

Санкт-Петербургский национальный исследовательский университет  
информационных технологий, механики и оптики

**Лабораторная работа №3**  
**Дисциплина «Системы управления базами данных»**  
**Вариант 111105**

**Выполнили:**

Съестов Дмитрий Вячеславович  
Баев Дмитрий Владимирович  
Группа Р3417

**Преподаватель:**

Николаев Владимир Вячеславович

Санкт-Петербург  
2019

Цель работы - настроить процедуру периодического резервного копирования базы данных, сконфигурированной в ходе выполнения лабораторной работы №2.

В процессе выполнения работы необходимо создать резервную копию БД на узле db105, настроить процесс репликации, и осуществить процедуру восстановления БД с резервной копии.

Репликацию необходимо организовать двумя способами (в два разных каталога):

1. С помощью Data Pump.
2. С помощью автоматического периодического экспорта / импорта файлов.

#### **Требования к настройке резервного копирования:**

- Вся логика сервиса, осуществляющего репликацию БД, должна быть реализована в виде shell-скриптов.
- Необходимо реализовать задачу для планировщика cron, осуществляющую периодический (например, раз в час) запуск скрипта репликации.
- Каталог, в котором будет создаваться резервная копия экземпляра Oracle, выбирается на усмотрение студента.
- Для того, чтобы можно было продемонстрировать корректность работы репликации, тестовая база не должна быть пустой. Т.е. предварительно в ней нужно создать тестовые таблицы и заполнить их тестовыми данными, осуществив несколько транзакций.

#### **Выполнение**

##### **1. Создание скриптов для заполнения БД**

create.sql

```
CREATE TABLE foo (  
    id NUMBER PRIMARY KEY,  
    value VARCHAR2(15) NOT NULL  
) TABLESPACE DRY_WHITE_LOVE;  
  
CREATE TABLE bar (  
    id NUMBER PRIMARY KEY,  
    value VARCHAR2(15) NOT NULL  
) TABLESPACE NICE_RED_MATH;
```

insert.sql

```
INSERT INTO foo VALUES (1, 'dry');  
INSERT INTO foo VALUES (2, 'white');  
INSERT INTO foo VALUES (3, 'love');  
  
INSERT INTO bar VALUES (1, 'nice');  
INSERT INTO bar VALUES (2, 'red');  
INSERT INTO bar VALUES (3, 'math');
```

##### **2. Подготовка директорий**

```
$ mkdir /u01/backup  
$ mkdir /u01/backup/datapump  
$ mkdir /u01/backup/auto
```

```
$ sqlplus
```

```
SQL> CREATE DIRECTORY DATAPUMP_DIR as '/u01/backup/datapump/';  
SQL> GRANT READ, WRITE ON DIRECTORY DATAPUMP_DIR TO SYSTEM;  
SQL> exit
```

### 3. Создание скриптов для Data Pump

#### expdp.sh

```
ORACLE_USER=admin
ORACLE_PASSWORD=oracle
DATE=$(date +"%Y.%m.%d_%H.%M")

expdp $ORACLE_USER/$ORACLE_PASSWORD full=y \
    directory=DATAPUMP_DIR \
    dumpfile=${DATE}_${ORACLE_SID}_full_export.dmp \
    logfile=${DATE}_${ORACLE_SID}_full_export.log \
    flashback_time=SYSTIMESTAMP

scp -i ~/.ssh/id_rsa /u01/aee11/backup/datapump/${DATE}* oracle@db192:
/u01/aee11/backup/datapump
```

#### impdp.sh

```
read -r filename
impdp admin/oracle content=all table_exists_action=replace full=y
directory=DATAPUMP_DIR dumpfile="$filename"
```

### 4. Создание скриптов для exp/imp

#### exp.sh

```
ORACLE_USER=admin
ORACLE_PASSWORD=oracle
DATE=$(date +"%Y.%m.%d_%H.%M")

exp ${ORACLE_USER}/${ORACLE_PASSWORD} full=y \
    file=/u01/aee11/backup/auto/${DATE}_${ORACLE_SID}_full_export.dmp \
    log=/u01/aee11/backup/auto/${DATE}_${ORACLE_SID}_full_export.log \
    consistent=y statistics=none

scp -i ~/.ssh/id_rsa /u01/aee11/backup/auto/${DATE}* oracle@db192:/u01/aee11/backup/auto
```

#### imp.sh

```
read -r filename
imp admin/oracle full=y file="/u01/aee11/backup/auto/$filename" constraints=n
ignore=y
```

### 5. Создание задачи для cron

```
0 * * * * /u01/aee11/shscripts/expdp.sh
0 * * * * /u01/aee11/shscripts/exp.sh
```

### Выводы

В ходе выполнения данной работы мы изучили такие средства для резервного копирования БД Oracle, как Data Pump и утилиты exp и imp, а также планировщиком задач cron, который позволил запускать скрипты автоматически.