



# **SAGE Computing Services**

Customised Oracle Training Workshops and Consulting

## ***The SQL and PL/SQL Results Cache***

***Is it a Dream Come True or Your Latest Nightmare?***

***Penny Cookson***

**Managing Director**

**penny@sagecomputing.com.au**



# ***Agenda***

***What it is?***

***How does it work?***

***What is it good for?***

**It takes a long  
time to fetch  
stuff from  
the shops**



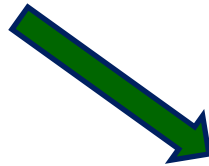


**So we keep stuff  
in the fridge  
ready to  
cook**

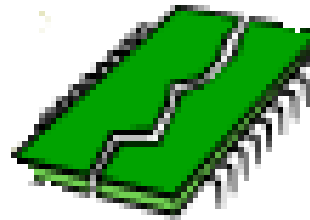
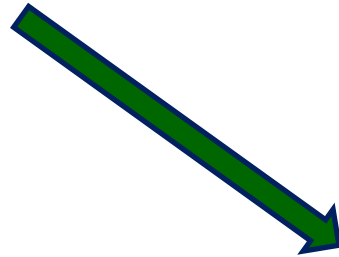
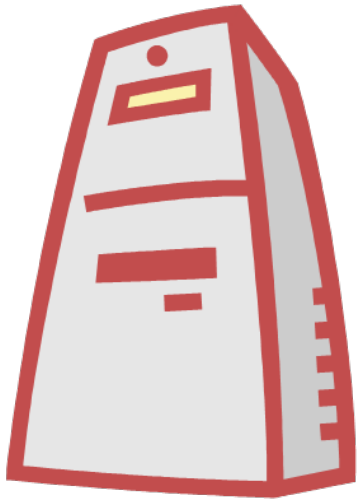
**But it  
still takes  
time to cook it**



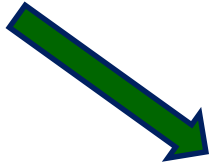
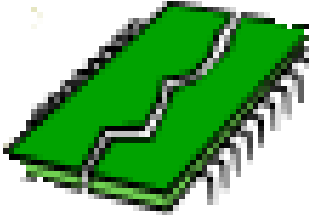
**So why not store  
it already  
prepared**



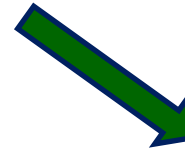
**Its much faster to  
get data from the  
SGA than from  
disk**



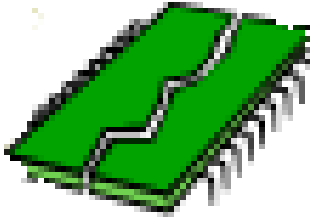
# But we still have to process it



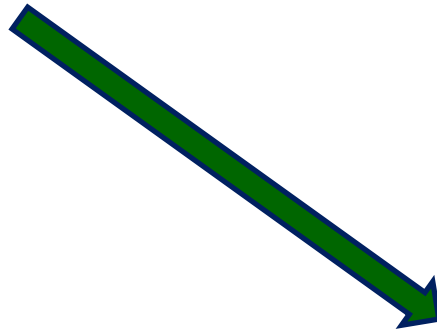
```
select col1, col2, pk1.func(col3),  
sum(col8)  
from table1, table2, table3  
where col1=col2  
and col3=col4  
and col4 > (select col5 from table4  
where col6 > col7)  
group by col1, col2 , pk1.func(col3)  
order by col1
```



# So why not store it already processed

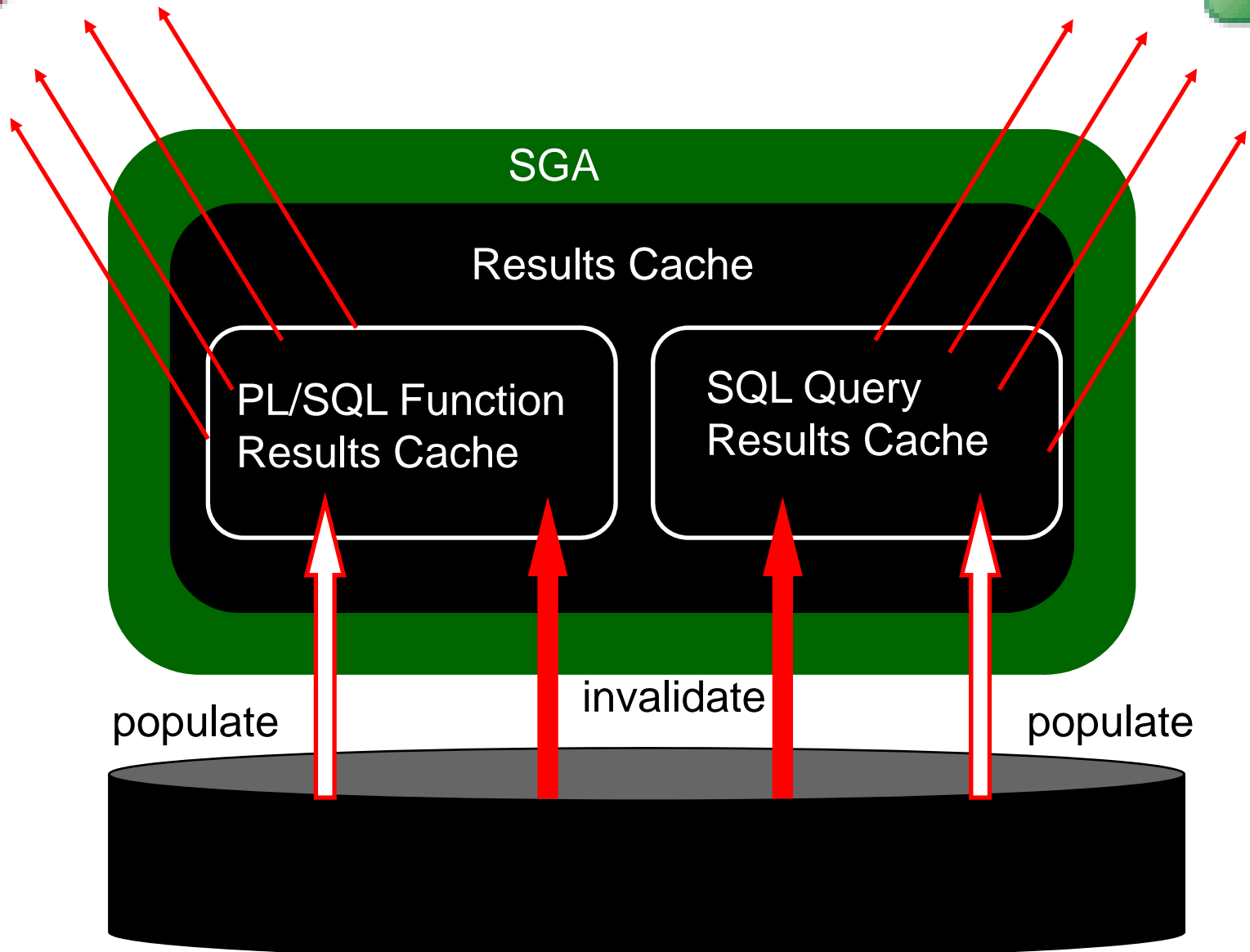


```
select col1, col2, pk1.func(col3),  
sum(col8)  
from table1, table2, table3  
where col1=col2  
and col3=col4  
and col4 > (select col5 from table4  
where col6 > col7)  
group by col1, col2 , pk1.func(col3)  
order by col1
```

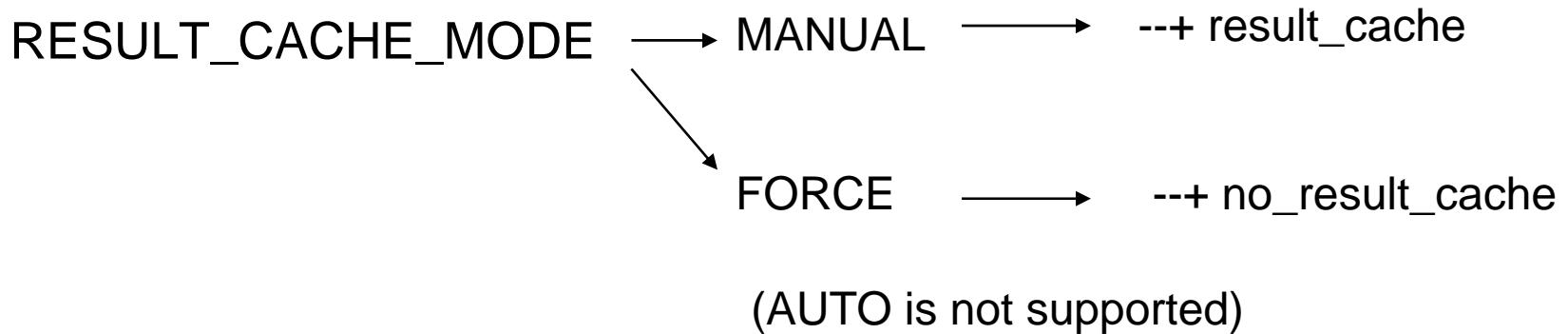




# Results Cache



# Query Results Cache



Will not cache if:-

- Dictionary tables

- Temporary tables

- Sequences

- Dates/times

- Non deterministic PL/SQL

# DBMS\_RESULT\_CACHE

DBMS\_RESULT\_CACHE.STATUS

DBMS\_RESULT\_CACHE.FLUSH

DBMS\_RESULT\_CACHE.MEMORY\_REPORT

DBMS\_RESULT\_CACHE.INVALIDATE

DBMS\_RESULT\_CACHE.BYPASS

## Monitoring the Results Cache

```
SELECT * FROM v$result_cache_memory
```

```
SELECT * FROM v$result_cache_objects
```

```
SELECT * FROM v$result_cache_statistics
```

```
SELECT * FROM v$result_cache_dependency
```

# Demo 1

**Disk v Buffer Cache v Result Cache**

\*\*\*\*\*

```
SELECT count(b.comments)
FROM train.events_large e, train.bookings_large b
WHERE e.org_id = :v1
AND e.event_no = b.event_no
AND e.comments = :v2
```

call	count	cpu	elapsed	disk	query	current	rows
Parse	1	0.00	0.00	0	0	0	0
Execute	1	0.04	0.03	0	0	0	0
Fetch	2	4.60	10.26	35341	36342	0	1
total	4	4.65	10.29	35341	36342	0	1

Misses in library cache during parse: 1  
 Misses in library cache during execute: 1  
 Optimizer mode: FIRST\_ROWS  
 Parsing user id: 102

Rows	Row Source Operation
1	SORT AGGREGATE (cr=36342 pr=35341 pw=35341 time=0 us)
175	HASH JOIN (cr=36342 pr=35341 pw=35341 time=16 us cost=10036 size=18420634 card=594214)
4	TABLE ACCESS FULL EVENTS_LARGE (cr=992 pr=0 pw=0 time=6 us cost=274 size=176545 card=10385)
5767168	TABLE ACCESS FULL BOOKINGS_LARGE (cr=35350 pr=35341 pw=35341 time=176016 us cost=9735 size=80740352 card=5767168)

Elapsed times include waiting on following events:

Event waited on	Times Waited	Max. Wait	Total Waited
SQL*Net message to client	2	0.00	0.00
direct path read	425	0.22	5.02
SQL*Net message from client	2	0.18	0.18

\*\*\*\*\*

\*\*\*\*\*

```
SELECT /*+ RESULT_CACHE */ count(b.comments)
FROM train.events_large e, train.bookings_large b
WHERE e.org_id = :v1
AND e.event_no = b.event_no
AND e.comments = :v2
```

# First Execution

call	count	cpu	elapsed	disk	query	current	rows
Parse	1	0.00	0.00	0	0	0	0
Execute	1	0.01	0.03	0	0	0	0
Fetch	2	4.04	10.33	35341	36342	0	1
total	4	4.06	10.37	35341	36342	0	1

Misses in library cache during parse: 1  
Misses in library cache during execute: 1  
Optimizer mode: FIRST\_ROWS  
Parsing user id: 102

Rows	Row Source Operation
1	RESULT CACHE fvczs5hza4svbc4r4mz1xwzhcz (cr=36342 pr=35341 pw=35341 time=0 us)
1	SORT AGGREGATE (cr=36342 pr=35341 pw=35341 time=0 us)
175	HASH JOIN (cr=36342 pr=35341 pw=35341 time=22 us cost=10036 size=18420634 card=594214)
4	TABLE ACCESS FULL EVENTS_LARGE (cr=992 pr=0 pw=0 time=6 us cost=274 size=176545 card=10385)
5767168	TABLE ACCESS FULL BOOKINGS_LARGE (cr=35350 pr=35341 pw=35341 time=169878 us cost=9735 size=80740352 card=5767168)

Elapsed times include waiting on following events:

Event waited on	Times Waited	Max. Wait	Total Waited
SQL*Net message to client	2	0.00	0.00
direct path read	425	0.22	6.03
SQL*Net message from client	2	0.16	0.16

\*\*\*\*\*



\*\*\*\*\*

```
SELECT /*+ RESULT_CACHE */ count(b.comments)
FROM train.events_large e, train.bookings_large b
WHERE e.org_id = :v1
AND e.event_no = b.event_no
AND e.comments = :v2
```

# Second Execution

call	count	cpu	elapsed	disk	query	current	rows
Parse	1	0.00	0.00	0	0	0	0
Execute	1	0.00	0.00	0	0	0	0
Fetch	2	0.00	0.00	0	0	0	1
total	4	0.00	0.00	0	0	0	1

Misses in library cache during parse: 0  
Optimizer mode: FIRST\_ROWS  
Parsing user id: 102

Rows	Row	Source	Operation
1	0	RESULT CACHE	fvczs5hza4svbc4r4mz1xwzhcz (cr=0 pr=0 pw=0 time=0 us)
0	0	SORT AGGREGATE	(cr=0 pr=0 pw=0 time=0 us)
0	0	HASH JOIN	(cr=0 pr=0 pw=0 time=0 us cost=10036 size=18420634 card=594214)
0	0	TABLE ACCESS FULL	EVENTS_LARGE (cr=0 pr=0 pw=0 time=0 us cost=274 size=176545 card=10385)
0	0	TABLE ACCESS FULL	BOOKINGS_LARGE (cr=0 pr=0 pw=0 time=0 us cost=9735 size=80740352 card=5767168)

Elapsed times include waiting on following events:

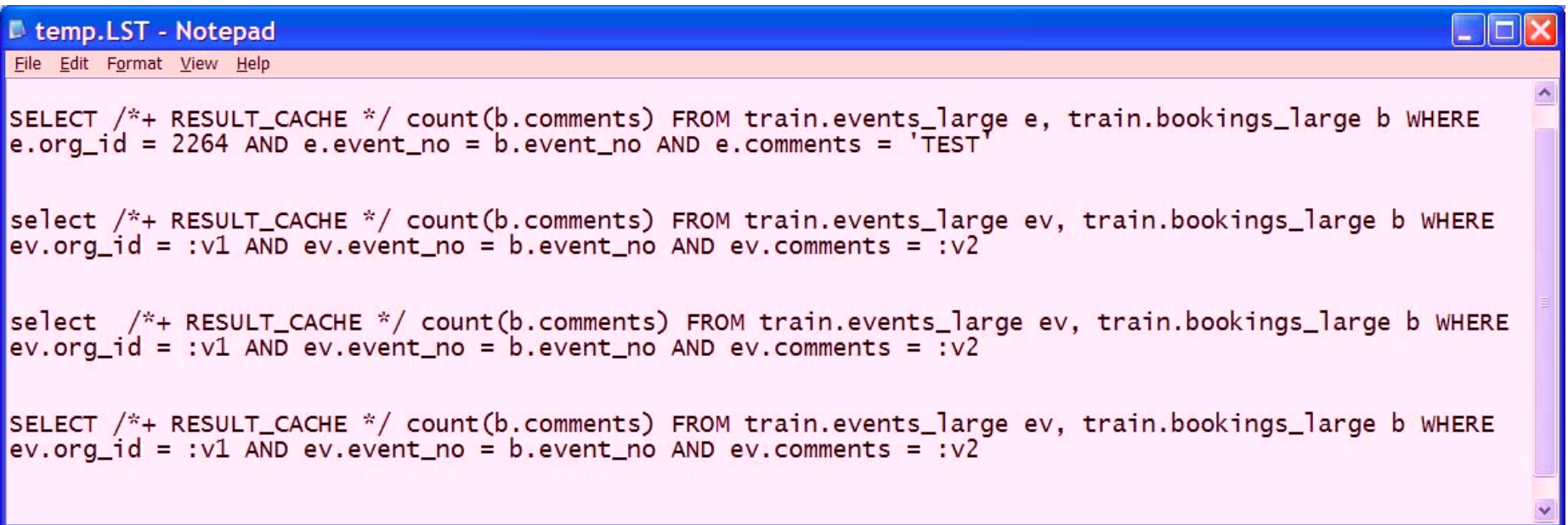
Event waited on	Times Waited	Max. Wait	Total Waited
SQL*Net message to client	2	0.00	0.00
SQL*Net message from client	2	0.01	0.01

\*\*\*\*\*



# V\$SQL

## Matching Criteria



A Notepad window titled "temp.LST - Notepad" with a menu bar (File, Edit, Format, View, Help). The text area contains four SQL queries. The first query is a full join. The second and third queries are left joins with different table aliases. The fourth query is a full join with different table aliases.

```
temp.LST - Notepad
File Edit Format View Help

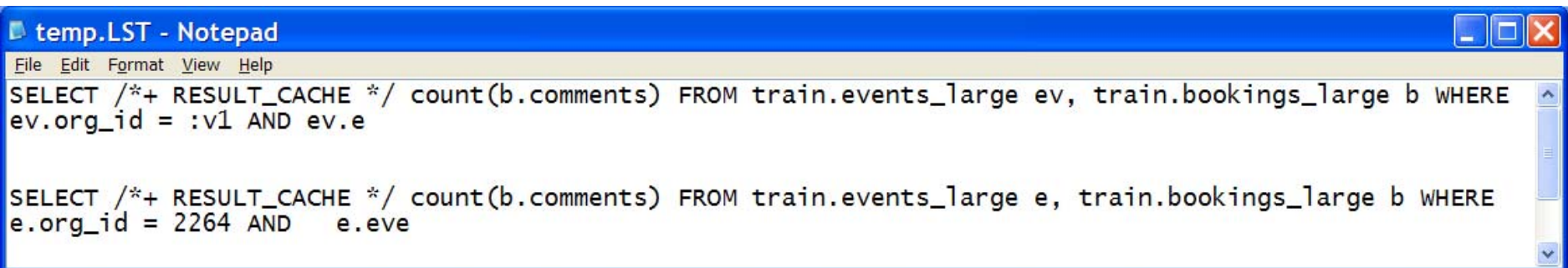
SELECT /*+ RESULT_CACHE */ count(b.comments) FROM train.events_large e, train.bookings_large b WHERE
e.org_id = 2264 AND e.event_no = b.event_no AND e.comments = 'TEST'

select /*+ RESULT_CACHE */ count(b.comments) FROM train.events_large ev, train.bookings_large b WHERE
ev.org_id = :v1 AND ev.event_no = b.event_no AND ev.comments = :v2

select /*+ RESULT_CACHE */ count(b.comments) FROM train.events_large ev, train.bookings_large b WHERE
ev.org_id = :v1 AND ev.event_no = b.event_no AND ev.comments = :v2

SELECT /*+ RESULT_CACHE */ count(b.comments) FROM train.events_large ev, train.bookings_large b WHERE
ev.org_id = :v1 AND ev.event_no = b.event_no AND ev.comments = :v2
```

# V\$\_RESULT\_CACHE\_OBJECTS



A Notepad window titled "temp.LST - Notepad" with a menu bar (File, Edit, Format, View, Help). The text area contains two SQL queries. The first query is a left join. The second query is a full join.

```
temp.LST - Notepad
File Edit Format View Help

SELECT /*+ RESULT_CACHE */ count(b.comments) FROM train.events_large ev, train.bookings_large b WHERE
ev.org_id = :v1 AND ev.e

SELECT /*+ RESULT_CACHE */ count(b.comments) FROM train.events_large e, train.bookings_large b WHERE
e.org_id = 2264 AND e.eve
```

# Flushing the Shared Pool and Buffer Cache

Enter SQL Statement:

```
ALTER SYSTEM FLUSH shared_pool;  
  
ALTER SYSTEM FLUSH buffer_cache;  
  
SELECT * FROM v$result_cache_objects;
```

Results Script Output Explain Autotrace DBMS Output OWA Output

Results:

	ID	TYPE	STATUS	BUCKET_NO	HASH	NAME
1	1	Dependency	Published	1524	4092503540	TRAIN.BOOKINGS_LARGE
2	0	Dependency	Published	1902	1859311470	TRAIN.EVENTS_LARGE
3	3	Result	Published	3150	2877217870	SELECT /*+ RESULT_CACHE */ count(b.comments)FROM train.events_large ev, train.bookings_large b\WHERE ..
4	2	Result	Published	511	1075298815	SELECT /*+ RESULT_CACHE */ count(b.comments)FROM train.events_large e, train.bookings_large b\WHERE e..

# Effect on Stats in the Shared Pool

## First Execution

select sql\_text, executions, child\_number,buffer\_gets, disk\_reads, fetches from v\$sql where sql\_id = '5fbry2cpl1t0kk'

Results | Script Output | Explain | Autotrace | DBMS Output | OWA Output

results:

SQL_TEXT	EXECUTIONS	CHILD_NUMBER	BUFFER_GETS	DISK_READS	FETCHES
1 SELECT /*+ RESULT_CACHE */ count(b.comments) FROM train.events_large e1, train....	1	0	36342	35341	1

After that

Enter SQL Statement:

select sql\_text, executions, child\_number,buffer\_gets, disk\_reads, fetches from v\$sql where sql\_id = '5fbry2cpl1t0kk'

Results | Script Output | Explain | Autotrace | DBMS Output | OWA Output

Results:

SQL_TEXT	EXECUTIONS	CHILD_NUMBER	BUFFER_GETS	DISK_READS	FETCHES
1 SELECT /*+ RESULT_CACHE */ count(b.comments) FROM train.events_large e1, train....	2	0	36342	35341	2
2 SELECT /*+ RESULT_CACHE */ count(b.comments) FROM train.events_large e1, train....	9	1	0	0	9

## Third Execution

select sql\_text, executions, child\_number,buffer\_gets, disk\_reads, fetches from v\$sql where sql\_id = '5fbry2cpl1t0kk'







Results | Script Output | Explain | Autotrace | DBMS Output | OWA Output

Results:

SQL_TEXT	EXECUTIONS	CHILD_NUMBER	BUFFER_GETS	DISK_READS	FETCHES
1 SELECT /*+ RESULT_CACHE */ count(b.comments) FROM train.events_large e1, train....	2	0	36342	35341	2
2 SELECT /*+ RESULT_CACHE */ count(b.comments) FROM train.events_large e1, train....	1	1	0	0	1

# Initialisation Parameters

```
SELECT * FROM v$parameter  
WHERE name like 'result_cache%'
```

	 NUM	 NAME	 TYPE	 VALUE	 DISPLAY_VALUE	 ISDEFAULT
1	1160	result_cache_mode	2	MANUAL	MANUAL	FALSE
2	1206	result_cache_max_size	6	10485760	10M	FALSE
3	1207	result_cache_max_result	3	5	5	FALSE
4	1208	result_cache_remote_expiration	3	0	0	TRUE

# Trace Event - 43905

```
ora11_ora_5560_FRED.trc - Notepad
File Edit Format View Help
Trace file c:\ora\diag\rdbms\ora11\ora11\trace\ora11_ora_5560_FRED.trc
Oracle Database 11g Enterprise Edition Release 11.1.0.6.0 - Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options
Windows XP Version V5.1 Service Pack 2
CPU : 2 - type 586
Process Affinity : 0x00000000
Memory (Avail/Total): Ph:1444M/3055M, Ph+PgF:5689M/7476M, VA:1243M/2047M
Instance name: ora11
Redo thread mounted by this instance: 1
Oracle process number: 27
Windows thread id: 5560, image: ORACLE.EXE (SHAD)

*** 2008-09-30 13:01:41.046
*** SESSION ID:(140.20) 2008-09-30 13:01:41.046
*** CLIENT ID:() 2008-09-30 13:01:41.046
*** SERVICE NAME:(ora11) 2008-09-30 13:01:41.046
*** MODULE NAME:(SQL*Plus) 2008-09-30 13:01:41.046
*** ACTION NAME:() 2008-09-30 13:01:41.046

Top level query block
Objects for this node : 79629 79637
-----
Query[len=168]: SELECT /*+ RESULT_CACHE */ count(b.comments)
FROM train.events_large e, train.bookings_large b
WHERE e.org_id = :v1
AND e.event_no = b.event_no
AND e.comments = :v2
Normalized Query[len=260]: 172 1?"COUNT" 225 1?"B" 226 1?"COMMENTS" 229 70 1?"TRAIN" 226
1?"EVENTS_LARGE" 1?"E" 219 1?"TRAIN" 226 1?"BOOKINGS_LARGE" 1?"B" 213 1?"E" 226 1?"ORG_ID" 221
218 1?"V1" 8 1?"E" 226 1?"EVENT_NO" 221 1?"B" 226 1?"EVENT_NO" 8 1?"E" 226 1?"COMMENTS" 221 218
1?"V2"
Cache id1: fvczs5hza4svbc4r4mz1xwzhcz
Cache id2: fvczs5hza4svbc4r4mz1xwzhcz
Column count: 1
NLS Dependent: YES User Referenced: NO Ordered: NO Auto: NO
Dependencies: (79629 - TRAIN.EVENTS_LARGE) (79637 - TRAIN.BOOKINGS_LARGE)
-----
```

```
BEGIN  DBMS_RESULT_CACHE.MEMORY_REPORT(true);  END;
```

## Result Cache Memory Report

### [Parameters]

Block Size = 1K bytes

Maximum Cache Size = 10M bytes (10K blocks)

Maximum Result Size = 10M bytes (10K blocks)

### [Memory]

Total Memory = 103536 bytes [0.056% of the Shared Pool]

... Fixed Memory = 5140 bytes [0.003% of the Shared Pool]

..... Cache Mgr = 112 bytes

..... Memory Mgr = 128 bytes

..... Bloom Fltr = 2K bytes

..... State Objs = 2852 bytes

... Dynamic Memory = 98396 bytes [0.053% of the Shared Pool]

..... Overhead = 65628 bytes

..... Hash Table = 32K bytes (4K buckets)

..... Chunk Ptrs = 12K bytes (3K slots)

..... Chunk Maps = 12K bytes

..... Miscellaneous = 8284 bytes

..... Cache Memory = 32K bytes (32 blocks)

..... Unused Memory = 16 blocks

..... Used Memory = 16 blocks




..... Dependencies = 4 blocks (4 count)

..... Results = 12 blocks

..... SQL = 4 blocks (1 count)

..... Invalid = 8 blocks (2 count)

## Explain Plan

 OPERATION	 OPTIONS	 OBJECT_NAME
SELECT STATEMENT	(null)	(null)
RESULT CACHE	(null)	bzurnr2sm456r4yctszyqks1s5
SORT	AGGREGATE	(null)
NESTED LOOPS	(null)	(null)
TABLE ACCESS	FULL	BOOKINGS_LARGE
INDEX	UNIQUE SCAN	EVTLG_PK

# So What Should We Use it For?

## Demo 2

**Single table – once**  
**Single table – multiple**  
**Join**  
**More complex statement**



# Changing the Data

```
SELECT /*+ RESULT_CACHE */ o.org_id, o.name, e.description, b.comments, b.cost
FROM organisations o, events_large e, bookings_large b
WHERE o.org_id = e.org_id
AND e.event_no = b.event_no
AND b.cost >
    (SELECT AVG(b10.cost) FROM bookings b10 WHERE b10.event_no = e.event_no)
AND o.org_id = 2264;
```

```
SELECT type, status, name, row_count, scan_count
FROM v$result_cache_objects;
```

ID	TYPE	STATUS	NAME	ID	ROW_COUNT	ID	SCAN_COUNT
1	Dependency	Published	TRAIN1.BOOKINGS_LARGE		0		0
2	Dependency	Published	TRAIN1.BOOKINGS		0		0
3	Dependency	Published	TRAIN1.EVENTS_LARGE		0		0
4	Dependency	Published	TRAIN1.ORGANISATIONS		0		0
5	Result	Published	SELECT /*+ RESULT_CACHE */ o.org_id, o.name, e.description, b.comments, b.costFROM organisati...		25		0

# Changing the Data




```
UPDATE events_large SET start_date = start_date -1;  
COMMIT;
```

```
SELECT type, status, name, row_count, scan_count , invalidations  
FROM v$result_cache_objects;
```

R2	TYPE	R2	STATUS	R2	NAME	R2	ROW_COUNT	R2	SCAN_COUNT	R2	INVALIDATIONS
1	Dependency		Published		TRAIN1.BOOKINGS_LARGE		0		0		0
2	Dependency		Published		TRAIN1.BOOKINGS		0		0		0
3	Dependency		Published		TRAIN1.EVENTS_LARGE		0		0		1
4	Dependency		Published		TRAIN1.ORGANISATIONS		0		0		0
5	Result		Invalid		SELECT /*+ RESULT_CACHE */ o.org_id, o.name,e.description, b.comme...		25		0		0

# Changing the Data

```
SELECT * FROM  
v$result_cache_statistics;
```

	 ID	 NAME	 VALUE
1	1	Block Size (Bytes)	1024
2	2	Block Count Maximum	10240
3	3	Block Count Current	32
4	4	Result Size Maximum (Blocks)	10240
5	5	Create Count Success	1
6	6	Create Count Failure	0
7	7	Find Count	0
8	8	Invalidation Count	1
9	9	Delete Count Invalid	0
10	10	Delete Count Valid	0

## So What Should We Use it For?

Queries that are:

Frequently accessed

Are not simple (Involve joins /aggregates/  
subqueries)

Are based on data which changes rarely

Have an acceptable response even without  
the result cache

# Which Products Should (NOT) Use It

WINDOW1

Booking No	179
Event No	579
Resource Code	VCR1
Chargeable	N
Made By	USER2
Quantity	20
Cost	420
Status	C
Comments	Lunch arranged for 12:30



```
CREATE OR REPLACE view  
bookings_large_vw  
AS  
SELECT /*+RESULT_CACHE */ *  
FROM bookings_large
```

Execute Query

SELECT type, status, name, row\_count,  
scan\_count, invalidations FROM  
v\$result\_cache\_objects;

	TYPE	STATUS	NAME	ROW_COUNT	SCAN_COUNT	INVALIDATIONS
1	Dependency	Published	TRAIN1.BOOKINGS_LARGE	0	0	0
2	Result	New	TRAIN1.BOOKINGS_LARGE_VW	0	0	0

# Second Form

The screenshot shows a software interface with a form titled "Second Form". The form contains several input fields: "Booking No", "Event No", "Resource Code", "Chargeable", and "Made By". Below these fields is a table with columns for "Status" and "Comments". The table is currently empty. At the bottom of the form, there is a button labeled "Execute Query" and a status bar showing "Record: 1/1".

Status	Comments
--------	----------



Execute Query

# Second Form

WINDOW1

Booking No	179
Event No	579
Resource Code	VCR1
Chargeable	N
Made By	USER2
Quantity	20
Cost	420
Status	C
Comments	Lunch arranged for 12:30

and eventually after

`_result_cache_timeout`

SELECT type, status, name, row\_count,  
scan\_count, invalidations FROM  
v\$result\_cache\_objects;

	TYPE	STATUS	NAME	ROW_COUNT	SCAN_COUNT	INVALIDATIONS
1	Dependency	Published	TRAIN1.BOOKINGS_LARGE	0	0	0
2	Result	Bypass	TRAIN1.BOOKINGS_LARGE_VW	0	0	0

# Why the Wait?

```
SELECT wait_time, event, state
FROM v$session_wait
WHERE sid=138;
```

WAIT_TIME	EVENT	STATE
0	enq: RC - Result Cache: Contention	WAITING

```
SELECT event, total_waits, time_waited,
average_wait, time_waited_micro
FROM v$system_event
WHERE event like 'enq: RC%'
ORDER BY time_waited
```

EVENT	TOTAL_WAITS	TIME_WAITED	AVERAGE_WAIT
enq: RC - Result Cache: Contention	12	12014	1001.13

if \_result\_cache\_timeout = 0 then no wait



Trace file c:\ora\diag\rdbms\ora11\ora11\trace\ora11\_ora\_4100.trc  
Oracle Database 11g Enterprise Edition Release 11.1.0.6.0 - Production  
With the Partitioning, OLAP, Data Mining and Real Application Testing options  
Windows XP Version V5.1 Service Pack 2  
CPU : 2 - type 586  
Process Affinity : 0x00000000  
Memory (Avail/Total): Ph:1445M/3055M, Ph+PgF:5789M/7476M, VA:1276M/2047M  
Instance name: ora11  
Redo thread mounted by this instance: 1  
Oracle process number: 32  
Windows thread id: 4100, image: ORACLE.EXE (SHAD)

\*\*\* 2008-09-30 08:41:19.203  
\*\*\* SESSION ID:(138.62) 2008-09-30 08:41:19.203  
\*\*\* CLIENT ID:() 2008-09-30 08:41:19.203  
\*\*\* SERVICE NAME:(ora11) 2008-09-30 08:41:19.203  
\*\*\* MODULE NAME:(frmweb.exe) 2008-09-30 08:41:19.203  
\*\*\* ACTION NAME:() 2008-09-30 08:41:19.203

WAIT #3: nam='enq: RC - Result Cache: Contention' ela= 10020510 name|mode=1380122628 chunkNo=1  
blockNo=2 obj#=83444 tim=12655511975

\*\*\* 2008-09-30 08:41:29.234  
WAIT #3: nam='enq: RC - Result Cache: Contention' ela= 10019793 name|mode=1380122628 chunkNo=1  
blockNo=2 obj#=83444 tim=12665543323

\*\*\* 2008-09-30 08:41:39.250  
WAIT #3: nam='enq: RC - Result Cache: Contention' ela= 10015302 name|mode=1380122628 chunkNo=1  
blockNo=2 obj#=83444 tim=12675558840

\*\*\* 2008-09-30 08:41:49.265  
WAIT #3: nam='enq: RC - Result Cache: Contention' ela= 10015428 name|mode=1380122628 chunkNo=1  
blockNo=2 obj#=83444 tim=12685574484

\*\*\* 2008-09-30 08:41:59.281  
WAIT #3: nam='enq: RC - Result Cache: Contention' ela= 10015461 name|mode=1380122628 chunkNo=1  
blockNo=2 obj#=83444 tim=12695590161

\*\*\* 2008-09-30 08:42:09.296  
WAIT #3: nam='enq: RC - Result Cache: Contention' ela= 10015198 name|mode=1380122628 chunkNo=1  
blockNo=2 obj#=83444 tim=12705605597  
WAIT #3: nam='direct path read' ela= 15044 file number=6 first dba=9068 block cnt=13 obj#=83444  
tim=12705621289  
WAIT #3: nam='SQL\*Net message to client' ela= 4 driver id=1413697536 #bytes=1 p3=0 obj#=83444  
tim=12705621356

## **Which Products Should Use It**

Only those that fetch all the rows

So never set it as a default

# PL/SQL Results Cache

```
CREATE OR REPLACE FUNCTION quantitybooked
(p_resource_code in resources.code%TYPE,p_event_date in date)
RETURN NUMBER
RESULT_CACHE RELIES_ON (BOOKINGS,EVENTS)
IS
    v_total_booked number := 0;
BEGIN
    SELECT      sum(b.quantity)
    INTO        v_total_booked
    FROM        bookings b, events e
    WHERE       e.event_no = b.event_no
    AND         p_event_date between e.start_date and e.end_date
    AND         b.resource_code = p_resource_code;

    book_pkg.add_one(v_total_booked, v_total_booked);

    RETURN (v_total_booked);
END;
```

# PL/SQL Results Cache

```
DECLARE
```

```
v_num number;
```

```
BEGIN
```

```
  v_num := quantitybooked('VCR2','06-NOV-2007');
```

```
  dbms_output.put_line(v_num);
```

```
END;
```

9

```
SELECT *
```

```
FROM v$result_cache_objects
```

NAME	TYPE	STATUS	NAME	PIN_CODE	SCAN_COUNT	INVALIDATIONS
NAME	TYPE	STATUS	NAME	PIN_CODE	SCAN_COUNT	INVALIDATIONS
(null)	Dependency	Published	TRAIN.QUANTITYBOOKED	0	0	0
(null)	Dependency	Published	TRAIN.BOOKINGS	0	0	0
PL (null)	Dependency	Published	TRAIN.EVENTS	0	0	0
PLSQL	Result	Published	"TRAIN"."QUANTITYBOOKED":8."QUANTITYBOOKED"#496944ef298963c5 #1	0	1	0

# PL/SQL Results Cache

```
UPDATE bookings SET quantity = quantity*2;
COMMIT;
```

N...	TYPE	STATUS	NAME	PIN_CO...	SCAN_COUNT	INVALIDATIONS
(null)	Dependency	Published	TRAIN.QUANTITYBOOKED	0	0	0
(null)	Dependency	Published	TRAIN.BOOKINGS	0	0	1
(null)	Dependency	Published	TRAIN.EVENTS	0	0	0
PLSQL	Result	Invalid	"TRAIN"."QUANTITYBOOKED":8."QUANTITYBOOKED"#496944ef298963c5 #1	0	1	0

```
DECLARE
```

```
v_num number;
```

```
BEGIN
```

```
  v_num := quantitybooked('VCR2','06-NOV-2007');
```

```
  dbms_output.put_line(v_num);
```

```
END;
```

17

N...	TYPE	STATUS	NAME	PIN_CO...	SCAN_COUNT	INVALIDATIONS
(null)	Dependency	Published	TRAIN.QUANTITYBOOKED	0	0	0
(null)	Dependency	Published	TRAIN.BOOKINGS	0	0	1
(null)	Dependency	Published	TRAIN.EVENTS	0	0	0
PLSQL	Result	Published	"TRAIN"."QUANTITYBOOKED":8."QUANTITYBOOKED"#496944ef298963c5 #1	0	0	0
PLSQL	Result	Invalid	"TRAIN"."QUANTITYBOOKED":8."QUANTITYBOOKED"#496944ef298963c5 #1	0	1	0

# PL/SQL Results Cache

```
CREATE OR REPLACE PACKAGE BODY book_pkg AS
PROCEDURE add_one (p_result IN NUMBER, p_result_out OUT NUMBER)
IS
BEGIN
    p_result_out := p_result+10;
END add_one;
END book_pkg;
/
```

N...	TYPE	STATUS	NAME	PIN_CO...	SCAN_COUNT	INVALIDATIONS
(null)	Dependency	Published	TRAIN.QUANTITYBOOKED	0	0	0
(null)	Dependency	Published	TRAIN.BOOKINGS	0	0	1
(null)	Dependency	Published	TRAIN.EVENTS	0	0	0
PLSQL	Result	Published	"TRAIN"."QUANTITYBOOKED"::8."QUANTITYBOOKED"#496944ef298963c5 #1	0	1	0
PLSQL	Result	Invalid	"TRAIN"."QUANTITYBOOKED"::8."QUANTITYBOOKED"#496944ef298963c5 #1	0	1	0

# PL/SQL Results Cache

```
DECLARE
```

```
v_num number;
```

```
BEGIN
```

```
  v_num := quantitybooked('VCR2','06-NOV-2007');
```

```
  dbms_output.put_line(v_num);
```

```
END;
```

17

N...	TYPE	STATUS	NAME	PIN_CO...	SCAN_COUNT	INVALIDATIONS
{null}	Dependency	Published	TRAIN.QUANTITYBOOKED	0	0	0
{null}	Dependency	Published	TRAIN.BOOKINGS	0	0	1
{null}	Dependency	Published	TRAIN.EVENTS	0	0	0
PLSQL	Result	Published	"TRAIN"."QUANTITYBOOKED"::8."QUANTITYBOOKED"#496944ef298963c5 #1	0	2	0
PLSQL	Result	Invalid	"TRAIN"."QUANTITYBOOKED"::8."QUANTITYBOOKED"#496944ef298963c5 #1	0	1	0

# PL/SQL Results Cache

```
CREATE OR REPLACE PACKAGE BODY book_pkg    AS
PROCEDURE add_one (p_result IN NUMBER, p_result_out OUT NUMBER)
IS
BEGIN
    p_result_out := p_result+10;
END add_one;
END book_pkg;
/
```

```
BEGIN
    dbms_result_cache.invalidate('TRAIN1','QUANTITYBOOKED');
END;
/
```

R#	N...	R#	TYPE	R#	STATUS	R#	NAME	R#	PIN_CO...	R#	SCAN_COUNT	R#	INVALIDATIONS
(null)			Dependency		Published		TRAIN.QUANTITYBOOKED		0		0		1
(null)			Dependency		Published		TRAIN.BOOKINGS		0		0		1
(null)			Dependency		Published		TRAIN.EVENTS		0		0		0
	0			1		0	PLSQL Result	Invalid					"TRAIN"."QUANTITYBOOKED":8."QUANTITYBOOKED"#496944ef298963c5 #1
	0			1		0	PLSQL Result	Invalid					"TRAIN"."QUANTITYBOOKED":8."QUANTITYBOOKED"#496944ef298963c5 #1



# PL/SQL Results Cache

# Don't do stupid stuff

## CREATE OR REPLACE FUNCTION non\_determ

RETURN VARCHAR2 RESULT\_CACHE

IS

# BEGIN

# RETURN

```
TO_CHAR(sysdate,'ddmmyyyy hh:mi:ss');
```

END;

[illegible]

# PL/SQL Results Cache - Example

## Demo 3

**Create and execute a RESULT CACHE function**

# Using the Results Cache – Design Example

ELEMENTS	
Constraints	Grants
Statistics	Trig...
Sort...	
SYMBOL	NAME
1 Fe	Iron
2 Al	Aluminium
3 Ca	Calcium
4 Ba	Barium
5 Cu	Copper
6 Na	Sodium
7 Pb	Lead
8 Zn	Zinc
9 P	Phospho...
10 K	Potassium

ELEMENT_RANGE						
Data	Constraints	Grants	Statistics	Triggers	Flashback	Dependencies
Sort...	Filter:					Actions...
ID	EL_SYMBOL	LOW_PCT	HIGH_PCT	START_DATE	END_DATE	
1	1 Al	1.6	4	01-JAN-08	02-JAN-08	
2	2 Ba	0.07	0.09	01-JAN-08	02-JAN-08	
3	3 Ca	0.11	0.15	01-JAN-08	02-JAN-08	
4	4 Cu	5.1	5.9	01-JAN-08	02-JAN-08	
5	5 Fe	51.5	60	01-JAN-08	02-JAN-08	
6	6 K	0.8	0.9	01-JAN-08	02-JAN-08	
7	7 Na	0.7	0.8	01-JAN-08	02-JAN-08	
8	8 P	1.1	1.5	01-JAN-08	02-JAN-08	
9	9 Pb	6	7	01-JAN-08	02-JAN-08	
10	10 Zn	3	4	01-JAN-08	02-JAN-08	
11	651 Al	1.6	4	02-JAN-08	03-JAN-08	
12	652 Ba	0.07	0.09	02-JAN-08	03-JAN-08	
13	653 Ca	0.11	0.15	02-JAN-08	03-JAN-08	
14	654 Cu	5.1	5.9	02-JAN-08	03-JAN-08	

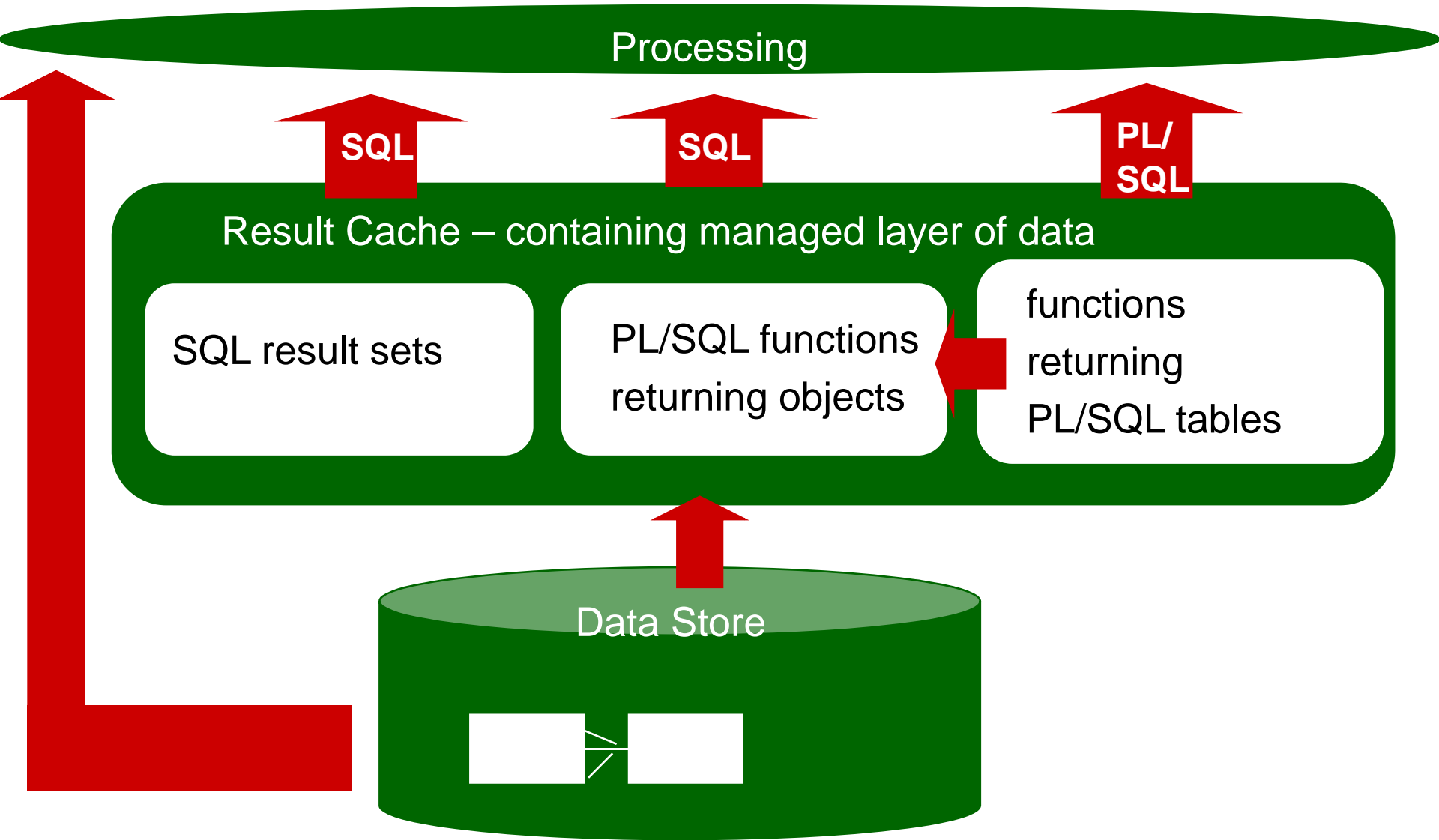
train1		sys		ELEMENT_VALUES	
Columns	Data	Constraints	Grants	Statistics	Triggers
Flashback	Depend				
Sort...	Filter:				
ID	MEASUREMENT_DATE	EL_SYMBOL	VALUE		
1 2366...	24-AUG-08	Al	4.01		
2 2366...	24-AUG-08	Ba	0.1		
3 2366...	24-AUG-08	Ca	0.16		
4 2366...	24-AUG-08	Cu	5.91		
5 2366...	24-AUG-08	Zn	4.01		
6 2366...	24-AUG-08	K	0.91		
7 2366...	24-AUG-08	Na	0.81		
8 2366...	24-AUG-08	P	1.51		
9 2366...	24-AUG-08	Pb	7.01		
10 2366...	24-AUG-08	Fe	60.01		
11 1973...	24-AUG-08	Zn	3.99		
12 1973...	24-AUG-08	Pb	6.99		
13 1973...	24-AUG-08	P	1.49		
14 1973...	24-AUG-08	Na	0.79		
15 1973...	24-AUG-08	K	0.89		
16 1973...	24-AUG-08	Fe	59.99		
17 1973...	24-AUG-08	Cu	5.89		
18 1973...	24-AUG-08	Ca	0.14		
19 1973...	24-AUG-08	Al	3.99		

# Using the Results Cache – Design Example

## Demo 4

### Designing a Results Cache Layer

# Designing for the Cross Session Results Cache



# Writing it in 10g

```
FUNCTION qty_booked(p_resource IN VARCHAR2
                    ,p_date IN DATE)
    RETURN NUMBER
$IF dbms_db_version.ver_le_10 $THEN
$ELSE RESULT_CACHE $END
IS
    li_total  PLS_INTEGER := 0;
BEGIN
    SELECT SUM(b.qty)
    INTO    li_total
    FROM    bookings b, events e
    WHERE   p_date BETWEEN e.start_date AND e.end_date
    AND     b.resource = p_resource;

    RETURN li_total;
END qty_booked;
```

# Caching Mechanisms

## Buffer Cache

Shared

Block level data stored

## PGA Memory

Coded by developers

Store per user data in PGA

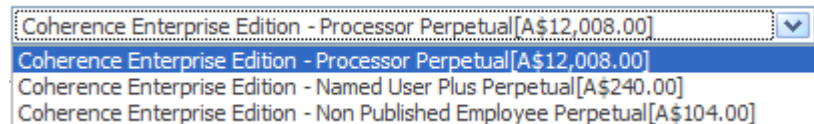
## Results Cache

Shared

Results stored

Clustered caching services

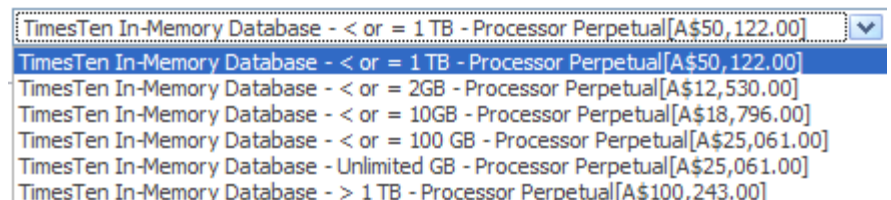
## Coherence



Coherence Enterprise Edition - Processor Perpetual[A\$12,008.00]	▼
Coherence Enterprise Edition - Processor Perpetual[A\$12,008.00]	
Coherence Enterprise Edition - Named User Plus Perpetual[A\$240.00]	
Coherence Enterprise Edition - Non Published Employee Perpetual[A\$104.00]	

## Times Ten database

In memory database



TimesTen In-Memory Database - < or = 1 TB - Processor Perpetual[A\$50,122.00]	▼
TimesTen In-Memory Database - < or = 1 TB - Processor Perpetual[A\$50,122.00]	
TimesTen In-Memory Database - < or = 2GB - Processor Perpetual[A\$12,530.00]	
TimesTen In-Memory Database - < or = 10GB - Processor Perpetual[A\$18,796.00]	
TimesTen In-Memory Database - < or = 100 GB - Processor Perpetual[A\$25,061.00]	
TimesTen In-Memory Database - Unlimited GB - Processor Perpetual[A\$25,061.00]	
TimesTen In-Memory Database - > 1 TB - Processor Perpetual[A\$100,243.00]	

# So What Should we Use it For?

## (Battle of the Experts)

The Results Cache causes contention in TPO systems



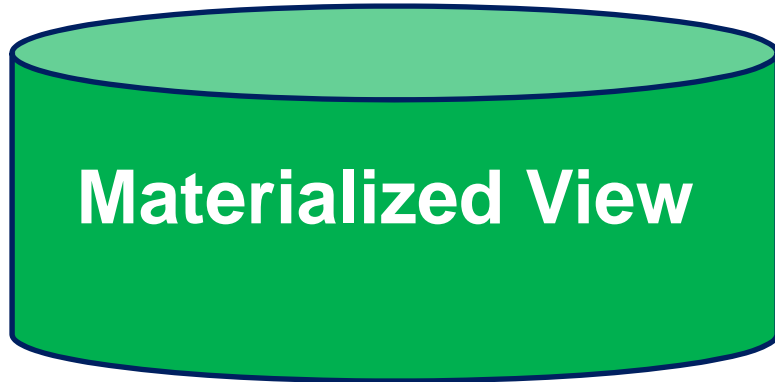
The feature was never meant for access to small simple data sets in TPO applications, its for aggregations of large data sets into small results – like a just in time MV

So why does Oracle use it for access to Apex meta data?





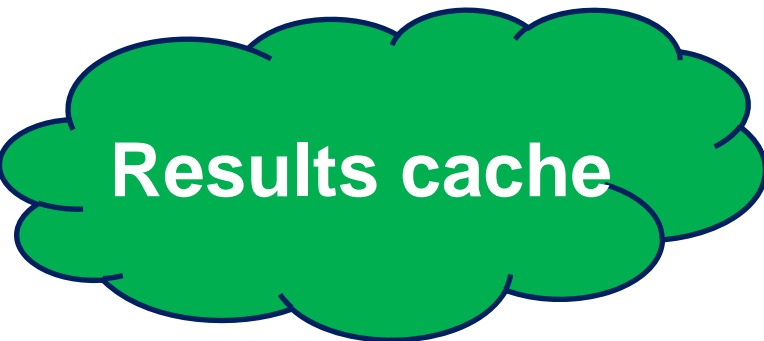
# Result Cache v Materialized Views



Aggregate large amounts of data into small results set

Store the results in a table

Users accesses the table/buffer cache

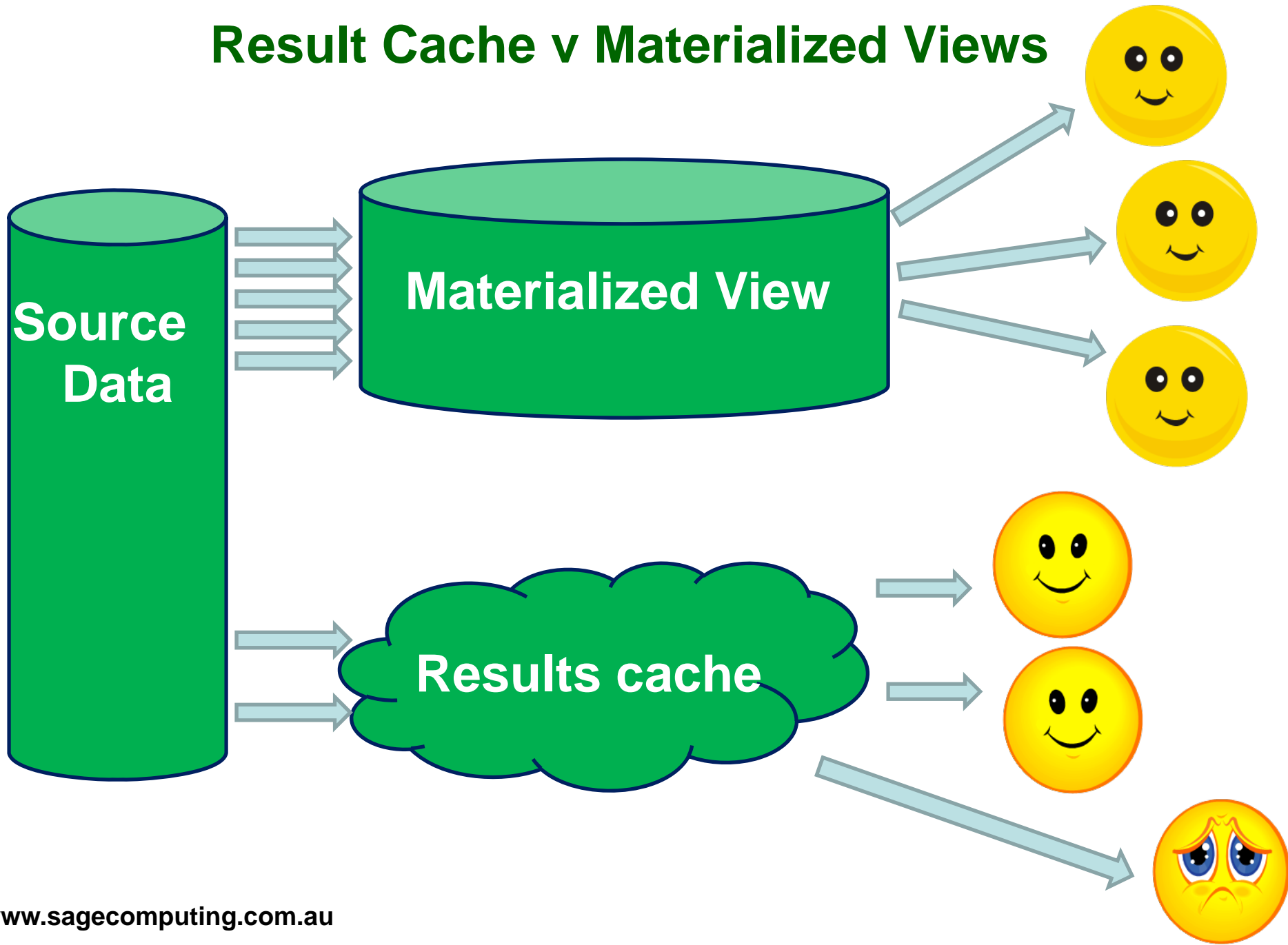


Aggregate large amounts of data into small results set

Store the results in the SGA

User accesses results in the SGA -  
maybe

# Result Cache v Materialized Views



# Monitoring Contention

```
SELECT name, gets, misses, sleeps,  
immediate_gets, immediate_misses  
FROM v$latch  
WHERE name like '%Result Cache%';
```

NAME	GETS	MISSES	SLEEPS	IMMEDIATE_GETS	IMMEDIATE_MISSES
Result Cache: Latch	41980	0	0	0	0
Result Cache: SO Latch	16	0	0	0	0

```
SELECT event, total_waits, time_waited,  
average_wait  
FROM v$system_event  
WHERE event like 'enq: RC%'
```

EVENT	TOTAL_WAITS	TIME_WAITED	AVERAGE_WAIT
enq: RC - Result Cache: Contention	18	18019	1001.05

# A Personal Opinion (or six)

If you need aggregation use a materialized view not the results cache

Leave `RESULT_CACHE_MODE` as `MANUAL`

Use the PL/SQL result cache for a managed layer of frequently used results, e.g metadata

Do not use the result cache when the time taken to access the data from the buffer cache is unacceptable (even for one user you don't like much)

Add the `RESULT_CACHE` hint to selected queries on reference data

or aggregates that are just acceptable but could be better



# So – To Complete the Metaphor

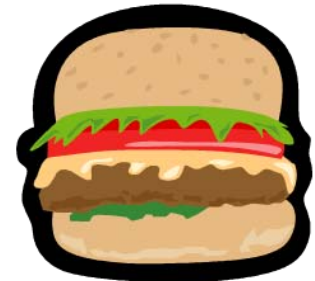
Don't store a whole load of watermelons – they will fill up your fridge



Its going to take too long to knock up a roast if you run out of them in the freezer



The PL/SQL results cache is designed to make fast food even faster





# **SAGE Computing Services**

Customised Oracle Training Workshops and Consulting

## **Information sources:**

The Oracle doco

Vladimir Begun: <http://vbegun.blogspot.com>

Alex Fatkulin: <http://www.pythian.com>

Connor McDonald: [OracleDBA.co.uk](http://OracleDBA.co.uk)

(who took the p\*ss at me using SQL\*Developer – thank you for getting me back to the command line, I promise not to lapse again)

And a whole load of playing around with the features



# **SAGE Computing Services**

Customised Oracle Training Workshops and Consulting

## **Questions and Answers?**



Presentations are available from our website:

*<http://www.sagecomputing.com.au>*

*[enquiries@sagecomputing.com.au](mailto:enquiries@sagecomputing.com.au)*

*[penny@sagecomputing.com.au](mailto:penny@sagecomputing.com.au)*



# **SAGE Computing Services**

## **(Unrepentant Marketing)**

**Register for the Next**  
**PL/SQL course:**  
**18<sup>th</sup> – 20<sup>th</sup> November**

Call in at the Sage Computing Services exhibition booth, or contact Penny

*[penny@sagecomputing.com.au](mailto:penny@sagecomputing.com.au)*

*0419904458*