">
    <!--
      var version = "12.2.0.1.0";
      var swf_base_path = "http://download.oracle.com/otn_software/";

      document.write('<script language="javascript" type="text/javascript" ' +
        'src="' + swf_base_path + 'emviewers/scripts/activeReportInit.js?' +
        Math.floor((new Date()).getTime()/(7*24*60*60*1000)) +
        '></' + 'script>');

    -->
  </script>
</head>
<body onload="sendXML();">
  <script type="text/javascript">
    writeIframe();
  </script>
  <script id="fxtmodel" type="text/xml">
    <!--FXTMODEL-->
    <report db_version="12.2.0.1.0" elapsed_time="1.24" cpu_time="1.07" cpu_cores="2" hyperthread="N" con_id="3" con_name="ORCL" timezone_offset="-18000" packs="2"
    encode="base64" compress="zlib">
      <report_id><![CDATA[orarep/sqlmonitor/main]]></report_id>
      eAhtXftz28iR/t1/xRzuKrH3bIvGmxuJtZREaZV0o06kd+NLPVAQCUPYgWQNgJav
      vz5fzwMYkAABSnaiTanKVSiwPe9+fNpDAx+Gnz1r4S/dyA+swFn50cS+OEHo+ssj
      o/6+YrDwIzzZkXNkVMwDs3lQrZhtvJv/bNR/ rNaM7ivGDku1a2UH9skJUJne4v3N
      evrJiaypv15G3drhQepZkLhL0H+xPSuM7CDqbdRPTzYoNmo6CxnGdXqWjWioGEE
      ZI7zEB3f0hFzLxjCcupY7uzIMDFtJ6RFEI+15Dl0Atf2joxm3ezQ6mD5qEb9U6NT
      n94v7uzPv30Wr52vzLTMK2vtqhoR77LZarWqZku8Xnn20rqzw7sjo1qrd1qdVq1D
      g0Bv87XnWVp5rVJvtuuteqNhsNmNtV66n9e0tCRWHBl+MPXM6pQXoEo0940FLBq4
      y/VX9rXdZM36uxs3Mphcnjs/jCTNl5tbZ4n5Tvk+YwnXmd2m2zVr2Bf1IDaEnrrj
      8bF4LzlggyheQ9l7+ErT0QAPsDkcnA1HGh0XpKvBvwUldTHblrc0f4LgW150rMfvf
      fPdL7c3hgSITLRb+b0053fH/DX64RqXDA/LC LGJsX9yp06X10DxQT7JMrnkfAUX
      hX9j4yPRnfseM40AiMf/PC/7Gpo/dI/mQxH1mTUuxqfDUeXTIoN+GAt008z2bv
      Z9aDYwdv2er9yroJ70XMfmvC9eK1500+vjjLtfMGS/3l1VngL3gVz106fjBzAua9
      1RqCzFkzd8Fm2kuIWMR2otwvP5lqqGr369cwKHGvTeoyveIMkt0+Jjop+fnIdX
      veMdyqqQD28xZsjPL75vEFFnSiq8H1oyeeEiqY6RQe3fkBtUwt/sFd+KfLs/PR
      f5tV8ZDU0y0BcwwxSLH3Lvuj i50eor465aXxkkpas9NpvToP/PWK3TzEhfp6Y4e1
      PZWsC0m01mH3dHjVZ697gwEbDX8dg43ke8ELgTMPnPb0qqwWqQz9hSCauwEkRjVE
      Lvg4U29l0IkGPLu4PkRmm0xUn60D08Ladc3Dg/i3KPJXGI77Dwips/wiJg6p5YqZ
      CY1gTYP3C55c0nZc0RvUfKfkrA0hCJzp0rDmnn0bGt3KLmJqxvMcz7LXKT/zV2XJ
      ZzOyQzPH6DpL+8ZzZqU6Qa3Zwivdh3Mb00ih3ATQsBzRzA3LD4nU/pr2B3vAaxnd
      KfG7paazsL9aMzLlImLD8LYXGbo+eECdub32ImuP9RYVY6/4rW3Zt1i+w9JLwmYa
      3Wq13q5U2F+g9fPHGvkrDPx0tRZihVop6s0DLA49PBADeDcekoCGLHpYz5ZjjaVM
      ERVJvnY8exU6M4vEEexqderJlaQKDQFIILpLIqw0mw0Q/ZhH4EpWrd224k6Wut
      Zi03Ww4Q5040lQrE60a53qznc7QM4AF5akLYVf/iF0dQh1xW3ejBCu3FChInJ6Je
      syk0RRHknKzWBIPLEh0Z7pItnAXUMBdpRcnX8UA96c0lfcycyHY9hneEYRi5oAvo
      iq9FvNhdYELpIQHvouPKrZhm6TcByfgyQhCzJmjfSUYZ9Y6rtiMDsEkSJY3hneBH
```


aHsgLIPpPKre3YHlgugucGzguyvFc6IttlwvbyAwe7AkZV67t7XxiZewy9urVD
aEJC5139ElvDxxLuEx0e+hgdtw0SqcF9AQAKG096k/5L/2pimIKVgKgZxUBR+IX
TDJmW21X0smop34Ydfk7AmL4LSwJhFI2n9Pdz73xzwYj0fWXaPV8NPxwzY4/1l6B
n2Ww+MV7NBuE7QAIsLKB1E3pJ+8QK795iJyw26rxEvGgriIH1TquD6VC6g0mwIg
fvkY1+NBK+WtRtU009VqvdNBR+qtIhM2vPJjBf9M7AJ/VoWfbzx/+qmL5f0fWEn5
pAqFgvI68N0saEzLuS/1FFC4PEXht6vvq+8r79EGEWTTQ4mGjhV075yFbXQP/+tv
J6e9Se9vBnc18fe/d3dWVScGDMcNbnIsKFULWjk+Z5SQYIGBd/ThryNdAu4007m
itW5g6gmM7s4vxq0+tbwenJxafUvj/unp/1T6+eLq8mYz5dTy8Um/LdRnde7+P/+
yDrr9yYrFv2x1b/qHQ/6r/+oLvuPb8o0dHoMeD8aXwyy9q0JKGkRlCzTy/DDZHBx
1bcG/d72658MzLJGqfGdFRgM4hrMODWeUnvwpNrrJ9W+3q82Vur04uo8c+rMSCbC
jGRYZNizkw/jvkXqT0slabnU9mS0kIznsS3s0Yvrv1p/HL5cWwCXg0L/9P155Laz
54zGv/auxYgurrq4/TMba1L5dS3uukdonq3d+Puqf9yYk8LteCeA005HhLQeGsd4n
DFtsLhntjzCQgBz5Q0JXpoHsNgstWLQsC1mpNXhZ2kZGzmJlhSsbbg7TrAorqr1T
Fis2pU3d3u9lSoGu041atUVunb1M6SpwZi45ChSet6dTuNmM7h8+r/3oTwpX5718
GlrXvdHkL/2P4vLI/L0EV2naXCwiLsQVqd4ZyZkL1jv5KQ/HmuA5uLqsn85HH1k
pHrFpgE1yk3Dr81N829+g7eoS0NGn+IhXnOC0bVqPurP5mGeNk4QCSrTdqLFOCIP
7dTbDRx6hxfhcsN2Lz7E0DNc6yR2qBt8/2hmA0yR5JmDR3PUB6xQbfqJfduI1v
kgV7rh12r3+S4E5sm3zLN6+YfchwaIKNI9AGjwAnpfjABPw2GwRdt8Fvp1JvtDoN
I000bHMMWFu1BmmEGKR7GA2qnXUyuWIFrzVzY7ZM2twan1foR1/uL7eFNpxSmJh
Vopmm0ekV500KROWTtpZkxFOYF8IK3nDxcEHW5Ijrd8en0HwsG5Z8grxyTmhCEH0
2CFSiFm7g/e509PpNBpocK9dKV6f8kqtGa8TVixnnWi7BhF90e9uobKUTdrmbHm
UAEwQUqhpdzS28ya1UdaoX2TPsqrs/E3VGcIR21I0aY665iVWR0TlSnVjlmDumuj
NENaoM+qe5AE+qwGTZUrNajaatQ69c7312fD0wL/hGN4XzEFxyqnKY12ahFBRPw2
maRYos1WjveI3xJFRoN2CBL0XIRZJhdFA+Z4dy9Nku7GFw0/U2eVT3sDLFq1Ne
n3XiVYJmy9FntFnW6cXL72yZsjRNWpttcuHXfm8k2ZbCZtt8mdVkwk9psnyuor0
KboqW2RwxpN0yMT3qQRGU155QHZCGEanA1ZuIF6RITVQbYBUNBM0iJ6p3UyKFGT1
AdJu0Hy52q3ZqrYIw5u/dBasRwLVlniuM2Xnp065j9vycqrHvj0lbW5e3zRPeQa
5CYlW/n83Tyv54EonTT+0Izn1zoedIeX0i1LDS7Gk9cZBa8LV706mrLfbBhhd+8
zW+ktgd66T7hqeQKH9EFTLPZtm9PfsJ1NeDw9K62HYEBmD0URJSC0JhKlXKqIm
Ar08PkevoecC5z0CRFDI8pkrz0TF4vNGFQeEkdCw0/AgRAzpyXaEE45yPeL30uKt
e+NUDEwkP2yHyCS4FbExL2JZvMPw04r+xLx80Vnls9JnHVNADThGhuJEEJSvBmUb
b0fNJfFekeKiV+ar0UfWVRFOHlTUIExTLuZIPUJqA89CmSJuzDMIcshpmZOMtoI0
0FZAKw0Axfxmyaa657hgIkSryXuhk2pxcKztDveqTndD5AwT4a42Eyx3bkhSrrr
IjYpdx0Pm7se5yUoPsBIteFfMUW5Dy10IiKsYCY1Qk6CrZgrj2/FqHlXbBJtKPCQ
HfKEAQ9iYgnSswck10KNNZVF8cLb4NsCtn06b2NzInUL5JGjHM2a0mzrRPEAUk
JySiAKZOREF6KSXzi7jD45lfI0/kcn9moELn/43jPxpUpf3WrsSPMAMji/kc8JB0J
USyP2i/C4TwtXY/EZUom1nU/9idHPjvTqfaxLFKJ0+ZCtQN7kMqDtkJYlKk6Dt
PXLU7a20BN6htCVCZI7XrjdjY2RTEgoPp0cG/Wb+nCH2h5QikNwXqzWljimTFUET
pBhXLdVKG6JMSQJAB0nHalrUC6RP72Xk37MTSpNXV3x9CLqLPDvQ4a87aRX1U0j
r3FEp0TjZ/byHeKX241CRURc9E0GpLUZ0p9YidziIn0AU4LZxbPJ7QbTFN2MPT/K
XCh7ibbeCeXGKJv93W8+FisEfeEkcGQXrU8ofZCJ3bi0x63mJEpFNhYtL9iXZH6q
n9ydwDLX7+ferVLNj03fEf0c0bGCMu6Ng01zGLLub9zLbeEwcGAUw7hEHqf7boXM
vmQQ+Iwc47wpx50F5gCdULRVcxieEbdV2D+ci/oyjFcuKnixDhi+w8Ceub0L/qRc
dhILmq4DXhk1wVZInF3EXjtfp94aY08W+00Jj/j1pnBI80SJI2343nqxtAPWX059
aiVKA+Q7BvatM8tbFCcm9RQpLyK5HwxImYYjxrsCb0LKhftMcooZA8uMAQ1v6FZB
mfH4krLsEHCvYUthCf0o1BVy32pm09wADWlL04Ubvvp2jZakVZDG200pVr19Hchic
7eQH0/WuTtLY2UBzZymSETf0Cc17Lzpcam8i/A+fByFQ+d/WreU8NxIPURT+uAMP
8Yix5BHesn748+9xT0ZcfxE/ZcgXa0Q8LNMCC+E8VDN500RedNrT8gVzAVctIJMAQ
+HljM5WA3iuwtSNhAGQKamXkHKD0BU09FyivIEQlsFGSwYZN3FY1eLmNjWdx75Fo
LnCUaj9pkAQjr5FsqQraHnaJhbfvubcu7oGkwxfmnqMt5igKQkf+f075NsxBgRKH
mAujcsZzTmDijHQto9fbD5xohSxByc+RnVK6hwQWbDQhknI70EVI+veJ2sG+DBz
vhb0IYZGZ/AV3bEpQzB8Tv1FBIyCnD6E51WNwm5iZMTBEDCYVpnhXpuw7bnLD8/v
o10M2b0NTELADefeCHljec0IWC8cVQ6UTx/PQMtD+XbBm6CljLNMCMG12ieHQMSV/
57TNE801s0i2t/GM2DnFBfwEsc2GqphNFQzi1QkXYErp9geXfKXlMA8okdfgoZ
bt5N+M0GhZ+SA8dMUodEHXhqqU/3TyL0yoB5dCMis/ttqLkVjYfwZFLyLckqy1Xd
aeztu90zsPLsaU46fLQit6gZCVlUpIzqzpwreCqzmp27BYIXy/pcLkVzbjbnPIgt
7EnajVdr1SguRMPLeqY4Ithg9zWr2+20cdv1w0UkyCuXTCr0/B5IcA34Y6mPB19
xHoP/16AT/H7B0DHFjY7mq2yAKWmIjVSD0wTN82hxpIfxzh2cfi/geXjliEnwfk
vTBP+70y8gyxjVz1bGQa0muLdnITCkGkVM52QIIX9TNC1E327Jc0nkkuS3Y54XN
E7bGS02jSBB87EYLe5Xpu+N0TLG8C7rheD9zPdmQAsZzcuSUHELzvfS0zlhvBo
PmRjJ9rGMdUeVkevgi/Ke9CYZbcw0esWMQx4ARAB8B8D/fXVNYIXVxkLzqKxDBe
9Ea3h/foip0X6ifG8aKf+0BT7NVLjkh1FT2fjhaIwb9odjFkn+BGds19v2y+Xk5x
XCHP4vaeKfrR6dwwJs0A1tt0M83gXbNZI49Hnj2DtBRQwGTWceKbWjUsxSb3c91
RWlwiWVLWKLUPRRp4c96FLu4x7qukuTVf3tnvATIv0uafAu7LrB3sRvV8g5TL14i

```

YrG03z5TppDiI86UqVT4PFSXlxMP9aU0LZtZ8V5k0N30xHc0KoSxNUPgheU91xQ
XrXSbrXqyPDNs1EZ0qzjbthG2e8EnLyLEMLvzGQNBxS0aPnfaaEi09TCzFCKXWm
LGdxE/Bd8kyJMNiTBGx8wSNP+Wtc9Ij1TjpXXdc5uVc6dH2zlc7+omvoI2HPJMtm
S5BLKl7UgVl703bb+JJKttYSqS0FBADWNf2TR3p+KjUAXJ0x3P1gNXztmDH3vy9
IsIEzb8frE7uIL2IPH26Jv1pL54Ly31OGMKXj0q9M6ozZIgwVH6cFdpZS5WJ1ca
U1NwSR0wAuvJr2RlHVazQHbq0wkY5r/GHZu+d5NnP3PvfGGcEr1v3fqI0Xed17s
AqG03rcviIHgBb0/F4uaid4fbVURXkk4MA9wI42q3qziA8y1jUDNvLFwk24+JEfi
JFKCDX518S1nFP/lyDdZPP2CRhK44ZkB+GxxzpXNF/Hln4Xd+67S7/e+0b9HfBuV
drvVr0LjL7qM7y2+MJU6+Ma3PPaCt+Qw+37wVrvm+1R8+yLYkbhW+c0vEuYERwTe
TgQ756QnYfnzu0j4WMG0oS9sQXyPVX54t+CruPAbF3wVVy0nvxIrL6tCat3UvdeN
G64CNKorsB13XTc+YZv5uVp+jLLT2f1hWkrCVpTKfidv9I8GY4XSFzLFnWB1h0u+
gnrDbVF5ooMThP5Xie6rfwL50Zcg
  
```

```

    </report>
  <!--FXTMODEL-->
</script>
</body>
</html>
  
```

```

SQL>
SQL> @10_vgb_im.sql
Connected.
SQL> set numwidth 20
SQL>
SQL> -- In-Memory Column Store query with In-Memory Aggregation enabled
SQL>
SQL> select d.d_year, c.c_nation, sum(lo_revenue - lo_supplycost) profit
  2      From      LINEORDER l, DATE_DIM d, PART p, SUPPLIER s, CUSTOMER C
  3      Where     l.lo_orderdate = d.d_datekey
  4      And       l.lo_partkey   = p.p_partkey
  5      And       l.lo_suppkey   = s.s_suppkey
  6      And       l.lo_custkey   = c.c_custkey
  7      And       s.s_region     = 'AMERICA'
  8      And       c.c_region     = 'AMERICA'
  9      Group by  d.d_year, c.c_nation
 10      Order by d.d_year, c.c_nation;
  
```

D_YEAR	C_NATION	PROFIT
1992	ARGENTINA	257114232147
1992	BRAZIL	265087838628
1992	CANADA	267029429200
1992	PERU	260091746389
1992	UNITED STATES	264098681570
1993	ARGENTINA	260475234675
1993	BRAZIL	264646147899
1993	CANADA	265113570471
1993	PERU	262628508817
1993	UNITED STATES	260411002935
1994	ARGENTINA	261149015641
1994	BRAZIL	263808033983
1994	CANADA	264598150413
1994	PERU	258595600981
1994	UNITED STATES	265282504206
1995	ARGENTINA	258498976118
1995	BRAZIL	269135848643
1995	CANADA	264654265482
1995	PERU	257451709833
1995	UNITED STATES	259660457396
1996	ARGENTINA	259361903850
1996	BRAZIL	265970119048
1996	CANADA	265333193889
1996	PERU	260916013039

```

1996 UNITED STATES      262339293224
1997 ARGENTINA          261099548066
1997 BRAZIL             266353055971
1997 CANADA             265036379243
1997 PERU               259114682243
1997 UNITED STATES     262208356128
1998 ARGENTINA          151054449013
1998 BRAZIL             153632348378
1998 CANADA             156899052279
1998 PERU               152297126350
1998 UNITED STATES     153937088695

```

35 rows selected.

Elapsed: 00:00:01.66

SQL>

SQL> set echo off

Hit enter ...

PLAN_TABLE_OUTPUT

SQL_ID czrxzt1nt4vpg, child number 0

```

select d.d_year, c.c_nation, sum(lo_revenue - lo_supplycost) profit
  From   LINEORDER l, DATE_DIM d, PART p, SUPPLIER s, CUSTOMER c
  Where  l.lo_orderdate = d.d_datekey          And
        l.lo_partkey   = p.p_partkey          And   l.lo_suppkey
        = s.s_suppkey   And   l.lo_custkey    = c.c_custkey
        And            s.s_region            = 'AMERICA' And
        c.c_region     = 'AMERICA'          Group by d.d_year,
        c.c_nation      Order by d.d_year, c.c_nation

```

Plan hash value: 2858170253

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT				17618 (100)	
1	TEMP TABLE TRANSFORMATION					
2	LOAD AS SELECT (CURSOR DURATION MEMORY)	SYS_TEMP_0FD9D6659_353EAF				
3	HASH GROUP BY		7	98	2 (50)	00:00:01
4	KEY VECTOR CREATE BUFFERED	:KV0000	7	98	1 (0)	00:00:01
5	TABLE ACCESS INMEMORY FULL	DATE_DIM	2556	25560	1 (0)	00:00:01
6	LOAD AS SELECT (CURSOR DURATION MEMORY)	SYS_TEMP_0FD9D665A_353EAF				
7	HASH GROUP BY		1	9	96 (27)	00:00:01
8	KEY VECTOR CREATE BUFFERED	:KV0001	1	9	75 (6)	00:00:01
9	TABLE ACCESS INMEMORY FULL	PART	800K	3906K	74 (5)	00:00:01
10	LOAD AS SELECT (CURSOR DURATION MEMORY)	SYS_TEMP_0FD9D665B_353EAF				
11	HASH GROUP BY		1	22	4 (25)	00:00:01
12	KEY VECTOR CREATE BUFFERED	:KV0002	1	22	3 (0)	00:00:01
* 13	TABLE ACCESS INMEMORY FULL	SUPPLIER	4000	72000	3 (0)	00:00:01
14	LOAD AS SELECT (CURSOR DURATION MEMORY)	SYS_TEMP_0FD9D665C_353EAF				
15	HASH GROUP BY		25	950	39 (21)	00:00:01
16	KEY VECTOR CREATE BUFFERED	:KV0003	25	950	36 (14)	00:00:01
* 17	TABLE ACCESS INMEMORY FULL	CUSTOMER	60000	1992K	36 (14)	00:00:01
18	SORT GROUP BY		62	9920	17477 (13)	00:00:01
* 19	HASH JOIN		62	9920	17476 (13)	00:00:01
20	TABLE ACCESS FULL	SYS_TEMP_0FD9D665C_353EAF	25	850	2 (0)	00:00:01
* 21	HASH JOIN		62	7812	17474 (13)	00:00:01
22	MERGE JOIN CARTESIAN		7	315	6 (0)	00:00:01
23	MERGE JOIN CARTESIAN		1	31	4 (0)	00:00:01
24	TABLE ACCESS FULL	SYS_TEMP_0FD9D665A_353EAF	1	9	2 (0)	00:00:01
25	BUFFER SORT		1	22	2 (0)	00:00:01

26	TABLE ACCESS FULL	SYS_TEMP_0FD9D665B_353EAF	1	22	2	(0)	00:00:01
27	BUFFER SORT		7	98	4	(0)	00:00:01
28	TABLE ACCESS FULL	SYS_TEMP_0FD9D6659_353EAF	7	98	2	(0)	00:00:01
29	VIEW	VW_VT_80F21617	62	5022	17468	(13)	00:00:01
30	VECTOR GROUP BY		62	2976	17468	(13)	00:00:01
31	HASH GROUP BY		62	2976	17468	(13)	00:00:01
32	KEY VECTOR USE	:KV0000	3566K	163M	17466	(13)	00:00:01
33	KEY VECTOR USE	:KV0001	3566K	149M	17466	(13)	00:00:01
34	KEY VECTOR USE	:KV0003	3566K	136M	17466	(13)	00:00:01
35	KEY VECTOR USE	:KV0002	11M	409M	17466	(13)	00:00:01
* 36	TABLE ACCESS INMEMORY FULL	LINEORDER	59M	1830M	17465	(13)	00:00:01

Predicate Information (identified by operation id):

```

13 - inmemory("S"."S_REGION"='AMERICA')
    filter("S"."S_REGION"='AMERICA')
17 - inmemory("C"."C_REGION"='AMERICA')
    filter("C"."C_REGION"='AMERICA')
19 - access("ITEM_16"=INTERNAL_FUNCTION("C0") AND "ITEM_17"="C2")
21 - access("ITEM_12"=INTERNAL_FUNCTION("C0") AND "ITEM_13"="C2" AND "ITEM_14"=INTERNAL_FUNCTION("C0") AND
    "ITEM_15"="C2" AND "ITEM_18"=INTERNAL_FUNCTION("C0") AND "ITEM_19"="C2")
36 - inmemory((SYS_OP_KEY_VECTOR_FILTER("L"."LO_SUPPKEY",:KV0002) AND
    SYS_OP_KEY_VECTOR_FILTER("L"."LO_CUSTKEY",:KV0003) AND SYS_OP_KEY_VECTOR_FILTER("L"."LO_PARTKEY",:KV0001) AND
    SYS_OP_KEY_VECTOR_FILTER("L"."LO_ORDERDATE",:KV0000)))
    filter((SYS_OP_KEY_VECTOR_FILTER("L"."LO_SUPPKEY",:KV0002) AND
    SYS_OP_KEY_VECTOR_FILTER("L"."LO_CUSTKEY",:KV0003) AND SYS_OP_KEY_VECTOR_FILTER("L"."LO_PARTKEY",:KV0001) AND
    SYS_OP_KEY_VECTOR_FILTER("L"."LO_ORDERDATE",:KV0000)))

```

Note

- vector transformation used for this statement

76 rows selected.

```

SQL>
SQL> @11_novgb_im.sql
Connected.
SQL> set numwidth 20
SQL>
SQL> -- In-Memory Column Store query with In-Memory Aggregation disabled
SQL>
SQL> select /*+ NO_VECTOR_TRANSFORM */
2      d.d_year, c.c_nation, sum(lo_revenue - lo_supplycost) profit
3      From    LINEORDER l, DATE_DIM d, PART p, SUPPLIER s, CUSTOMER C
4      Where   l.lo_orderdate = d.d_datekey
5      And    l.lo_partkey   = p.p_partkey
6      And    l.lo_suppkey   = s.s_suppkey
7      And    l.lo_custkey   = c.c_custkey
8      And    s.s_region     = 'AMERICA'
9      And    c.c_region     = 'AMERICA'
10     Group by d.d_year, c.c_nation
11     Order by d.d_year, c.c_nation;

```

D_YEAR	C_NATION	PROFIT
1992	ARGENTINA	257114232147
1992	BRAZIL	265087838628
1992	CANADA	267029429200
1992	PERU	260091746389
1992	UNITED STATES	264098681570
1993	ARGENTINA	260475234675

```

1993 BRAZIL                264646147899
1993 CANADA                265113570471
1993 PERU                  262628508817
1993 UNITED STATES        260411002935
1994 ARGENTINA            261149015641
1994 BRAZIL                263808033983
1994 CANADA                264598150413
1994 PERU                  258595600981
1994 UNITED STATES        265282504206
1995 ARGENTINA            258498976118
1995 BRAZIL                269135848643
1995 CANADA                264654265482
1995 PERU                  257451709833
1995 UNITED STATES        259660457396
1996 ARGENTINA            259361903850
1996 BRAZIL                265970119048
1996 CANADA                265333193889
1996 PERU                  260916013039
1996 UNITED STATES        262339293224
1997 ARGENTINA            261099548066
1997 BRAZIL                266353055971
1997 CANADA                265036379243
1997 PERU                  259114682243
1997 UNITED STATES        262208356128
1998 ARGENTINA            151054449013
1998 BRAZIL                153632348378
1998 CANADA                156899052279
1998 PERU                  152297126350
1998 UNITED STATES        153937088695

```

35 rows selected.

Elapsed: 00:00:10.57
SQL>
SQL> set echo off
Hit enter ...

PLAN_TABLE_OUTPUT

SQL_ID 80sjuhrs2yq2, child number 0

```

select /*+ NO_VECTOR_TRANSFORM */      d.d_year, c.c_nation,
sum(lo_revenue - lo_supplycost) profit  From    LINEORDER
l, DATE_DIM d, PART p, SUPPLIER s, CUSTOMER C      Where
l.lo_orderdate = d.d_datekey              And    l.lo_partkey   =
p.p_partkey                                And    l.lo_suppkey   = s.s_suppkey
      And    l.lo_custkey   = c.c_custkey       And
s.s_region = 'AMERICA'                    And    c.c_region
      = 'AMERICA'                          Group by d.d_year, c.c_nation
Order by d.d_year, c.c_nation

```

Plan hash value: 2501995747

Id	Operation	Name	Rows	Bytes	TempSpc	Cost (%CPU)	Time
0	SELECT STATEMENT					69847 (100)	
1	SORT GROUP BY		124	12276		69847 (4)	00:00:03
* 2	HASH JOIN		3566K	336M		69746 (4)	00:00:03
3	TABLE ACCESS INMEMORY FULL	DATE_DIM	2556	25560		1 (0)	00:00:01
* 4	HASH JOIN		3566K	302M	12M	69735 (4)	00:00:03
5	TABLE ACCESS INMEMORY FULL	PART	800K	3906K		74 (5)	00:00:01
* 6	HASH JOIN		3566K	285M	2696K	52810 (5)	00:00:03

* 7	TABLE ACCESS INMEMORY FULL	CUSTOMER	60000	1992K		36	(14)	00:00:01
* 8	HASH JOIN		11M	568M		17639	(14)	00:00:01
9	JOIN FILTER CREATE	:BF0000	4000	72000		3	(0)	00:00:01
* 10	TABLE ACCESS INMEMORY FULL	SUPPLIER	4000	72000		3	(0)	00:00:01
11	JOIN FILTER USE	:BF0000	59M	1830M		17465	(13)	00:00:01
* 12	TABLE ACCESS INMEMORY FULL	LINEORDER	59M	1830M		17465	(13)	00:00:01

Predicate Information (identified by operation id):

- 2 - access("L"."LO_ORDERDATE"="D"."D_DATEKEY")
- 4 - access("L"."LO_PARTKEY"="P"."P_PARTKEY")
- 6 - access("L"."LO_CUSTKEY"="C"."C_CUSTKEY")
- 7 - inmemory("C"."C_REGION"='AMERICA')
filter("C"."C_REGION"='AMERICA')
- 8 - access("L"."LO_SUPPKEY"="S"."S_SUPPKEY")
- 10 - inmemory("S"."S_REGION"='AMERICA')
filter("S"."S_REGION"='AMERICA')
- 12 - inmemory(SYS_OP_BLOOM_FILTER(:BF0000,"L"."LO_SUPPKEY"))
filter(SYS_OP_BLOOM_FILTER(:BF0000,"L"."LO_SUPPKEY"))

46 rows selected.

SQL>

SQL> @i2_vgb_buffer.sql

Connected.

SQL>

SQL> alter session set inmemory_query = disable;

Session altered.

Elapsed: 00:00:00.00

SQL>

SQL> -- Buffer Cache query with the column store disabled via the inmemory_query parameter

SQL>

SQL> SELECT

```

2      d.d_year, c.c_nation, sum(lo_revenue - lo_supplycost) profit
3  From    LINEORDER l, DATE_DIM d, PART p, SUPPLIER s, CUSTOMER C
4  Where   l.lo_orderdate = d.d_datekey
5  And     l.lo_partkey    = p.p_partkey
6  And     l.lo_suppkey    = s.s_suppkey
7  And     l.lo_custkey    = c.c_custkey
8  And     s.s_region      = 'AMERICA'
9  And     c.c_region      = 'AMERICA'
10 Group by d.d_year, c.c_nation
11 Order by d.d_year, c.c_nation;
```

D_YEAR	C_NATION	PROFIT
1992	ARGENTINA	257114232147
1992	BRAZIL	265087838628
1992	CANADA	267029429200
1992	PERU	260091746389
1992	UNITED STATES	264098681570
1993	ARGENTINA	260475234675
1993	BRAZIL	264646147899
1993	CANADA	265113570471
1993	PERU	262628508817
1993	UNITED STATES	260411002935
1994	ARGENTINA	261149015641
1994	BRAZIL	263808033983
1994	CANADA	264598150413

```

1994 PERU 258595600981
1994 UNITED STATES 265282504206
1995 ARGENTINA 258498976118
1995 BRAZIL 269135848643
1995 CANADA 264654265482
1995 PERU 257451709833
1995 UNITED STATES 259660457396
1996 ARGENTINA 259361903850
1996 BRAZIL 265970119048
1996 CANADA 265333193889
1996 PERU 260916013039
1996 UNITED STATES 262339293224
1997 ARGENTINA 261099548066
1997 BRAZIL 266353055971
1997 CANADA 265036379243
1997 PERU 259114682243
1997 UNITED STATES 262208356128
1998 ARGENTINA 151054449013
1998 BRAZIL 153632348378
1998 CANADA 156899052279
1998 PERU 152297126350
1998 UNITED STATES 153937088695

```

35 rows selected.

```

Elapsed: 00:00:10.60
SQL>
SQL> set echo off
Hit enter ...

```

PLAN_TABLE_OUTPUT

SQL_ID a0x06yzufd67k, child number 0

```

SELECT          d.d_year, c.c_nation, sum(lo_revenue - lo_supplycost)
profit          From    LINEORDER l, DATE_DIM d, PART p, SUPPLIER s,
CUSTOMER C      Where    l.lo_orderdate = d.d_datekey
And             l.lo_partkey = p.p_partkey          And
l.lo_suppkey   = s.s_suppkey          And          l.lo_custkey
= c.c_custkey  And             s.s_region = 'AMERICA'
              And             c.c_region = 'AMERICA'  Group by
d.d_year, c.c_nation          Order by d.d_year, c.c_nation

```

Plan hash value: 2858170253

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT				126K(100)	
1	TEMP TABLE TRANSFORMATION					
2	LOAD AS SELECT (CURSOR DURATION MEMORY)	SYS_TEMP_0FD9D665D_353EAF				
3	HASH GROUP BY		7	98	16 (7)	00:00:01
4	KEY VECTOR CREATE BUFFERED	:KV0000	7	98	15 (0)	00:00:01
5	TABLE ACCESS FULL	DATE_DIM	2556	25560	15 (0)	00:00:01
6	LOAD AS SELECT (CURSOR DURATION MEMORY)	SYS_TEMP_0FD9D665E_353EAF				
7	HASH GROUP BY		1	9	1913 (2)	00:00:01
8	KEY VECTOR CREATE BUFFERED	:KV0001	1	9	1892 (1)	00:00:01
9	TABLE ACCESS FULL	PART	800K	3906K	1891 (1)	00:00:01
10	LOAD AS SELECT (CURSOR DURATION MEMORY)	SYS_TEMP_0FD9D665F_353EAF				
11	HASH GROUP BY		1	22	70 (2)	00:00:01
12	KEY VECTOR CREATE BUFFERED	:KV0002	1	22	69 (0)	00:00:01
* 13	TABLE ACCESS FULL	SUPPLIER	4000	72000	69 (0)	00:00:01
14	LOAD AS SELECT (CURSOR DURATION MEMORY)	SYS_TEMP_0FD9D6660_353EAF				

15	HASH GROUP BY		25	950	840	(1)	00:00:01
16	KEY VECTOR CREATE BUFFERED	:KV0003	25	950	838	(1)	00:00:01
* 17	TABLE ACCESS FULL	CUSTOMER	60000	1992K	838	(1)	00:00:01
18	SORT GROUP BY		62	9920	123K	(1)	00:00:05
* 19	HASH JOIN		62	9920	123K	(1)	00:00:05
20	TABLE ACCESS FULL	SYS_TEMP_0FD9D6660_353EAF	25	850	2	(0)	00:00:01
* 21	HASH JOIN		62	7812	123K	(1)	00:00:05
22	MERGE JOIN CARTESIAN		7	315	6	(0)	00:00:01
23	MERGE JOIN CARTESIAN		1	31	4	(0)	00:00:01
24	TABLE ACCESS FULL	SYS_TEMP_0FD9D665E_353EAF	1	9	2	(0)	00:00:01
25	BUFFER SORT		1	22	2	(0)	00:00:01
26	TABLE ACCESS FULL	SYS_TEMP_0FD9D665F_353EAF	1	22	2	(0)	00:00:01
27	BUFFER SORT		7	98	4	(0)	00:00:01
28	TABLE ACCESS FULL	SYS_TEMP_0FD9D665D_353EAF	7	98	2	(0)	00:00:01
29	VIEW	VM_VT_80F21617	62	5022	123K	(1)	00:00:05
30	VECTOR GROUP BY		62	2976	123K	(1)	00:00:05
31	HASH GROUP BY		62	2976	123K	(1)	00:00:05
32	KEY VECTOR USE	:KV0000	3566K	163M	123K	(1)	00:00:05
33	KEY VECTOR USE	:KV0001	3566K	149M	123K	(1)	00:00:05
34	KEY VECTOR USE	:KV0003	3566K	136M	123K	(1)	00:00:05
35	KEY VECTOR USE	:KV0002	11M	409M	123K	(1)	00:00:05
* 36	TABLE ACCESS FULL	LINEORDER	59M	1830M	123K	(1)	00:00:05

Predicate Information (identified by operation id):

```

13 - filter("S"."S_REGION"='AMERICA')
17 - filter("C"."C_REGION"='AMERICA')
19 - access("ITEM_16"=INTERNAL_FUNCTION("C0") AND "ITEM_17"="C2")
21 - access("ITEM_12"=INTERNAL_FUNCTION("C0") AND "ITEM_13"="C2" AND "ITEM_14"=INTERNAL_FUNCTION("C0") AND
"ITEM_15"="C2" AND "ITEM_18"=INTERNAL_FUNCTION("C0") AND "ITEM_19"="C2")
36 - filter((SYS_OP_KEY_VECTOR_FILTER("L"."LO_SUPPKEY",:KV0002) AND
SYS_OP_KEY_VECTOR_FILTER("L"."LO_CUSTKEY",:KV0003)))

```

Note

- vector transformation used for this statement

70 rows selected.

```

SQL>
SQL> @i3_novgb_buffer.sql
Connected.
SQL>
SQL> alter session set inmemory_query = disable;

```

Session altered.

Elapsed: 00:00:00.00

```

SQL>
SQL> -- Buffer Cache query with the column store disables via the inmemory_query parameter
SQL>
SQL> SELECT /*+ NO_VECTOR_TRANSFORM */
2      d.d_year, c.c_nation, sum(lo_revenue - lo_supplycost) profit
3      From   LINEORDER l, DATE_DIM d, PART p, SUPPLIER s, CUSTOMER C
4      Where  l.lo_orderdate = d.d_datekey
5      And    l.lo_partkey   = p.p_partkey
6      And    l.lo_suppkey   = s.s_suppkey
7      And    l.lo_custkey   = c.c_custkey
8      And    s.s_region     = 'AMERICA'
9      And    c.c_region     = 'AMERICA'
10     Group by d.d_year, c.c_nation

```


11 Order by d.d_year, c.c_nation;

D_YEAR	C_NATION	PROFIT
1992	ARGENTINA	257114232147
1992	BRAZIL	265087838628
1992	CANADA	267029429200
1992	PERU	260091746389
1992	UNITED STATES	264098681570
1993	ARGENTINA	260475234675
1993	BRAZIL	264646147899
1993	CANADA	265113570471
1993	PERU	262628508817
1993	UNITED STATES	260411002935
1994	ARGENTINA	261149015641
1994	BRAZIL	263808033983
1994	CANADA	264598150413
1994	PERU	258595600981
1994	UNITED STATES	265282504206
1995	ARGENTINA	258498976118
1995	BRAZIL	269135848643
1995	CANADA	264654265482
1995	PERU	257451709833
1995	UNITED STATES	259660457396
1996	ARGENTINA	259361903850
1996	BRAZIL	265970119048
1996	CANADA	265333193889
1996	PERU	260916013039
1996	UNITED STATES	262339293224
1997	ARGENTINA	261099548066
1997	BRAZIL	266353055971
1997	CANADA	265036379243
1997	PERU	259114682243
1997	UNITED STATES	262208356128
1998	ARGENTINA	151054449013
1998	BRAZIL	153632348378
1998	CANADA	156899052279
1998	PERU	152297126350
1998	UNITED STATES	153937088695

35 rows selected.

Elapsed: 00:00:20.79

SQL>

SQL> set echo off

Hit enter ...

PLAN_TABLE_OUTPUT

SQL_ID bgm86vz7ghjsj, child number 0

```
-----  
SELECT /*+ NO_VECTOR_TRANSFORM */      d.d_year, c.c_nation,  
sum(lo_revenue - lo_supplycost) profit  From      LINEORDER l,  
DATE_DIM d, PART p, SUPPLIER s, CUSTOMER C      Where  
l.lo_orderdate = d.d_datekey      And      l.lo_partkey      =  
p.p_partkey      And      l.lo_suppkey      = s.s_suppkey  
      And      l.lo_custkey      = c.c_custkey      And  
s.s_region      = 'AMERICA'      And      c.c_region  
      = 'AMERICA'      Group by d.d_year, c.c_nation  
Order by d.d_year, c.c_nation
```

Plan hash value: 3639618634

Id	Operation	Name	Rows	Bytes	TempSpc	Cost (%CPU)	Time
0	SELECT STATEMENT					178K (100)	
1	SORT GROUP BY		124	12276		178K (1)	00:00:07
* 2	HASH JOIN		3566K	336M		178K (1)	00:00:07
3	TABLE ACCESS FULL	DATE_DIM	2556	25560		15 (0)	00:00:01
* 4	HASH JOIN		3566K	302M	12M	178K (1)	00:00:07
5	TABLE ACCESS FULL	PART	800K	3906K		1891 (1)	00:00:01
* 6	HASH JOIN		3566K	285M	2696K	159K (1)	00:00:07
* 7	TABLE ACCESS FULL	CUSTOMER	60000	1992K		838 (1)	00:00:01
* 8	HASH JOIN		11M	568M		123K (1)	00:00:05
* 9	TABLE ACCESS FULL	SUPPLIER	4000	72000		69 (0)	00:00:01
10	TABLE ACCESS FULL	LINEORDER	59M	1830M		123K (1)	00:00:05

Predicate Information (identified by operation id):

- 2 - access("L"."LO_ORDERDATE"="D"."D_DATEKEY")
- 4 - access("L"."LO_PARTKEY"="P"."P_PARTKEY")
- 6 - access("L"."LO_CUSTKEY"="C"."C_CUSTKEY")
- 7 - filter("C"."C_REGION"='AMERICA')
- 8 - access("L"."LO_SUPPKEY"="S"."S_SUPPKEY")
- 9 - filter("S"."S_REGION"='AMERICA')

40 rows selected.

SQL>