



SQL PLAN BASELINE

ADVANCED COMPRESSION

OPTIMIZER FEATURES

REAL APPLICATION TESTING

RESULT CACHE

IN MEMORY DATABASE CACHE

ORACLE VM

ACTIVE DATA GUARD

SQL PERFORMANCE ANALYZER

TOTAL RECALL

PERFORMANCE

# ORACLE®

## Тестирование

Геннадий Сигалаев  
Oracle CIS

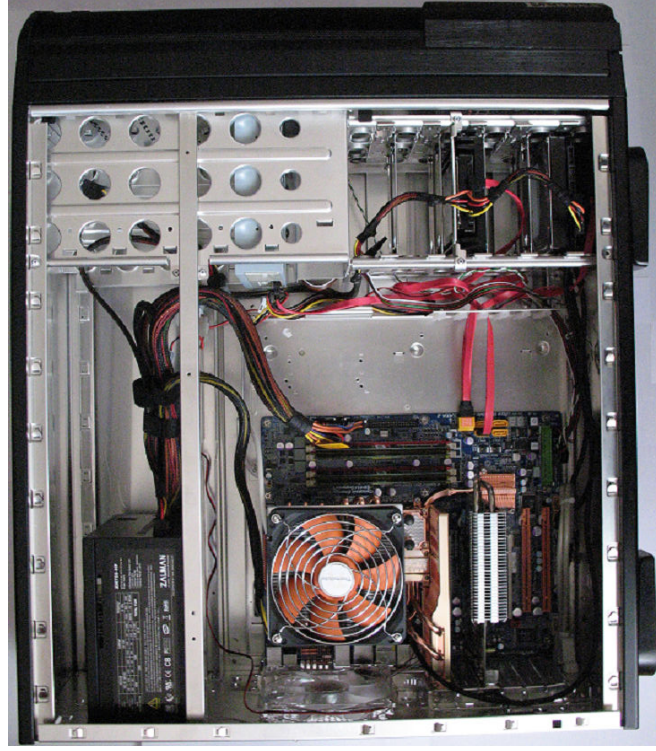


## Цель тестирования

- Определение средней скорости DML операций
- Определение средней скорости операции SELECT
- Влияние Durable commit на производительность
- Влияние Client/Server connection на производительность

# DB Options Home PC

## Тестовый стенд



- CPU: Intel Core 2 Duo CPU 2.33 ГГц
- RAM: 4 GB DDR2
- HDD 100 Gb SATA-II
- OC Fedora 8 x64 (Ядро Linux 2.6.23.1-42.fc6)
- Oracle TimesTen 7.0.5.0.0 (64 bit Linux)



# Data store

## *Тестовый сценарий*

```
[dsdemo]  
Driver=/app/oracle/product/7.0.5/TimesTen/tt705/lib/libtten.so  
DataStore=/home/oracle/testtimesten/datastore/dsdemo  
DatabaseCharacterSet=CL8MSWIN1251  
PermSize=3100  
TempSize=200  
UID=tttest  
PWD=tttest  
...
```

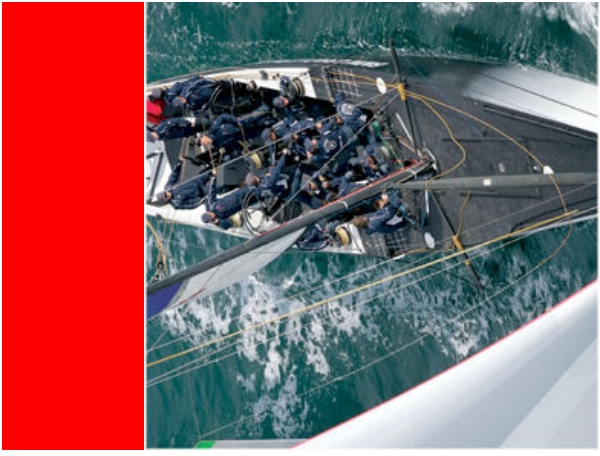
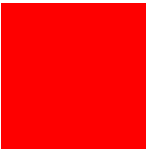


# Data store

## *Тестовый скрипт*

```
...
String query = "INSERT INTO ACCOUNTS (id, tel_no, plan_id, amount, status)
               VALUES (accounts_seq.nextval,?, ?, ?, ?)";

DriverManager.registerDriver(new com.timesten.jdbc.TimesTenDriver());
conn = DriverManager.getConnection( "jdbc:timesten:direct:server_tt" , "tttest", "tttest");
//   conn.setAutoCommit(false);
CallableStatement stmt = conn.prepareCall(query);
start = System.currentTimeMillis();
for (count= 1 ; count<=10000; count++) {
    stmt.setString(1,Integer.toString(count));
    ...
        stmt.execute();
    }
//   conn.commit();
end = System.currentTimeMillis();
...
```



**INSERT**



# INSERT

- DIRECT CONNECTION
  - DURABLE COMMIT = OFF
  - DURABLE COMMIT = ON
- CLIENT/SERVER CONNECTION
  - DURABLE COMMIT = OFF
  - DURABLE COMMIT = ON



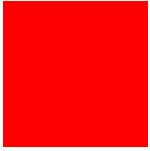
# DIRECT, durable commit = off

Rows	1 commit	every row commit	amount
10 000	0,234867	0,261467	30
50 000	1,024867	1,137533	30
100 000	1,999967	2,125667	30
500 000	12,6667	10,036467	15
1 000 000	32,4622	20,4118	10
30 000 000		603,8724	5



# DIRECT, durable commit = on

Rows	1 commit	every row commit	amount
10 000	0,252067	3,490367	30
50 000	1,1641	16,462033	30
100 000	2,370567	33,436233	30
500 000	13,343	165,05345	20
1 000 000	32,0661	327,0143	10



# C/S, durable commit = off

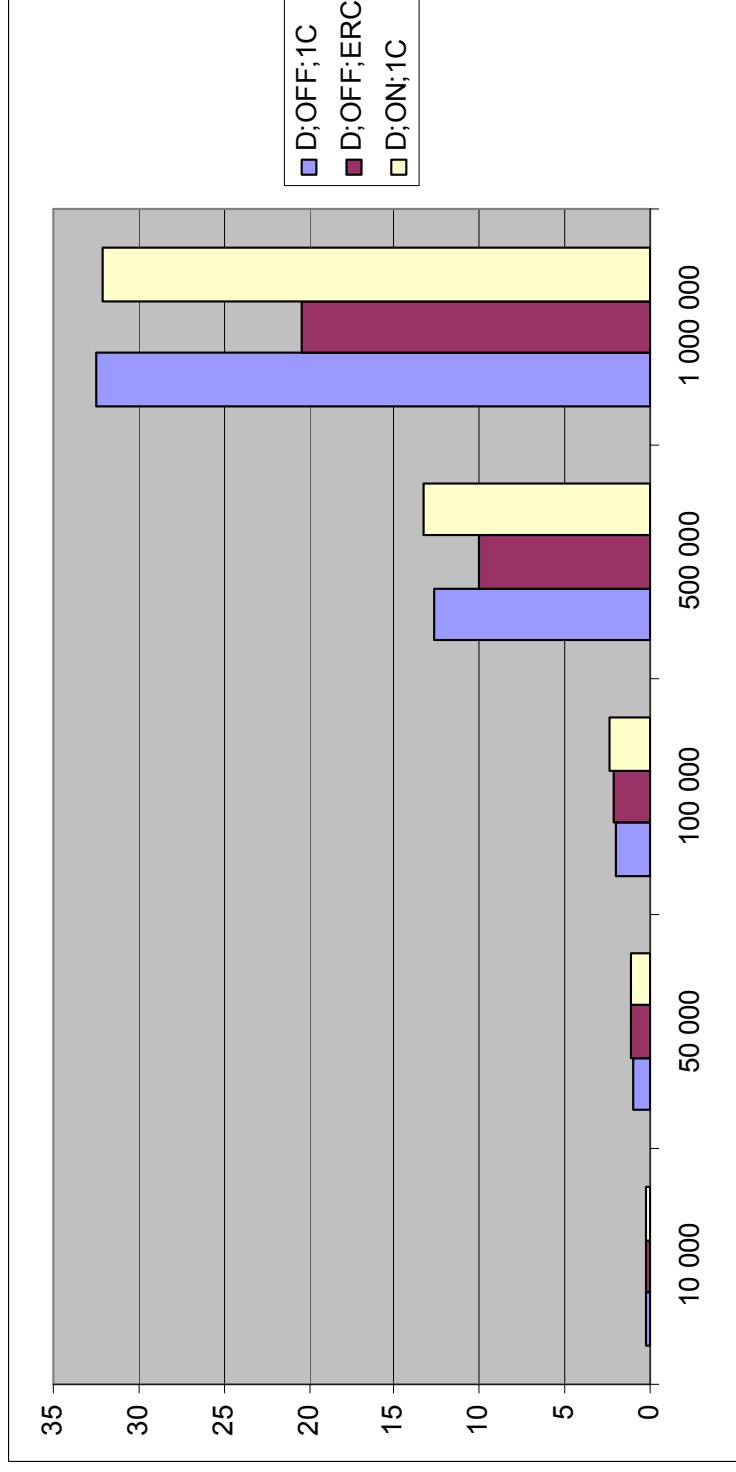
Rows	1 commit	every row commit	amount
10 000	4,9166	4,721367	30
50 000	21,035533	30,9239	30
100 000	46,38585	49,2521	30
500 000	317,4798	291,0406	15
1 000 000	475,7015	581,8797	10



# C/S, durable commit = on

Rows	1 commit	every row commit	amount
10 000	6,635867	10,2328	30
50 000	31,722433	49,9432	30
100 000	65,8609	95,5796	10
500 000	290,8516	455,9017	10
1 000 000	589,9516	1000,42426	10

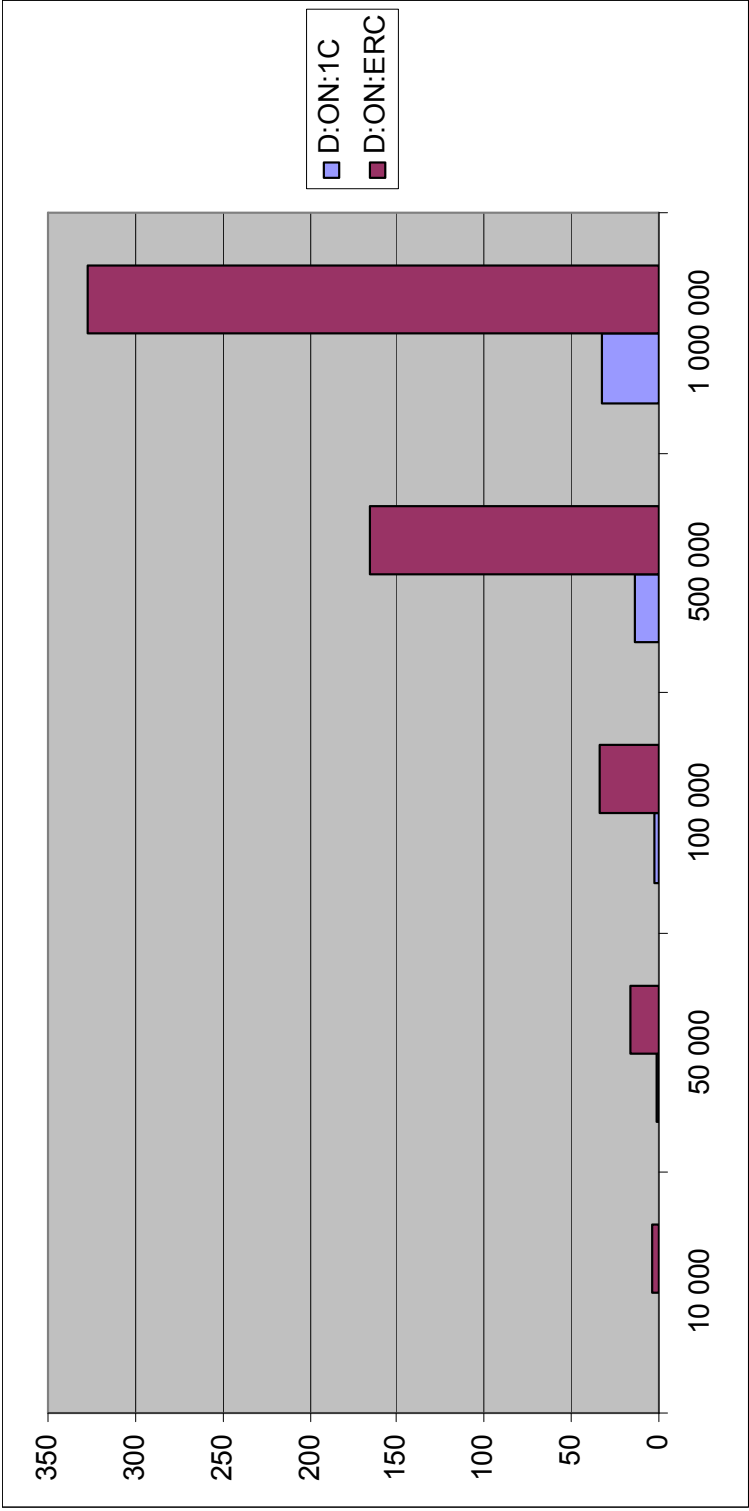
# ИТОГИ



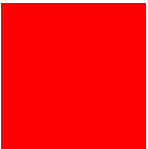
Влияние Durable commit при прямом подключении и пакетной фиксации транзакций практически не сказывается на производительности



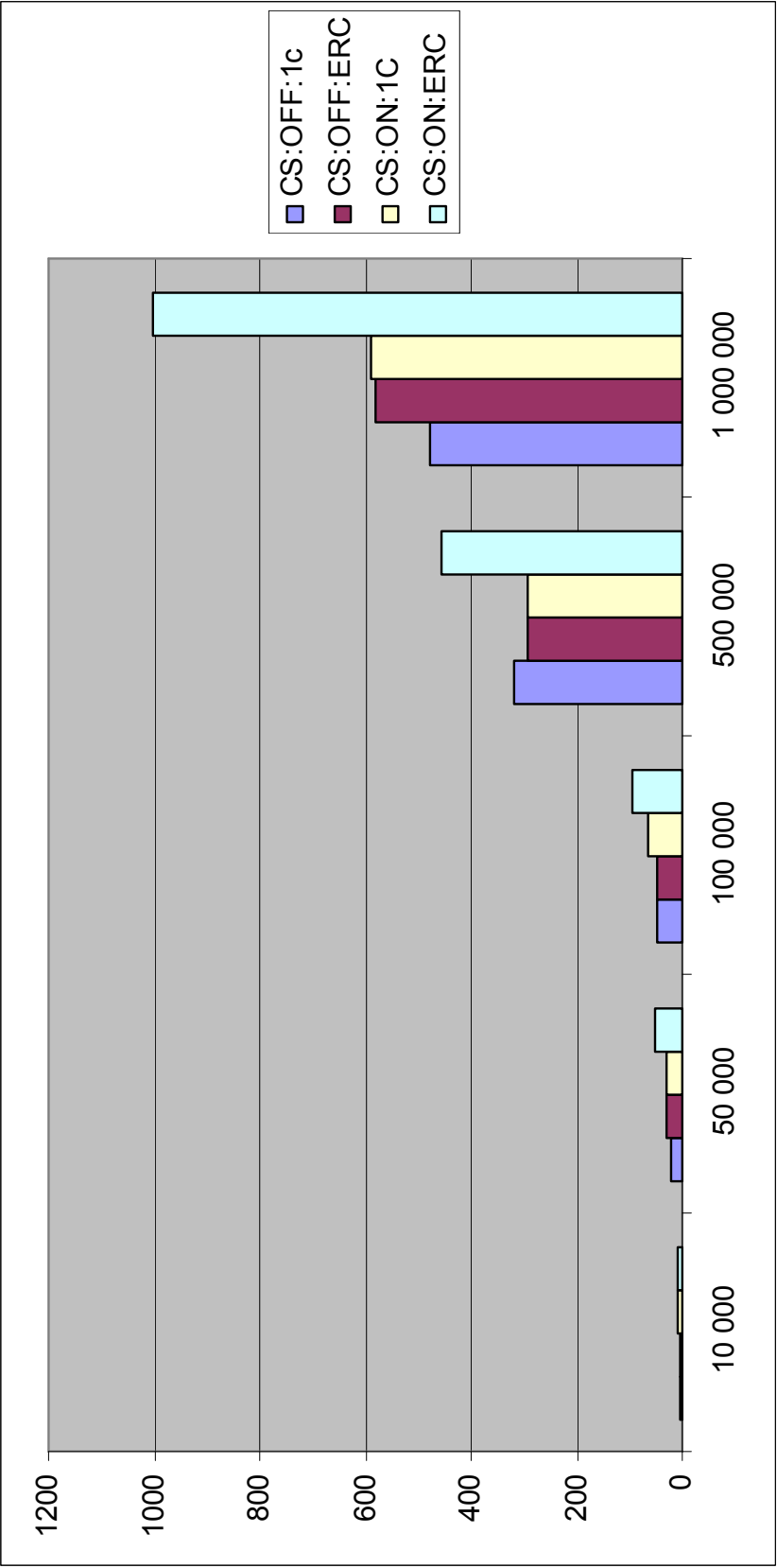
# ИТОГИ



Влияние Durable commit при прямом подключении и фиксации транзакции после каждой операции КАТАСТРОФИЧЕСКИ сказывается на производительности

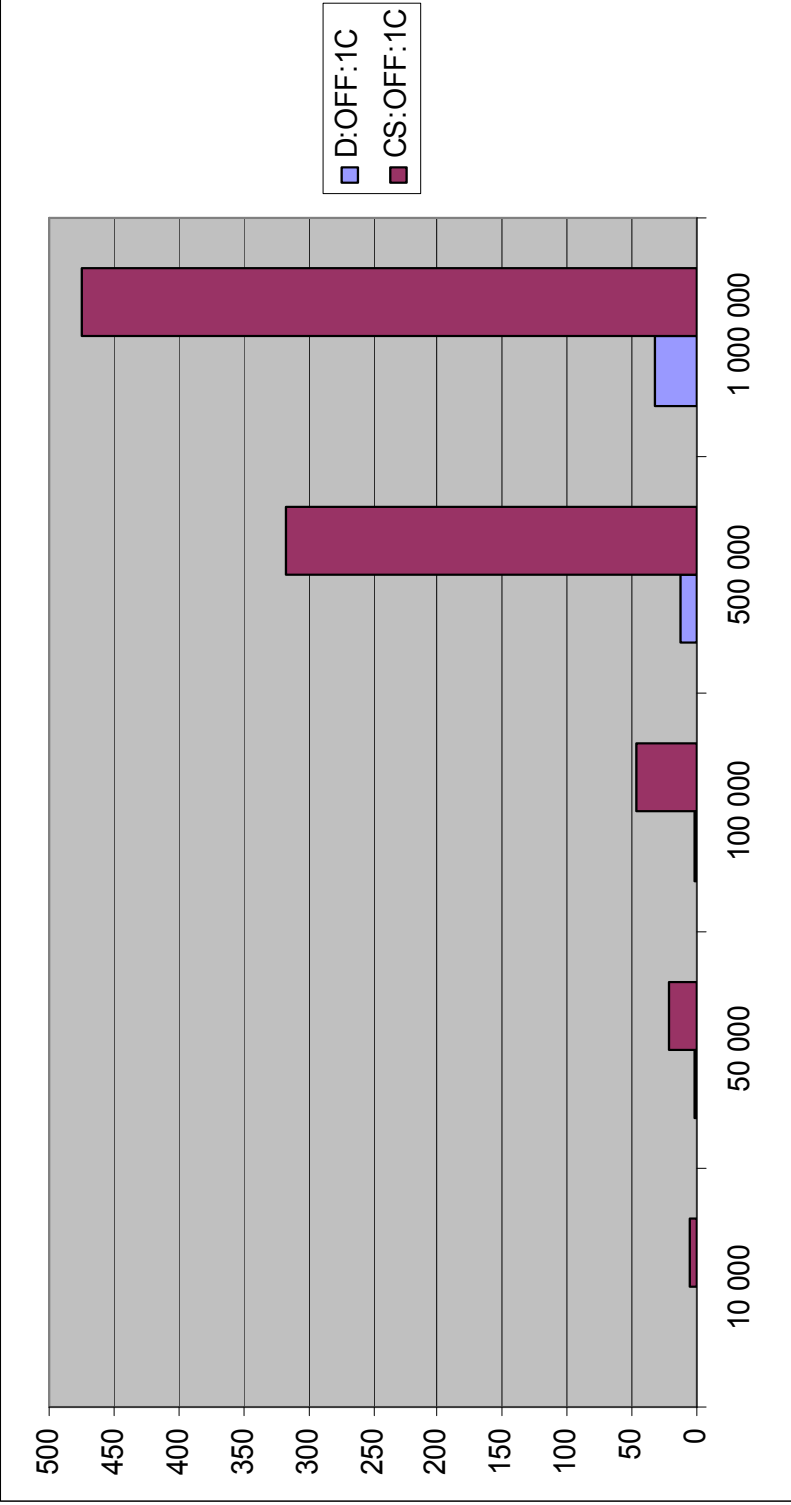


# ИТОГИ

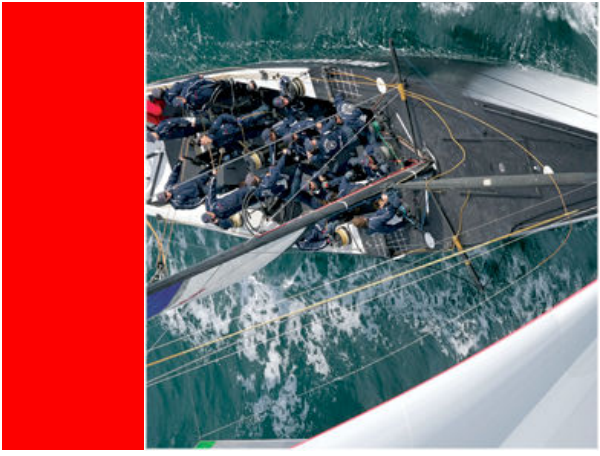
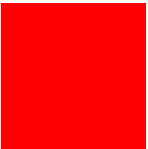




# ИТОГИ



При подключении Client/Server, достаточно сильно теряем в производительности (~10 раз), но при этом очень незначительно нагружаем процессор



# SELECT



# SELECT

- Среднее время FULL SCAN (30 000 000)
- Среднее время доступа RowLkTreeScan
- Среднее время доступа RowLkHashScan



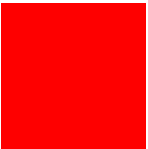
# SELECT (FULL SCAN)

```
Command> select count(*) from accounts;
```

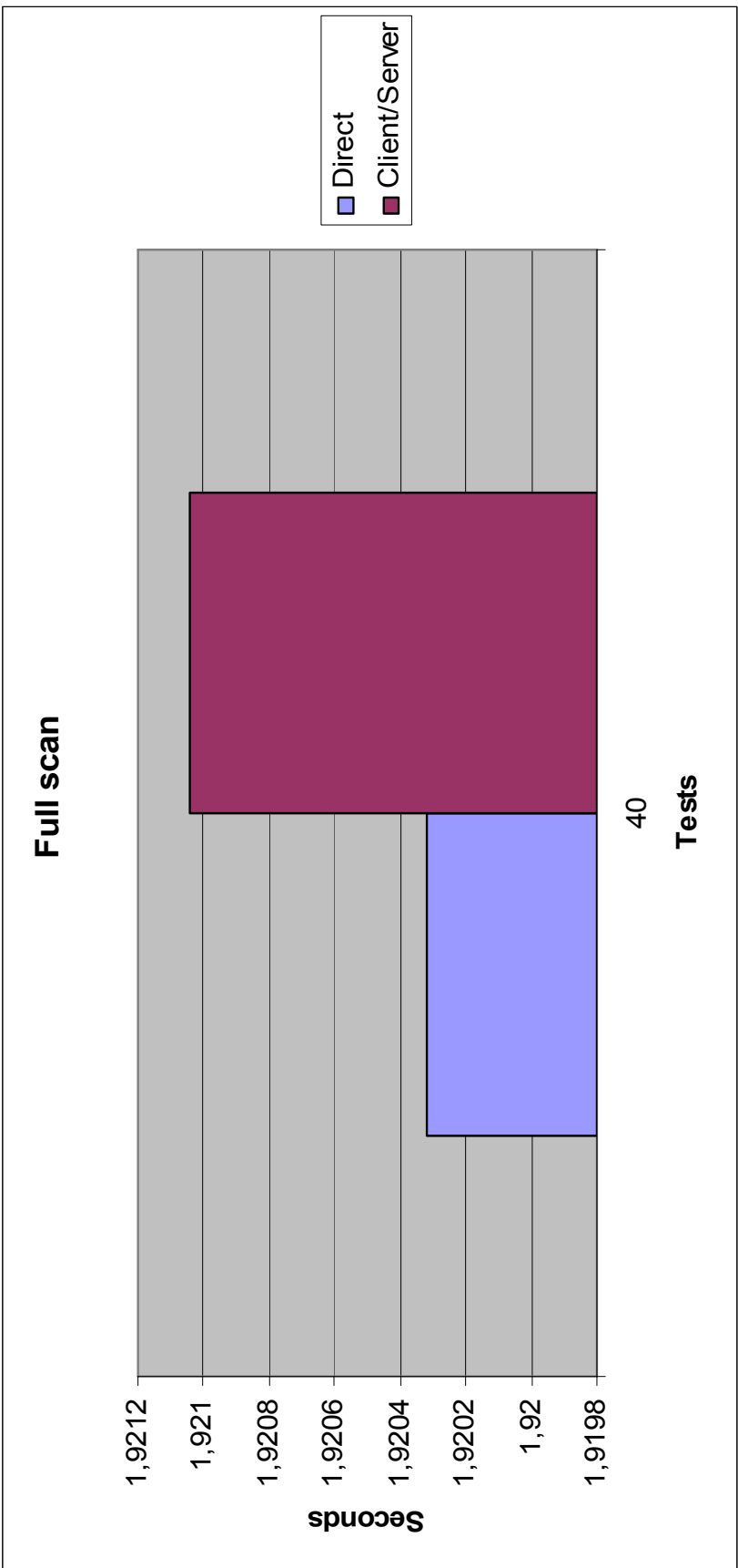
```
Query Optimizer Plan:
```

```
STEP:          1
LEVEL:         1
OPERATION:     TblLkSerialScan
TBLNAME:       ACCOUNTS
IXNAME:        <NULL>
INDEXED CONDITION: <NULL>
NOT INDEXED:   <NULL>
```

```
< 30000000 >
1 row found.
```



# SELECT (FULL SCAN 30 000 000)



AVG time (direct): 1,920321 sec.

AVG time (client/server): 1,921004 sec



# SELECT (RowLkTtreeScan)

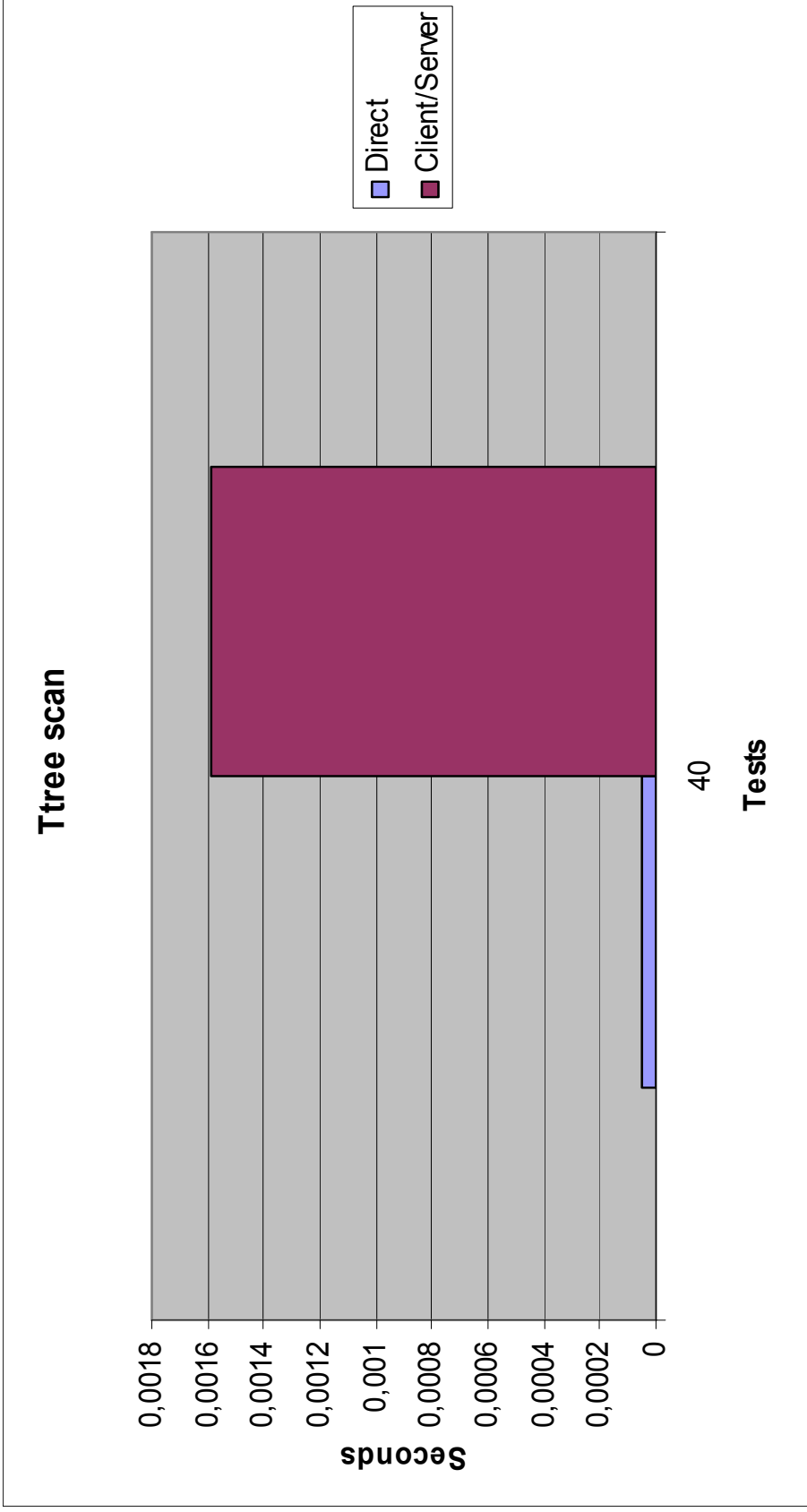
```
Command> select * from accounts where id = 100;
```

```
Query Optimizer Plan:
```

```
STEP:          1
LEVEL:         1
OPERATION:     RowLkTtreeScan
TBLNAME:      ACCOUNTS
IXNAME:       ACCOUNTS
INDEXED CONDITION:  ACCOUNTS.ID = 100
NOT INDEXED:  <NULL>

< 100, 100, 100, 100.000000000000, YES >
1 row found.
```

# SELECT (RowIdTreeScan)



AVG time (direct): 0,000047 sec.

AVG time (client/server): 0,001592 sec.



# SELECT (RowLkHashScan)

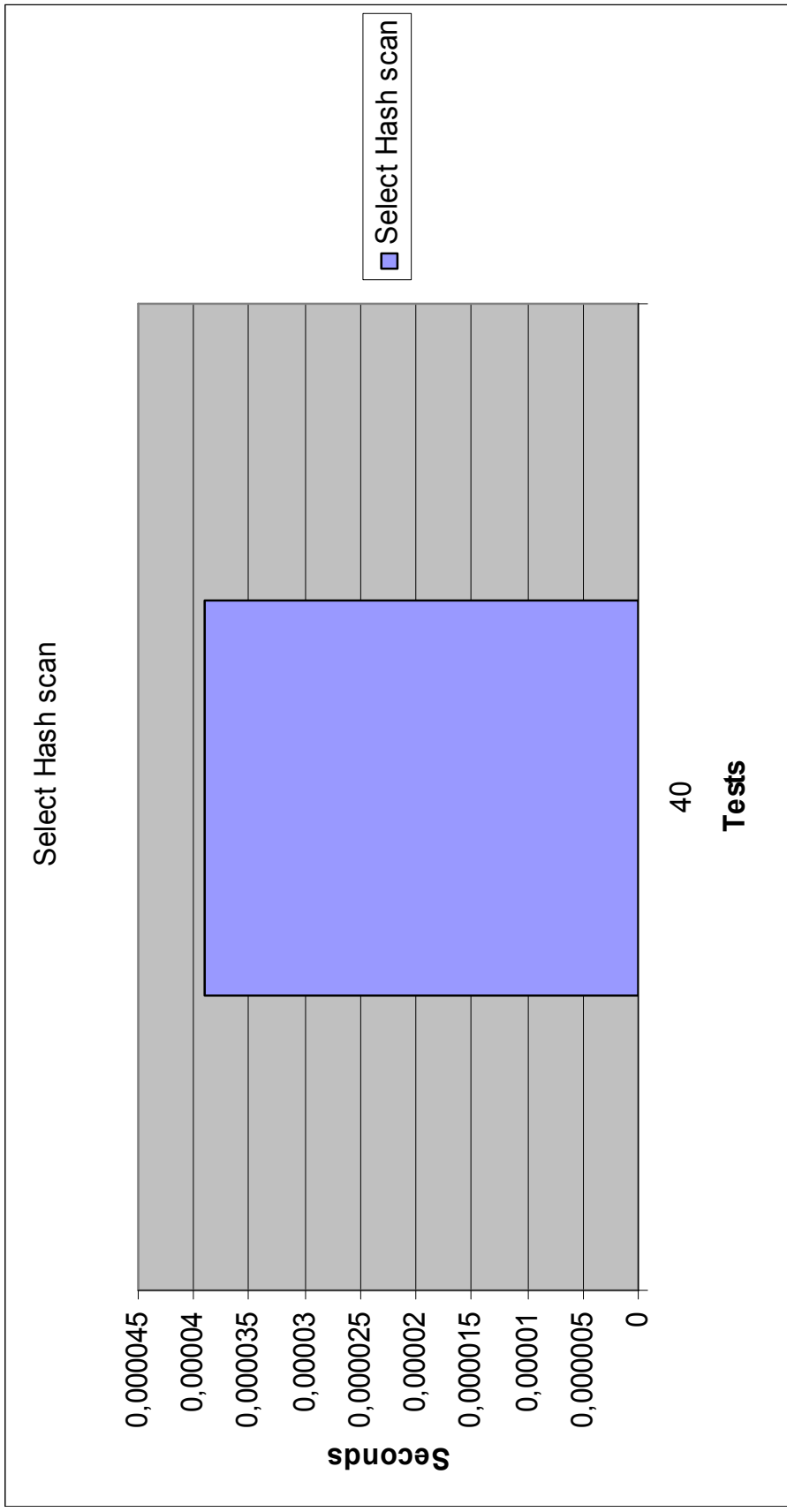
```
Command> select * from accounts where id = 100;
```

```
Query Optimizer Plan:
```

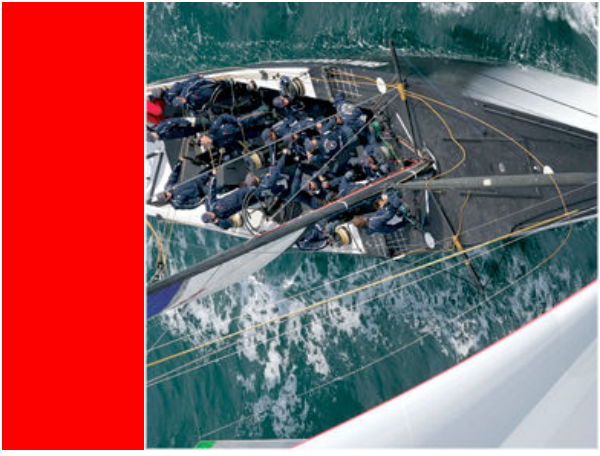
```
STEP:                1
LEVEL:               1
OPERATION:           RowLkHashScan
TBLNAME:             ACCOUNTS
IXNAME:              ACCOUNTS
INDEXED CONDITION:  ACCOUNTS.ID = 100
NOT INDEXED:        <NULL>

< 100, 100, 100, 100.000000000000, YES >
1 row found.
```

# SELECT (RowIdkHashScan)



AVG time: 0,000039 sec.



# BACKUP, EXP/IMP

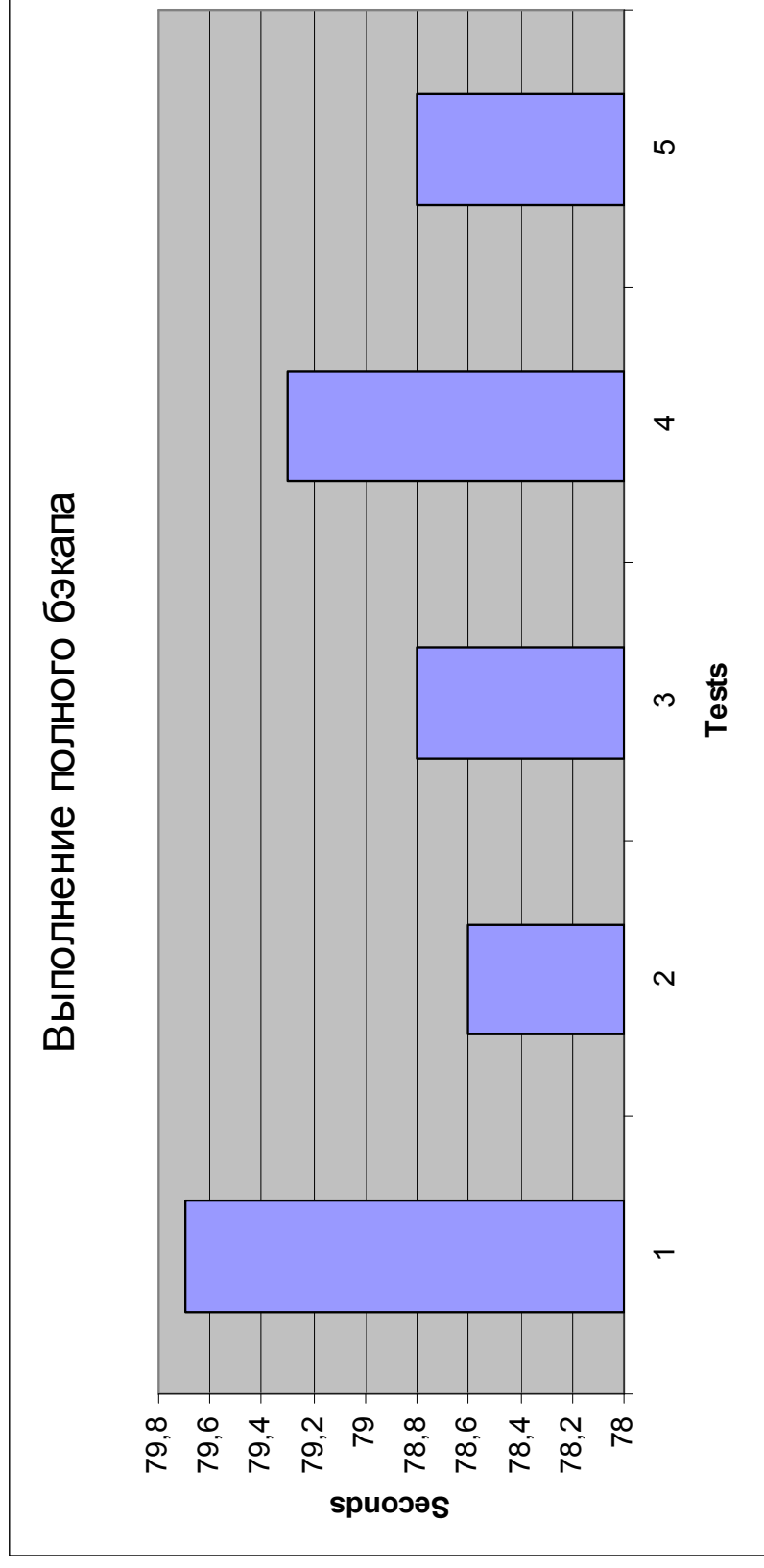
# Выполнение операции backup

- Использование утилиты `ttBackup`

```
Command> dssize;
```

```
PERM_ALLOCATED_SIZE:      3174400
PERM_IN_USE_SIZE:        2599036
PERM_IN_USE_HIGH_WATER:  2599036
TEMP_ALLOCATED_SIZE:     20480
TEMP_IN_USE_SIZE:        6523
TEMP_IN_USE_HIGH_WATER:  6523
```

# Выполнение операции backup



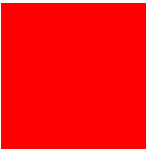
AVG time: 79,04 sec.

# Выполнение операции restore

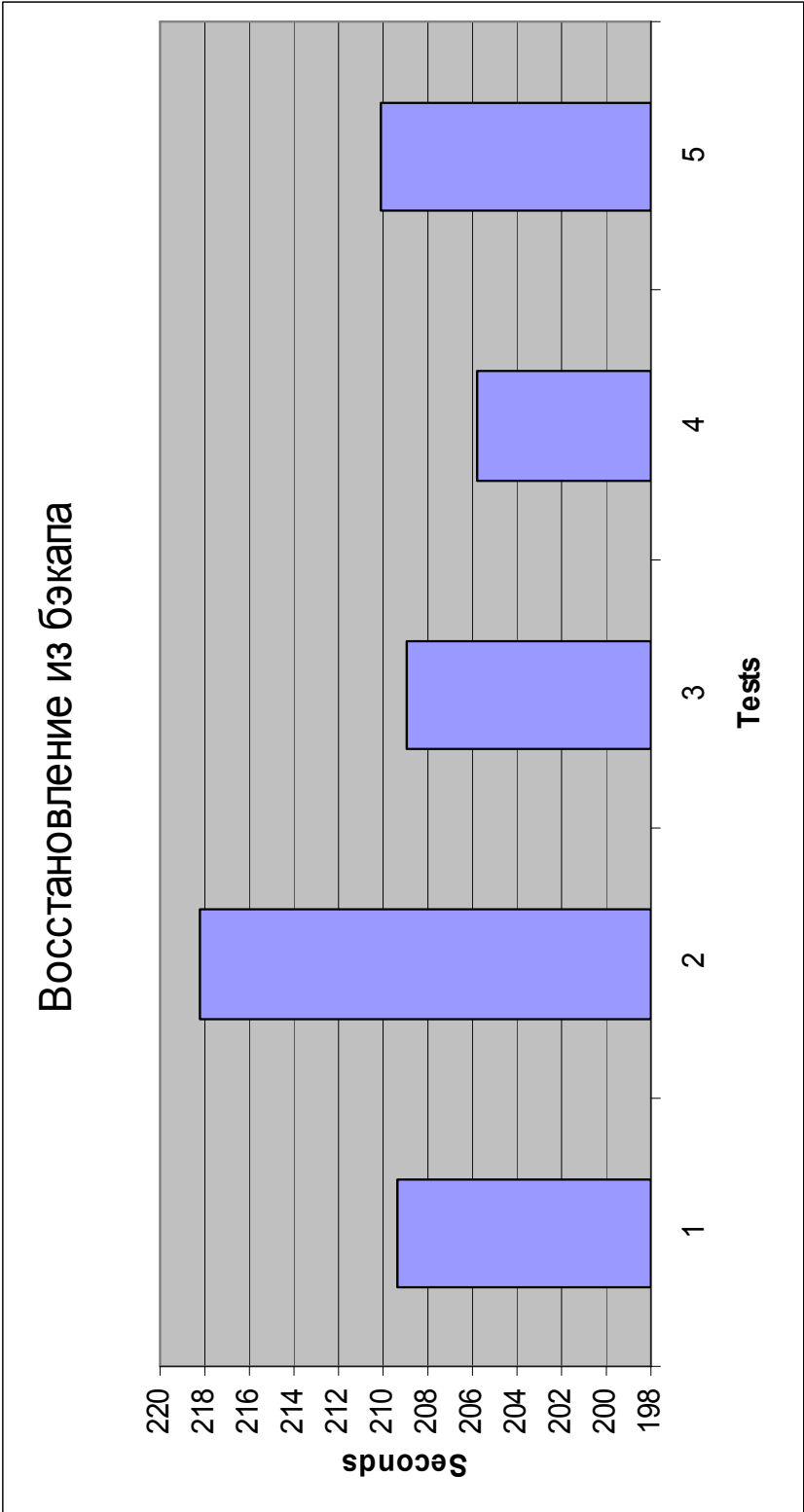
- Использование утилиты ttRestore

```
Command> dssize;
```

```
PERM_ALLOCATED_SIZE:      3174400
PERM_IN_USE_SIZE:         2599036
PERM_IN_USE_HIGH_WATER:  2599036
TEMP_ALLOCATED_SIZE:      20480
TEMP_IN_USE_SIZE:         6523
TEMP_IN_USE_HIGH_WATER:  6523
```

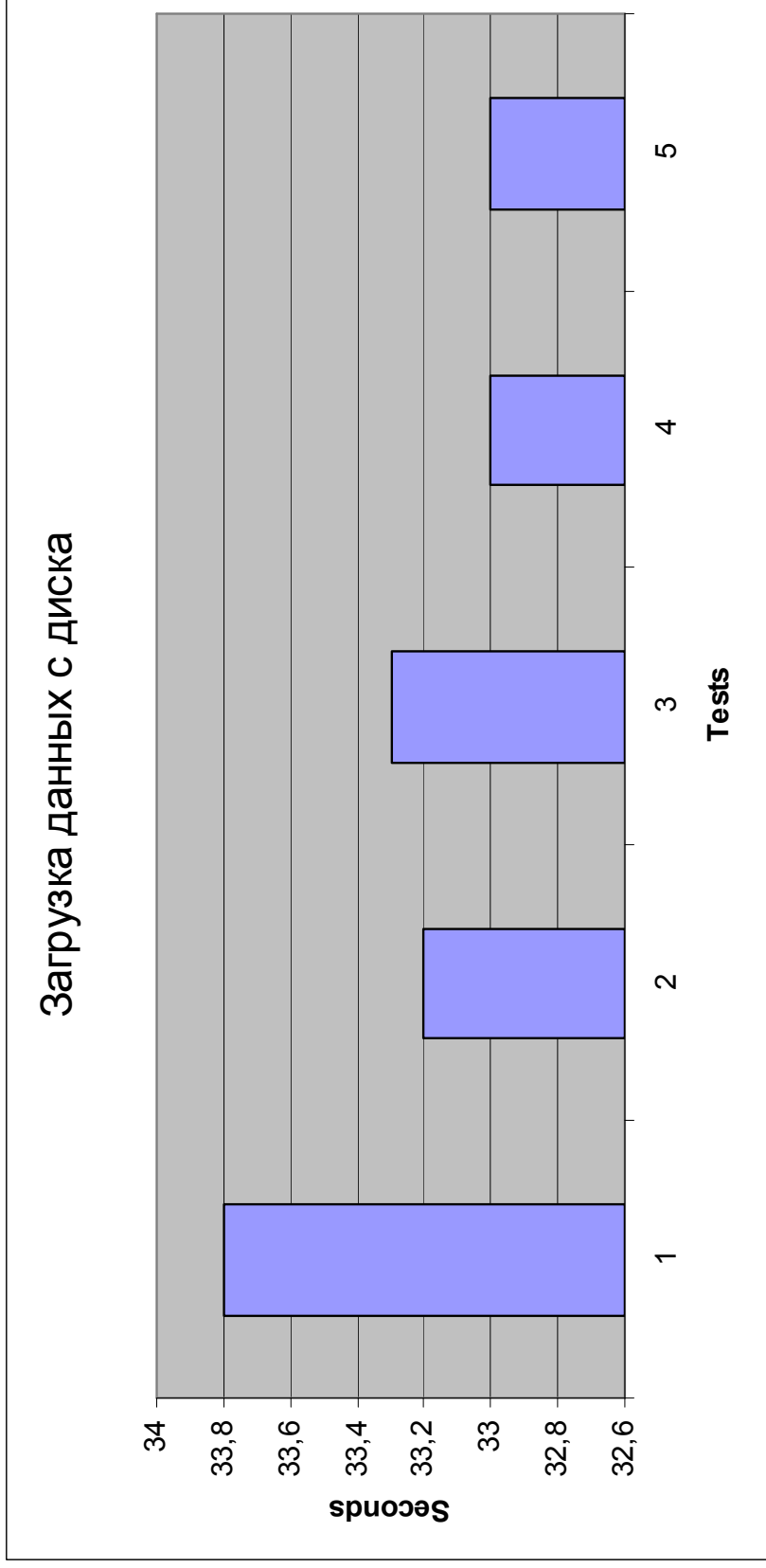


# Выполнение операции restore



AVG time: 210,5 sec.

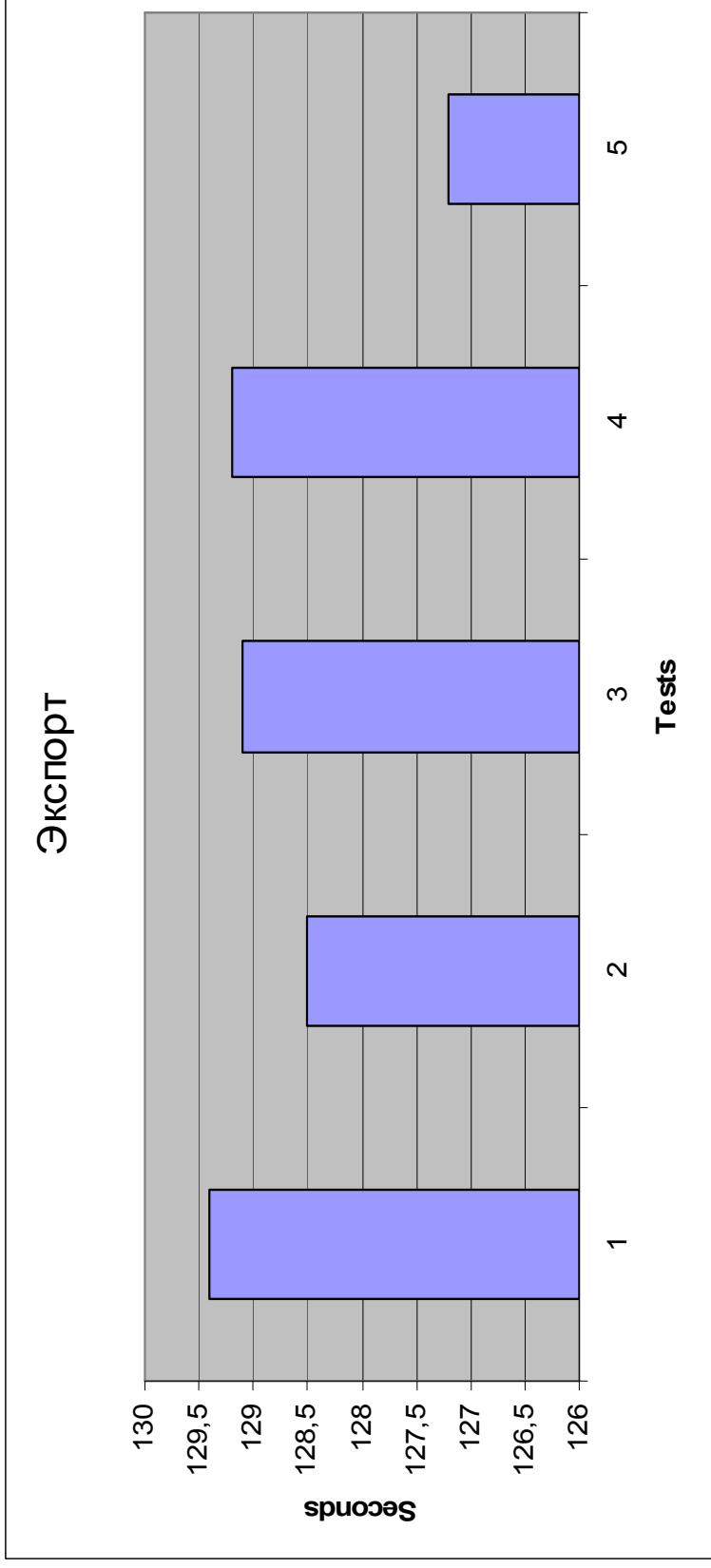
# Загрузка данных с диска



AVG time: 33,26 sec.

# Экспорт данных (30 000 000)

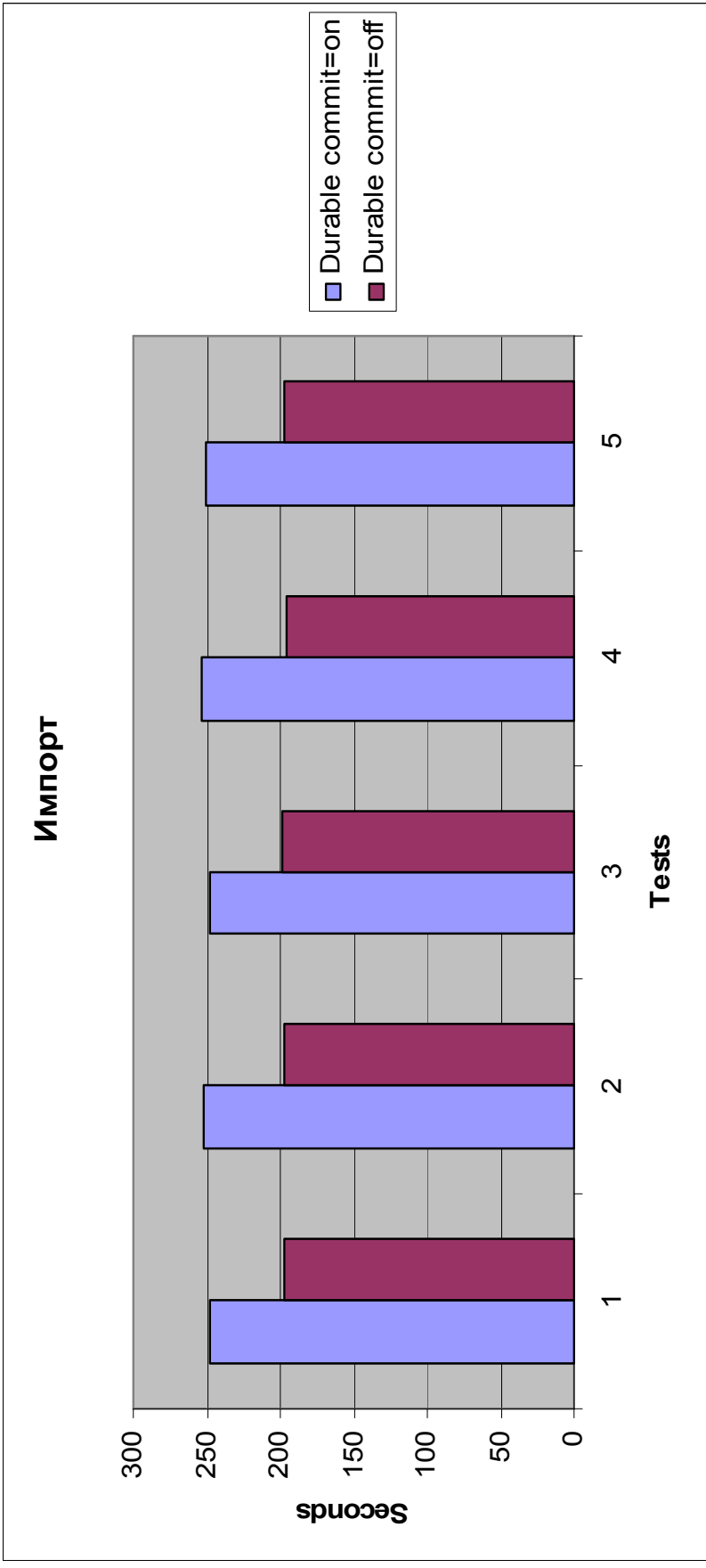
Использование утилиты `ttBulkSp`



AVG time: 128,68 sec.

# Импорт данных (30 000 000)

Использование утилиты `ttBulkCp`



AVG time(durable commit on) : 250,24 sec.

AVG time(durable commit off) : 197,34 sec.



Геннадий Сигалаев  
Старший консультант, Oracle СНГ

Email : [Gennady.Sigalaev@oracle.com](mailto:Gennady.Sigalaev@oracle.com)

Phone : +7 (495) 641 14 00

Direct: +7 (495) 795 22 28

Mobile: +7 (985) 169 75 66