

# **Администрирование баз данных и приложений**

**Архитектура ORACLE: Области  
оперативной памяти**

**Лекция 5**

# При запуске экземпляра

---

- ▶ выделяется область разделяемой памяти, называемая SGA
  - ▶ запускаются фоновые процессы
- 
- ▶ Процесс - это механизм в операционной системе, который может выполнить последовательность шагов



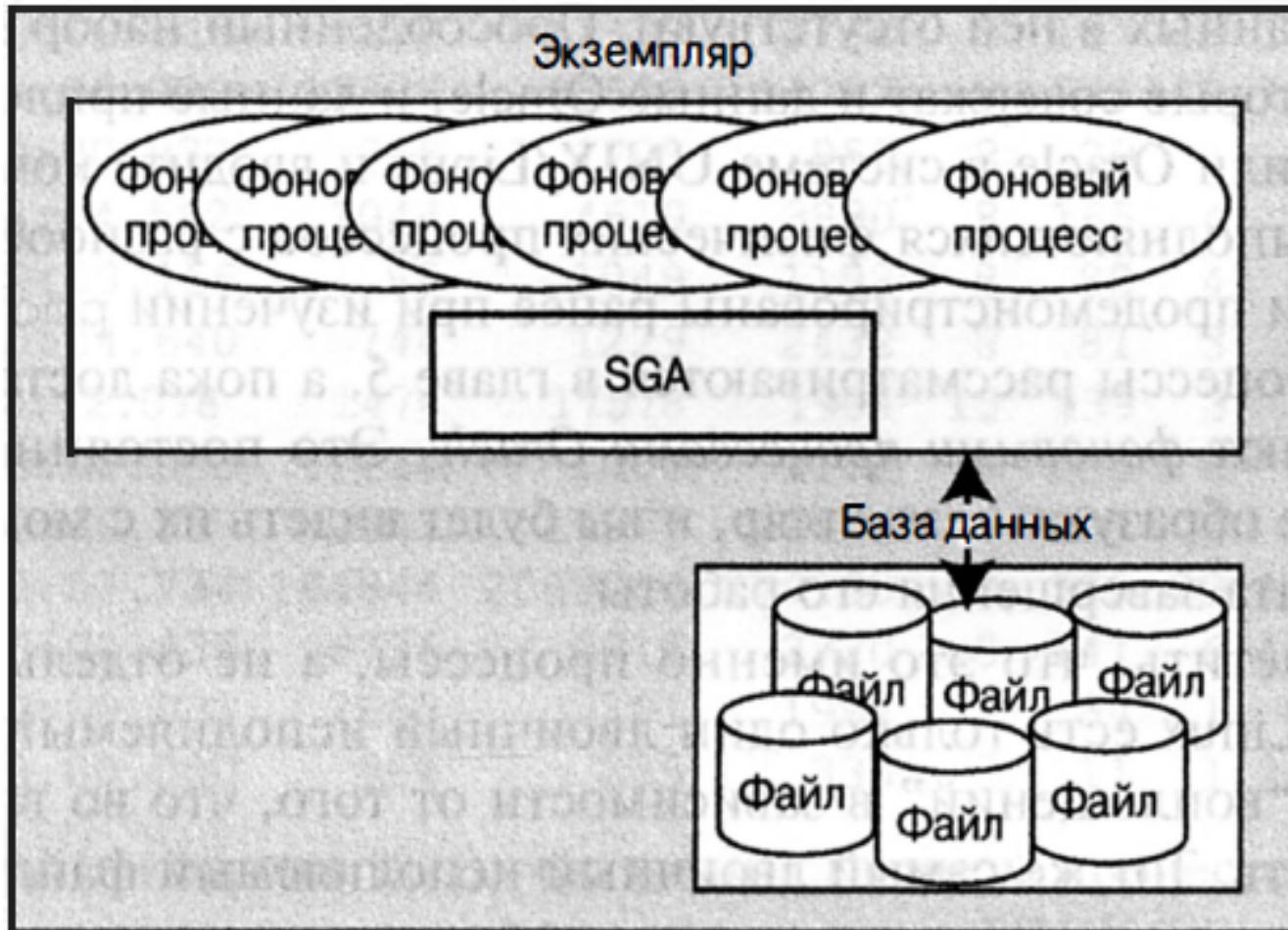
# Хранение и использование данных

---

- ▶ выполняемый код программы
- ▶ данные, которые совместно используются
- ▶ частные области данных для каждого подключенного пользователя



# Схема экземпляра



# Области оперативной памяти

---

- ▶ Системная Глобальная область - SGA
- ▶ Глобальные программные области - PGA



# SGA

---

- ▶ группа областей разделяемой памяти
- ▶ содержат данные и управляющую информацию для одного экземпляра Oracle
- ▶ совместно используется всеми серверными и фоновыми



# PGA

---

- ▶ создаваемая Oracle неразделяемая память при запуске серверного или фонового процесса
- ▶ области памяти, которые содержат данные и управляющую информацию для серверного или фонового процесса, такие как пространство стека или информация сеанса
- ▶ доступ к PGA является монопольным для серверного процесса
- ▶ у каждого серверного процесса и фонового процесса есть собственная PGA
- ▶ запись в PGA выполняется только кодом Oracle, действующим по запросу PGA



# PGA

---

- ▶ содержит стековое пространство
- ▶ содержит пользовательскую глобальную область (UGA)



# UGA

---

- ▶ Область курсоров
- ▶ Область хранения данных сеанса
- ▶ Рабочие области SQL:
  - ▶ Область сортировки для функций, таких как ORDER BY и GROUP BY
  - ▶ Область хеша для выполнения соединения таблиц
  - ▶ Область создания битового массива
  - ▶ Область слияния битовых массивов, используемую для того, чтобы разрешать план выполнения индекса битового массива



# UGA – User Global Area

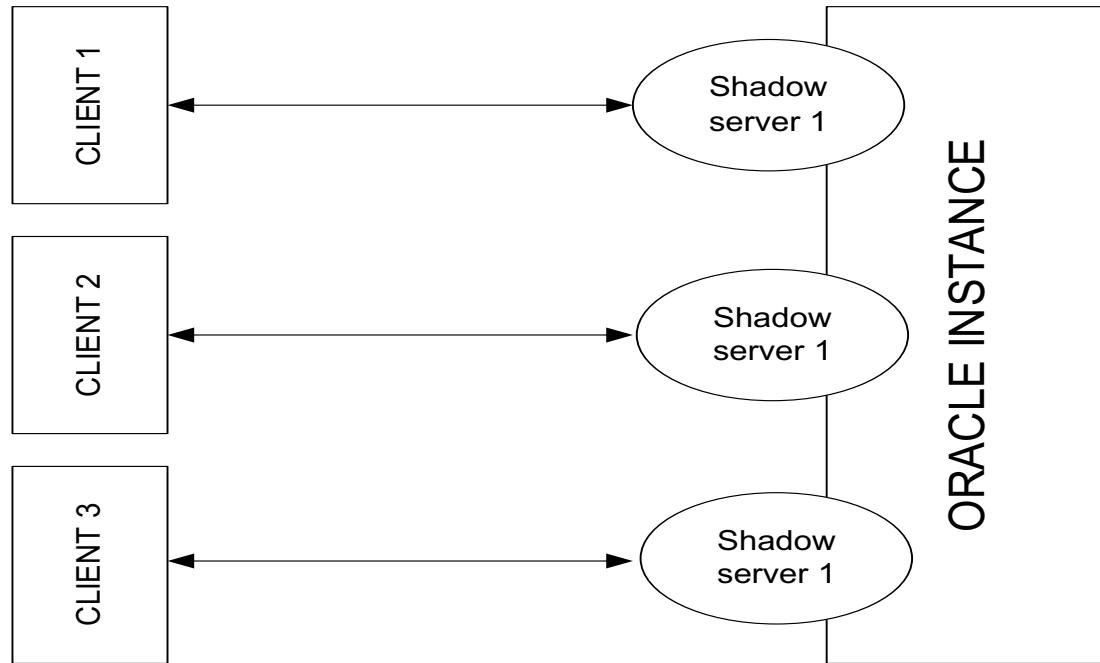
---

- ▶ Располагается в области SGA, если подключение выполнено посредством разделяемого сервера, в PGA остается только стековое пространство
- ▶ Располагается в области PGA, если подключение выполнено через выделенный сервер



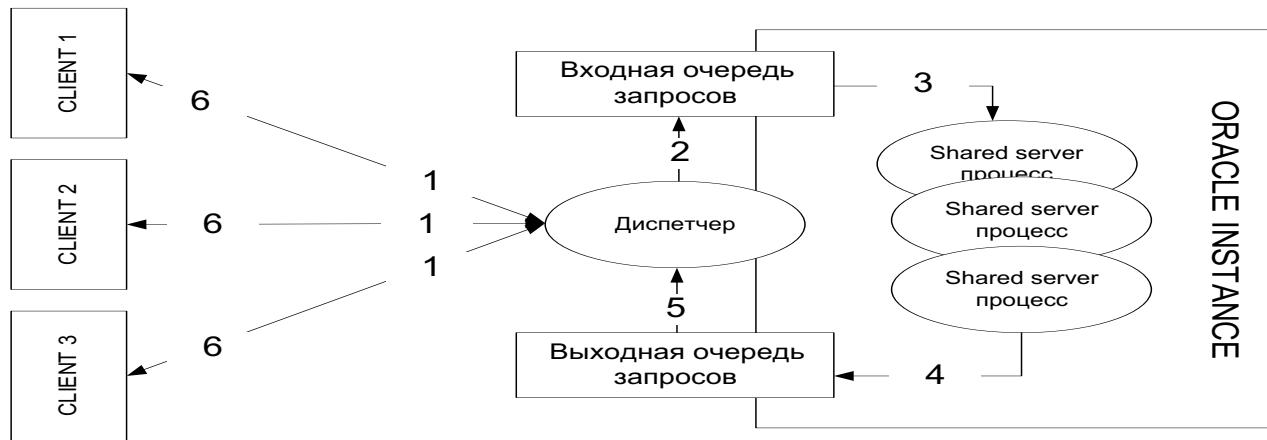
# Режим выделенного сервера

- ▶ **Dedicated server** (режим по умолчанию) – для каждого клиента выделяется отдельный выделенный серверный процесс (обработчик запросов, dedicated server process) который называется shadow process (теневой процесс).



# Режим разделяемого сервера

- ▶ **Shared server** – обрабатывает программа **dispatcher**:
- ▶ 1) получает запрос от клиента,
- ▶ 2) помещает их во входную очередь к разделяемым серверам;
- ▶ 3) незанятый разделяемый сервер извлекает и обрабатывает запрос;
- ▶ 4) после обработки разделяемый сервер помещает результат обработки в выходную очередь;
- ▶ 5) из очереди результат извлекает диспетчер;
- ▶ 6) диспетчер пересыпает результат клиенту



# Режимы соединений

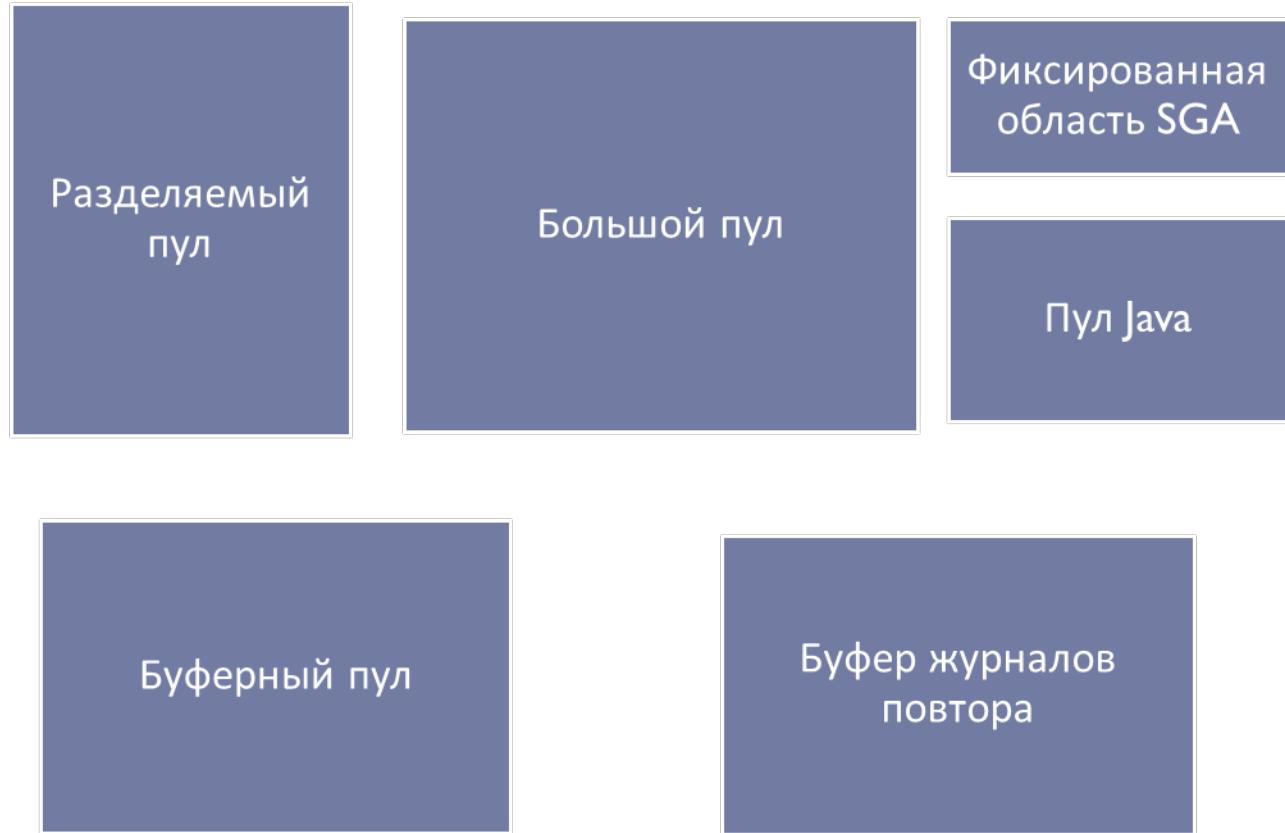
```
select username, service_name, server from v$session where username is not null;
```

Query Result SQL | All Rows Fetched: 6 in 0 seconds

USERNAME	SERVICE_NAME	SERVER
1 SYS	pdb_a.be.by	DEDICATED
2 SYS	SYS\$USERS	DEDICATED
3 SYSTEM	pdb_a.be.by	DEDICATED
4 SYS	pdb_a.be.by	DEDICATED
5 C##USER_1	SYS\$USERS	DEDICATED
6 C##USER_1	pdb_a.be.by	DEDICATED

# Структура SGA

---



# Основные пулы области SGA

---

- ▶ Java pool
- ▶ Large pool
- ▶ Shared pool
- ▶ Streams pool
- ▶ “Null” pool



# Структура оперативной памяти Oracle 12c

```
SQL> startup open;
ORACLE instance started.

Total System Global Area 1670221824 bytes
Fixed Size          2403352 bytes
Variable Size        1006633960 bytes
Database Buffers     654311424 bytes
Redo Buffers          6873088 bytes
Database mounted.
Database opened.
```

Query Result x

SQL | All Rows Fetched: 4 in 0 seconds

NAME	VALUE	CON_ID
1 Fixed Size	2403352	0
2 Variable Size	1056965608	0
3 Database Buffers	603979776	0
4 Redo Buffers	6873088	0

Query Result x

SQL | All Rows Fetched: 1 in 0 seconds

SUM(VALUE)
1 1670221824

# Представления словаря

---

- ▶ v\$sga
- ▶ V\$sgastat
- ▶ v\$sga\_dynamic\_components
- ▶ v\$sga\_dynamic\_free\_memory



# Гранулы – granule

---

- ▶ Память различным пулам в SGA выделяется блоками, которые называются гранулами
- ▶ Одна гранула (granule) - это область памяти размером 4, 8 или 16 Мбайт
- ▶ Гранула является наименьшей единицей выделения памяти



# Представления словаря

```
select
    component,
    current_size,
    max_size,
    last_oper_mode,
    last_oper_time,
    granule_size,
    current_size/granule_size as Ratio
from v$sga_dynamic_components
where current_size > 0;
```

Query Result x

SQL | All Rows Fetched: 5 in 0 seconds

COMPONENT	CURRENT_SIZE	MAX_SIZE	LAST_OPER_MODE	LAST_OPER_TIME	GRANULE_SIZE	RATIO
1 shared pool	419430400	419430400	DEFERRED	04.10.16	16777216	25
2 large pool	33554432	150994944	DEFERRED	27.09.16	16777216	2
3 java pool	16777216	16777216	(null)	(null)	16777216	1
4 DEFAULT buffer cache	553648128	771751936	DEFERRED	04.10.16	16777216	33
5 Shared IO Pool	50331648	50331648	IMMEDIATE	27.09.16	16777216	3

# Представления словаря

```
select
sum(min_size),
sum(max_size),
sum(current_size)
from v$sga_dynamic_components;
```

Query Result

SQL | All Rows Fetched: 1 in 0,015 seconds

	SUM(MIN_SIZE)	SUM(MAX_SIZE)	SUM(CURRENT_SIZE)
1	855638016	1409286144	1073741824

```
select current_size from v$sga_dynamic_free_memory;
```

Query Result

SQL | All Rows Fetched: 1 in 0 seconds

	CURRENT_SIZE
1	587202560

# Параметры SGA

- ▶ **SGA\_MAX\_SIZE** – указывает максимальный размер памяти

## SGA\_MAX\_SIZE

Property	Description
Parameter type	Big integer
Syntax	<code>SGA_MAX_SIZE = integer [K   M   G]</code>
Default value	Initial size of SGA at startup, dependent on the sizes of different pools in the SGA, such as buffer cache, shared pool, large pool, and so on.
Modifiable	No
Range of values	0 to operating system-dependent

# Параметры SGA

- ▶ SGA\_TARGET – указывает текущий (возможный) размер памяти

## SGA\_TARGET

Property	Description
Parameter type	Big integer
Syntax	SGA_TARGET = <code>integer [K   M   G]</code>
Default value	0 (SGA autotuning is disabled for DEFERRED mode autotuning requests, but allowed for IMMEDIATE mode autotuning requests)
Modifiable	<code>ALTER SYSTEM</code>
Range of values	64 MB to operating system-dependent
Basic	Yes

# Параметры SGA – изменение

```
SQL> alter system set sga_target=2G;
alter system set sga_target=2G
*
ERROR at line 1:
ORA-02097: parameter cannot be modified because specified value is invalid
ORA-00823: Specified value of sga_target greater than sga_max_size
```

```
SQL>
SQL>
SQL> show parameter sga
```

NAME	TYPE	VALUE
lock_sga	boolean	FALSE
pre_page_sga	boolean	TRUE
sga_max_size	big integer	1648M
sga_target	big integer	0
unified_audit_sga_queue_size	integer	1048576

```
SQL>
SQL>
SQL> alter system set sga_target=128m;
```

```
System altered.
```

```
SQL> show parameter sga
```

NAME	TYPE	VALUE
lock_sga	boolean	FALSE
pre_page_sga	boolean	TRUE
sga_max_size	big integer	1648M
sga_target	big integer	128M
unified_audit_sga_queue_size	integer	1048576

```
select sum(bytes) from v$sgastat;
SUM(BYTES)
1267342504
```

# Структура SGA



# Буферный пул

---

- ▶ Буферный пул (буферный кэш) – область SGA, которая содержит образы блоков, считанные из файлов данных или созданные динамически, чтобы реализовать модель согласованного чтения
- ▶ Совместно используется всеми пользователями



# Буферный пул

---

- ▶ Пользовательский процесс БД требует определенный фрагмент данных
- ▶ Поиск данных в буферном пуле
- ▶ Если данные обнаружены, то их можно считать прямо из памяти
- ▶ Если данные не обнаружены, то придется скопировать блок данных из файла данных на диске в буфер, чтобы получить доступ к данным



# Буферный пул

---

- ▶ Буферный пул (буферный кэш) хранит блоки данных табличных пространств
- ▶ Блок – единица обмена информацией между оперативной памятью и диском
- ▶ С каждым блоком связан счетчик использования
- ▶ Списки блоков упорядочены по количеству обращений к блоку



# Буферный пул

---

- ▶ Поддерживается два списка блоков:
  - ▶ список грязных блоков (отличаются от своей копии на диске и должны быть записаны в табличное пространство)
  - ▶ список чистых блоков (не измененные блоки)



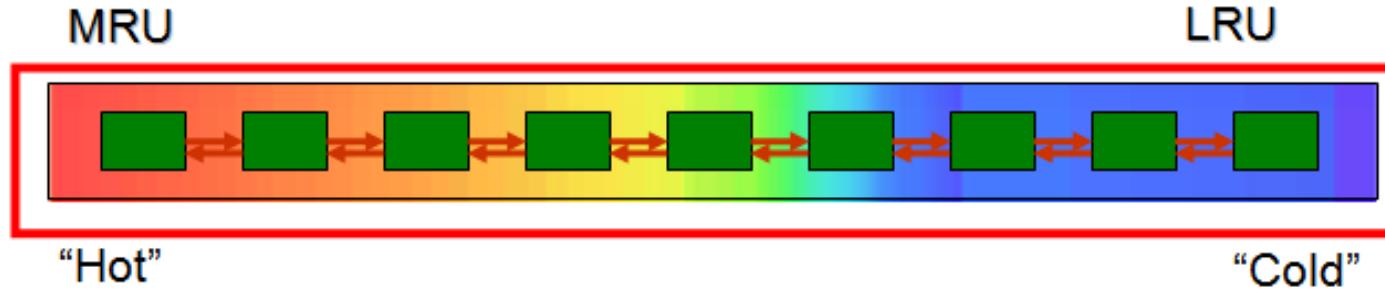
# Буферный пул

---

- ▶ Алгоритм LRU (least recently used) – первыми вытесняются блоки с наименьшим значением счетчика
- ▶ Запись грязных блоков на диск осуществляется в 4x случаях:
  - ▶ 1) истечение тайм-аута (3 сек);
  - ▶ 2) контрольная точка;
  - ▶ 3) превышение длины грязных блоков заданного лимита;
  - ▶ 4) процесс не может обнаружить свободный блок.



# Буферный пул – вытеснение блоков

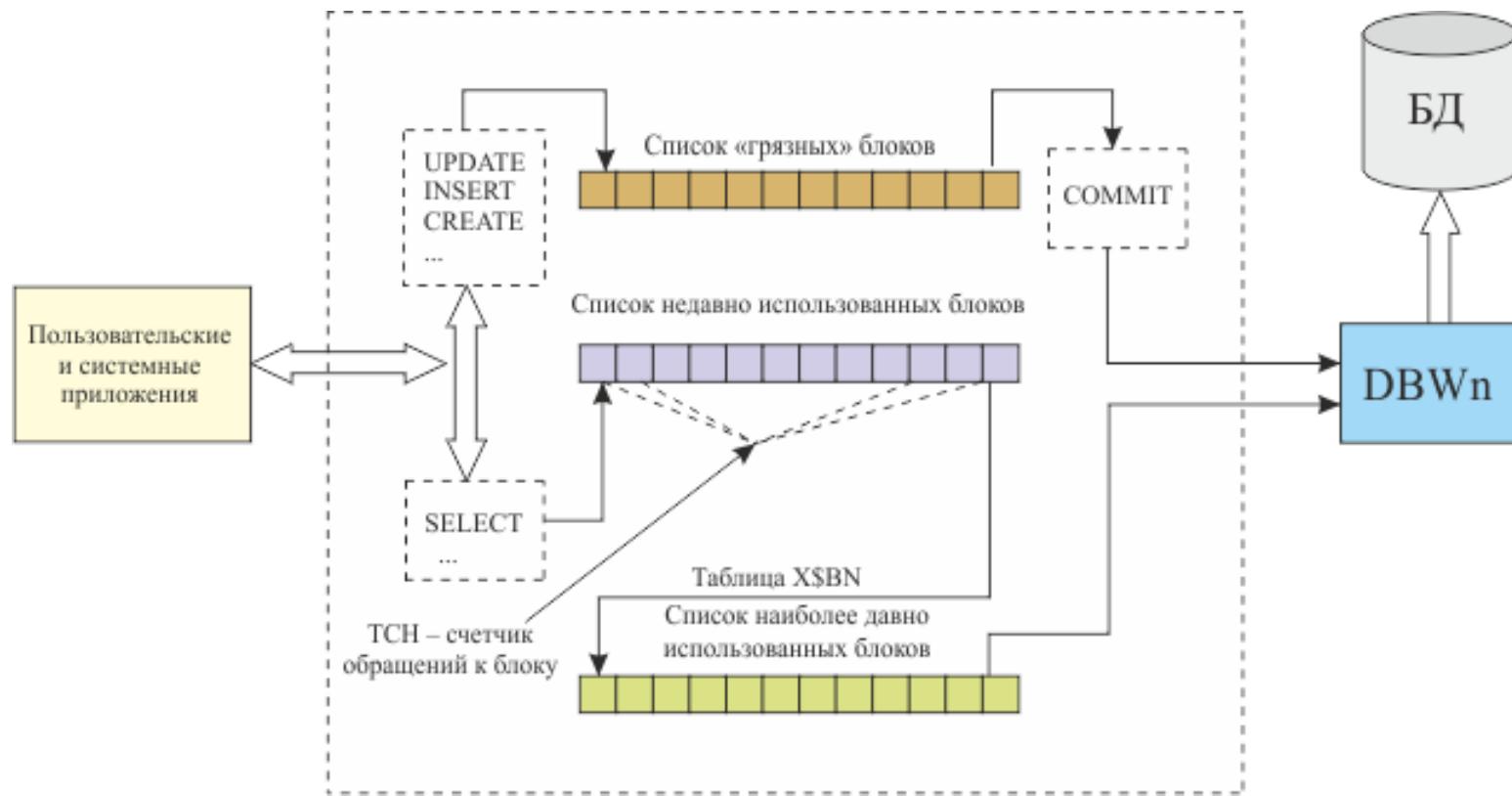


LRU = Least Recently Used

MRU = Most Recently Used



# Буферный пул – вытеснение блоков



# Пулы буферного кэша

- ▶ DEFAULT
- ▶ KEEP
- ▶ RECYCLE

```
select
component,
min_size,
current_size
from v$sga_dynamic_components;
```

Query Result | All Rows Fetched: 15 in 0 seconds

COMPONENT	MIN_SIZE	CURRENT_SIZE
1 shared pool	251658240	419430400
2 large pool	33554432	33554432
3 java pool	16777216	16777216
4 streams pool	0	0
5 DEFAULT buffer cache	553648128	553648128
6 KEEP buffer cache	0	0
7 RECYCLE buffer cache	0	0
8 DEFAULT 2K buffer cache	0	0
9 DEFAULT 4K buffer cache	0	0
10 DEFAULT 8K buffer cache	0	0
11 DEFAULT 16K buffer cache	0	0
12 DEFAULT 32K buffer cache	0	0
13 Shared IO Pool	0	50331648

# Параметры пулов буферного кэша

- ▶ Устанавливаются alter system set ...
- ▶ В файле параметров

## DB\_KEEP\_CACHE\_SIZE

Property	Description
Parameter type	Big integer
Syntax	DB_KEEP_CACHE_SIZE = <i>integer</i> [K   M   G]
Default value	0 (DB_KEEP_CACHE_SIZE is not configured by default)
Modifiable	ALTER SYSTEM
Range of values	Minimum: 0 (values greater than zero are automatically modified to be either the granule size * number of processor groups, or 4 MB * number of CPUs, whichever is greater) Maximum: operating system-dependent
Basic	No

## DB\_CACHE\_SIZE

Property	Description
Parameter type	Big integer
Syntax	DB_CACHE_SIZE = <i>integer</i> [K   M   G]
Default value	If SGA_TARGET is set: If the parameter is not specified, then the default is 0 (internally determined by the Oracle Database). If the parameter is specified, then the user-specified value indicates a minimum value for the memory pool. If SGA_TARGET is not set, then the default is either 48 MB or 4 MB * number of CPUs, whichever is greater
Modifiable	ALTER SYSTEM
Basic	No

## DB\_RECYCLE\_CACHE\_SIZE

Property	Description
Parameter type	Big integer
Syntax	DB_RECYCLE_CACHE_SIZE = <i>integer</i> [K   M   G]
Default value	0 (DB_RECYCLE_CACHE_SIZE is not configured by default)
Modifiable	ALTER SYSTEM
Range of values	Minimum: 0 (values greater than zero are automatically modified to be either the granule size * number of processor groups, or 4 MB * number of CPUs, whichever is greater) Maximum: operating system-dependent
Basic	No

# Пулы буферного кэша

---

- ▶ DEFAULT
- ▶ KEEP
- ▶ RECYCLE

```
alter system set db_cache_size=600m scope=spfile;
alter system set db_keep_cache_size=100m scope=spfile;
alter system set db_recycle_cache_size=100m scope=spfile;
```



# Пулы буферного кэша

```
select component, min_size, current_size, max_size  
from v$sga_dynamic_components where component like '%cache%' ;
```

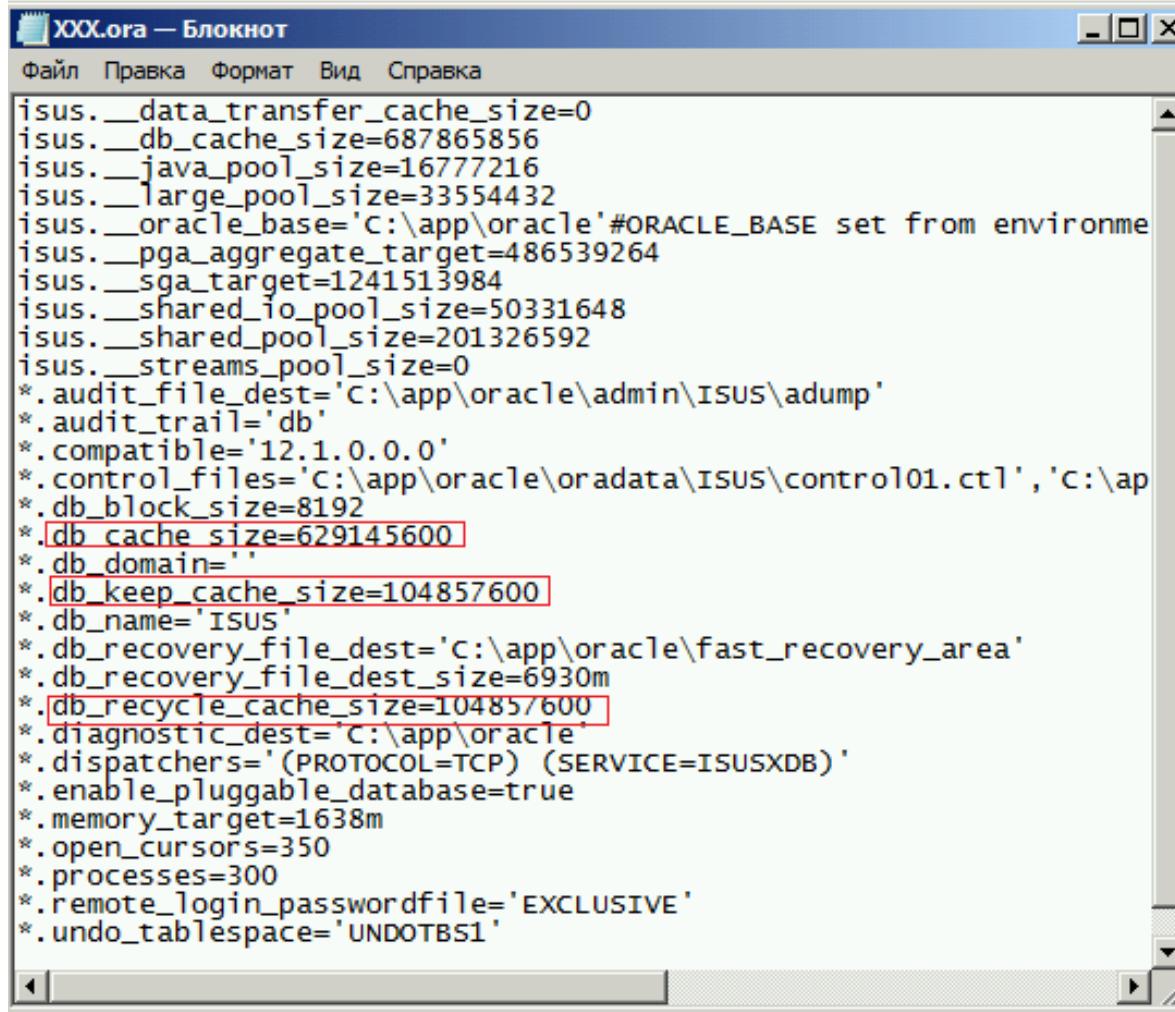
COMPONENT	MIN_SIZE	CURRENT_SIZE	MAX_SIZE
DEFAULT buffer cache	637534208	754974720	771751936
KEEP buffer cache	117440512	117440512	117440512
RECYCLE buffer cache	117440512	117440512	117440512
DEFAULT 2K buffer cache	0	0	0
DEFAULT 4K buffer cache	0	0	0
DEFAULT 8K buffer cache	0	0	0
DEFAULT 16K buffer cache	0	0	0
DEFAULT 32K buffer cache	0	0	0

```
select name, resize_state, block_size, buffers, prev_buffers from v$buffer_pool
```

NAME	RESIZE_STATE	BLOCK_SIZE	BUFFERS	PREV_BUFFERS
KEEP	STATIC	8192	13769	0
RECYCLE	STATIC	8192	13769	0
DEFAULT	STATIC	8192	80546	86548

# Пулы буферного кэша

```
create pfile='XXX.ora' from spfile;
```



XXX.ora – Блокнот

Файл Правка Формат Вид Справка

```
isus.__data_transfer_cache_size=0
isus.__db_cache_size=687865856
isus.__java_pool_size=16777216
isus.__large_pool_size=33554432
isus.__oracle_base='c:\app\oracle'#ORACLE_BASE set from environment
isus.__pga_aggregate_target=486539264
isus.__sga_target=1241513984
isus.__shared_io_pool_size=50331648
isus.__shared_pool_size=201326592
isus.__streams_pool_size=0
*.audit_file_dest='C:\app\oracle\admin\ISUS\adump'
*.audit_trail='db'
*.compatible='12.1.0.0.0'
*.control_files='c:\app\oracle\oradata\ISUS\control01.ctl','c:\ap
*.db_block_size=8192
*.db_cache_size=629145600
*.db_domain=''
*.db_keep_cache_size=104857600
*.db_name='ISUS'
*.db_recovery_file_dest='C:\app\oracle\fast_recovery_area'
*.db_recovery_file_dest_size=6930m
*.db_recycle_cache_size=104857600
*.diagnostic_dest='C:\app\oracle'
*.dispatchers='(PROTOCOL=TCP) (SERVICE=ISUSXDB)'
*.enable_pluggable_database=true
*.memory_target=1638m
*.open_cursors=350
*.processes=300
*.remote_login_passwordfile='EXCLUSIVE'
*.undo_tablespace='UNDOTBS1'
```

# Помещение таблицы в определенный пул

```
create table XXX (k int) storage(buffer_pool keep)    tablespace users;
create table YYY (k int) storage(buffer_pool recycle) tablespace users;
create table zzz (k int) storage(buffer_pool default) tablespace users; -- по умолчанию

insert into XXX values(1);
insert into YYY values(1);
insert into ZZZ values(1);
commit;

select segment_name, segment_type,tablespace_name, buffer_pool from user_segments
where segment_name in('XXX','YYY','ZZZ');
```

SEGMENT_NAME	SEGMENT_TYPE	TABLESPACE_NAME	BUFFER_POOL
XXX	TABLE	USERS	KEEP
YYY	TABLE	USERS	RECYCLE
ZZZ	TABLE	USERS	DEFAULT

# CACHE

CACHE – помещение таблицы в конец LRU-списка (для малых таблиц) обычно в default pool

```
create table CCC  (k int) cache    tablespace users;
create table CCC1 (k int) cache     storage(buffer_pool keep) tablespace users;
create table CCC2 (k int) cache     storage(buffer_pool recycle) tablespace users;
```

```
insert into CCC values(1);
insert into CCC1 values(1);
insert into CCC2 values(1);
commit;
```

```
select segment_name, segment_type,tablespace_name, buffer_pool from user_segments
where segment_name in('CCC', 'CCC1', 'CCC2');
```

SEGMENT_NAME	SEGMENT_TYPE	TABLESPACE_NAME	BUFFER_POOL
CCC	TABLE	USERS	DEFAULT
CCC1	TABLE	USERS	KEEP
CCC2	TABLE	USERS	RECYCLE

# Структура SGA

---



# Буфер журналов повторного выполнения

---

- ▶ Буфер журнала повторного выполнения предназначен для временного циклического хранения данных журнала повтора
- ▶ Позволяет ускорить работу сервера за счет буферизации
- ▶ Содержимое сбрасывается на диск (в журнал повтора) в 4-х случаях:
  - ▶ 1) каждые три секунды;
  - ▶ 2) при фиксации транзакции;
  - ▶ 3) при заполнении буфера на 1/3;
  - ▶ 4) если в буфере более 1м данных журнала повтора.



# Параметр LOG\_BUFFER

## LOG\_BUFFER

Property	Description
Parameter type	Integer
Default value	5 MB to 32 MB, depending on the size of the SGA, CPU count, and whether the operating system is 32-bit or 64-bit
Modifiable	No
Range of values	2 MB to operating system-dependent
Basic	No

```
show parameter log_buffer
```

NAME	TYPE	VALUE
log_buffer	integer	6344704



# Ожидание освобождения буфера журналов повтора

- ▶ Redo buffer allocation retries – статистика – количество случаев ожидания процессами освобождения буфера журнала повтора:
  - ▶ буфер занят
  - ▶ процесс LGWR не сбросил данные на диск
  - ▶ процесс ждет освобождения буфера для записи информации об изменении базы данных

Значение должно быть равно 0, иначе надо увеличивать размер буфера

select name, value from v\$sysstat where name like '%redo%retries'	
NAME	VALUE
redo buffer allocation retries	0



# Структура SGA



# Фиксированная область SGA

---

- ▶ хранит переменные, указывающие на другие области памяти, значения параметров;
- ▶ представляет собой загрузочный бинарный код;
- ▶ размер области зависит от платформы, версии операционной системы;
- ▶ размером фиксированной области SGA управлять нельзя.



# Структура SGA



# Разделяемый пул

---

- ▶ Библиотечный кэш
- ▶ Разделяемую область SQL
- ▶ Кэш словаря данных
- ▶ Управляющие структуры



# Поддержка словаря данных

---

- ▶ Кэш словаря данных – содержит данные в виде строк
  - ▶ Библиотечный кэш
- 
- ▶ Все пользовательские процессы Oracle совместно используют эти два кэша для доступа к информации о словаре данных



# Поддержка словаря данных

---

- ▶ Каждый SQL-оператор распознается
- ▶ Для нового SQL-оператора выделяется память в разделяемом пуле
- ▶ Когда два пользователя выполняют одно и то же предложение SQL, оператор используется повторно из разделяемой области SQL
- ▶ Аналогично обрабатываются программные блоки



# Разделяемый пул

```
select
  component,
  min_size,
  current_size,
  max_size
  from v$sga_dynamic_components
  where component = 'shared pool';
```

Query Result X

SQL | All Rows Fetched: 1 in 0 seconds

	COMPONENT	MIN_SIZE	CURRENT_SIZE	MAX_SIZE	
1	shared pool	251658240	419430400	419430400	

# Управление размером разделяемого пула

- ▶ Устанавливаются alter system set ...
- ▶ В файле параметров

```
SQL> show parameter shared_pool
NAME          TYPE        VALUE
shared_pool_reserved_size  big integer 9227468
shared_pool_size          big integer 0
SQL> -
```

## SHARED\_POOL\_SIZE

Property	Description
Parameter type	Big integer
Syntax	SHARED_POOL_SIZE = integer [K   M   G]
Default value	If SGA_TARGET is set: If the parameter is not specified, then the default is 0 (internally determined by the Oracle Database). If the parameter is specified, then the user-specified value indicates a minimum value for the memory pool. If SGA_TARGET is not set (32-bit platforms): 64 MB, rounded up to the nearest granule size.  If SGA_TARGET is not set (64-bit platforms): 128 MB, rounded up to the nearest granule size.
For considerations when dealing with database instances using ASM, see "SHARED_POOL_SIZE and Automatic Storage Management".	
Modifiable	ALTER SYSTEM
Range of values	Minimum: the granule size Maximum: operating system-dependent

```
SQL>
SQL> alter system set shared_pool_size=180m;
System altered.
```

```
SQL> show parameter shared_pool
```

```
NAME          TYPE        VALUE
shared_pool_reserved_size  big integer 9227468
shared_pool_size          big integer 192M
SQL>
```



# Управление размером разделяемого пула

```
SQL> alter system set shared_pool_size=208m;
```

```
System altered.
```

```
SQL> show parameter shared_pool
```

NAME	TYPE	VALUE
shared_pool_reserved_size	big integer	9227468
shared_pool_size	big integer	208M

```
SQL> _
```

```
select component, min_size, current_size, max_size  
from v$sga_dynamic_components where component = 'shared pool' ;
```

COMPONENT	MIN_SIZE	CURRENT_SIZE	MAX_SIZE
shared pool	167772160	218103808	218103808



# Объекты в разделяемом пуле

```
select pool, name, bytes from v$sgastat where pool = 'shared pool';
```

POOL	NAME	BYTES
1	shared pool db_files	453448
2	shared pool KCFIS SGA	8
3	shared pool KTC txn rsrc cnt	88
4	shared pool os statistics	13096
5	shared pool kdlxdup swapp	24
6	shared pool KGLH0	10407544
7	shared pool KQF optimizer stats table	2224
8	shared pool namrec_kfdsg	24
9	shared pool object temp hash table la	163840
.0	shared pool keomm:emptyMon_keommSg	7168
.1	shared pool object level stat table	1056
.2	shared pool kgnfs gid list	104
.3	shared pool HM_MESSAGE	1504
.4	shared pool PQ/ResSched	32784
.5	shared pool generic process shared st	16
.6	shared pool KTSJ state object	1408
.7	shared pool SGA structure for kelr	2456
.8	shared pool KKJ SHRD WRQS	240
.9	shared pool zafwctx	40608
.10	shared pool z1llab Group Tree Heap De	160
.11	shared pool wewr sra	160

```
select count(*) from v$sgastat  
where pool = 'shared pool';
```

Query Result x

SQL | All Rows Fetched: 1 in 0 seconds

COUNT(*)
1 1201

# Структура SGA



# Большой пул

---

- ▶ Большой пул - область памяти SGA, применяемая для хранения больших фрагментов памяти
- ▶ В этой области не применяется вытеснение по алгоритму LRU
- ▶ память становится свободной сразу после того, как перестает использоваться
- ▶ аналог RECYCLE (разделяемый пул – KEEP)
- ▶ хранятся данные при резервном копировании (RMAN), специальные области UGA и пр.



# Большой пул

```
select
pool,
name,
bytes
from v$sgastat
where pool = 'large pool';
```

Query Result x

SQL | All Rows Fetched: 10 in 0 seconds

POOL	NAME	BYTES
1 large pool	PX msg pool	491520
2 large pool	SWRF Metric Eidbuf	368640
3 large pool	SWRF Metric CHBs	1802240
4 large pool	SWRF Metric Eidbuf	368640
5 large pool	SWRF Metric CHBs	1802240
6 large pool	SWRF Metric CHBs	1802240
7 large pool	free memory	24379392
8 large pool	SWRF Metric CHBs	1802240
9 large pool	SWRF Metric Eidbuf	368640
10 large pool	SWRF Metric Eidbuf	368640

```
select
component,
min_size,
current_size,
max_size
from v$sga_dynamic_components
where component = 'large pool';
```

Query Result x

SQL | All Rows Fetched: 1 in 0 seconds

COMPONENT	MIN_SIZE	CURRENT_SIZE	MAX_SIZE
1 large pool	33554432	33554432	150994944

# Большой пул – параметры

## LARGE\_POOL\_SIZE

Property	Description
Parameter type	Big integer
Syntax	<code>LARGE_POOL_SIZE = integer [K   M   G]</code>
Default value	If <code>SGA_TARGET</code> is set, but a value is not specified for <code>LARGE_POOL_SIZE</code> , then the default is 0 (internally determined by the Oracle database). If <code>LARGE_POOL_SIZE</code> is specified, then the user-specified value indicates a minimum value for the memory pool. If <code>SGA_TARGET</code> is not set, 0 if both of the following are true: <ul style="list-style-type: none"><li>• The pool is not required by parallel execution</li><li>• <code>DBWR_IO_SLAVES</code> is not set</li></ul> Otherwise, derived from the values of <code>PARALLEL_MAX_SERVERS</code> , <code>PARALLEL_THREADS_PER_CPU</code> , <code>CLUSTER_DATABASE_INSTANCES</code> , <code>DISPATCHERS</code> , and <code>DBWR_IO_SLAVES</code> .
Modifiable	<code>ALTER SYSTEM</code>
Range of values	Minimum: the granule size Maximum: operating system-dependent
Basic	No
<pre>show parameter large_pool</pre>	
NAME	TYPE VALUE
-----	-----
<code>large_pool_size</code>	big integer 0



# Большой пул – параметры

```
SQL>
SQL>
SQL> alter system set large_pool_size=2m;
System altered.

SQL> show parameter large_pool
NAME                      TYPE         VALUE
large_pool_size            big integer 16M
SQL> alter system set large_pool_size=18m;
System altered.

SQL> show parameter large_pool
NAME                      TYPE         VALUE
large_pool_size            big integer 32M
SQL> _
```

```
select component, min_size, current_size, max_size
from v$sga_dynamic_components where component = 'large pool' ;

```

COMPONENT	MIN_SIZE	CURRENT_SIZE	MAX_SIZE
large pool	33554432	33554432	150994944

# Структура SGA

---



# Java-пул

- ▶ Java-пул предназначен для работы Java-машины;

```
select
component,
min_size,
current_size,
max_size
from v$sga_dynamic_components
where component = 'java pool';
```

Query Result x

All Rows Fetched: 1 in 0 seconds

COMPONENT	MIN_SIZE	CURRENT_SIZE	MAX_SIZE
1 java pool	16777216	16777216	16777216

```
select
pool,
name,
bytes
from v$sgastat
where pool = 'java pool';
```

Query Result x

All Rows Fetched: 1 in 0 seconds

POOL	NAME	BYTES
1 java pool	free memory	16777216

# Параметры Java-пула

## JAVA\_POOL\_SIZE

Property	Description
Parameter type	Big integer
Syntax	JAVA_POOL_SIZE = <i>integer</i> [K   M   G]
Default value	If SGA_TARGET is set: If the parameter is not specified, then the default is 0 (internally determined by the Oracle Database). If the parameter is specified, then the user-specified value indicates a minimum value for the memory pool. If SGA_TARGET is not set: 24 MB, rounded up to the nearest granule size
Modifiable	ALTER SYSTEM
Range of values	Minimum: 0 (values greater than zero are rounded up to the nearest granule size) Maximum: operating system-dependent
Basic	No

```
show parameter java_pool_size
NAME                                     TYPE        VALUE
java_pool_size                           big integer 0
```

```
SQL> alter system set java_pool_size=18m;
System altered.

SQL> show parameter java_pool
NAME                                     TYPE        VALUE
java_pool_size                           big integer 32M
SQL>
SQL>
```



# Вопросы?

---

